Privacy-Preserving Integrity Evidence for Student-Society Voting-Adjacent Workflows: A Phase C Pilot of Project Simurgh at Macquarie University

Mohammad Raouf Abedini
Department of Computing
Macquarie University
Sydney, Australia
mohammadraouf.abedini@students.mq.edu.au
https://raoufabedini.dev

***Abstract*—In a Phase C pilot of 31 consented sessions (30 submitted, 1 withdrawn) alongside a real student-society voting event, a consent-based shadow-mode system collected session-integrity metadata without recording ballot choices. This boundary was enforced through explicit submit-call JSON construction, client-side ballot-field exclusion safeguards, and server-side forbidden-field rejection, rather than by policy alone. Existing cryptographic schemes primarily address outcome integrity, while the telemetry-collection question studied here is narrower: whether a shadow-mode system can produce evidence that it collected only what it disclosed. Project Simurgh addresses this question with an HMAC-SHA-256 audit chain [3], a server-side forbidden-field guard, and a collection-closure flag that returns `HTTP 410 Gone` on all write routes once data collection ends. The report builder emits `ballot_choice_recorded_by_simurgh: false` as an implementation-enforced invariant; the closeout gate suite verified this invariant after the 30 submitted sessions: 359/359 automated tests, 8/8 smoke gates, 10/10 security-audit gates, and 5/5 closure gates passed at closeout. This study is voting-adjacent: it does not implement ballot cryptography, eligibility verification, coercion resistance, tally protection, or public-election certification; the ballot-choice exclusion boundary is enforced by structural design, not participant behavior.**

***Index Terms*—data privacy, message authentication, audit trails, electronic voting, consent management, student governance**

I. Introduction

Digital voting tools support event preference polls, committee elections, and budget decisions in student societies, including at Macquarie University where this pilot was conducted. These tools leave a gap that existing cryptographic solutions do not address: session integrity in the presence of a potential covert collector: whether the system collected only what it disclosed, and no more.

End-to-end verifiable voting schemes [4], [5] solve the *outcome* integrity problem (did the tally reflect all ballots?) but do not address *session* integrity (was the participant's interaction free of covert collection?). Project Simurgh is a research prototype that targets session integrity through privacy-preserving metadata collection and tamper-evident audit chains [1]. It was developed in response to the Invisible Window attack [6], which demonstrated that OS-level display affinity APIs allow application windows to evade browser-based screen capture entirely, undermining the display-fidelity assumption on which WebRTC proctoring relies. This paper is a companion to the main Project Simurgh architecture paper [1]; it reports the Phase C voting-adjacent pilot as a bounded, application-specific case study of that system.

This paper reports a Phase C pilot in which Project Simurgh ran in shadow mode alongside a real MQ Persian Society event preference poll, collecting session-level integrity metadata in parallel with no effect on the official result. We use *voting-adjacent* to mean the system ran entirely separate from the official ballot-counting infrastructure, with no access to ballot choices or influence on official outcomes. The pilot was not a study of election security, voting-system integrity, or electoral fraud prevention.

Contributions:

- A consent-to-submit pilot with 30 submitted sessions verifying `ballot_choice_recorded_by_simurgh: false` as an implementation-enforced invariant in every session, backed by smoke, security-audit, and privacy-audit gate suites (Sec. V).
- A verifiable end-of-collection posture in which the collection-closed middleware executes before authentication on all write routes, preventing token-bearing clients from bypassing closure and verifiable at the API layer by any caller able to reach the pilot endpoint (Sec. III).
- A gate-verified evidence methodology pairing implementation-enforced structural invariants with independently runnable smoke, security-audit, and privacy-audit suites, archived at tag `v0.5.0-voting-pilot-phase-c-closeout`

Voting-adjacent case study, not a production election-security claim. Research prototype only. Source: https://github.com/Raoof128/Project-Simurgh [1]. Tag: `v0.5.0-voting-pilot-phase-c-closeout`. DOI: 10.5281/zenodo.20549736 [2].

for reproducing the implementation-enforced gate checks and extending the methodology to future privacy-preserving integrity evidence work in voting-adjacent contexts.

II. Related Work

A. End-to-End Verifiable Voting

Systems such as Helios [4] use homomorphic tallying or mix-net shuffles so voters can verify their encrypted ballot was counted without revealing the plaintext choice. Civitas [5] adds coercion resistance; STAR-Vote [7] adds risk-limiting-audit integration.

Project Simurgh differs in scope. It does not implement ballot encryption, voter-verifiable tallying, or public election certification. Where end-to-end verifiable voting proves the integrity of the *outcome*, Project Simurgh studies the integrity of the *participation session*: whether the system collected only what it disclosed and nothing more.

B. Remote and Internet Voting Security

Remote electronic voting introduces risks beyond ordinary web application security: endpoint compromise, coercion, eligibility verification, ballot secrecy, and tally integrity. NIST guidance [8] identifies threat categories for remote electronic voting and recommends caution around internet-connected election infrastructure. The National Academies report [9] emphasizes that vote collection, alteration, destruction, counting, and reporting each require separate mitigations.

This pilot addresses none of those concerns. It is a student-society research case study, not a production online-voting system.

C. Voting Standards, Privacy, and Auditability

The U.S. Election Assistance Commission adopted VVSG 2.0 in 2021 [10], establishing updated requirements for privacy, accessibility, auditability, and software independence in voting systems. Modern voting-system guidance consistently requires audit records that do not reveal individual ballot choices.

Project Simurgh aligns with this direction: the HMAC audit chain records session lifecycle events without ballot content, but remains outside public-election certification scope.

D. Australian Technology-Assisted Voting

Several Australian jurisdictions have piloted technology-assisted voting, including iVote in New South Wales. The NSW Electoral Commission's technology-assisted voting review [11] examined privacy, accessibility, and security trade-offs, recommending careful scoping and ongoing evaluation before broader deployment. The NSW Electoral Commission's evidence-driven approach makes small, bounded pilots the appropriate vehicle for evaluating session-integrity mechanisms at the consent-privacy boundary.

E. Privacy-Preserving Telemetry and Data Minimization

Project Simurgh's ballot-field exclusion architecture follows data-minimization principles [12]: the system collects the minimum metadata required for a session-integrity signal and structurally prevents ballot-choice data from reaching the server. Data that is structurally excluded from the server cannot be leaked by the server; residual metadata (e.g., submission timestamps) are minimized and out of scope for ballot-choice inference. The contribution is not a new voting protocol; it is evidence that shadow-mode session-integrity collection is achievable at student-governance scale without access to ballot choices or influence on official outcomes.

III. System Design

Fig. 1 shows the Phase C session lifecycle.

A. Consent Flow

The consent page disclosed two categories: data Project Simurgh may collect across its deployments (session timestamps, focus-loss and paste counts, audit-chain validity signal, and an anonymous HMAC-hashed session code) and data it never collects (ballot choices, screen recordings, webcam/audio, typed or pasted content, raw process names, window titles, or device serial identifiers). In Phase C, implemented collection comprised only session timestamps, audit-chain lifecycle events, and an anonymous participant code; focus-loss and paste counts are exam-integrity features of the broader Project Simurgh system and were not active in the voting pilot. This conservative disclosure posture means participants consented to a superset of what was implemented.

On consent, the server issued an anonymous participant code and a signed session token. No direct identifiers such as names, student identifiers, email addresses, or ballot choices were collected.

B. Ballot-Field Exclusion

The submit handler constructed the JSON request body explicitly, containing only `pilot_session_id` and `submit_intent: true`; no form fields were serialized into the body. As a future-code-change safeguard, the handler also zeroed all radio button value properties (`el.value = ""`) before the fetch call; this measure ensures ballot data cannot appear through a form-serialization path if the construction were ever changed, though the current implementation never serializes form fields. (The consent `fetch` occurs before any ballot interaction and carries no ballot field.) The server provided a second layer: the `FORBIDDEN_BALLOT_FIELDS` set rejected any submission body containing a known ballot-choice key with `HTTP 400`, with no echo of the supplied value.

C. HMAC Audit Chain

Each session maintained an append-only HMAC-SHA-256 audit chain [3] recording lifecycle events in sequence: `CONSENT_ACCEPTED` and `STARTED` (both appended at the consent/accept call), then `SUBMITTED` at submit, and

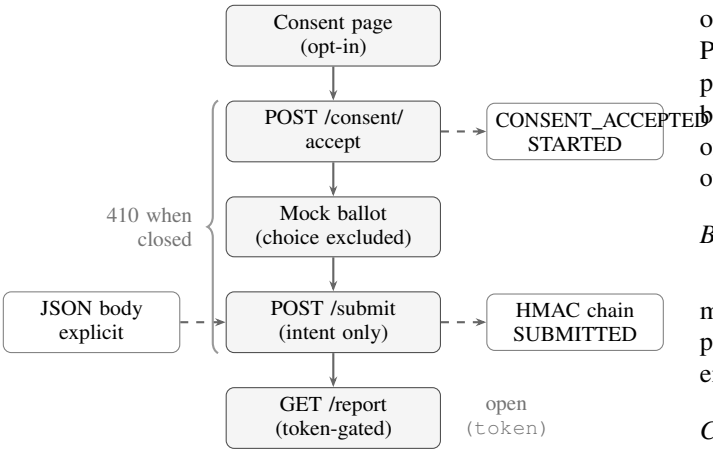

Fig. 1. Phase C session lifecycle. The submit-call JSON body is constructed explicitly without form serialization, so ballot data is structurally absent (left). Two HMAC chain events append at consent (`CONSENT_ACCEPTED`, `STARTED`) and one at submit (`SUBMITTED`) (right). Under the collection-closed flag, write routes return `HTTP 410 Gone`; the report endpoint remains token-protected.

optionally `BALLOT_FIELD_REJECTED`, `WITHDRAWN`, or `REPORT_EXPORTED`. The chain initialized on consent; the server verified its integrity on report export.

D. Collection-Closure Posture

After reaching the 30-session target, we set the environment variable `SIMURGH_VOTING_PILOT_COLLECTION_CLOSED` to `true`. With this flag active:

- `POST /api/voting-pilot/consent/accept` returns `410 Gone`.
- `POST /api/voting-pilot/submit` returns `410 Gone`.
- `POST /api/voting-pilot/withdraw` returns `410 Gone`.
- `GET /api/voting-pilot/:id/report` remains available, token-protected.

The closure middleware executes before authentication on write routes; a valid session token cannot bypass it.

IV. Phase C Pilot

A. Study Design

The pilot ran in shadow mode alongside the MQ Persian Society 2026 event preference poll, asking members to select one of four proposed social events (Nowruz cultural night, Persian movie night, career/networking night, food and music night). Project Simurgh ran in parallel without access to or influence on the official ballot-counting system. Each participant who joined the pilot completed two steps: a consent page explaining data collection and a mock ballot page mirroring the official event poll. The mock ballot choice was excluded from the submit-call JSON body; Project Simurgh received only a submit-intent signal.

Participants were members of the society and were recruited through society communication channels; participation was opt-in until the 30 submitted-session target was reached. Participation was voluntary and governed by the Phase C participant notice and data management addendum approved by the society executive on 2026-06-04. No incentive was offered. The society's official outcome was determined entirely outside Project Simurgh. Session counts are reported in Table I.

B. Withdrawn Session

One participant (`vp_[redacted]`[1]) withdrew before submitting. Withdrawn sessions are excluded from report export, privacy assertions, and primary analysis; that session's report endpoint returns `HTTP 403`.

C. Governance and Ethics

Data collection was scoped to avoid directly identifying personal information under the Privacy Act 1988 (Cth) [13]: no names, student identifiers, email addresses, ballot choices, demographic attributes, or content-level data were collected. Because anonymous timestamps and session metadata may still carry contextual privacy risk in a small cohort, this paper reports only aggregate technical outcomes. No individual-level analysis, disciplinary action, re-identification, or subgroup analysis was performed. Participation was fully voluntary with no consequence for declining or withdrawing. This paper does not claim formal institutional human-research ethics approval; it reports a society-approved internal pilot with aggregate technical outcomes only.

D. Data Management

The server held live session data in memory only. The application persisted no live pilot session records; aggregate gate evidence and closeout artifacts were separately archived in the repository. The store reset on server restart, consistent with the data management addendum, which specified no persistent storage of individual pilot session records.

V. Results

A. Dataset

Phase C recorded 31 consented pilot sessions, of which 30 completed the submit step and formed the primary analysis dataset; one session was withdrawn before submission. Table I summarizes session counts.

The Phase C human sessions are reported as aggregate closeout outcomes. Individual live session records were not persisted; retained evidence consists of closeout logs, gate outputs, source-level invariants, and the archived implementation baseline at tag `v0.5.0-voting-pilot-phase-c-closeout`.

B. Privacy Assertions

Table II summarizes the privacy assertions across all 30 submitted sessions.

The `ballot_choice_recorded_by_simurgh` field is a structural guarantee: the value is hardcoded as `false`

[1]Ephemeral pseudonymous token; retained in the Phase C closeout log.

TABLE I
SESSION COUNTS—PHASE C PILOT

Metric	Count
Consented pilot sessions	31
Submitted sessions (primary analysis)	30
Withdrawn sessions	1

TABLE II
PRIVACY ASSERTIONS ACROSS ALL 30 SUBMITTED SESSIONS. (BALLOT_CHOICE_RECORDED ABBREVIATES BALLOT_CHOICE_RECORDED_BY_SIMURGH.)

Assertion	Result
`ballot_choice_recorded`	`false`
`official_vote_impact`	`false`
Screen capture	No
Webcam / audio	No
Typed content	No
Pasted content	No
Personal device identifiers	No
Raw process names	No
Raw window titles	No
Privacy audit (`privacy-audit.mjs`)	PASS

TABLE III
SAFETY GATES—CLOSEOUT BASELINE

Gate	Result
Node tests (`npm test`)	359/359 pass
npm audit (`--audit-level=high`)	0 high/critical
Privacy audit	PASS (code + 52 evidence files)
`smoke-voting-pilot.sh`	8/8 pass
`security-audit-voting-pilot.sh`	10/10 pass
`smoke-voting-pilot-closed.sh`	5/5 pass

in the session report builder and verified by the automated smoke suite at every gate run. Because the session store was in-memory and cleared after collection, individual session reports are not available post-hoc; the guarantee holds by implementation-enforced design, not by retrospective per-session inspection. The privacy audit (`privacy-audit.mjs`) performs static string-pattern searches across source files and 52 pre-existing evidence export files to confirm the absence of ballot-choice patterns in persisted data; Phase C sessions were never persisted and are therefore not included in that file count.

C. Integrity Flow

All 30 submitted sessions completed the consent-to-submit flow. The HMAC audit chain recorded `CONSENT_ACCEPTED` and `STARTED` on consent and `SUBMITTED` on submit for each. The withdrawn session's chain recorded `WITHDRAWN`; its report endpoint returned `HTTP 403`, confirming exclusion from analysis.

D. Safety Gates

Table III shows all closeout gate results.

Gate evidence files are archived in the repository under the Phase C closeout evidence directory at tag `v0.5.0-voting-pilot-phase-c-closeout`.

E. Collection Closure

After collection reached 30 sessions, all write endpoints returned `HTTP 410 Gone`. The closure smoke suite (5/5 gates) confirmed that consent/accept, submit, and withdraw are unreachable under the closed configuration, while report export remains available with a valid token, verifiable at the API layer by any caller able to reach the pilot endpoint, independently of the client UI state.

VI. DISCUSSION

A. Interpretation

The central finding is that structural enforcement, building privacy into the system's data flow rather than relying on participant compliance, produces a verifiable guarantee even without post-hoc per-session audit records. The ballot-choice exclusion holds at the server layer regardless of client behavior because the JSON body is constructed explicitly; browser-layer error or tampering cannot introduce ballot data into an accepted submission, as attempts to include ballot-choice fields are rejected by the server. This makes the guarantee independent of participant trust, which matters for adoption: a system that requires participants to trust both a policy and their own correct behavior is weaker than one that enforces the boundary structurally.

The server-side collection-closure mechanism extends this approach: closure is enforced at the infrastructure layer, specifically pre-authentication middleware, and verifiable at the API layer by any caller able to reach the pilot endpoint. Together, these design choices demonstrate a pattern applicable to any shadow-mode telemetry deployment where the operator must demonstrate what was and was not collected: both collection boundaries and collection closure can be made API-observable and gate-verifiable.

B. Limitations

- The 30-session dataset is a proof-of-concept pilot. No statistical power calculation was performed; results should not be generalized beyond this scope.
- Sessions were anonymous; no demographic or behavioral analysis of participants is possible or intended.
- The pilot ran in shadow mode only. No claim extends to a non-shadow deployment.
- The in-memory store means session data did not survive server restarts. This suits a research prototype but is not a production data-management posture.
- `ballot_choice_recorded_by_simurgh: false` is an implementation-enforced structural guarantee verified by the automated gate suite; per-session audit reports are not available post-hoc due to in-memory storage.
- The HMAC audit chain is keyed with a server-side secret; chain validity is operator-verifiable but not independently

verifiable by participants without the signing key. This is consistent with the prototype scope but means the chain does not provide third-party attestation.
- The pilot was conducted in one student society at one university; results may not transfer to larger or more formal governance contexts.
- The pilot did not measure usability, accessibility, completion time, comprehension, or participant trust.
- The pilot did not include a formal adversarial participant study; malicious ballot-field attempts were tested through scripted security gates.

C. Non-Claims

This pilot does not claim to:

- Secure or audit the official MQ Persian Society election result.
- Detect or prevent election fraud.
- Provide hardware-rooted attestation.
- Prevent or detect screen-capture evasion techniques (including display-affinity or GPU-layer approaches) on participant devices.
- Replace or improve the society's own voting system.
- Generalize to national, governmental, or binding elections.

VII. Conclusion

Thirty sessions completed the full consent-to-submit flow in Phase C of Project Simurgh. In every session, the system collected no ballot choice, screen recording, webcam or audio data, typed content, pasted content, or personal device identifier. After data collection ended, all write endpoints returned `HTTP 410 Gone` under a server-side flag, confirmed at the API layer independently of the client UI by the 5/5 closure smoke suite.

The result demonstrates that privacy-preserving session-integrity evidence is structurally achievable alongside a real student-society voting event. The primary constraint for generalization is the operator-held HMAC signing key: participant-verifiable integrity rather than operator-verifiable integrity would require either a disclosed symmetric key or a public-key chain. That extension, alongside a non-shadow deployment with a larger cohort, is the natural next step for this research program.

References

[1] M. R. Abedini, "Project simurgh: Privacy-preserving device integrity proofs for capture-resistant high-stakes sessions," Zenodo preprint, 2026, https://doi.org/10.5281/zenodo.20374849. Source: https://github.com/Raoof128/Project-Simurgh.

[2] ——, "Privacy-preserving integrity evidence for student-society voting-adjacent workflows: A phase C pilot of Project Simurgh at Macquarie University," Zenodo preprint, 2026, https://doi.org/10.5281/zenodo.20549736. Tag: v0.5.0-voting-pilot-phase-c-closeout.

[3] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-hashing for message authentication," Internet Engineering Task Force, RFC 2104, Feb. 1997, https://www.rfc-editor.org/rfc/rfc2104.

[4] B. Adida, "Helios: Web-based open-audit voting," in *Proceedings of the 17th USENIX Security Symposium*. USENIX Association, 2008, pp. 335–348.

[5] M. R. Clarkson, S. Chong, and A. C. Myers, "Civitas: Toward a secure voting system," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy*. IEEE, 2008, pp. 354–368.

[6] M. R. Abedini, "The invisible window: Exploiting OS-level display affinity to bypass WebRTC proctoring systems," Preprint, 2026, https://zenodo.org/records/20376495.

[7] S. Bell, J. Benaloh, M. Byrne, D. DeBeauvoir, B. Eakin, G. Fisher, P. Kortum, N. McBurnett, J. Montoya, M. Parker, O. Pereira, P. Stark, D. Wallach, and M. Winn, "STAR-Vote: A secure, transparent, auditable, and reliable voting system," in *Proceedings of the 2013 EVT/WOTE Workshop on Electronic Voting Technology*. USENIX Association, 2013.

[8] N. Hastings, R. Peralta, S. Popoveniuc, and A. Regenscheid, "Security considerations for remote electronic UOCAVA voting," National Institute of Standards and Technology, Gaithersburg, MD, NIST Interagency Report 7770, Feb. 2011.

[9] National Academies of Sciences, Engineering, and Medicine, *Securing the Vote: Protecting American Democracy*. Washington, DC: The National Academies Press, 2018.

[10] U.S. Election Assistance Commission, "Voluntary voting system guidelines (VVSG) 2.0," https://www.eac.gov/voting-equipment/voluntary-voting-system-guidelines, 2021, adopted February 2021.

[11] NSW Electoral Commission, "Technology assisted voting: Final review report," NSW Electoral Commission, Tech. Rep., Nov. 2023, https://elections.nsw.gov.au/technology-assisted-voting-review/review-papers (archived: https://web.archive.org/web/2024/https://elections.nsw.gov.au/technology-assisted-voting-review/review-papers).

[12] A. Cavoukian, "Privacy by design: The 7 foundational principles," Information and Privacy Commissioner of Ontario, Tech. Rep., 2009, https://www.ipc.on.ca/wp-content/uploads/resources/7foundationalprinciples.pdf.

[13] Australian Government, "Privacy act 1988 (cth)," https://www.legislation.gov.au/Series/C2004A03712, 1988, compilation current at time of access; Australian Privacy Principles (Schedule 1) govern collection, use, and disclosure of personal information.