

Eight eMudhra Intermediate CA Keys Compromised; Browser Vendors Respond



Mozilla and Google's Chrome Root Program confirmed on October 20 that private keys for eight eMudhra intermediate CA certificates are compromised, allowing certificates for any website that browsers would accept.

- Mozilla and Google's Chrome Root Program confirmed the compromise on October 20.
- Eight eMudhra intermediate CA certificates are affected.
- An intermediate CA signs the certificates websites present, so a key holder could issue certificates for any website that browsers would accept.
- A researcher reported the issue; the source describes emergency fixes across browsers and software.

What happened

Mozilla and Google's Chrome Root Program confirmed on October 20 that the private keys of eight intermediate certificate authority (CA) certificates operated by eMudhra are compromised. A researcher reported the problem, and the source describes emergency fixes being issued across browsers and software.

Why it matters

An intermediate CA signs the certificates that websites present to users. Anyone holding one of the compromised keys could therefore issue a certificate for any website, and browsers would accept it. This undermines the trust that TLS connections rely on, and could allow impersonation of sites or interception of traffic.

The source excerpt does not give further detail on the specific certificates, the scope of misuse, or the fix timelines, so organisations should rely on vendor updates for those specifics.

What to do next

The practical response is to apply browser and software updates once released, and to check whether your environment trusts certificates chained to eMudhra. Teams that manage trust stores, proxies or certificate pinning should review them now rather than wait for an incident.

What to do now

1. Apply browser, operating system and security software updates as vendors release them.
2. Check whether any internal or customer-facing systems trust or use certificates chained to eMudhra.
3. Review trust stores, TLS inspection proxies and certificate pinning configurations.
4. Monitor vendor and root program advisories for scope and remediation details.
5. Add certificate trust incidents to your incident response playbooks if not already covered.