AEGISMESH

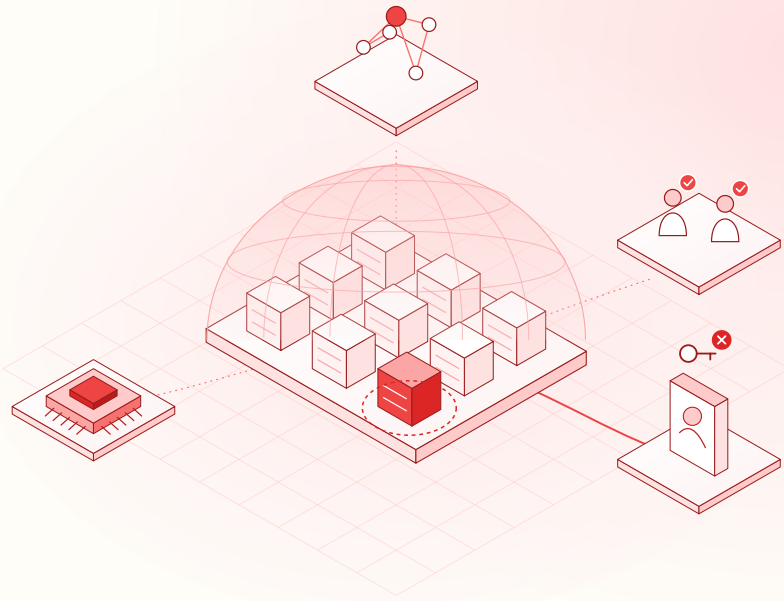
AegisMesh: Architettura Tecnica

Panoramica pubblica dell'architettura

Più veloce dell'attacco, più saggia del panico: la sonda decide al bordo, il cervello centrale giudica, e sugli asset critici decidono due persone.

AegisMesh è una piattaforma di cybersecurity Zero-Trust che unisce la telemetria del kernel all'identità cloud federata. Rileva con sensori eBPF, normalizza nello standard OCSF, correla in un grafo temporale e contiene la minaccia su due fronti: congelamento del container sull'host e revoca delle sessioni sull'identity provider.

La catena completa è stata **validata in staging** su un nodo dedicato (fasi P4–P7). La configurazione di produzione si definisce per ogni progetto (vedi §8 e §9).



Il gap Zero-Trust

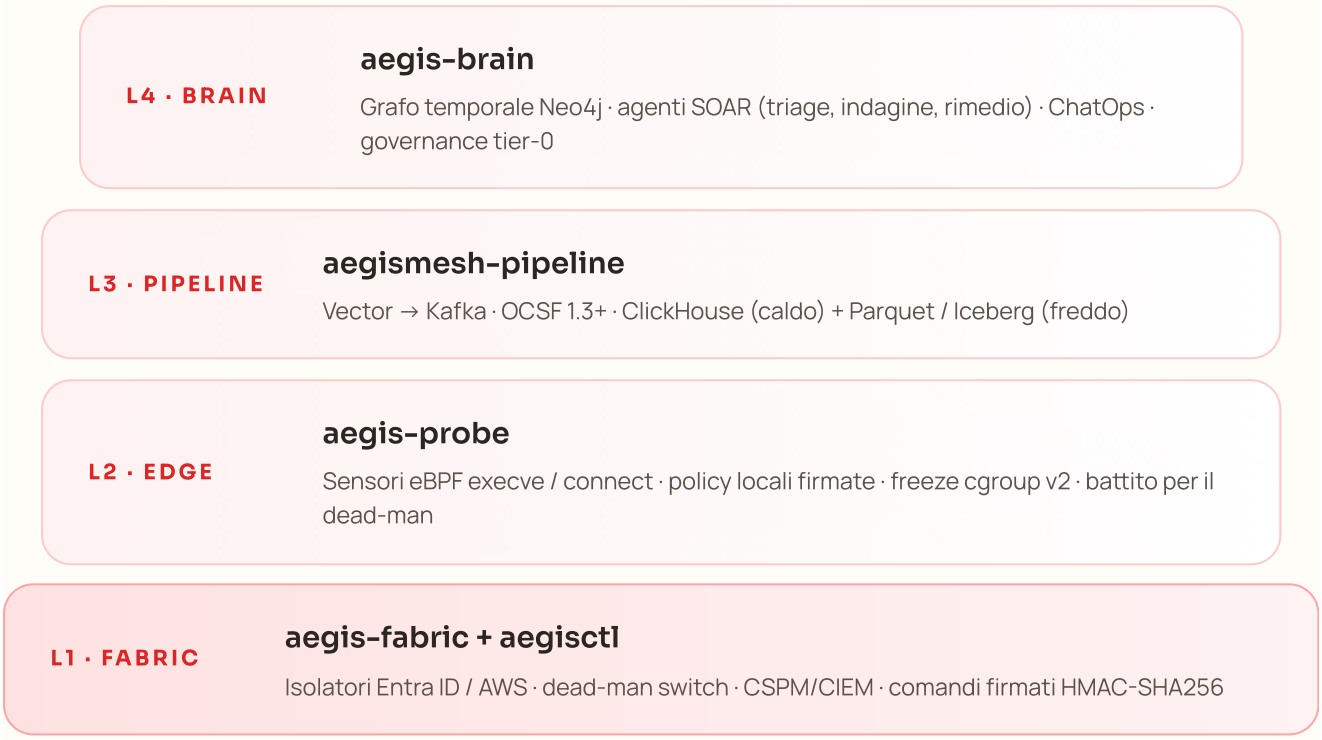
Il perimetro da difendere è l'identità, non più la rete. Una compromissione su un nodo Linux può impiegare minuti o ore a riflettersi sui controlli IAM del cloud: bloccare solo il processo lascia validi i token federati, terminare i workload distrugge le evidenze forensi. AegisMesh tratta host e identità come **un solo incidente**.

FRONTE	SENZA COORDINAMENTO	CON AEGISMESH
Host / container	Kill o spegnimento: le evidenze si perdono	Freeze non distruttivo del cgroup: memoria e processi restano analizzabili
Identità cloud	Token e sessioni restano validi	Revoca delle sessioni sull'identity provider (Microsoft Entra ID)
Decisione	Manuale e lenta, oppure automatica e rischiosa	Automatica sugli asset standard, due approvatori sugli asset tier-0

1. Cosa fa e cosa non fa

AEGISMESH FA	AEGISMESH NON FA
Rileva exec e connessioni in uscita con sonde eBPF e applica policy locali firmate	Sostituire il SOC: sugli asset critici la decisione resta alle persone
Normalizza la telemetria in OCSF 1.3+ e la conserva su due livelli (caldo e freddo)	Eseguire azioni distruttive automatiche sugli asset tier-0
Correla host, processi, identità e permessi in un grafo e riconosce tecniche MITRE ATT&CK	Dichiarare certificazioni o conformità normative (non attestate da questo documento)
Contiene con comandi firmati: freeze, isolamento host, revoca delle sessioni cloud	Annullare la revoca con una sola chiamata: il ripristino dell'accesso passa dalle policy dell'identity provider

2. Architettura



Tassonomia a quattro livelli (fonti: README.md, ADR-0001)

AegisMesh adotta un'**elaborazione asimmetrica Edge/Cloud** (ADR-0001): al bordo la sonda è autonoma e lavora anche offline; nel cloud ingestione, grafo e SOAR lavorano in modo asincrono e restano l'autorità per la governance del raggio d'impatto, il quorum tier-0 e la correlazione tra host. In sintesi: *la sonda è la sopravvivenza, il cervello è il giudizio*.

Archivio a due livelli (ADR-0002). Vector è il punto di normalizzazione obbligato: i formati proprietari vengono adattati *prima* di Kafka. L'archivio caldo ClickHouse (`aegis_lake`) serve la caccia interattiva e il SOAR; l'archivio freddo in Parquet / Apache Iceberg su object storage riceve i dati oltre la finestra di conservazione calda (predefinita 7 giorni).

Budget temporale di progetto (matrice ATT&CK): blocco locale al bordo a 0 ms · ingestione OCSF in 1–3 s · correlazione a grafo in 5–10 s · poi SOAR con comandi firmati.

3. Contratti e flussi

CONTRATTO	DETTAGLIO
Sonda → pipeline	Classi OCSF `1007` (Process Activity) e `4001` (Network Activity); il contratto di progetto prevede Protobuf/gRPC su mTLS con attestazione TPM
Bus Kafka	**aegis.telemetry.critical** (obiettivo di progetto P99 < 250 ms) · **aegis.telemetry.audit** (micro-batch) · **aegis.telemetry.quarantine**
Comandi di controllo	**aegis.control.commands** : payload Protobuf firmato HMAC-SHA256, con ID chiave per la rotazione e scadenza; la sonda rifiuta comandi manomessi o con sfasamento oltre 30 s
Azioni previste dal protocollo	**ISOLATE_HOST** · **FREEZE_CONTAINER** · **REVOKE_IDP_SESSION** · **REVERT_IAM_POLICY**
Battito della sonda	**aegis.probe.heartbeat** ogni 3–5 s, con hash del bundle di policy, stato dei driver e sequenza

4. Governance: raggio d'impatto e regola delle due persone

Le regole sono codificate in `BlastRadiusPolicy` e `MitigationDispatcher` (ADR-0003), non affidate a un manuale.

CLASSE	COMPORTAMENTO
`EDGE_LOCAL`	Kill o freeze sul nodo, senza approvazione cloud
Asset standard	Invio automatico tramite **aegisctl** , comando firmato HMAC
Asset `tier-0`	Proposta in stato `PENDING_APPROVAL` : servono **2 approvatori distinti** (API o ChatOps); la stessa persona non conta due volte
Impatto di massa	Sempre soggetto ad approvazione; interruttori automatici sugli errori delle dipendenze

Tetto predefinito: isolamento di **max 2% dei nodi attivi all'ora** senza override amministrativo. In ChatOps (Slack in Socket Mode, connessione solo in uscita, nessuna porta pubblica) la card della proposta si aggiorna sul posto: in attesa → approvato 1/2 → eseguito. È prevista una procedura break-glass con doppia custodia per quando identity provider, ChatOps o API non sono disponibili.

5. Dead-man switch

Se la sonda tace oltre **15 secondi**, o segnala manomissione (attestazione fallita, driver non funzionante, regressione della sequenza), il monitor di aegis-fabric pubblica `AGENT_TAMPERING_SUSPECTED` sul topic di quarantena e attiva l'isolamento di rete del nodo (blocco del traffico in uscita tranne la gestione).

6. Contenimento duale

Quando una connessione in uscita da shell verso un C2 (MITRE T1059.004, reverse shell) si correla a un'identità federata, il SOAR propone `FREEZE_AND_REVOKE_SESSION` (ADR-0006); senza identità cloud la proposta riguarda solo l'host.

GAMBA	MECCANISMO	MODALITÀ
Host / container	`aegisctl mitigate --kind FREEZE_CONTAINER` → bus di controllo Kafka → sonda	Asincrona, comando firmato
Identità cloud	Microsoft Entra ID `revokeSignInSessions` (Graph API)	Sincrona

Tempi misurati sulla demo di staging (P7): AttackPath in circa 2,3 s dall'evento kernel · revoca Entra in circa 350–410 ms · emissione del comando host sotto i 10 ms. Sono misure di staging, non livelli di servizio garantiti.

7. Modello di minaccia (MVP)

TECNICA	NOME	CONTROLLO PREVISTO
T1059	Command and Scripting Interpreter	Policy locale firmata + eBPF execve / kill del processo
T1486	Data Encrypted for Impact	Kill sul dispositivo + quarantena della storyline
T1078.004	Cloud Accounts	Rilevazioni CIEM + revoca firmata della sessione
T1530	Data from Cloud Storage	CSPM agentless: bucket pubblici
T1068	Privilege Escalation	Deriva a runtime via eBPF + freeze del cgroup
T1021	Remote Services	Percorso nel grafo + isolamento host con limiti di frequenza
T1048	Exfiltration Over Alternative Protocol	Caccia sull'archivio caldo + blocco di rete firmato

La matrice è quella di progetto (MVP): lo stato effettivo di ogni componente è al §8.

8. Cosa è vero oggi

Validato in staging (nodo dedicato Ubuntu 24.04, kernel 6.8+, cgroup v2)

- **P4:** compilazione eBPF e aggancio su kernel reale, ingestione OCSF 1007, freeze del cgroup verificato, stack Kafka / Vector / ClickHouse / Neo4j / SOAR, ciclo di controllo end-to-end; perimetro chiuso (porte dei servizi solo in locale).
- **P5:** revoca live delle sessioni Entra ID su utente canary, correlazione OCSF → identità nel grafo, rilevazione → approvazione a due persone → revoca.
- **P6:** quorum a due persone via Slack (Socket Mode) con aggiornamento della card fino a «eseguito».
- **P7:** contenimento duale live su canary: freeze via bus Kafka + revoca Entra; poi ripristino della modalità dry-run, che in staging resta quella predefinita.
- CI su ogni modifica: test Go e Python, verifica dei contratti Protobuf e del chart Helm (versione 0.1.0).

Predisposto, non ancora collaudato live

- Quarantena AWS via security group: codice e test presenti, collaudo live rinviato per scelta operativa.
- Attestazione TPM della sonda: oggi simulata (MVP); mTLS nel contratto di progetto.
- CSPM/CIEM agentless e segnali di rischio ITDR verso Entra: componenti in forma di stub.
- Archivio freddo: in staging su object storage compatibile S3 di test.
- Pacchetto di rilevazione volutamente ridotto (T1059, T1078, reverse shell); ChatOps su Teams tramite Bot Framework da completare.

9. Percorso di adozione e avvertenze

Laboratorio (Docker Compose o chart Helm con dipendenze di laboratorio) → **Staging** (account dedicati mai di produzione, risorse marcate per lo staging, flag live a zero finché i controlli non sono verdi, dry-run di default) → **Canary live** (finestra esplicita, utente canary, poi ripristino del dry-run) → **Produzione** (Kafka, ClickHouse e Neo4j gestiti, segreti in un secret store, copertura on-call per il quorum).

Avvertenze. Questa panoramica è un documento tecnico informativo. Non è una certificazione né un parere di conformità normativa. Le metriche citate provengono dall'ambiente di staging. Microsoft Entra ID, Slack e AWS sono marchi dei rispettivi titolari.

Contatti · GLOBAL CONSULTANT di Roberto Bonadies · info@globalconsultant.eu · +39 393 5756044 · globalconsultant.eu