§ TLS 1.3 SESSION RESUMPTION · AN UNDEFINED REQUIREMENT, AND WHAT IT IS ALREADY COSTING

TLS 1.3 never tells the server what a resumed connection must re-check.

Post-quantum makes resumption the mechanism that decides whether authentication is affordable, and it is being switched off because nobody can say what it is allowed to skip.

01 Five deployments where a replayed verdict is not re-checked

A full handshake sends the certificate chain and the endpoint validates it. A resumption sends none of that: both sides carry the earlier result forward, which is why it is cheap. Nothing tells the server what it must re-establish before relying on that carried-forward verdict. In practice:

DEPLOYMENT	WHAT THE RESUMED CONNECTION SKIPS	SOURCE
Two virtual hosts on one server, one behind mutual TLS	A ticket minted at the public host is offered back under the protected host's name. The server resumes and never asks for a client certificate.	CVE-2025-23419, nginx and independently Cloudflare
A client certificate is revoked	Revocation is not consulted on the resumption path, so the holder keeps access until the ticket expires.	Cloudflare, January 2023
A client certificate expires	The resumed session is accepted; the control full handshake with the same certificate is rejected.	Measured, OpenSSL 3.6.3
A CA is removed from the trust store	Sessions minted while it was trusted keep resuming, because the store is consulted at the full handshake and not again.	CVE-2025-68121, Go, since fixed
A mesh issues 24-hour workload certificates	The point of a short certificate is that a compromised workload loses access quickly, but the ticket key outlives the identity. The window is bounded at about twelve hours by accident, not design.	Istio and Envoy, as shipped

These are five of **eleven identifiers across eight pieces of software, 2014 to 2026**. In five of the eleven a party outside the current trust scope gains standing access needing no network position, and in all five the replayed verdict is a client-certificate verdict. One is a revocation bypass. The corpus, its inclusion test, and the provenance for every row are in supporting document 1.

02 Why post-quantum makes this decisive

PROFILE	CERTIFICATE MATERIAL	SERVER'S FIRST FLIGHT	VS 14,600 B INITIAL CONGESTION WINDOW
Classical ECDSA P-256, web	3,465 B	3,763 B	fits, one round trip
Post-quantum ML-DSA, web, two SCTs	**18,408 B**	**19,762 B**	**exceeds by 5,162 B: two round trips**
Resumption, either of the above	**No certificate is sent at all. About 1 to 2 kB, always fits, always one round trip.**		

Classically, resuming saves about 3.5 kB and nobody agonises over it. Post-quantum it saves 18 kB and a round trip. The same optimisation stops being a nicety and becomes the thing that decides whether post-quantum authentication is affordable at all.

Constants from FIPS 203 and 204; X.509 encoding measured against live chains. The full model, and what Merkle Tree Certificates and KEM-based authentication change, are in supporting document 3; what conforming gives an operator back, per connection and at fleet scale, is in supporting document 4. **This is not a forecast:** on 27 and 28 August 2026, eleven of fourteen public endpoints we measured already selected X25519MLKEM768 alongside the classical groups on offer. The key exchange has migrated. Authentication, which carries the bytes above, has not.

03 This was written down once, on the server, and then removed

RFC 6066 section 3, for TLS 1.2, put the rule on the party that could enforce it: *"A server that implements this extension MUST NOT accept the request to resume the session if the server_name extension contains a different name."* TLS 1.3

removed it. RFC 9846 section 4.3.11 records the decision: *"there is no need for the server to associate an SNI value with the ticket."*

What remains is one normative certificate-scope rule at resumption, RFC 9846 section 4.7.1, and it is written for the client: *"Clients MUST only resume if the new SNI value is valid for the server certificate presented in the original session."* On the time bound the RFC is normative but unquantified: it is *RECOMMENDED* that implementations limit the lifetime of resumption keying material against the peer certificate's lifetime and the likelihood of intervening revocation. **On scope, server-side, there is nothing.** Every entry in the 2025 cluster is a server-side failure that violated no TLS-layer requirement, because nothing was asked of the server.

One profile does ask, which is the proof this can be written. RFC 9190 section 5.7 calls it a time-of-check time-of-use vulnerability and states that decisions made on information that has since changed *"MUST be reevaluated."* Section 05 generalises that from one profile to TLS.

04 Who has turned it off, and what that costs

Faced with a requirement the specification does not state, operators land on the answer that is safe without further analysis: do not resume under client authentication. **As last measured or documented, four parties do not resume** under mutual TLS, and where the switch was thrown it was thrown network-wide rather than per customer. The population paying for it is the one using client certificates: service meshes, API gateways, business-to-business APIs and zero-trust estates, which will also meet post-quantum certificate sizes first. Its global size has never been measured.

One vendor has now done this twice, which is the argument in one history. In January 2023 Cloudflare found revoked client certificates being honoured on resumption. It disabled resumption for mutual TLS, then built a real fix, checking revocation status at resumption, and re-enabled the feature within weeks. That fix was correct and it was specific: it bound revocation. Two years later CVE-2025-23419 was a *scope* failure, which the 2023 fix did not address and was never meant to. The switch went off again, network-wide, and has stayed off. A vendor that solves this class correctly still has no way to know what else the class contains, because nothing defines it. That is the difference between a patch and a specification.

Resumption is offered almost everywhere it could be: **94.8%** of reachable TLS domains issue a session ticket, 1,108,322 of 1,169,527, from a scan that probed by sending an HTTP request (Hebrok et al., USENIX Security 2025, Table 3 and section 5.1.1). Exploitation is not what is expensive here: the same scan classified 176 of 4,370 manually reviewed resumptions as vulnerable. The cost paid today is the retreat, not the breach.

05 The fix

> **An endpoint MUST NOT rely on a replayed authentication verdict for any decision that a full authentication, performed when the verdict is first relied upon, would decide differently.**

It binds the decision, not the handshake path, and it holds in either direction: the server holding a client-certificate verdict, which is every 2025 entry above, and the client holding a server-identity verdict, which RFC 9846 already binds. Completing a resumed handshake is transport, not reliance. Four conforming ways to satisfy it, only one expensive: **re-establish** from the cached session, one chain validation; **refuse to resume**, the state the four parties above are in and what this paper asks the IETF to stop making anyone settle for; **defer the decision**, as Apache does with a 421 after resuming; or **over-approximate**, binding trust configuration and authentication policy into the resumption scope so any change forces a full handshake, paired with a lifetime cap. None requires a protocol change.

The conditions a verdict depends on are validity across the whole cached path, scope meaning identity *and* authentication policy, and continued acceptance by the current trust configuration. Bind only the host, as RFC 9846's client-side rule does, and CVE-2025-23419 conforms to the rule, which is why policy has to be in scope. The full normative text, a survey of how nginx, BoringSSL, Go, HAProxy, Envoy and Fastly bind a session today, and the adjudication of which of them conform, are in supporting document 2. **None of the six fully conforms.** That is not an indictment of the implementers. It is what an unwritten requirement produces, and it is the argument.

Reproducible: the experiments, the panel and the per-host run files are public in quantakrypto/tls-resumption-tests, and the same state is measured daily on a larger panel by quantakrypto/pqc-observatory. **Where follow-up should occur:** the TLS working group, or UTA if the group judges this deployment guidance rather than protocol. **Remaining limits:** no global measurement of the mutual-TLS population exists, so the affected share is large but unsized; and if Merkle Tree Certificates or trust-anchor negotiation shrink post-quantum chains, the per-connection saving shrinks with them.