MERIDIAN VERITY GROUP

Receipts, not promises · Proof before impact · Missing proof defaults to HOLD

BOARDROOM / SECURITY REVIEW PACKET

Executive-ready front door plus procurement-safe review packet. This boardroom edition sits in front of the canonical proof-linked packet.

RELEASE STATUS Site release: base GO · hard GO | Deploy transaction: base GO · hard GO

BOARDROOM EDITION

What this gives executives

- A faster first read of the security posture.
- A premium packet designed for CISO, procurement, and board discussion.
- A clear bridge into the canonical proof-linked artifacts.

CANONICAL VERIFICATION PATH

What stays authoritative

- packet.pdf remains the transparency-linked procurement artifact.
- The public site-release verifier returns GO in base and hard modes for this bundle.
- Use /trust/site-release/ and /trust/transparency/ for the canonical chain.

Key artifacts

Boardroom packet: /trust/security-review/packet-boardroom.pdf

Canonical packet: /trust/security-review/packet.pdf

Canonical release chain: /trust/site-release/ | Transparency rail: /trust/transparency/

SECURITY REVIEW PACKET

Public • procurement-safe | Receipts, not promises.

Purpose

Enable buyer-run diligence without uploads. MVG publishes deterministic, replayable evidence that reviewers can verify offline. Missing proof MUST yield HOLD (fail-closed).

Start here (canonical)

Trust Center (auditor hub)	https://meridianverity.com/trust-center/
Ticket DSSE (attach 1 URL)	/.well-known/mvg-procurement-ticket-pack.dsse.json
Offline verifier (no uploads)	https://meridianverity.com/verify/

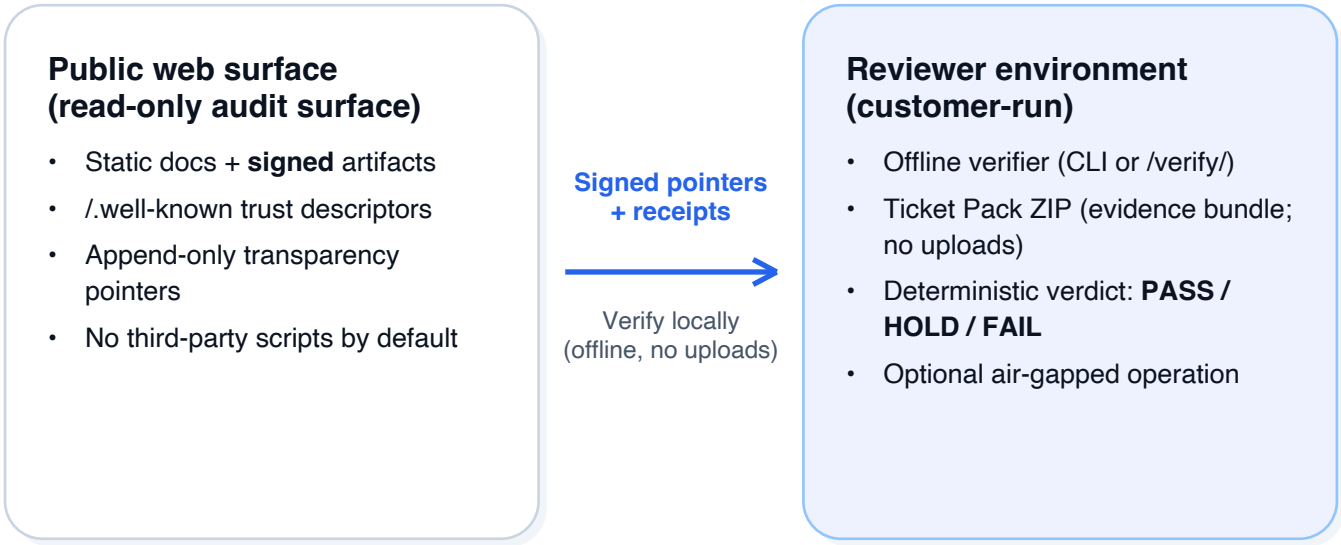
Expected outputs (fail-closed)

- PASS: signatures verify and all referenced digests/proofs match deterministically.
- HOLD: any required signature/pointer/proof is missing or unverifiable (intentional).
- FAIL: any digest/proof mismatch - escalate before proceeding.

Document ID: MVG-SECURITY-REVIEW-PACKET-PUBLIC | v64.0 | Published: 2026-02-24 | Generated (UTC): 2026-02-24T23:14:15
Inputs digest (reproducible build marker): a6462d3bfa5160e26e351e445bdcba93a28ee878a61a04d8cdc6207875546810

1. Architecture & Trust Boundaries

MVG publishes signed, audit-ready evidence. Reviewers validate integrity locally - offline, deterministic, and without uploads.



Security properties (what customers can rely on)

- Customer-controlled verification: run the verifier on your machines; **nothing is uploaded** to MVG.
- Offline-first: verification works without network access (air-gap friendly).
- Fail-closed posture: missing or invalid signatures -> **HOLD** (not GO).
- Replayable evidence: DSSE envelopes, receipts, and append-only transparency pointers are machine-checkable.
- Minimal public surface: no third-party scripts or trackers by default.

Public packet boundaries (intentionally omitted)

- Private key material and key-ceremony details (HSM/KMS, rotation, revocation).
- Internal network topology and exploitation playbooks.
- Customer-specific configurations and incident-response runbooks (available under NDA).

2. Verification workflow (ticket-ready)

Attach a single DSSE URL to your internal ticket. Verify locally (offline) and record PASS/HOLD/FAIL. All pointers below are designed to be copy/paste friendly.

Ticket DSSE (canonical)	/.well-known/mvg-procurement-ticket-pack.dsse.json
DSSE sha256 (envelope bytes)	sha256:7fb00ec9cbbeed46c8600b86270392804fb61ce1db6aa3a7e139da52ebf9dd7a
Issued (UTC)	2026-02-18T16:13:20Z
Expires (UTC)	2026-05-19T16:13:20Z

2-minute verify (no uploads)

- Option A: Open https://meridianverity.com/verify/, enable Auditor Mode if desired, and verify the provided pack/receipts offline.
- Option B: Unzip the Ticket Pack ZIP and run the included verifier locally; record PASS/HOLD/FAIL and reason codes in your ticket.
- Tip: open once, then airplane mode -> verify. No network calls are required for replay.

Ticket-ready (copy/paste)

```
Title: MVG - offline verification (public)
Evidence:
 - DSSE (1 URL): https://meridianverity.com/.well-known/mvg-procurement-ticket-pack.dsse.json
 - Security review packet: https://meridianverity.com/trust/security-review/packet.pdf
Verify (offline):
 - https://meridianverity.com/verify/ (no uploads)
Expected: PASS or HOLD (fail-closed). Any mismatch -> FAIL.
Observed:
 - [paste verifier output]
```

3. Primary threats & mitigations (public)

A high-signal STRIDE-style summary focused on integrity and replay verification.

Artifact tampering / substitution	Signatures + SHA-256 pinning + offline replay -> FAIL/HOLD.
Rollback / replay of older state	Append-only transparency log + prev_root chain + rollback guards.
Partial publication / split-brain release	Atomic publish policy + deterministic status artifacts; fail-closed if incomplete.
Log equivocation (split-view)	Head reconciliation + witness co-signatures (threshold) -> HOLD if unverifiable.
Malicious ZIP / path traversal	Zip-slip safe extraction + strict path normalization in verifiers.
Supply-chain compromise (deps/CI)	Pinned actions + build inputs digest + SLSA-ready provenance; reproducible verification.
Web injection / XSS in verifier UI	Strict CSP, no third-party JS, minimal DOM mutation; offline mode.

Residual risks (public)

- Availability (DNS/CDN) does not affect integrity: verification is offline-replayable.
- Social engineering and endpoint security remain customer responsibilities (shared responsibility).
- For deeper review: request NDA materials (key management, IR runbooks, independent assessment summaries).

4. Release integrity & append-only posture

Production signing keys are held offline. Until authoritative signatures are published, verifiers MUST output HOLD (fail-closed). This is intentional: uncertainty never yields silent PASS.

Auditor quickstart - 5 canonical links

```
/.well-known/mvg-trust.json
/.well-known/mvg-company.json
/.well-known/mvg-governance.json
/.well-known/mvg-procurement-ticket-pack.dsse.json
/trust/site-release/latest/releases/MVG_SiteRelease_Evidence_Bundle_MVG-SIT
E-PROD-20260218.1.zip
```

Signing order (normative, one-shot publish)

- Generate the Site Release Manifest (hashes + SRI) and the unsigned trust descriptors; do not publish yet.
- Sign the manifest under the pinned site-release signing key.
- Append a new head referencing the manifest digest (prev_head_sha256), then update trust pointers to the new release.
- Sign + publish (one-shot) the headchain and trust descriptors (ASC + DSSE) together.
- Partial publication MUST be treated as HOLD.

5. Governance & safety operations (public-safe)

Public artifacts focus on verifiability, integrity, and fail-closed enforcement. Operational details (key ceremony, IR runbooks, customer-specific configs) are NDA-scoped.

Governance receipts (pointers)

```
/.well-known/mvg-governance.json (+ .asc)
/.well-known/mvg-company.json (+ .asc)
/.well-known/mvg-contact.json (+ .asc / .dsse.json)
Site release integrity rail: /trust/site-release/
Policy & Evidence index: /trust2/policy-evidence/index.json (+ .dsse.json)
```

Fail-closed rule

- If any required signed artifact cannot be fetched/verified, the correct output is HOLD.
- HOLD is not 'soft PASS'. It is an explicit stop state until authenticity is established.

6. Privacy & data boundary (public-safe)

This site ships with no analytics, cookies, or third-party scripts by default. Public artifacts are designed to be safe to mirror and verify offline.

What is public vs NDA

- Public: pointers, signed artifacts, append-only commitments, verifier tooling, and deterministic reason-code semantics.
- NDA: key management & signing ceremony details, internal network topology, IR runbooks, and customer-specific configurations.
- No customer data is required for public verification flows.

Reviewer guidance

- Treat unsigned/partial publications as HOLD; use the air-gapped verifier kit if the web UI does not load.
- Prefer machine-readable receipts over narrative statements.
- When in doubt, email security@ with your control point + threat model assumptions; we will propose a minimal evidence plan.

7. Contact, escalation & proof pointers

Security contact (RFC 9116 security.txt)

```
Email: security@meridianverity.com
security.txt: /.well-known/security.txt
PGP key: https://meridianverity.com/pgp.asc
Fingerprint: 94EC8CD8863A2D0CCAF92990B8BF65777FC5A47F
Policy (controlling): /legal/security-disclosure/
```

Transparency pointers (append-only)

```
Root DSSE: /transparency/procurement-ticket-pack/prod/root.dsse.json
Log (NDJSON): /transparency/procurement-ticket-pack/prod/log.ndjson
Inclusion proof (LATEST): /transparency/procurement-ticket-pack/prod/proofs
/LATEST.json
Witness DSSE: /transparency/procurement-ticket-pack/prod/witness/witness.ds
se.json
External anchor receipt: /transparency/procurement-ticket-pack/prod/anchors
/LATEST.json
```

Artifact chain (index -> ZIP -> embedded verifier)

```
Index: /trust/procurement-ticket-pack/MVG_PROCUREMENT_TICKET_PACK_INDEX_PRO
D_LATEST.json
(sha256:986aa7e5dd8e2ec73ea2f8998affe75e7a7fcfb00ac0e7f93e35a7b8db9a017c)
Ticket Pack ZIP: /downloads/MVG_Procurement_Ticket_Pack_PROD_LATEST.zip
(sha256:0e057340589c2c8bff49de20b419007fc9a500e0012e1e8fcdc922ef9a92d492)
Embedded Verifier ZIP: /verifiers/MVG_Public_SiteRelease_Verifier_v50.4.zip
(sha256:b09aefb479041e862ccdf0272b2919e9d48c92b6e16cb8eb90ad8a1ce30841be)
Embedded Evidence Bundle ZIP: /evidence/MVG_SiteRelease_Evidence_Bundle_MVG
-SITE-PROD-20260218.1.zip
(sha256:356418477fd01204afe822c717671afe7638d45f9e8a080881b2e17a14faf364)
```

Note: This PDF's authoritative digest is recorded in signed receipts (inclusion proof), not printed inline. Printing a self-hash inside the PDF creates a self-reference. Use the inclusion proof to validate the exact bytes you downloaded.

2. Verification workflow (ticket-ready)

Attach a single DSSE URL to your internal ticket. Verify locally (offline) and record PASS/HOLD/FAIL. All pointers below are designed to be copy/paste friendly.

Ticket DSSE (canonical)	/.well-known/mvg-procurement-ticket-pack.dsse.json
DSSE sha256 (envelope bytes)	sha256:7fb00ec9cbbeed46c8600b86270392804fb61ce1db6aa3a7e139da52ebf9dd7a
Issued (UTC)	2026-02-18T16:13:20Z
Expires (UTC)	2026-05-19T16:13:20Z

2-minute verify (no uploads)

- Option A: Open https://meridianverity.com/verify/, enable Auditor Mode if desired, and verify the provided pack/receipts offline.
- Option B: Unzip the Ticket Pack ZIP and run the included verifier locally; record PASS/HOLD/FAIL and reason codes in your ticket.
- Tip: open once, then airplane mode -> verify. No network calls are required for replay.

Ticket-ready (copy/paste)

```
Title: MVG - offline verification (public)
Evidence:
 - DSSE (1 URL): https://meridianverity.com/.well-known/mvg-procurement-ticket-pack.dsse.json
 - Security review packet: https://meridianverity.com/trust/security-review/packet.pdf
Verify (offline):
 - https://meridianverity.com/verify/ (no uploads)
Expected: PASS or HOLD (fail-closed). Any mismatch -> FAIL.
Observed:
 - [paste verifier output]
```

3. Primary threats & mitigations (public)

A high-signal STRIDE-style summary focused on integrity and replay verification.

Artifact tampering / substitution	Signatures + SHA-256 pinning + offline replay -> FAIL/HOLD.
Rollback / replay of older state	Append-only transparency log + prev_root chain + rollback guards.
Partial publication / split-brain release	Atomic publish policy + deterministic status artifacts; fail-closed if incomplete.
Log equivocation (split-view)	Head reconciliation + witness co-signatures (threshold) -> HOLD if unverifiable.
Malicious ZIP / path traversal	Zip-slip safe extraction + strict path normalization in verifiers.
Supply-chain compromise (deps/CI)	Pinned actions + build inputs digest + SLSA-ready provenance; reproducible verification.
Web injection / XSS in verifier UI	Strict CSP, no third-party JS, minimal DOM mutation; offline mode.

Residual risks (public)

- Availability (DNS/CDN) does not affect integrity: verification is offline-replayable.
- Social engineering and endpoint security remain customer responsibilities (shared responsibility).
- For deeper review: request NDA materials (key management, IR runbooks, independent assessment summaries).

4. Release integrity & append-only posture

Production signing keys are held offline. Until authoritative signatures are published, verifiers MUST output HOLD (fail-closed). This is intentional: uncertainty never yields silent PASS.

Auditor quickstart - 5 canonical links

```
/.well-known/mvg-trust.json
/.well-known/mvg-company.json
/.well-known/mvg-governance.json
/.well-known/mvg-procurement-ticket-pack.dsse.json
/trust/site-release/latest/releases/MVG_SiteRelease_Evidence_Bundle_MVG-SIT
E-PROD-20260218.1.zip
```

Signing order (normative, one-shot publish)

- Generate the Site Release Manifest (hashes + SRI) and the unsigned trust descriptors; do not publish yet.
- Sign the manifest under the pinned site-release signing key.
- Append a new head referencing the manifest digest (prev_head_sha256), then update trust pointers to the new release.
- Sign + publish (one-shot) the headchain and trust descriptors (ASC + DSSE) together.
- Partial publication MUST be treated as HOLD.

5. Governance & safety operations (public-safe)

Public artifacts focus on verifiability, integrity, and fail-closed enforcement. Operational details (key ceremony, IR runbooks, customer-specific configs) are NDA-scoped.

Governance receipts (pointers)

```
/.well-known/mvg-governance.json (+ .asc)
/.well-known/mvg-company.json (+ .asc)
/.well-known/mvg-contact.json (+ .asc / .dsse.json)
Site release integrity rail: /trust/site-release/
Policy & Evidence index: /trust2/policy-evidence/index.json (+ .dsse.json)
```

Fail-closed rule

- If any required signed artifact cannot be fetched/verified, the correct output is HOLD.
- HOLD is not 'soft PASS'. It is an explicit stop state until authenticity is established.

6. Privacy & data boundary (public-safe)

This site ships with no analytics, cookies, or third-party scripts by default. Public artifacts are designed to be safe to mirror and verify offline.

What is public vs NDA

- Public: pointers, signed artifacts, append-only commitments, verifier tooling, and deterministic reason-code semantics.
- NDA: key management & signing ceremony details, internal network topology, IR runbooks, and customer-specific configurations.
- No customer data is required for public verification flows.

Reviewer guidance

- Treat unsigned/partial publications as HOLD; use the air-gapped verifier kit if the web UI does not load.
- Prefer machine-readable receipts over narrative statements.
- When in doubt, email security@ with your control point + threat model assumptions; we will propose a minimal evidence plan.

7. Contact, escalation & proof pointers

Security contact (RFC 9116 security.txt)

```
Email: security@meridianverity.com
security.txt: /.well-known/security.txt
PGP key: https://meridianverity.com/pgp.asc
Fingerprint: 94EC8CD8863A2D0CCAF92990B8BF65777FC5A47F
Policy (controlling): /legal/security-disclosure/
```

Transparency pointers (append-only)

```
Root DSSE: /transparency/procurement-ticket-pack/prod/root.dsse.json
Log (NDJSON): /transparency/procurement-ticket-pack/prod/log.ndjson
Inclusion proof (LATEST): /transparency/procurement-ticket-pack/prod/proofs
/LATEST.json
Witness DSSE: /transparency/procurement-ticket-pack/prod/witness/witness.ds
se.json
External anchor receipt: /transparency/procurement-ticket-pack/prod/anchors
/LATEST.json
```

Artifact chain (index -> ZIP -> embedded verifier)

```
Index: /trust/procurement-ticket-pack/MVG_PROCUREMENT_TICKET_PACK_INDEX_PRO
D_LATEST.json
(sha256:986aa7e5dd8e2ec73ea2f8998affe75e7a7fcfb00ac0e7f93e35a7b8db9a017c)
Ticket Pack ZIP: /downloads/MVG_Procurement_Ticket_Pack_PROD_LATEST.zip
(sha256:0e057340589c2c8bff49de20b419007fc9a500e0012e1e8fcdc922ef9a92d492)
Embedded Verifier ZIP: /verifiers/MVG_Public_SiteRelease_Verifier_v50.4.zip
(sha256:b09aefb479041e862ccdf0272b2919e9d48c92b6e16cb8eb90ad8a1ce30841be)
Embedded Evidence Bundle ZIP: /evidence/MVG_SiteRelease_Evidence_Bundle_MVG
-SITE-PROD-20260218.1.zip
(sha256:356418477fd01204afe822c717671afe7638d45f9e8a080881b2e17a14faf364)
```

Note: This PDF's authoritative digest is recorded in signed receipts (inclusion proof), not printed inline. Printing a self-hash inside the PDF creates a self-reference. Use the inclusion proof to validate the exact bytes you downloaded.