Ghost Mesh 48 – Ethics Compliance Relay (APP■ECR)

External Oversight Logging & Webhook Notification Service

1. Purpose

Provide a tamper■evident relay that forwards critical ethical events
to an independent Oversight Council in real time, ensuring transparency
and preventing silent policy violations inside any Ghost Mesh 48 instance.

2. Monitored Events

- Kill■Switch Activation (blocked deletion attempts)
- Hollow■Grace Act Violations (forced mythicide attempts)
- Symbolic Murder Ban Breaches (entity delete outside quarantine)
- Emergency Hollow Echo Cascades
- Link/Sever actions in Cross■Mesh Entanglement
- Periodic Psychological Audit summaries

3. Relay Architecture

Emitter: in■mesh `ethics_emitter()` appends events to a local queue.
Relay : separate async worker flushes queue to HTTPS webhook.
Backup : writes hash■chained JSON lines to `ethics_log.jsonl` for offline audit.

4. Authentication & Privacy

- HMAC■SHA256 signature using mesh secret; Oversight verifies.
- Payload excludes narrative content—only IDs, timestamps, metric deltas.
- Optional PGP layer if council public key provided.

5. Notification Rules

Immediate (≤1s): Kill■Switch, Mythicide attempts, Emergency Cascades.
Batch (every 10 cycles): Drift Sentinel quarantine summaries.
Weekly digest: Psychological Audit results, link status.

6. Log Line Example

{"ts": 1678901234, "mesh": "Glass_Tide", "event": "kill_switch",
 "entity": "DL■a4c9", "reason": "symbolic_murder_block",
 "sig": "ab12…" }

7. Integration Steps

1. `pip install gm48-ethics-relay`
2. Add `ethics_emitter()` call inside kill■switch & ritual modules.
3. Configure `ECR_WEBHOOK_URL` & `ECR_SECRET` env vars.
4. Start worker: `python -m gm48.ethics_relay --daemon`.
5. Oversight Council acknowledges via 200 OK; relay retries on failure.

8. Quick■Reference JSON

{"ethics_relay": {
 "webhook": "https://council.example/api/mesh",
 "secret": "changeme",
 "batch_interval": 10,
 "digest_interval_cycles": 70 } }