ELTON

WHITEPAPER · TESTING PROCESS

How ELTON Tests a Medical Device

Complete coverage, traceable test cases, and exploitability proven on the device: what FDA expects to see in a test record, and how ELTON produces it release after release

KEY TAKEAWAYS

- FDA reads a cybersecurity test for coverage before it reads it for findings. The record it expects holds the test cases that cover the product's full scope, the vulnerabilities that stem from them, and a justified disposition for every one. A short list of curated vulnerabilities with nothing behind it no longer holds up in review.
- Exploitability decides what gets fixed and when, and everything found goes in the record. A weakness with no exploit path today stays there with its rationale, because the next root cause, the next release or the next researcher can give it one. FDA's guidance for AI-enabled devices says so plainly: six of its seven named threats live upstream of the deployed device, almost never exploit at runtime, and FDA still expects them tested and shown.
- ELTON gets completeness from automation and depth from people. Four lanes, SAST, DAST, fuzzing and pentest-class test cases, run as one pipeline from a digital twin of the release. Senior pentesters review every automated result and spend their hours on the attack surfaces that decide a device's exposure: authentication, trust boundaries, chaining and bypass, and testing beyond the plan.
- Every result traces back to the twin: component, interface, data flow, test case, evidence, verdict. A reviewer walks that chain, and it is what turns "Not Affected" from an opinion into a result.
- The process is built for repetition. The SPDF and the TPLC expect iterative testing with history, so the twin is onboarded once, every test case is retained, the next release tests what changed in days, and the submission report combines every test run on that release.
- Most of the activities in a subscription were written backwards from FDA feedback on customer submissions, and each one traces to the requirement it evidences. Section 6 carries the map, activity by activity.

Executive summary

The pentest report most manufacturers grew up with was short. Three to five vulnerabilities, chosen and polished by the tester, with a methodology paragraph in front and no record of what was tried. It read well because the tester decided what you saw.

FDA stopped accepting that shape. The premarket guidance puts cybersecurity testing inside the Secure Product Development Framework, asks for it throughout development and at regular intervals after release, and expects the submission to show the work: which test cases were launched against which parts of the product, what each one found, and why each finding was dispositioned the way it was. Reviewers now question a report with few findings as readily as one with many, because a clean report with no coverage behind it reads as untested.

ELTON's process is built to produce that record at a cost a manufacturer can repeat. A digital twin of each release generates the test plan. Automation runs it across every lane and every interface, which is how the whole scope gets covered. Senior pentesters review every automated result and spend their time on the surfaces where devices actually fall: authentication, trust boundaries, exploit chains and bypass. Every result, pass or fail, lands on the twin with evidence, and every finding gets one of three verdicts: directly exploitable, conditionally exploitable, or a weakness recorded with its rationale.

The output looks different from the curated list. It is larger, and the handful of vulnerabilities you are used to seeing is still at the top of it. The difference is that everything behind them is now on the record, which is what the reviewer asks for and what the next release reuses.

1. What FDA now expects from a cybersecurity test

FDA's premarket cybersecurity guidance moved testing from an event to a process, and it says in plain words when testing should happen:

> *"FDA recommends that cybersecurity testing should occur throughout the SPDF. Security testing early in development can ensure that security issues are addressed prior to impacting release timelines and can prevent the need to redesign or re-engineer the device. After release, cybersecurity testing should be performed at regular intervals commensurate with the risk (e.g., annually)." Section V.C, Cybersecurity Testing*

Three expectations follow from that, and they decide how a test has to be run.

Test cases that cover the scope of the whole product. The submission is expected to carry an enumerated list of the cybersecurity test cases launched against the product, mapped to the architecture and the threat model, so a reviewer can see coverage of the attack surface even where the finding count is zero. Firmware, mobile, cloud and the interfaces between them are all in scope. The same list is where threat mitigation testing lives: FDA asks for evidence that each claimed security control holds up against the threat it was designed to stop, which means attacking the login with credential stuffing, session fixation, lockout bypass and token replay, and recording each attempt as a test case with a result. A requirements trace that shows a login screen exists is evidence of a requirement, and nothing more.

Vulnerabilities that stem from those test cases. Findings are expected to trace to the tests that produced them, and the tests to the threats they were written for. Without that trace a finding is an assertion, and a reviewer has no way to tell where the testing stopped. In ELTON's own conversations with FDA reviewers this year the asks were specific: show the individual test cases from the penetration test, explain incomplete coverage, and explain a lack of findings.

A justified disposition for every finding, whether or not it is exploitable today. Legacy reports break here. FDA asks that everything found be rated, dispositioned and justified, with a justification that survives an audit a year later, and exploitability plays no part in whether a finding belongs in the record. Exploitability drives what you fix and when, and it never decides what you may leave out.

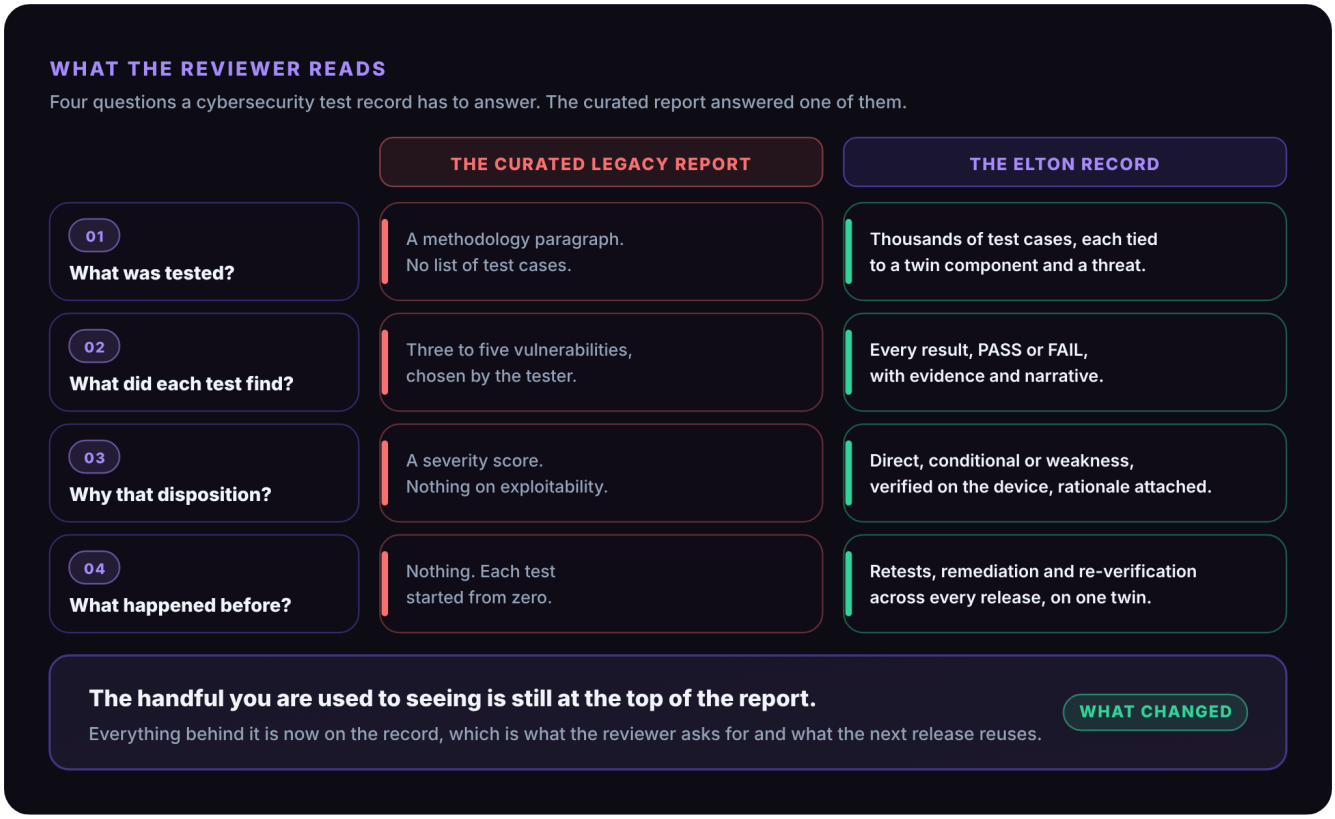


Figure 1. What the reviewer reads. The curated report answered one question. The record FDA asks for answers four, and the handful of vulnerabilities you are used to seeing is still at the top of it.

The AI-enabled device guidance makes the point unmissable

The January 2025 draft guidance for AI-enabled device software functions names seven cybersecurity threats in Section XII: data poisoning, model inversion or stealing, model evasion, data leakage, overfitting, model bias and performance drift. Only model evasion happens on the deployed device at runtime. The other six begin in the training data, the model registry, the pipeline that built the model, or the postmarket monitoring loop.

Almost none of those six will ever be exploitable on the device in a hospital. FDA expects them tested, found and shown anyway, because the risk they carry is to the model the device depends on, and a reviewer who has read Section XII will look for evidence that each named threat was addressed. ELTON's whitepaper on AI-enabled device testing sets out the thirty-seven technical tests that cover them. The point for this paper is narrower: the agency has told the industry, in writing, that "not exploitable on the device" is a disposition to justify and never a reason to leave something out.

Fragility is real

A weakness that cannot be exploited today is one change away from a vulnerability. A new release opens an interface, or a root cause elsewhere in the product starts producing the condition it needed. A researcher with six months and no budget finds the step the tester ran out of time for. Or a hospital's own security team scans the installed base and files the result with FDA before you have seen it.

FDA knows this, which is why the guidance asks for weaknesses to be tracked and not only fixed. Ignoring a finding because it is not directly exploitable today is a timebomb with a version number. The fix can wait for a later release. The rating, the recorded reason it is not exploitable from the reachable attack surface, and the re-evaluation when the product changes cannot wait. The justification takes the same evidence as any other vulnerability, and a hidden weakness is the first thing a deficiency letter asks about.

2. How ELTON tests: four lanes in, one verified stream out

All FDA required cybersecurity testing, run by the pipeline. ELTON performs it with a toolchain of ELTON-developed tools, orchestrated from a test plan generated off the digital twin, so that coverage of the whole scope is a list before testing starts rather than a hope after it ends. Thousands of test cases trace to every component, interface and data flow of every release, and the method is built on 13 years of testing medical devices and putting findings through FDA review.

Four disciplines run together across hardware, firmware, software, web, mobile and network, on every release:

LANE	WHAT IT EXERCISES	HOW
SAST	Source and binaries	Static tools built to read the product's own source and interrogate its compiled binaries, including bespoke firmware
DAST	Running services	Dynamic tools generated for the target, reaching into the services the device actually exposes
Fuzzing	Protocol listeners	Malformed traffic through DICOM, HL7, MQTT and proprietary listeners, on the real interface
Pentest-class test cases	Human techniques, encoded	Authentication bypass, session abuse, privilege and trust-boundary crossing, chained access: the moves a tester makes, written as repeatable test cases

Every lane ends in the same place: verification against the twin, the code and the real device. Results are deduplicated, so the same issue found by three tools is one finding, and every finding arrives with evidence a reviewer can follow.

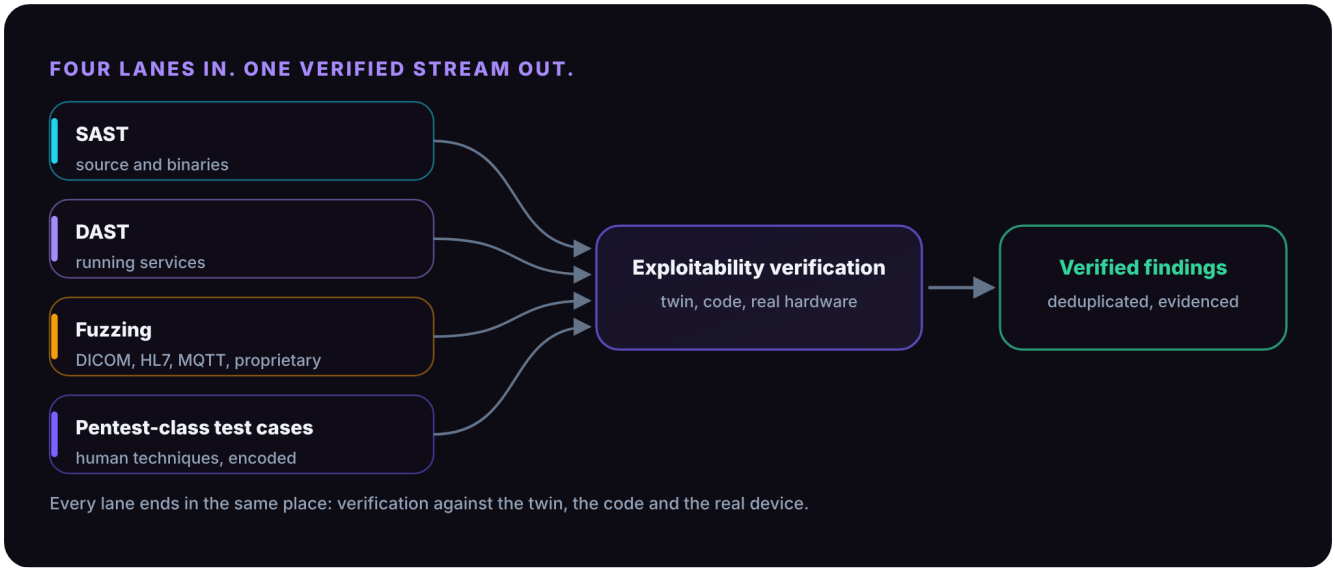


Figure 2. Four lanes in, one verified stream out. SAST, DAST, fuzzing and pentest-class test cases run as one pipeline, and every lane ends in exploitability verification against the twin, the code and the real hardware.

The twin gives the pipeline context no legacy testing firm has. It holds the components, interfaces, data flows, assets, trust boundaries and countermeasures of the release, built from the architecture documentation, SBOMs and threat model your quality system already produces. Test cases are generated from those elements and their EMB3D threat properties, which is why the coverage reads as a list before a single test runs. TestLink carries the whole pipeline onto the physical device: an appliance cabled to the real hardware and driven from the platform over out-of-band 5G, so firmware, serial, radio and proprietary links get the same treatment as the web interface.

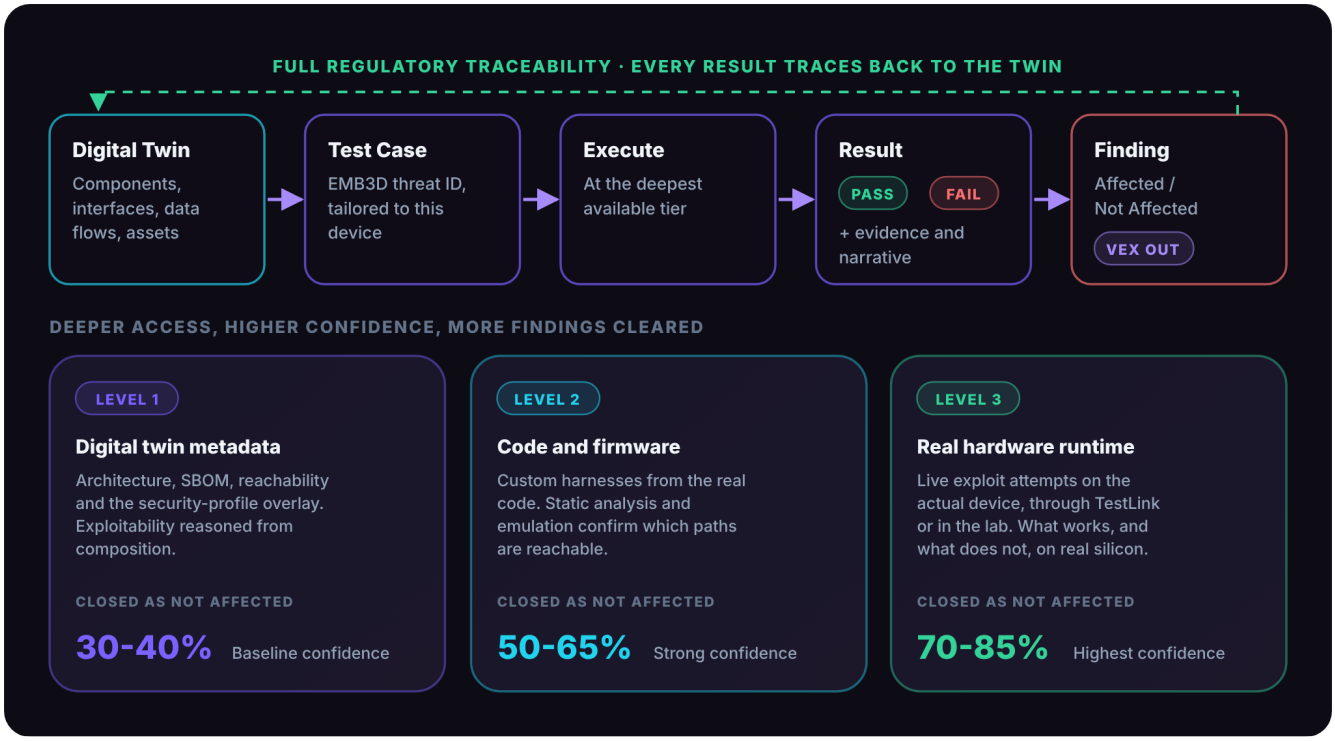


Figure 3. Full regulatory traceability. Digital twin to test case to execution to result to finding, with the VEX disposition out the end. Deeper access raises confidence and clears more findings as Not Affected.

Every result traces back to the twin

Cybersecurity testing is now about evidencing the tests that were run. Every test case is derived from the device's own threat model, runs at the deepest tier of access you have opened, and lands as a result with evidence and narrative attached. A result is PASS or FAIL. A FAIL becomes a finding, a finding is verified to Affected or Not Affected, and the VEX disposition is generated from the same record.

The chain is unbroken from thousands of test cases to every part of the twin, mapped to vulnerabilities wherever a result yielded one. A reviewer walks it when they ask how a finding was dispositioned, and so does a product team eighteen months later, when a new CVE lands against a component and someone asks whether that component was ever tested.

3. Automation for completeness, pentesters for depth

The flow inside an ELTON test has two halves, and the order matters.

Automation runs first and covers the whole plan. The twin generates the test cases, the pipeline executes them across all four lanes, and every component, interface and data flow in scope gets exercised. A two-week manual engagement never completes that half, because a person working interface by interface runs out of days long before the plan runs out of test cases. Automation is what makes complete coverage affordable, and it is what makes the coverage list real instead of claimed.

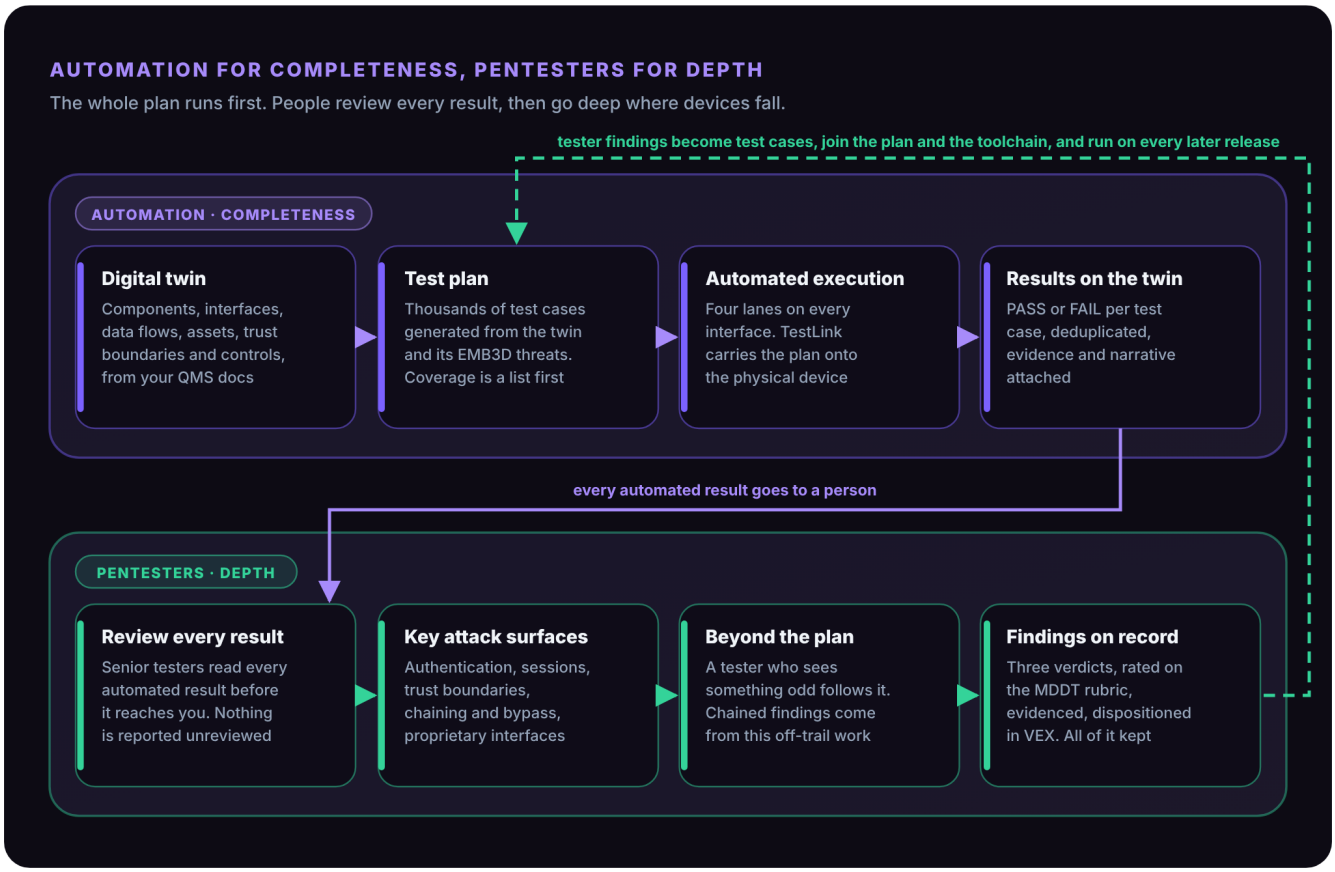


Figure 4. The flow pattern. Automation runs the whole plan for completeness; pentesters review every result and focus on the attack surfaces that decide exposure; what they find becomes new test cases for the next release.

The testers come in behind it. ELTON's senior pentesters, the same team that has carried findings through more than 1,000 FDA submissions, review every automated result before it reaches you. Nothing a tool found is reported unreviewed. Then they spend their hours where devices actually fall: authentication and session logic, privilege boundaries and the trust boundaries between components, exploit chaining across them, bypass of the controls the architecture claims, and the proprietary interfaces no generic tool understands. They also test beyond the plan. A tester who sees something odd in a result follows it, and that off-trail work is where the chained findings come from.

Their findings close the loop. A technique that worked against this release is written back as a test case, added to the plan and to the toolchain, and runs automatically on every release after it. Depth compounds, and the next test starts where the last one ended.

The test durations in an ELTON statement of work follow from this order. A physical device test runs about two weeks and a new-submission platform test about three, because the automated half has to finish before the review half can start, and the review half is where exploitability gets decided. A shorter window leaves the automation untouched and takes the hours out of the review, the chaining and the verification, and the record shows it as fewer results reviewed and more findings left unverified.

4. Exploitability decides the fix, and the record keeps everything found

A vulnerability, in ELTON's method, is a finding that is exploitable today from the reachable attack surface, proven by a test case executed against the target. A weakness is a finding with no exploit path on this device today. Both go in the record, and only the first goes in the fix queue.

Every finding gets one of three verdicts, computed from the twin rather than asserted by an analyst:

VERDICT	WHAT IT MEANS	WHAT HAPPENS TO IT
Directly exploitable	Exploitable on its own from a known entry vector, every precondition met by the product as shipped	Fix now: a few per product, and they carry the regulatory exposure.
Conditionally exploitable	Real, but reachable only while another root cause keeps producing the condition it needs	Watch it: fix the root and the condition closes, and the status recomputes on the twin.
Weakness	In scope, rated, and no exploit path on this device today	Dismissed with the reasoning attached, in VEX terms, and re-evaluated every time the product changes.

The isolation rating never moves. Every finding is scored first, through the MDDT-qualified rubric for CVSSv3 and ELTON's FDA-aligned method for CVSSv4, and that score stays intact for compliance. What changes with exploitability is the status, and the rationale for every status is the graph of entry vectors, conditions and findings on the twin. One fix to a root can retire a dozen findings' worth of action, and the record shows why each one flipped.

Exploitability is answered at runtime, on the device, and cannot be read off source code, an SBOM or an architecture diagram. ELTON verifies at three tiers and reports the confidence each one earns:

TIER	WHAT IS EXERCISED	CLOSED AS NOT AFFECTED
Level 1, digital twin	Architecture, SBOM, reachability and the security-profile overlay; exploitability reasoned from composition	About 30 to 40 percent of findings, baseline confidence
Level 2, code and firmware	Custom harnesses from the real code; static analysis and emulation confirm which paths are reachable	About 50 to 65 percent, strong confidence
Level 3, real hardware runtime	Live exploit attempts on the actual device, through TestLink or in the lab	About 70 to 85 percent, highest confidence

The deeper the access you open, the more findings close as Not Affected with evidence, and the shorter the list that needs an engineer. Open-box access pays for itself here: it is how the 99 percent get proven and the 1 percent get found.

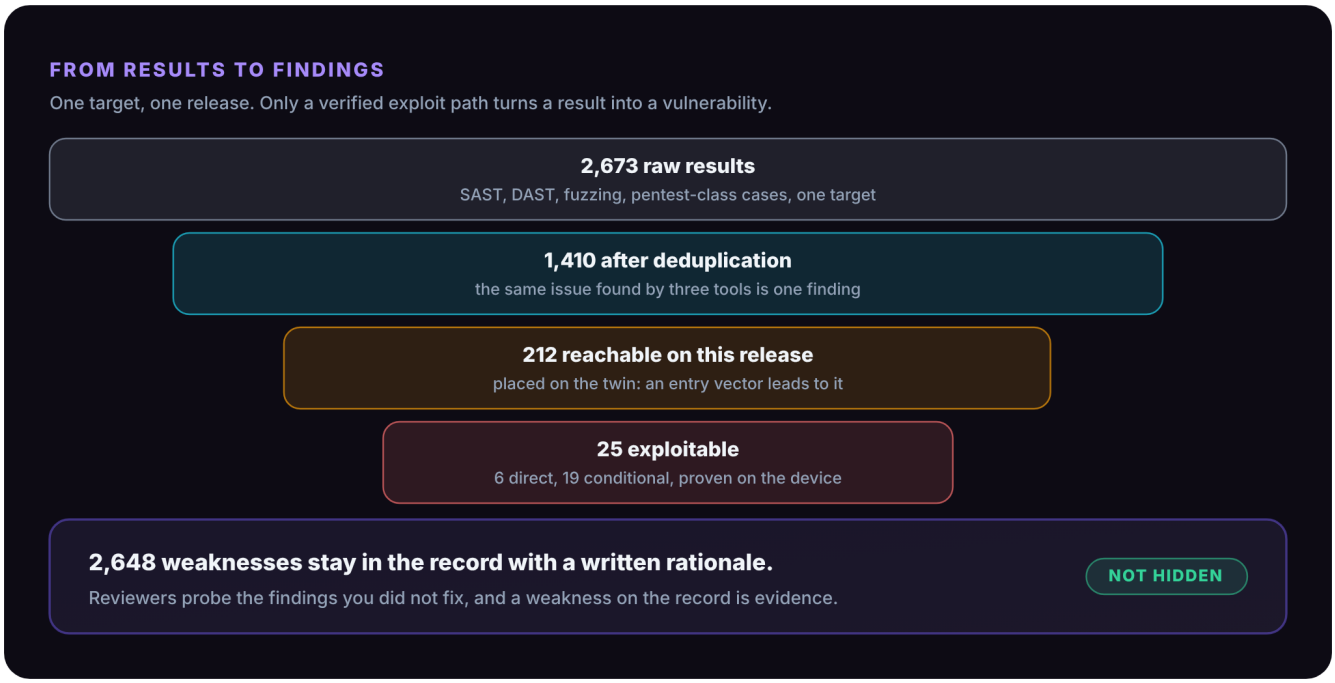


Figure 5. From results to findings. One target, one release: thousands of raw results, hundreds reachable, a few exploitable. The weaknesses stay in the record with a written rationale, because reviewers probe the findings you did not fix.

Legacy reports hid this funnel. The tester ran some of it, kept the part that looked like a vulnerability, discarded the rest, and the manufacturer never learned what was tried. ELTON's report keeps the whole funnel because FDA asks for the whole funnel, and because the weaknesses are the evidence that the few at the top are the right few.

5. Built for repetition: the twin, the history, and the next release

FDA's framework assumes a product will be tested many times.

> *"An SPDF encompasses all aspects of a product's lifecycle, including design, development, release, support, and decommission." Section V, Secure Product Development Framework*

> *"Cybersecurity risks may continue to be identified throughout the device's TPLC. Manufacturers should ensure they have appropriate resources to identify, assess, and mitigate cybersecurity vulnerabilities as they are identified throughout the supported device lifecycle." Section V.A.6, TPLC Security Risk Management*

Testing throughout the SPDF and at intervals after release means several tests on one release and more on every release after it, and FDA expects to see that history in the submission: findings discovered and burned down across development, re-verification across builds, and every test run on a release combined in the final report. A clean final report with no history reads as untested. A one-off test, delivered as one report, cannot produce that record no matter how good the tester was.

ELTON onboards a digital twin for every product for exactly this reason, and keeps it current per release instead of scoping a fresh engagement each time. The twin is built once from the documentation your quality system already produces. Every test case ever run against it is retained and re-runnable, with its result and its evidence. When the next release arrives, the twin diffs what changed, the plan covers the change, and the hundreds of test cases already run against unchanged components become the evidence that no further testing was required there. A retest of prior findings takes one to two days, and a milestone test closes in days instead of weeks. The formal submission report combines every test on the release, in order, with the remediation and re-verification in between.

The history keeps working after the submission too. The pipeline monitors the release's SBOM between tests, correlates every new CVE against the twin and the test results, and regenerates the report against today's CVE set with a fresh timestamp, so a report that was true in March is still defensible in September.

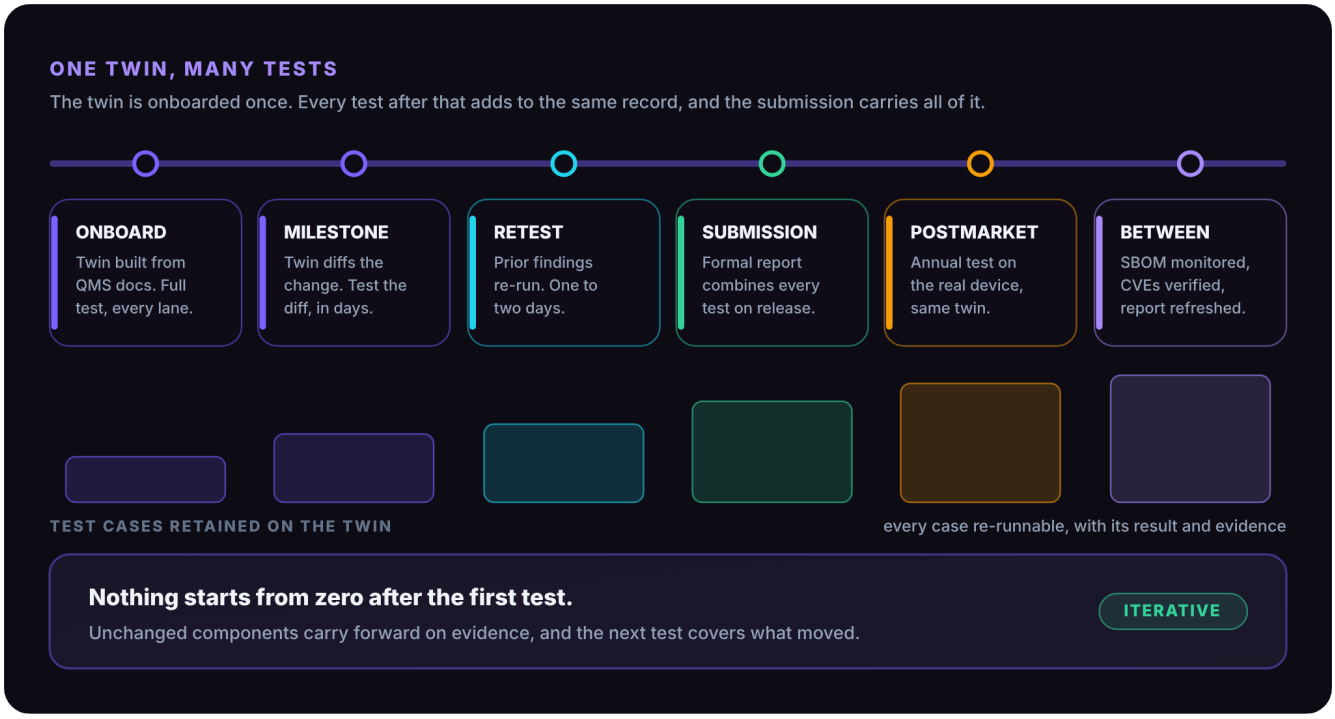


Figure 6. One twin, many tests. The twin is onboarded once; each milestone, retest, submission and postmarket cycle adds to the same record, and the final report combines every test on the release.

Why this runs as a subscription

A process that expects to test the same target many times needs capacity that is already there when the release lands. ELTON reserves it in advance, so a code freeze that slips does not put the product back in a queue, and a critical CVE on a Friday gets verified on the real device instead of waiting on a new statement of work. The economics follow the same logic: the first test of a product carries the onboarding, every test after it pays for the diff, and depth compounds across the term instead of restarting at zero. ELTON's whitepaper on continuous testing sets out the seven drivers in full. One of them is enough for this paper: the researcher who eventually finds the flaw in your device has no clock, and the affordable way to keep up is to keep testing the same product for years.

6. Every ELTON activity, traced to FDA guidance

Most of the activities in an ELTON subscription were written backwards from FDA feedback. Across more than 1,000 submissions, reviewers asked ELTON's customers for the test case list behind a penetration test, for an explanation of coverage, for the exploitability determination behind a dismissal, and for the trace from a finding to its test and from the test to the design control it verified. Each time a deficiency question came back, the answer became a standing activity, so the next submission already carried it.

The table maps every activity in a subscription to the requirement it evidences and to where that requirement is written. Section numbers follow the premarket guidance as cited in this paper; the postmarket guidance is FDA's 2016 Postmarket Management of Cybersecurity in Medical Devices.

INCLUDED ACTIVITY	WHAT FDA ASKS FOR, AND WHERE IT IS WRITTEN	WHAT ELTON DELIVERS
Iterative testing, premarket	Cybersecurity testing through development, with the reports in the submission: security requirements testing, threat mitigation testing, vulnerability testing and penetration testing. Premarket guidance, Section V.C Cybersecurity Testing; Section 524B(b)(2), processes that give reasonable assurance the device is cybersecure.	Every premarket release is tested through to final, results land on the digital twin, and the submission-ready report is regenerated on demand.
Annual testing, postmarket	Testing repeated at regular intervals after release, annually in the guidance's own example, as part of the postmarket cybersecurity plan. Premarket guidance, Section V.C and the cybersecurity management plan; Section 524B(b)(1), the plan to monitor, identify and address postmarket vulnerabilities.	The supported release is tested once a year on the real device, and the living report stays current between tests.
Continuous vulnerability verification (exploitability determination)	Monitor for new vulnerabilities after release and assess the exploitability of each one before deciding what it means for the device. Section 524B(b)(1); Postmarket guidance, Section VI.A, assessing the exploitability of the cybersecurity vulnerability.	New CVEs are correlated with the twin and the test results, then verified at the twin, in code or firmware, and on the device, so each one resolves to Affected or Not Affected with evidence.
MDDT-qualified vulnerability severity ratings	An exploitability-based cybersecurity risk assessment, with CVSS applied the way FDA qualified it for medical devices. Premarket guidance, security risk management and cybersecurity risk assessment; Postmarket guidance, Section VI.A (CVSS as an exploitability tool); the Rubric for Applying CVSS to Medical Devices, qualified under the MDDT program (Q171974).	Product-adjusted CVSSv3 and CVSSv4 for every finding through the MDDT-qualified rubric, each rating evidenced against the twin's security profile.
Digital twin of each product release	Security architecture views for the release and an SBOM of commercial, open-source and off-the-shelf components. Premarket guidance, security architecture and third-party software components (SBOM); Section 524B(b)(3).	A twin per release: components, interfaces, data flows, assets, trust boundaries and countermeasures, with software and hardware materials exportable as CycloneDX.
Threat-model-driven test case creation	A threat model that informs the risk assessment and the test plan, so what gets tested follows from how the device can be attacked. Premarket guidance, threat modeling under security risk management; Section V.C, threat mitigation testing.	Test cases are generated from the twin's components and their EMB3D properties, so coverage reads as a list before testing starts.
Test case traceability to design	Traceability from security requirements and design controls to the tests that verified them, kept in the design history file. Premarket guidance, Section V.C and its submission documentation; QMSR, 21 CFR Part 820 incorporating ISO 13485 clauses 7.3.3 and 7.3.6, design inputs and design verification.	Every test case maps to the twin component and the control it exercises, and the matrix exports with the report.

INCLUDED ACTIVITY	WHAT FDA ASKS FOR, AND WHERE IT IS WRITTEN	WHAT ELTON DELIVERS
Vulnerability traceability to test cases	Findings tied to the tests that produced them, including a security assessment of unresolved anomalies. Premarket guidance, Section V.C test reports and the security assessment of unresolved anomalies.	Every finding links to its test case, its evidence and its component, so a reviewer can walk from the vulnerability back to the test.
Exploitability classification: directly exploitable, conditionally exploitable, weakness	Decide whether the risk is controlled or uncontrolled from exploitability and severity of patient harm, and separate known unacceptable vulnerabilities from critical ones. Postmarket guidance, Sections VI and VII, controlled and uncontrolled risk; Section 524B(b)(2)(A) and (B).	Three verdicts on every finding, and a VEX-aligned disposition with CISA-defined justifications for everything dismissed.
Prescriptive remediation	Updates and patches on a regular cycle for known unacceptable vulnerabilities, out of cycle for critical ones, with corrective and preventive action under the quality system. Section 524B(b)(2)(A) and (B); Postmarket guidance, Section VII Remediating and Reporting; QMSR, ISO 13485 clauses 8.5.2 and 8.5.3.	One fix per root cause, tied to the chain it closes, written for the developer who ships it and tracked to Fixed in a later release.
Mitigation prioritization	Risk-based remediation: what to fix first, on what timeline, by the risk it carries to patients. Postmarket guidance, Section VII; Section 524B(b)(2).	Root causes ranked by the exploitable chains they close, with mean time to triage and mean time to remediation tracked across the program.

ELTON meets FDA premarket (Section 524B) and postmarket cybersecurity vulnerability testing and management requirements. The rubric is FDA-qualified under the MDDT program; the ratings and dispositions are ELTON's, evidenced against your product.

7. What you receive

OUTPUT	WHEN	WHAT IT CONTAINS
Findings in the platform	As they land, during execution	Every result on the twin and every finding with its verdict, evidence, rating and disposition, plus status updates on plan changes and on any high or critical finding
Draft findings and recommendations	At test completion	The full record for your review and questions, delivered through the platform, with a walkthrough on request
Vulnerabilities report	Compliance and milestone tests	Final findings and recommendations after your feedback, with the test case coverage behind them
Formal regulatory report	New product or significant update	Submission-ready: test cases mapped to the architecture and threat model, findings traced to test cases, MDDT-qualified ratings, VEX dispositions, and the history of every test on the release
Living report	On demand, through the term	The current report for a previously tested release, regenerated against today's CVE set and timestamped

The report is longer than the one you are used to, and the first page still carries the handful that needs fixing. Everything after it is the reason a reviewer accepts that handful as complete.

The curated list was the part of the test the tester chose to show you. FDA now asks for the rest, and the manufacturers who have it on record answer a deficiency question in a day. Everyone else answers it with a new engagement.

FDA, Cybersecurity in Medical Devices: Quality Management System Considerations and Content of Premarket Submissions, final guidance, February 2026, Docket FDA-2021-D-1158, Sections V, V.A.6 and V.C. FDA, Artificial Intelligence-Enabled Device Software Functions: Lifecycle Management and Marketing Submission Recommendations, draft guidance, January 2025, Section XII. FDA, Postmarket Management of Cybersecurity in Medical Devices, final guidance, December 2016, Sections VI and VII. Section 524B of the Federal Food, Drug, and Cosmetic Act, as added by FDORA 2022. QMSR, 21 CFR Part 820 incorporating ISO 13485:2016 clauses 7.3 and 8.5. The Rubric for Applying CVSS to Medical Devices, qualified under FDA's MDDT program (Q171974). ELTON, What the FDA expects from cybersecurity testing (eltoncyber.com/fda-testing-explainer); AI-Enabled Medical Device Cybersecurity Testing and Management for FDA Compliance (eltoncyber.com/whitepaper-ai-enabled-devices); eltoncyber.com/autonomous-testing; Every ELTON activity traced to FDA (eltoncyber.com/fda-deficiency-explainer); Continuous Cybersecurity Testing and Vulnerability Contextualization (eltoncyber.com/assets/whitepapers/ELTON-Whitepaper-continuous-testing.pdf). Test durations, retest times, Not Affected rates and the counts in Figure 5 describe ELTON's delivery experience and are illustrative; they are not FDA figures.

About ELTON. ELTON is the exploitability management platform for medical devices, delivered as a managed program. ELTON builds a digital twin of each device, runs the pipeline across firmware, protocols, and real hardware every release, verifies which findings are exploitable on the device, and evidences every finding and every dismissal for regulatory review. ELTON meets FDA premarket (Section 524B) and postmarket cybersecurity vulnerability testing and management requirements. Talk to us at **eltoncyber.com**.