© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723404

Cyber Resilience of Financial Market Data and Electronic Trading Integrations in Saudi Banks: Securing APIs, FIX Adaptors, Market Feeds, and Transaction Workflows

ZAHED ALI SOHAIL MOHAMMED

Abstract- ***Saudi banks increasingly connect treasury, brokerage, custody, payment, risk and core-banking functions to external markets through application programming interfaces, Financial Information eXchange adaptors, real-time market-data feeds and automated transaction workflows. These interfaces create business agility but also form tightly coupled paths through which data corruption, credential compromise, latency shocks, third-party outages and malicious message injection can propagate from an edge connection into trading decisions, settlement records and liquidity positions. This review examines cyber resilience at the integration layer rather than treating cybersecurity as a perimeter problem. It synthesises literature published from 2020 to 2025 on banking cyber risk, open-banking APIs, microservice security, zero-trust architecture, algorithmic trading, market-data dependence, secure trading mechanisms and operational resilience. The analysis distinguishes four resilience properties: trustworthy identity, message and data integrity, controlled availability, and recoverable transaction state. It argues that Saudi banks require controls that remain effective under partial failure, preserve deterministic evidence of what occurred, and support rapid reconciliation after service restoration. The review develops an integrated architecture in which API gateways, FIX sessions, feed handlers and workflow engines are protected by mutually reinforcing identity, cryptographic, observability, isolation and reconciliation controls. It further proposes lifecycle metrics that connect technical events with operational and financial consequences. The resulting framework contributes a bank-oriented model for securing electronic-trading integrations while preserving performance, auditability and continuity. Research gaps remain in empirical testing of FIX-specific attacks, cross-interface dependency modelling, market-data integrity benchmarks and Saudi-specific recovery exercises across banks, vendors and market infrastructures.***

Keywords: cyber resilience; Saudi banks; electronic trading; FIX protocol; API security; market data; transaction workflows; operational resilience

I. INTRODUCTION

Digital banking resilience is increasingly determined by what happens between systems rather than within a single application. A modern Saudi bank may obtain reference prices from external vendors, subscribe to exchange feeds, route orders through electronic gateways, expose services through APIs, and post resulting transactions into risk, ledger, settlement and reporting platforms. Each connection can be secure in isolation yet still contribute to a fragile end-to-end chain when authentication assumptions, data semantics, recovery procedures or timing dependencies differ. Bank stability research already shows that cyber technology investment and cyber risk are linked to operational performance, while system-level analysis demonstrates that disruption at a highly connected institution can propagate through payment and market infrastructure [1,2]. For that reason, resilience must encompass prevention, containment, continuity, recovery and trustworthy reconstruction of transaction state, not merely the absence of a successful intrusion.

The integration problem is especially important for electronic trading. Banking and financial-sector threat surveys describe malware, credential theft, denial of service, exploitation and insider misuse as persistent attack classes [3]. Banking digitalisation also introduces legacy-system integration, vendor dependence, access-control complexity and continuous-monitoring requirements [4]. In an electronic market, however, the consequence of an integrity or availability failure may unfold in milliseconds: a stale price can alter a valuation, duplicated execution reports can distort positions, an incorrectly sequenced message can break an order state machine, and a compromised service identity can authorise actions that appear syntactically valid.

Automated markets are tightly coupled and complex, so local safeguards can fail to prevent system-wide instability when dependencies interact unexpectedly [5]. Algorithmic trading further increases the importance of rapid and accurate information incorporation, making trustworthy market data and dependable execution channels part of the bank's operational risk boundary [6].

The title of this review deliberately combines market data and transaction integration because these flows are inseparable in practice. The Financial Information eXchange protocol remains a core electronic-trading messaging mechanism for orders, executions and market information, with session behaviour, message sequencing and application semantics that differ from ordinary web services [7]. In parallel, banks increasingly adopt external APIs to improve data portability, partner connectivity and service composition [8]. Formal analysis of open-banking protocols shows that authentication and authorisation properties must be assessed across complete protocol interactions rather than inferred from the use of standard components [9]. Recent API-security research likewise identifies broken authentication, broken authorisation, injection and object-level access weaknesses as recurring risks that require design-time and runtime controls [10]. The resulting environment is therefore heterogeneous: REST-style services, event streams, FIX sessions, vendor feeds, microservices and legacy transaction engines coexist, each with distinct failure modes.

This review aims to develop a technically grounded cyber-resilience model for financial market data and electronic-trading integrations in Saudi banks. Four objectives guide the analysis. First, it identifies how cyber risk enters through APIs, FIX adaptors, market feeds and workflow interfaces. Second, it evaluates controls that preserve confidentiality, integrity, availability and transaction recoverability without ignoring latency and throughput requirements. Third, it explains how identity, observability, reconciliation and third-party governance should operate as cross-cutting capabilities rather than isolated products. Fourth, it translates international evidence into deployment considerations relevant to Saudi banking, where rapid digitalisation and market development place a premium on secure interoperability. The review does not rank products or prescribe a single technology stack; it focuses on control properties and evidence that can be tested across architectures.

II. REVIEW METHODOLOGY

A structured integrative review was used because the research question spans cybersecurity, banking, software architecture and market microstructure. The attached reference paper demonstrates a rigorous review pattern based on systematic searching, categorisation, critical appraisal and synthesis rather than a purely descriptive catalogue. This study adapts that logic to financial-market integrations while avoiding bibliometric claims that cannot be supported by a narrowly bounded corpus. Searches were framed around combinations of bank, financial market, cyber resilience, operational resilience, API security, open banking, microservice security, zero trust, electronic trading, algorithmic trading, FIX, market data, transaction workflow, third-party risk and recovery. The principal academic discovery domains were Scopus and Web of Science, with targeted publisher and DOI checks used to verify bibliographic records.
Eligibility was restricted to English-language sources published from 2020 through 2025 with a resolvable DOI and a direct contribution to at least one analytical theme: banking cyber risk; API or service-to-service security; electronic-trading or market-data dependence; identity and authorisation; secure transaction processing; governance, third-party risk or operational recovery. Studies focused on other sectors were retained only where the mechanism was demonstrably transferable to bank integration architecture. Cryptocurrency evidence was used selectively for cross-market cyber-impact mechanisms rather than as a substitute for banking evidence. Non-peer-reviewed commentary, undated material and sources lacking a valid DOI were excluded from the 30-item reference corpus.

The synthesis proceeded in two stages. First, each source was coded against integration surface, threat or failure mechanism, control objective, resilience phase and evidence type. Second, findings were compared across four technical surfaces: APIs and service meshes; FIX adaptors and order gateways; market-

data ingestion and normalisation; and transaction workflows linking trading, risk, ledger, settlement and reporting. Cross-cutting themes were then derived for identity, cryptography, monitoring, isolation, recoverability and governance. The review therefore treats resilience as an end-to-end systems property. It does not perform statistical meta-analysis because outcome measures, architectures and study designs are too heterogeneous for a defensible pooled effect. Instead, the contribution is a critical architecture synthesis and a set of testable design principles for Saudi banks.

III. SECURING APIS AND SERVICE-TO-SERVICE INTEGRATIONS

Microservice adoption changes the security boundary from a small number of applications to a network of independently deployable components. A multivocal review of microservice security found that authentication and authorisation dominate proposed safeguards, while comparatively little work addresses reaction and recovery after compromise [11]. That imbalance is important for banks because a trading incident is rarely resolved when an attacker is blocked; the institution must determine which requests were accepted, which messages were duplicated or lost, and which downstream books require repair. OAuth research similarly shows that flexible authorisation designs can become insecure through implementation and configuration choices, even when the underlying protocol is widely accepted [12]. A systematic review of microservice authentication and authorisation reinforces the need for consistent identity propagation and policy enforcement across services rather than ad hoc service-level decisions [13].

For trading APIs, resilience begins with explicit trust boundaries. Internet-facing or partner-facing requests should terminate at controlled gateways that authenticate the client, validate scopes and claims, enforce schema and rate controls, and attach an internal identity context that downstream services can verify. Service-to-service calls should not inherit trust merely because they originate inside a bank network. Runtime studies of cloud microservices show that distributed architectures enlarge attack surfaces through dependencies, container configuration, network exposure and insecure communication, while available security tools still leave material coverage gaps [14]. Therefore, banks need layered enforcement: strong client authentication; short-lived, audience-bound tokens; mutual authentication for privileged internal paths; least-privilege service identities; and policy checks close to sensitive business functions.

Open-banking research is useful because it exposes the organisational as well as protocol dimensions of API risk. Comparative analysis of open-banking requirements highlights third-party access, standard alignment and the need to connect security controls with regulatory obligations [15]. A Gulf-region FinTech framework further shows that effective cybersecurity requires adaptation to local institutions, business models and supervisory expectations rather than simple importation of generic frameworks [16]. In a Saudi bank, this means an electronic-trading API should be classified not only by technical endpoint but by the economic authority it conveys: read-only market-data retrieval, order entry, cancellation, allocation, cash movement, reference-data maintenance and administrative actions warrant different authentication strengths and approval patterns.

Zero-trust principles are therefore relevant but must be applied pragmatically. Comprehensive zero-trust research emphasises continuous verification, micro-segmentation, strong authentication, context-aware access and policy automation [17]. Finance-specific work proposes integrating identity, device, network and data controls into an adaptive trust model [18]. For low-latency trading, this should not imply heavy interactive checks on every message. Instead, trust decisions should be pre-established through hardened sessions, short-lived credentials, hardware-backed secrets where appropriate, network segmentation, deterministic policy and continuous telemetry. The design goal is to reduce implicit trust while keeping critical-path latency predictable.

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723404

Table 1. Integration attack surface, control objectives and resilience evidence.

Integration surface	Representative attack / failure modes	Control objectives	Resilience evidence
APIs and service calls	Stolen machine credentials; broken object authorisation; injection; rate exhaustion; dependency failure	Strong service identity; least privilege; schema and policy enforcement; bounded retries	Gateway logs; token claims; policy decisions; request IDs; dependency health
FIX adaptors and sessions	Session hijack; message injection; replay; sequence gaps; resend loops; unintended reset	Authenticated encrypted sessions; semantic validation; sequence controls; limits; independent confirmation	Raw FIX logs; sequence history; session state; drop copy; order and execution reconciliation
Market-data feeds	Stale or manipulated prices; dropped packets; source outage; mapping or reference-data errors	Source authentication; provenance; sequence and timestamp validation; cross-source checks; safe degradation	Raw feed capture; quality flags; source latency; mapping version; divergence alerts
Transaction workflows	Duplicate processing; inconsistent state; queue backlog; unauthorised state change; rollback mismatch	Idempotency; state-machine controls; checkpoint integrity; replay-safe recovery; multilevel reconciliation	Business IDs; event lineage; checkpoints; break reports; position, cash and ledger proofs
Third-party dependencies	Common cloud, carrier or identity failure; vendor compromise; delayed notification	Dependency mapping; concentration analysis; tested failover; contractual incident evidence; exit capability	Supplier recovery tests; architecture attestations; shared-dependency register; notification timestamps

IV. FIX ADAPTORS AND ELECTRONIC-TRADING GATEWAYS

FIX adaptors differ from general APIs because they implement a stateful protocol in which message sequence, session recovery and business semantics are central to correctness. The protocol's role in electronic trading means that a bank may maintain multiple persistent sessions to brokers, venues, liquidity providers or internal execution services [7]. A cyber-resilience model must therefore protect not only transport confidentiality but session identity, message ordering, replay behaviour and application-level state. An attacker who can inject a syntactically valid order, alter an account field, replay a cancel request or manipulate sequence recovery may create outcomes that are difficult to distinguish from operational error. Conversely, an overly aggressive security control that resets sessions or drops traffic can itself generate business disruption.

Controls should be layered around the adaptor rather than concentrated solely in the network. Session endpoints require mutually authenticated encrypted transport, strict certificate lifecycle management, source restrictions and protected configuration. The adaptor should validate message type, mandatory tags, enumerated values, account entitlements, instrument permissions, price and quantity boundaries, expected sender/target identifiers and temporal plausibility before forwarding business instructions. Sequence numbers and resend handling need explicit safeguards against anomalous gaps, repeated recovery loops and unexpected resets. Drop-copy or independent execution-report channels can provide an evidential path that is separate from the primary order route,

improving the bank's ability to detect discrepancies and reconstruct state after an incident.

The literature on automated markets explains why this control depth matters. Algorithmic-trading environments combine high reliability practices with systemic tight coupling; a local technical failure can interact with other participants and feedback mechanisms in ways that amplify disturbance [5]. Evidence from futures markets indicates that algorithmic trading can improve liquidity resilience under normal conditions, yet this benefit presupposes functioning data and execution infrastructure [6]. Secure trading research also demonstrates that confidentiality and integrity can be engineered into market mechanisms without making practical throughput impossible; secure multi-party computation has been implemented for block trading with protection against privileged information misuse [25]. The broader lesson is that security and performance are not inherently contradictory, but controls must be designed with market microstructure and timing requirements in mind.

For Saudi banks, FIX resilience should be operationalised through pre-defined failure states. Each critical session needs an owner, business purpose, dependency map, recovery-time objective, acceptable message-loss tolerance and reconciliation procedure. During an outage, the bank should know whether to queue, reject, reroute or suspend orders; how to prevent duplicate submission when connectivity returns; and how to compare local order state with broker or venue records. Recovery scripts must be tested under partial failure, not only total disconnection. A successful reconnection without position and execution reconciliation is incomplete recovery.

V. MARKET-DATA INTEGRITY, AVAILABILITY AND PROVENANCE

Market data is frequently treated as an input rather than a protected transaction dependency. That distinction is misleading. Pricing, pre-trade risk, collateral, valuation, best-execution logic and algorithmic decisions may all depend on external prices, order-book updates, reference data and trading-status messages. The growing adoption of APIs by banks illustrates how external data access has become embedded in core information flows [8], while research on technology-enabled market-data access shows that APIs can materially change trading behaviour [27]. If an attacker or faulty vendor corrupts a feed, the bank may continue operating normally from a systems perspective while making economically wrong decisions. Integrity controls must therefore detect plausible but incorrect data, not merely malformed packets.

A resilient feed architecture should separate acquisition, normalisation, validation, distribution and consumption. At acquisition, the bank should authenticate the provider and protect transport integrity. During normalisation, source-specific messages should be mapped to controlled internal schemas with version tracking. Validation should combine sequence continuity, timestamp checks, stale-data thresholds, price-band and volume plausibility, cross-source comparison and trading-status logic. Distribution should preserve provenance so that downstream services can identify source, receipt time, normalisation version and quality flags. Critical consumers should define behaviour for degraded data, such as switching to a secondary source, widening controls, entering a safe mode or suspending automated action rather than silently accepting uncertain prices.

Market microstructure research supports the importance of timely information quality. Evidence around changes in algorithmic trading shows that automation can improve the speed with which order-flow information is incorporated into prices [26]. This same speed creates exposure when a corrupted or stale feed reaches automated strategies before a human can intervene. A cyber-resilience design therefore needs machine-speed validation and a clear hierarchy of trust among sources. When two feeds disagree, the correct response is not always to select the numerically closest value; the system should consider venue status, source latency, instrument state, sequence integrity and known incidents. Data-quality alarms should be

coupled to business controls rather than sent only to technology monitoring.

Observability is the bridge between integrity and recoverability. Banks should retain immutable or strongly protected records of raw inbound data around incidents, normalised outputs, quality decisions and consumer actions. This evidence allows post-event reconstruction without assuming that the transformed database is authoritative. The same principle applies to reference-data changes: symbol mappings, tick sizes, calendars and corporate-action adjustments should be versioned because an apparently small mapping error can alter a large set of orders or valuations. Market-data resilience is thus a data-governance problem, a security problem and a trading-control problem at the same time.

VI. TRANSACTION WORKFLOWS, RECONCILIATION AND RECOVERABILITY

Electronic-trading integration becomes most difficult after an order leaves the front-end. A single transaction can pass through pre-trade limits, execution management, broker or venue connectivity, confirmations, allocations, risk aggregation, general ledger, settlement, regulatory reporting and client statements. Cyber resilience depends on preserving the relationship among these states even when components fail at different times. Banking cyber-risk studies increasingly emphasise quantification and business impact because technology severity alone does not reveal the financial significance of an incident [19]. Risk-management evidence also suggests that more mature cyber governance is associated with firm value and more explicit cyber-risk awareness [20]. For transaction workflows, the relevant unit of control is therefore the business process, not the server.

Idempotency and reconciliation are foundational. Any command that may be retried should carry a stable business identifier so that the receiver can distinguish a legitimate retry from a new instruction. Events should include monotonic or causally meaningful sequence information where feasible, and state transitions should be validated against permitted workflow paths. Critical systems should avoid relying on exactly-once delivery claims that are difficult to guarantee across heterogeneous platforms; instead, they should be designed so duplicate delivery is detectable and harmless. Reconciliation should operate at multiple levels: message counts, order and execution state, position and cash movements, ledger postings and external confirmations. Breaks need materiality thresholds, ownership and escalation rather than passive reporting.

Recovery also requires security-sensitive checkpointing. Restoring a database snapshot may reintroduce revoked credentials, obsolete entitlements or transaction states that no longer match external counterparties. A resilience runbook should specify which security configuration is authoritative after rollback, how secrets are reissued, how queued events are handled, and how the bank proves that recovered state is complete. Board-level cybersecurity research shows that governance becomes ineffective when reporting is unclear or directors cannot connect technical findings with risk decisions [21]. Workflow recovery metrics should therefore translate technical evidence into business exposure: number and value of unreconciled trades, duration of stale market data, outstanding settlement breaks, privileged actions during the incident, and time to validated position restoration.

Governance evidence from MENA banks also points to the value of formal cyber oversight and disclosure structures [22]. Within a Saudi bank, the equivalent operational practice is explicit decision rights during an integration incident. Security operations may detect anomalous credentials, trading operations may see rejects, technology teams may observe queue growth, and risk teams may identify position deviations. A single incident command model must combine these signals and define who can disable a gateway, fail over a feed, cancel open orders, invoke a manual process or accept temporary control degradation. Without those

rights, recovery can be delayed by organisational ambiguity even when technology is available.

VII. INTEGRATED CYBER-RESILIENCE ARCHITECTURE FOR SAUDI BANKS

Saudi banking adds a specific deployment context: high digital adoption, expanding financial-market participation, ambitious service modernisation and a regulatory environment that expects strong security and continuity. Saudi-focused mobile-banking research highlights the importance of confidentiality, authentication and device trust in sustaining secure digital use [23]. Although retail mobile channels differ from institutional trading gateways, the underlying lesson is transferable: confidence depends on both technical protection and credible assurance that identities and devices are trustworthy. For market integrations, that assurance extends to service accounts, certificates, vendor endpoints, privileged operators and automated processes.

A bank-wide cyber-resilience architecture should therefore organise controls around shared capabilities rather than duplicate them separately for each interface. Identity services should issue and revoke machine credentials consistently. Key and secret management should support rotation without avoidable outages. Central policy should define network and service segmentation, but enforcement should remain close enough to each workload to prevent lateral movement. Telemetry should preserve high-resolution events from API gateways, FIX engines, feed handlers, workflow brokers, privileged administration and security tools using synchronized time. An integrated banking risk framework is valuable precisely because fragmented controls leave gaps between threat identification, risk assessment and mitigation [24].

Third-party dependence is a particular concern. Trading connectivity may rely on market-data vendors, telecommunications carriers, cloud services, software suppliers, brokers, exchanges and specialised managed services. Contractual availability targets are necessary but insufficient because a vendor can meet aggregate uptime while still failing during a critical market window. Due diligence should therefore test architecture, recovery evidence, subcontractor concentration, security incident notification, cryptographic control, access paths, data retention and exit arrangements. Operational-resilience analysis of financial market infrastructures stresses the need to manage third-party cyber risk as part of the service outcome rather than as a procurement checklist [30]. Banks should identify where several supposedly independent services share the same cloud region, network carrier, identity provider or software component, because common dependencies can invalidate redundancy assumptions.

Figures 1 and 2 translate these principles into an architecture and control loop. The first separates external connectivity from controlled integration zones and downstream transaction systems while showing the cross-cutting services needed to establish identity, integrity, observability and recovery. The second treats resilience as a continuous cycle: establish trust, validate data and messages, detect deviation, contain impact, recover service, reconcile economic state and learn from evidence. This is intentionally more demanding than a conventional incident-response sequence because a financial service is not considered restored until business state is reconciled.

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723404

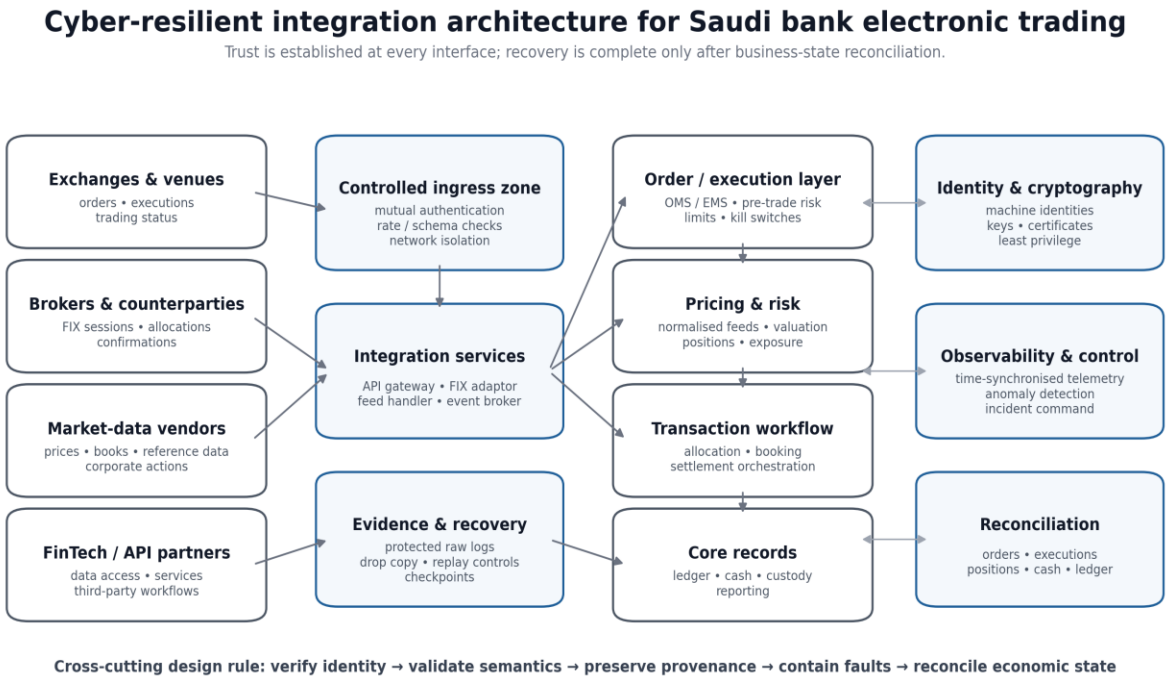

Figure 1. Cyber-resilient integration architecture for Saudi bank electronic-trading ecosystems.

VIII. DISCUSSION: FROM CYBERSECURITY CONTROLS TO MEASURABLE RESILIENCE

A way to test the framework is to trace failure paths from an external connection to a financial consequence. One path begins with a compromised API credential, continues through an apparently authorised order request, and ends in an execution that is valid at the protocol level but outside the customer or desk mandate. Another begins with a manipulated market-data update that passes syntactic checks, changes an algorithmic decision, and creates an unintended position before cross-source validation detects divergence. A third starts with a FIX session disruption, triggers resend activity and duplicate downstream processing, and only becomes visible when broker records disagree with the bank's internal state. These examples show why threat modelling should include technical transitions and business-state transitions in the same diagram. Controls can then be assigned to prevention, detection, containment and reconstruction points, with explicit evidence requirements. This approach also exposes single controls that protect several pathways, such as service identity, message provenance, independent execution confirmation and immutable audit records. Conversely, it reveals controls that appear strong locally but fail when a shared identity provider, common network route or message broker becomes unavailable. Resilience testing should therefore exercise combinations of degraded components rather than isolated device failures.

The reviewed evidence supports five design propositions. First, interface security should be specified in terms of business authority. An API that returns indicative data is not equivalent to an API that can submit an order or move cash, even if both use the same authentication stack. Second, protocol validity does not guarantee business validity. FIX, REST and event-stream messages require semantic controls for account, instrument, quantity, price, sequence, entitlement and workflow state. Third, resilience depends on independent evidence. Drop copies, secondary data sources, protected logs and external confirmations provide alternate truth sources when the primary path is suspected. Fourth, recovery must include reconciliation. Restored compute capacity without reconciled trades, positions and ledgers leaves

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723404

hidden financial exposure. Fifth, dependency concentration should be measured across suppliers and technology layers rather than inferred from the number of contracts.

These propositions also clarify the relationship between cybersecurity and market outcomes. Cyber events can generate measurable negative market reactions for affected firms [28], while cryptocurrency-related incidents have been associated with changes in returns, volatility and trading volume across both digital and traditional assets [29]. Such findings do not imply that every integration incident becomes systemic, but they reinforce the economic value of rapid containment and credible recovery. A bank that cannot establish what data were trusted, which orders were accepted and whether downstream books are correct will face longer decision delays and potentially larger operational losses.

Metrics should therefore be designed around resilience evidence rather than generic control counts. Preventive metrics include percentage of privileged machine identities using strong authentication, credential age, policy exceptions and unencrypted external sessions. Detection metrics include time to identify sequence anomalies, stale feeds, authorisation failures and cross-source price divergence. Containment metrics include time to isolate a compromised adaptor and the number of dependent services affected. Recovery metrics should capture time to restore a controlled service, but reconciliation metrics are more important: time to validated order state, number of unresolved execution breaks, value of unreconciled positions and time to trusted ledger closure. Risk quantification can then map these states to financial exposure rather than treating every security alert as equally material [19].

Performance engineering remains a legitimate constraint. Electronic markets reward speed, and excessive inline inspection can create latency variance or failure modes of its own. The answer is not to remove controls but to separate those that must occur synchronously from those that can be asynchronous. Cryptographic session establishment, entitlement checks, schema validation and hard risk limits belong on the critical path. Deep behavioural analytics, long-horizon anomaly correlation and forensic enrichment can run asynchronously provided that fast kill-switches exist when thresholds are crossed. The architecture should be benchmarked under normal load, burst traffic, partial vendor failure and attack conditions so security capacity is included in performance planning rather than added after deployment.

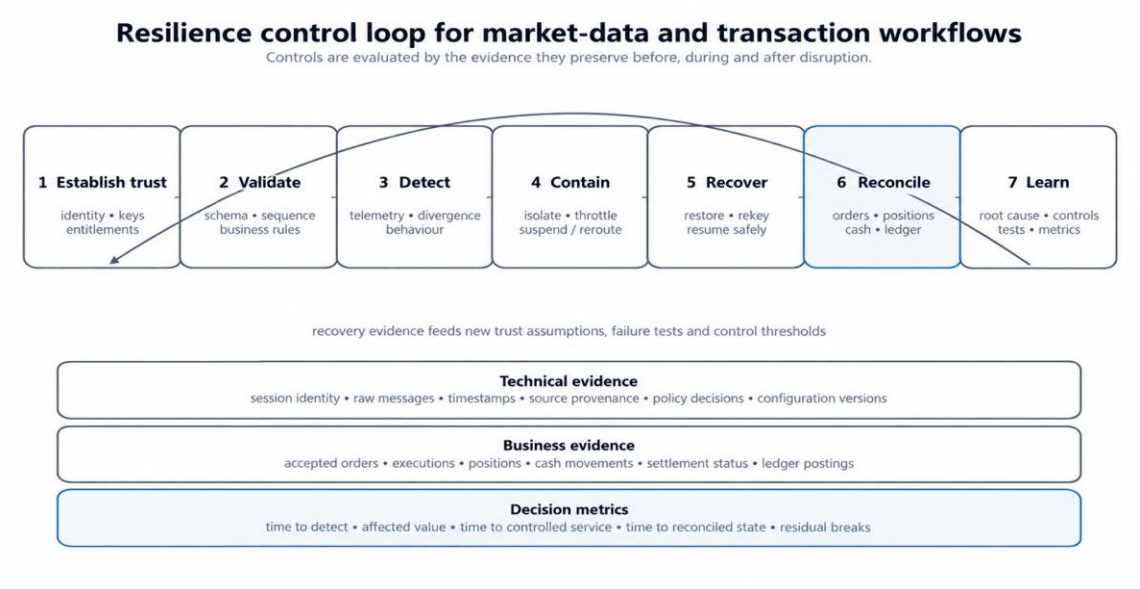

Figure 2. Resilience control loop linking technical restoration to financial-state reconciliation.

Table 2. Review-derived resilience metrics for electronic-trading integrations.

Resilience phase	Primary question	Indicative measures	Decision use
Prevent	Is unauthorised or invalid activity difficult to initiate?	Privileged machine identities with strong authentication; credential age; policy exceptions; unencrypted external sessions	Prioritise control debt and high-authority interfaces
Detect	Can abnormal data, message or identity behaviour be recognised quickly?	Time to sequence anomaly; stale-feed detection; cross-source divergence; authorisation-failure correlation	Trigger investigation before financial exposure expands
Contain	Can the affected path be isolated without uncontrolled propagation?	Time to isolate adaptor; dependent services affected; open orders at suspension; reroute success	Choose kill, throttle, failover or safe-mode actions
Recover	Can controlled service resume without recreating compromise?	Time to trusted service; credential reissue completion; queue recovery; validated configuration state	Measure technical restoration quality
Reconcile	Is the economic state complete and correct?	Unresolved execution breaks; position and cash differences; time to validated ledger; value at risk from breaks	Determine whether recovery is genuinely complete
Learn	Are controls improved using incident evidence?	Root-cause closure; scenario retest; dependency updates; control-threshold changes	Reduce repeat failure and hidden concentration

IX. RESEARCH GAPS AND LIMITATIONS

Several research gaps are evident. FIX-specific cybersecurity literature remains thin relative to the protocol's operational importance, and much available guidance is implementation-oriented rather than empirically evaluated. Controlled experiments are needed on message injection, session hijacking, replay, malformed recovery sequences and coordinated failures across order and drop-copy channels. Market-data research should develop public benchmarks for integrity attacks that preserve realistic latency, sequencing and microstructure behaviour. API-security work should move beyond endpoint vulnerability detection to assess cascading business consequences when authorised but malicious requests traverse multiple services.

Saudi-specific evidence is also limited for institutional electronic-trading resilience. Future work should examine common dependency concentration across banks, brokers, data vendors, cloud platforms and market infrastructures while respecting confidentiality. Sector exercises could test simultaneous degradation of market data, execution connectivity and identity services, with outcomes measured in reconciled business state rather than infrastructure uptime alone. Research should also evaluate whether zero-trust controls introduce latency or operational complexity that changes risk in high-throughput environments. Finally, cyber-risk quantification models need better linkage to market and settlement exposures so that investment decisions can compare the cost of resilience controls with plausible loss pathways.

This review has limitations. Its 30-source corpus was deliberately bounded to DOI-verified literature from 2020–2025 and therefore excludes useful standards, vendor guidance and older foundational work. The evidence base is heterogeneous and includes banking studies, security reviews, market-microstructure research and operational-resilience analysis; conclusions are consequently architectural rather than statistical. Some controls proposed here are synthesis-derived and require validation in production-scale

Saudi banking environments. These limits are consistent with the paper's purpose: to create a defensible integration-layer framework that can guide empirical testing rather than to claim a universal effectiveness estimate.

X. CONCLUSION

Cyber resilience for financial market integrations is an end-to-end property of identities, messages, data, transaction state, dependencies and recovery evidence. Saudi banks cannot secure electronic trading by protecting APIs, FIX adaptors, market feeds and workflows as unrelated technical components. The interfaces jointly determine whether the bank can trust an instruction, detect corrupted information, contain a failure, continue essential activity and reconstruct the economic state afterwards. The reviewed literature supports a control model built on explicit service identity, least privilege, authenticated and validated sessions, provenance-aware market data, deterministic workflow identifiers, independent evidence channels, layered observability and multi-level reconciliation.

The practical implication is a shift from uptime-centred resilience to state-centred resilience. An interface is not recovered simply because packets flow again; it is recovered when the bank can demonstrate that authorised orders, executions, positions, cash movements and ledger entries are complete and consistent. This standard also improves governance because incident decisions can be tied to measurable business exposure. For Saudi banks expanding digital and market connectivity, the architecture proposed here offers a basis for design reviews, control testing, third-party assurance and joint recovery exercises. The next research priority is empirical validation under realistic trading loads and coordinated failure scenarios, especially where common external dependencies link multiple institutions.

REFERENCES

[1] M. H. Uddin, S. Mollah, and M. H. Ali, "Does cyber tech spending matter for bank stability?," *International Review of Financial Analysis*, vol. 72, Art. no. 101587, 2020, doi: 10.1016/j.irfa.2020.101587. ScienceDirect

[2] T. M. Eisenbach, A. Kovner, and M. J. Lee, "Cyber risk and the U.S. financial system: A pre-mortem analysis," *Journal of Financial Economics*, vol. 145, no. 3, pp. 802–826, 2022, doi: 10.1016/j.jfineco.2021.10.007. ScienceDirect

[3] A. A. Darem, A. A. Alhashmi, T. M. Alkhaldi, A. M. Alashjaee, S. M. Alanazi, and S. A. Ebad, "Cyber Threats Classifications and Countermeasures in Banking and Financial Sector," *IEEE Access*, vol. 11, pp. 125138–125158, 2023, doi: 10.1109/ACCESS.2023.3327016. Crossref

[4] S. Wang, M. Asif, M. F. Shahzad, and M. Ashfaq, "Data privacy and cybersecurity challenges in the digital transformation of the banking sector," *Computers & Security*, vol. 147, Art. no. 104051, 2024, doi: 10.1016/j.cose.2024.104051. ScienceDirect

[5] B. H. Min and C. Borch, "Systemic failures and organizational risk management in algorithmic trading: Normal accidents and high reliability in financial markets," *Social Studies of Science*, vol. 52, no. 2, pp. 277–302, 2022, doi: 10.1177/03063127211048515. Crossref

[6] A. Frino, D. Gerace, and M. Behnia, "The impact of algorithmic trading on liquidity in futures markets: New insights into the resiliency of spreads and depth," *Journal of Futures Markets*, vol. 41, no. 8, pp. 1301–1314, 2021, doi: 10.1002/fut.22224. Wiley

[7] S. Kognole, "FIX Protocol: The Backbone of Financial Trading," *International Journal of Computer Science & Information Technology*, vol. 16, no. 4, pp. 97–114, 2024, doi: 10.5121/ijcsit.2024.16408.

[8] Lin, S. S. Zhang, and M. Zachariadis, "Open data and API adoption of U.S. banks," *Journal of Financial Intermediation*, vol. 63, Art. no. 101162, 2025, doi: 10.1016/j.jfi.2025.101162. ScienceDirect

[9] P. Modesti, L. Freitas, Q. Shotomiwa, and A. Almehrej, "Security analysis of the open banking account and transaction API protocol," *Cyber Security and Applications*, vol. 3, Art. no. 100097, 2025, doi: 10.1016/j.csa.2025.100097. ScienceDirect

[10] F. Tanveer, F. Iradat, W. Iqbal, and A. Ahmad, "Towards Secure APIs: A Survey on RESTful API Vulnerability Detection," *Computers, Materials & Continua*, vol. 84, no. 3, pp. 4223–4257, 2025, doi: 10.32604/cmc.2025.067536. Crossref

[11] A. Pereira-Vale, E. B. Fernandez, R. Monge, H. Astudillo, and G. Marquez, "Security in microservice-based systems: A multivocal literature review," *Computers & Security*, vol. 103, Art. no. 102200, 2021, doi: 10.1016/j.cose.2021.102200. ScienceDirect

[12] J. Singh and N. K. Chaudhary, "OAuth 2.0: Architectural design augmentation for mitigation of common security vulnerabilities," *Journal of Information Security and Applications*, vol. 65, Art. no. 103091, 2022, doi: 10.1016/j.jisa.2021.103091. ScienceDirect

[13] M. G. de Almeida and E. D. Canedo, "Authentication and Authorization in Microservices Architecture: A Systematic Literature Review," *Applied Sciences*, vol. 12, no. 6, Art. no. 3023, 2022, doi: 10.3390/app12063023. MDPI

[14] F. Minna and F. Massacci, "SoK: Run-time security for cloud microservices. Are we there yet?," *Computers & Security*, vol. 127, Art. no. 103119, 2023, doi: 10.1016/j.cose.2023.103119. ScienceDirect

[15] M. Gounari, G. Stergiopoulos, K. Pipyros, and D. Gritzalis, "Harmonizing open banking in the European Union: an analysis of PSD2 compliance and interrelation with cybersecurity frameworks and standards," *International Cybersecurity Law Review*, vol. 5, pp. 79–120, 2024, doi: 10.1365/s43439-023-00108-8. Springer

[16] S. AlBenJasim, H. Takruri, R. Al-Zaidi, and T. Dargahi, "Development of cybersecurity framework for FinTech innovations: Bahrain as a case study," *International Cybersecurity Law Review*, vol. 5, pp. 501–532, 2024, doi: 10.1365/s43439-024-00130-4. Springer

[17] N. F. Syed, S. W. Shah, A. Shaghaghi, A. Anwar, Z. Baig, and R. Doss, "Zero Trust Architecture (ZTA): A Comprehensive Survey," *IEEE Access*, vol. 10, pp. 57143–57179, 2022, doi: 10.1109/ACCESS.2022.3174679. IEEE

[18] C. Daah, A. Qureshi, I. Awan, and S. Konur, "Enhancing Zero Trust Models in the Financial Industry through Blockchain Integration: A Proposed Framework," *Electronics*, vol. 13, no. 5, Art. no. 865, 2024, doi: 10.3390/electronics13050865. Crossref

[19] S. Pollmeier, I. Bongiovanni, and S. Slapničar, "Designing a financial quantification model for cyber risk: A case study in a bank," *Safety Science*, vol. 159, Art. no. 106022, 2023, doi: 10.1016/j.ssci.2022.106022. ScienceDirect

[20] N. Gatzert and M. Schubert, "Cyber risk management in the US banking and insurance industry: A textual and empirical analysis of determinants and value," *Journal of Risk and Insurance*, vol. 89, no. 3, pp. 725–763, 2022, doi: 10.1111/jori.12381. Wiley

[21] M. Gale, I. Bongiovanni, and S. Slapničar, "Governing cybersecurity from the boardroom: Challenges, drivers, and ways ahead," *Computers & Security*, vol. 121, Art. no. 102840, 2022, doi: 10.1016/j.cose.2022.102840. ScienceDirect

[22] D. H. Elsayed, T. H. Ismail, and E. A. Ahmed, "The impact of cybersecurity disclosure on banks' performance: the moderating role of corporate governance in the MENA region," *Future Business Journal*, vol. 10, Art. no. 115, 2024, doi: 10.1186/s43093-024-00402-9.

[23] E. Shafie, "Securing mobile banking in Saudi Arabia: key insights on confidentiality, authentication, and device trust," *Journal of Umm Al-Qura University for Engineering and Architecture*, vol. 16, pp. 1401–1416, 2025, doi: 10.1007/s43995-025-00175-4. Springer

[24] Y. T. Y. Azura, M. A. Azad, and Y. Ahmed, "An integrated cyber security risk management framework for online banking systems," *Journal of Banking and Financial Technology*, vol. 9, pp. 85–104, 2025, doi: 10.1007/s42786-025-00056-3. Springer

[25] J. Cartlidge, N. P. Smart, and Y. Talibi Alaoui, "Multi-party computation mechanism for anonymous equity block trading: A secure

implementation of Turquoise Plato Uncross," *Intelligent Systems in Accounting, Finance and Management*, vol. 28, no. 4, pp. 239–267, 2021, doi: 10.1002/isaf.1502. Wiley

[26] D. Yuferova, "Algorithmic trading and market efficiency around the introduction of the NYSE Hybrid Market," *Journal of Financial Markets*, vol. 69, Art. no. 100909, 2024, doi: 10.1016/j.finmar.2024.100909. ScienceDirect

[27] T. Havakhor, M. S. Rahman, T. Zhang, and C. Zhu, "Tech-Enabled Financial Data Access, Retail Investors, and Gambling-Like Behavior in the Stock Market," *Management Science*, vol. 71, no. 2, pp. 1646–1670, 2025, doi: 10.1287/mnsc.2021.01379. INFORMS

[28] E. Akyildirim, T. Conlon, S. Corbet, and Y. G. Hou, "HACKED: Understanding the stock market response to cyberattacks," *Journal of International Financial Markets, Institutions and Money*, vol. 97, Art. no. 102082, 2024, doi: 10.1016/j.intfin.2024.102082. ScienceDirect

[29] H. Cheraghali, P. Molnár, M. Storsveen, and F. Veliqi, "The impact of cryptocurrency-related cyberattacks on return, volatility, and trading volume of cryptocurrencies and traditional financial assets," *International Review of Financial Analysis*, vol. 95, Art. no. 103439, 2024, doi: 10.1016/j.irfa.2024.103439. ScienceDirect

[30] D. Duggan, "The impact of the Digital Operational Resilience Act on financial market infrastructures in Europe," *Journal of Securities Operations & Custody*, vol. 16, no. 4, pp. 344–350, 2024, doi: 10.69554/KHFM3582.