

Nobody Can Lint a Merkle Tree Certificate *Yet*



Corey Bonnell
TurboLight Solutions

Transparency.dev Summit 2026 · ⚡ Lightning Talk

What is certificate linting?

Certificate linting

Automated checks/tooling to flag *malformed* or *non-compliant* certificates

Think of a linter for programming code, but for PKI artifacts

Two lessons from the Web PKI

Certificate transparency

is needed to make *visible* what CAs sign with their private keys

Certificate linting

is needed to constrain what CAs *can* sign with their private keys

Linting can run:

pre-issuance a gate on to-be-signed messages before the key signs

post-issuance anyone can audit at scale by fetching CT logs

Why lint?

[1] interop

Relying parties
shouldn't need lax
parsers to tolerate
malformed messages

*Postel's Law Considered
Harmful*

[2] security

Enforce security
requirements such as
industry-required key
quality and known weak
key checks

[3] compliance

Ensure CAs actually
follow the Baseline
Requirements and root
program policies

Both arrived late

- The Web PKI was in wide use long before CT or linting existed
- By then, preventable problems were common and requirements had gone unenforced at issuance
- Having both from the outset would have caught issues right away

```
$ cat payoff.txt
```

Widely used libraries like `pyca/cryptography` can now ship *strict* parsers, thanks to cleaner certificates from CAs, driven by linting tools and problems exposed via CT.

MTCs are transparency-native

In Merkle Tree Certificates, issuance is the logging operation.

```
$ lint entry.der  
error: no linter supports TBSCertificateLogEntry
```

MTC should be lint-native, too.

Learn the lessons of the X.509 Web PKI: lint from the outset.

The deadline is coming anyway

notAfter: 270915000000Z

Chrome's quantum-resistant root policy requires **pre-issuance** linting as of September 15th, 2027

But why wait?

Linting now gets you more

All of the previous benefits (interop, security, and compliance), **plus:**

implementation guidance

Interop testing is much easier when tooling exists to check adherence to the specs

standardization guidance

Writing checks for every lintable aspect of a spec surfaces issues in the spec itself

Want to see what a MTC linter might look like?



<https://infrared.turbolightsolutions.com>