Scorecard 001: The “Quantum Safe” Claim

How the five-category scorecard grades the broadest claim in the market — and why its biggest problem is not that it’s false, but that nothing existing today can prove or falsify it.

CLAIM UNDER REVIEW

“Our product is quantum safe.”

As commonly used in vendor marketing, sales decks, and RFP responses. This example grades the generic claim, not any specific vendor or product.

The structural problem, stated up front

No cryptographically relevant quantum computer (CRQC) exists. That means “quantum safe” cannot be demonstrated by running the attack it defends against — and it cannot be refuted that way either. Today’s NISQ hardware cannot break RSA-2048 or ECC-256, so it cannot meaningfully “fail to break” a PQC scheme. A claim that no available hardware can prove or falsify must instead be graded on its **evidence structure**: what kind of evidence backs it, where the claim tends to die in practice, and what a buyer can actually purchase. That is what the scorecard below does.

The scorecard

CATEGORY	GRADE	FINDING
Hardware reality QPU, simulator, or neither?	NOT DEMONSTRABLE	There is no hardware on which this claim can be tested. Corollary: any vendor offering a live “quantum resistance demo” is demonstrating theater — the demo would “validate” unpatched 1990s crypto just as convincingly. Treat a hardware demo of quantum-safety as a negative signal, not a positive one.
Signal integrity What evidence actually exists?	PARTIAL — MATH, NOT DEMOS	Legitimate evidence exists, but it is mathematical and adversarial, not empirical: security reductions to hard problems, an eight-year public NIST competition, and continuous cryptanalysis. “Quantum safe” honestly means “no known quantum or classical attack — yet.” First fork in the audit: is the algorithm a NIST standard (FIPS 203 / 204 / 205, HQC) or proprietary? Proprietary post-quantum math that skipped public cryptanalysis is a near-automatic reject.
Post-processing & implementation dependence Is the product safe, or just the algorithm?	WHERE THE CLAIM DIES	A standardized algorithm inside a product does not make the product quantum safe. Side-channel leakage in the implementation, key management, incorrect hybrid-mode composition, and the rest of the stack still running ECC/RSA (TLS termination, firmware signing, third-party dependencies) all void the marketing claim in practice. This is the category where most real “quantum safe” claims fail an audit — for entirely classical reasons.
Threat relevance Does this matter to risk timing?	REAL — ACT ANYWAY	Unprovability is not an excuse for inaction. Harvest-now-decrypt-later makes migration urgent for any data with a confidentiality lifetime beyond a few years, NIST says organizations should not wait, and the 2026 executive order sets a Dec 31, 2030 PQC deadline for high-impact federal systems — a timeline that will cascade through supply chains. The threat model is sound even though the safety claim is untestable.
Business decision Ignore, monitor, test, fund, or reject?	SPLIT VERDICT	**Reject the phrase; fund the migration.** Reject “quantum safe” as an absolute assurance — it is not a purchasable property. Fund migration to NIST-standardized algorithms in hybrid mode, and make **cryptographic agility** the contractual requirement: the vendor property you can actually buy is the ability to swap algorithms quickly when — not if — cryptanalysis moves.

WHY “AGILITY, NOT ASSURANCE” IS THE VERDICT

The Rainbow / SIKE lesson

During NIST’s own standardization process, two prominent post-quantum candidates were broken — Rainbow, a third-round signature finalist, and SIKE, a fourth-round encryption candidate. Neither fell to a quantum computer. Both fell to **classical** attacks; SIKE was broken on an ordinary desktop CPU in roughly an hour. Every “quantum safe” scheme that has died so far has died classically. This is the single most important fact a buyer can hold while reading a PQC pitch: the risk you can plan for is not the arrival of a CRQC, it is the ordinary, ongoing motion of cryptanalysis — and the only defense you can purchase against that is the ability to change algorithms without re-architecting your product. “Quantum safe” is a snapshot. Agility is a property.

The falsifiability note (epistemic integrity)

On the Firebringer checklist, every claim must name the result that would kill it. “Quantum safe” has a kill condition — a successful attack, quantum or classical — but no **confirmation** condition reachable with any hardware that exists. It can be falsified someday; it can be proven never. Claims with this shape are not worthless, but they must be sold as what they are: standardization plus surviving scrutiny plus agility. Any vendor selling it as a settled, demonstrated property is selling a category error.

The ten questions to ask before accepting the claim

01 · Which algorithm, exactly?
FIPS 203 (ML-KEM), FIPS 204 (ML-DSA), FIPS 205 (SLH-DSA), HQC — or something proprietary? If proprietary: why should it survive where Rainbow and SIKE, with years of public cryptanalysis behind them, did not?

02 · Hybrid or pure?
Is PQC composed with classical crypto (e.g., ML-KEM + ECDH) so that both must fail before security does? If pure PQC, what is the contingency when cryptanalysis moves?

03 · What is your crypto-agility story?
Concretely: how long, and what does it cost the customer, to swap the algorithm? Is agility in the architecture or in the roadmap slide?

04 · Whose implementation?
Which library, what version, audited by whom, and are known side-channel findings against that implementation patched? ‘We use Kyber’ is not an answer.

05 · Where does classical crypto still live?
TLS termination, code and firmware signing, session resumption, third-party SDKs, hardware roots of trust. A product is only as post-quantum as its least-migrated dependency.

06 · What does ‘quantum safe’ mean in the contract?
Is it a marketing phrase or an SLA? What are the vendor’s obligations if the algorithm is deprecated or broken?

07 · Have you run a hardware ‘quantum resistance’ demo?
A deliberate trap: a ‘yes’ reveals either a misunderstanding of NISQ capability or a willingness to sell theater. Either answer is audit-relevant.

08 · What breaks downstream?
PQC keys and signatures are larger and slower. What happens to your certificate chains, embedded devices, packet sizes, and latency budgets?

09 · What is your key-lifetime and HNDL posture?
Which of the customer’s data classes have confidentiality lifetimes long enough that harvest-now-decrypt-later already applies?

10 · Who attests it?
Independent audit, FIPS validation status, or self-assessment? ‘Trust us’ is the answer this scorecard exists to replace.

Worked example for illustration. Grades the generic marketing claim as commonly used; no specific vendor, product, or standardized algorithm is being evaluated or disparaged. NIST-standardized PQC algorithms represent the current best available practice — that is precisely why the honest framing is standardization + agility, not “safe.”