# BLADE-SWARM Governance Simulator

## Simulation User Guide

Version: v1.0
Document: SWARM-SIM-UG-001
Date: May 2026
Revision: 1.0
Author: Burak Oktenli
Institution: Georgetown University, School of Continuing Studies
Program: M.P.S. Applied Intelligence (STEM)
ORCID: 0009-0001-8573-1667
Related Artifact: DOI: 10.5281/zenodo.20351198

## Document Control

| | |
|---|---|
| Document ID | SWARM-SIM-UG-001 |
| Simulation Version | v1.0 |
| Author | Burak Oktenli — Georgetown University — ORCID 0009-0001-8573-1667 |
| Date / Revision | May 2026 / Rev 1.0 — Initial Release |
| Classification | UNCONTROLLED — Research Artifact |
| Related Artifact | DOI: 10.5281/zenodo.20351198 |

## Table of Contents



 

# 1. Purpose, Scope, and Assumptions

This guide provides operating procedures for the BLADE-SWARM (Attritable Swarm Authority Governance) simulation. It implements the seven-stage AUTHREX governance pipeline at swarm scale (SENSE -> SATA -> ADARA+IFF -> HMAA -> MAIVA -> FLAME -> ERAM+CARA), where each agent runs a Byzantine-fault-tolerant two-phase consensus gated by peer trust (SATA), four-tier authority (HMAA), and weighted voting (MAIVA) before the swarm commits to a coordinated action. The simulation is parameterised over N = 10 (physical testbed), N = 50 (combined operation), and N = 500 (DAWG-class). It governs decision authority and audit; it does not govern weapons.

**Intended Audience:** EB2-NIW petition evaluators, defense/aerospace and AI-security reviewers, academic peers, and technical collaborators seeking independent verification of behavior.

**Scope:** Operation of the BLADE-SWARM Governance Simulator. Does not cover mathematical theory (see the published paper) or hardware specifications (see the Zenodo deposit and the Interface Control Document, ICD-SWARM-001).

## 1.1 Assumptions and Constraints

- The user has a modern browser with JavaScript enabled and HTTPS access.
- All parameter values are synthetic research placeholders, not calibrated against physical hardware.
- The simulation models distributed consensus and governance logic only. Flight dynamics are simplified.
- Cryptographic operations (ECDSA P-256, SHA-256 audit chain) use the WebCrypto API, not a hardware secure element.
- Results are valid for demonstrating architectural behavior, not for operational safety assessment.

> **IMPORTANT**
>
> This simulation is a research prototype. Not for operational planning, safety-critical decisions, or system certification. BLADE-SWARM governs decision authority and audit only; it is not a swarm autopilot and contains no weapon. All parameters are synthetic.

# 2. Quick Start

**Step 1.** Open the simulation HTML file in Chrome over HTTPS.

**Step 2.** Select a swarm scale (N = 10, 50, or 500) and a scripted scenario.

**Step 3.** Review the per-node tier array, the consensus panel, and the distributed audit ledger.

**Step 4.** Click RUN/START to begin the consensus loop and observe the seven-stage pipeline.

**Step 5.** Use the export/download buttons to save session data as JSON for verification.

> **NOTE**
>
> All computation runs client-side with the WebCrypto API. No data leaves your browser. Requires HTTPS (not file://).

# 3. System Requirements and Security Considerations

| Requirement | Specification |
|---|---|
| Browser | Chrome 90+, Firefox 88+, Safari 15+, Edge 90+ |
| Protocol | HTTPS required (WebCrypto API for ECDSA / SHA-256 / HKDF) |
| Display | Min 1280x720; recommended 1920x1080+ |
| CPU/Memory | Any modern processor; ~200MB RAM peak for batch runs |
| Network | Internet for initial CDN load. All computation client-side after load. |
| Installation | None — zero install, no login, no backend, no database, no cookies |

## 3.1 Security Considerations

- No data exfiltration: all computation runs in the browser. No data is sent to any server.
- CDN dependencies load from cdnjs.cloudflare.com with Subresource Integrity hashes where available.
- Audit integrity: ECDSA P-256 signatures and a SHA-256 hash chain via WebCrypto. The VERIFY button recomputes the chain.
- No authentication: the simulation has no login system. All state is ephemeral in browser memory.

# 4. Interface Layout and Navigation

## 4.1 Panel Layout

The interface presents the swarm-scale selector (N = 10/50/500), the per-node tier array (each node T3/T2/T1/T0), the consensus panel (proposing node, sub-quorum membership, votes, and commit/abort outcome), the seven-stage pipeline status (SENSE through CARA, each PASS/FAIL/SKIP), the Byzantine-bound readout (f = (N-1)/3 per quorum), and the growing hash-chained distributed audit ledger. Controls include START/STOP, scale and scenario selection, fault injection (Byzantine, Sybil, RF denial), and a Monte Carlo batch.

> **TIP**
>
> Hover over interface elements for tooltips. Gauges include ARIA labels for screen-reader accessibility.

# 5. Operating Procedures

## 5.1 Startup

1. Navigate to the simulation URL or click Launch Simulation from burakoktenli.com.
2. Wait for loading (2-5 seconds). CDN scripts load from cdnjs.cloudflare.com.
3. Verify the interface loads completely. All panels should be visible.

## 5.2 Standard Operation

1. Open the simulation via Launch Simulator or navigate to blade-swarm-simulation.html.
2. Select a swarm scale and scenario, then click START. Agents form sub-quorums and propose actions.
3. Monitor the seven-stage pipeline (SENSE -> SATA -> ADARA+IFF -> HMAA -> MAIVA -> FLAME -> ERAM+CARA).
4. SATA scores each peer; ADARA and IFF flag Sybil and spoofed-identity peers and exclude them from quorum.
5. HMAA evaluates the authority tier; escalation requires an intersecting quorum, downgrade is unilateral.
6. MAIVA performs weighted sub-quorum voting with the quorum-intersection bound; FLAME runs the two-phase commit.
7. Inject faults (single Byzantine agent, Sybil probe, RF denial) and observe safe-halt-by-default behavior.
8. Run the Monte Carlo batch for statistical outcomes across scenarios and scales.
9. CARA isolates a misbehaving agent and writes a signed corrective entry to the distributed ledger.

## 5.3 Shutdown

1. Export session data. 2. Close the browser tab (all state discarded).

> **IMPORTANT**
>
> State is not persisted. Export before closing to preserve results.

# 6. Parameter Reference

Parameters are pre-configured per scenario and scale (quorum size, Byzantine bound f = (N-1)/3, SATA trust thresholds, FLAME window durations, MAIVA vote weights). See Section 4 for interface controls and the published paper for parameter derivation. The Byzantine bound and quorum-intersection size scale with N.

# 7. Scenario Reference

Five scripted scenarios across three scales. Select a scale and scenario from the control panel before starting.

| Scenario | Description |
|---|---|
| S1 Nominal | Healthy swarm, all agents T3, clean RF; consensus reaches an intersecting quorum and commits. |
| S2 Single Byzantine agent | One compromised node attempts an unauthorized commit; quorum intersection rejects it. |
| S3 Sybil probe | Spoofed-identity peers attempt to join; SATA and attested identity exclude them from quorum. |
| S4 Contested RF | Degraded mesh; FLAME contracts the deliberation window and the swarm safe-halts by default. |
| S5 Denied / degraded | Link loss beyond the Byzantine bound; tier downgrade to T0 with a signed audit entry. |

# 8. Metrics, Formulas, and Verification

## 8.1 Key Metrics

| Metric | Definition |
|---|---|
| Pipeline Status | Seven-stage status: SENSE, SATA, ADARA+IFF, HMAA, MAIVA, FLAME, ERAM+CARA. |
| Per-Node Tier | HMAA-computed authority tier per agent (T3/T2/T1/T0). |
| Consensus Outcome | Commit or abort, with the proposing node, sub-quorum membership, and vote tally. |
| Byzantine Bound | f = (N-1)/3 tolerated compromised agents per quorum; quorum-intersection size. |
| Audit Ledger | Hash-chained distributed ledger depth and per-entry ECDSA signature status. |

## 8.2 Verification Checklist

| Action | Expected Result |
|---|---|
| Start simulation (RUN/START) | Interface loads; agents form sub-quorums and the pipeline begins. |
| Observe nominal state (S1) | All agents T3; consensus commits with an intersecting quorum. |
| Inject one Byzantine agent (S2) | The unauthorized commit is rejected; safety invariant S2 holds. |
| Inject a Sybil probe (S3) | Spoofed peers are excluded by SATA and attested identity. |
| Deny RF (S4/S5) | FLAME contracts and the swarm safe-halts; tier downgrades to T0 with a signed entry. |
| Reload with same seed | Same seed + params + scale yield identical outputs and a valid audit chain. |

# 9. Data Export and Reproducibility

Click export/download to save a session JSON with parameters, scale, consensus history, and the ECDSA-signed distributed audit ledger. To verify: 1) export JSON; 2) note the PRNG seed and scale; 3) reload with the same seed, parameters, and scale; 4) confirm a bit-exact match and a valid audit chain.

## 9.1 Reproducibility Guarantee

| Property | Value |
|---|---|
| PRNG | Deterministic seeded generator (32-bit) |
| Math.random() | Zero calls in computation paths |
| Cross-Browser | Verified: Chrome, Firefox, Safari, Edge |
| Audit Chain | ECDSA P-256 + SHA-256 via WebCrypto (SubtleCrypto API) |
| Scales | N = 10 / 50 / 500 parameterised in a single simulator |

# 10. Limitations and Threat Considerations

| Limitation | Description |
|---|---|
| Simulation-Only Evidence | Browser-based computation. No physical flight data or hardware measurements. |
| Uncalibrated Parameters | All values are synthetic research parameters, not empirically derived. |
| No Real-Time Guarantees | The JavaScript engine provides no timing guarantees for safety-critical operations. |
| Simulated Cryptography | ECDSA/SHA-256 use WebCrypto. Secure-element operations are modeled, not hardware-backed. |
| Reduced-Scale Verification | The TLA+ properties are model-checked on a reduced-scale instance, not proven for arbitrary N. |

## 10.1 Threat Considerations

• CDN compromise: dependencies load from cdnjs.cloudflare.com. Mitigation: Subresource Integrity hashes where available.

• Browser extensions could modify the simulation DOM/state. Mitigation: test in Incognito mode for clean results.

• Local modification via DevTools is possible. Exported data should be verified against the published source on burakoktenli.com.

**ASSURANCE BOUNDARY**

This is a research demonstrator (TRL 3-4 simulator and formal spec; TRL 2 physical testbed design). No FIPS, Common Criteria, EAL, NSA, NASA, or DoD endorsement, validation, or certification of any kind is claimed. BLADE-SWARM governs decision authority and audit and emits only signed authority decisions.

# 11. Troubleshooting

| Problem | Likely Cause | Solution |
|---|---|---|
| Black screen after loading | Render error or CSP violation | Open F12 Console for details. Try Chrome Incognito. |
| Simulation runs slowly at N=500 | CPU-intensive large-scale batch | Reduce the scale to N=50 or N=10. Close other tabs. |
| Controls not responding | Tab lost focus | Click inside the simulation window. Ensure the tab is active. |
| Export not working | Download blocked | Allow downloads from the simulation domain in browser settings. |
| Loading never completes | CDN scripts blocked | Disable ad blockers. Allow cdnjs.cloudflare.com. |

# 12. Glossary and References

## 12.1 Glossary

| Term | Definition |
|---|---|
| BLADE-SWARM | Beam-Layer Authority for Directed Engagements, Swarm Node |
| SATA | Sensor/Peer Attestation and Trust Anchoring — per-peer trust scoring |
| HMAA | Human-Machine Authority Architecture — four-tier authority arbitration |
| MAIVA | Multi-Agent Integrity and Verification Architecture — weighted sub-quorum voting |
| Byzantine bound (f) | Maximum compromised agents per quorum tolerated, f = (N-1)/3 |
| Quorum intersection | Safety bound guaranteeing any two quorums share an honest agent |
| DAWG-class | Large-scale (N=500) attritable autonomous swarm operating regime |
| ECDSA P-256 | Elliptic-curve signature over NIST P-256 for the distributed audit ledger |

## 12.2 References

[1] Oktenli, B. (2026). BLADE-SWARM Governance Node. DOI: 10.5281/zenodo.20351198.

[2] Oktenli, B. (2026). Simulator artifact. https://burakoktenli.com/blade-swarm-simulation.html

[3] Oktenli, B. (2026). Research Portfolio. https://burakoktenli.com

[4] AUTHREX_SWARM.tla formal specification and verification report (this deposit).

[5] DoDD 3000.09; FY26 NDAA; NIST AI Risk Management Framework 1.0.

## 12.3 Contact

Burak Oktenli — Georgetown University, School of Continuing Studies
Website: burakoktenli.com | ORCID: 0009-0001-8573-1667
For questions about this simulation or the governance architecture research program, use the contact form at burakoktenli.com.

*End of Document*