Project Progress Update

Date: February 13, 2026 Project: FoodProject SOC Platform (Wazuh + Shuffle + IRIS-web + SOC Integrator)

1) Executive Summary

The MVP platform is operational and running end-to-end in the lab environment. Core integrations are in place:

- Detection: Wazuh
- Automation: Shuffle
- Case management: IRIS-web (replacing DFIRTrack)
- Escalation (MVP): PagerDuty Stub
- Orchestration/API layer: soc-integrator

All major containers are currently up, and key health checks are passing.

2) Completed Work

Platform orchestration and operations

- Combined stack runner created and improved (`run-combined-stack.sh`)
- Added command support for:
  - `up` , `down` , `logs` , `status` , `help`
  - per-target control (`wazuh` , `iris` , `shuffle` , `pagerduty` , `integrator`)
- Added consolidated health/status script (`soc-status.sh`)

Integration architecture

- Connected Wazuh, Shuffle, IRIS-web, PagerDuty Stub, and soc-integrator on shared network
- Resolved startup conflicts and runtime issues (port, compose, routing compatibility)

SOC Integrator (MVP)

- Added/validated integration APIs for:
  - Wazuh

    - Shuffle
    - IRIS-web
    - PagerDuty Stub
- Implemented MVP orchestration endpoints:
    - `POST /mvp/incidents/ingest`
    - `POST /mvp/ioc/evaluate`
    - `POST /mvp/vpn/evaluate`
    - `GET /mvp/config/policies`
    - `PUT /mvp/config/policies`
    - `GET /mvp/health/dependencies`
- Added internal API-key protection for mutation endpoints

Persistence layer

- Added PostgreSQL service for soc-integrator (`soc-integrator-db`)
- Added incident/policy/audit schema and startup initialization
- Enabled deduplication and audit tracking for incident processing

Testing utilities and documentation

- Added Wazuh test-event injection script:
    - `scripts/send-wazuh-test-events.sh`
- Added root project docs:
    - `README.md`
- Added root ignore rules:
    - `.gitignore`

3) Current Live Status (Lab)

Current stack status: **UP**

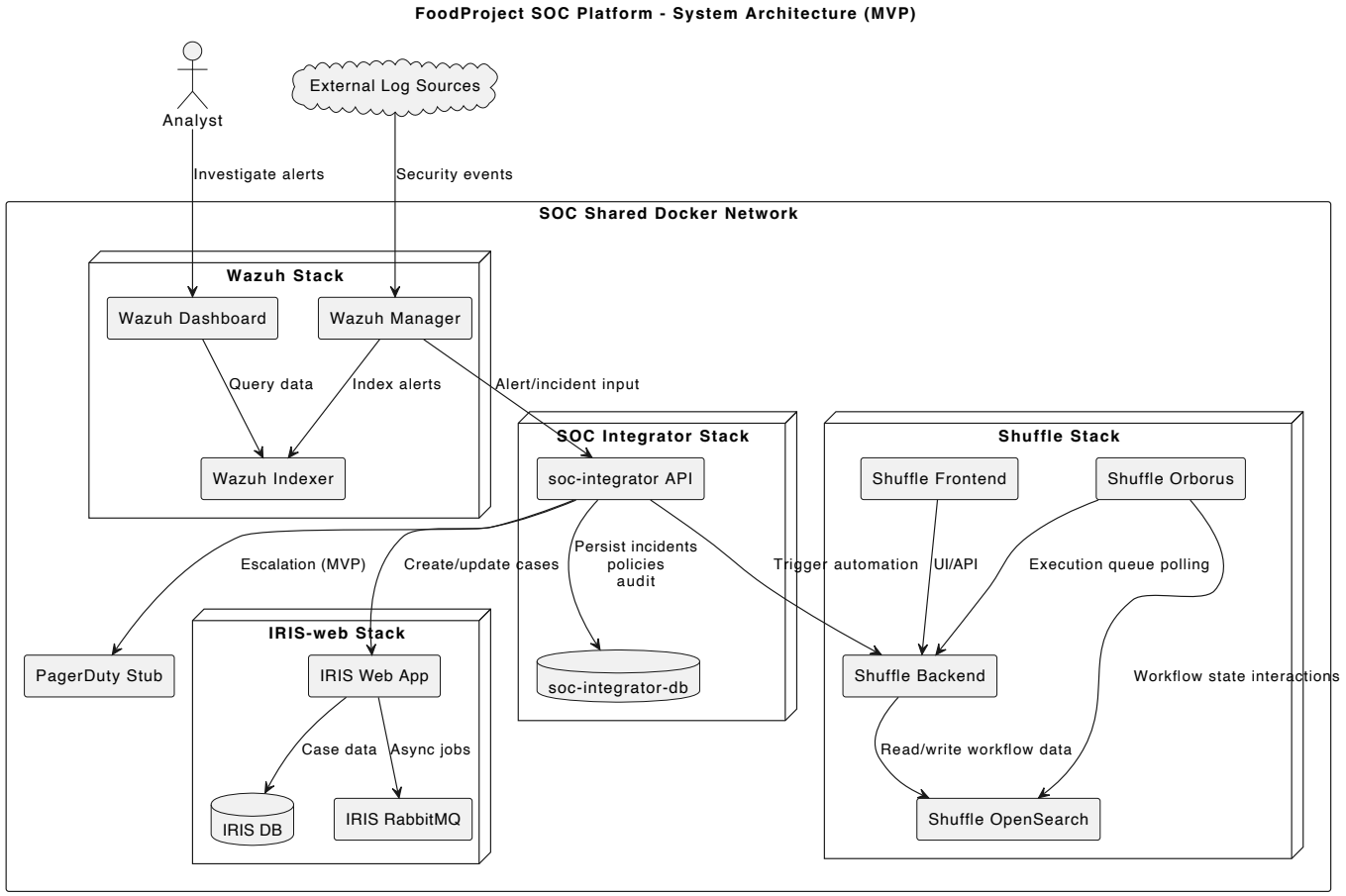
Healthy/available components:

- Wazuh manager, indexer, dashboard
- IRIS-web app/nginx/worker/db/rabbitmq
- Shuffle backend/frontend/opensearch/orborus
- PagerDuty Stub
- soc-integrator + soc-integrator-db

Endpoint checks:

- Wazuh Dashboard: OK
- Wazuh API: OK (auth-protected, expected 401 on unauthenticated root)
- IRIS Web: OK
- Shuffle Frontend: OK
- Shuffle Backend: reachable
- Shuffle OpenSearch: reachable (auth-protected)
- PagerDuty Stub: OK
- soc-integrator `/health`: OK

4) System Architecture Diagram (PlantUML)



5) In Progress / Remaining for Customer UAT

1. Detection content tuning

- Fine-tune Wazuh rules/decoders for customer log patterns and false-positive reduction

2. Use-case calibration

- Validate risk/severity mapping per approved use cases
- Tune exception list and threshold logic (especially VPN geo anomaly)

3. UAT evidence package

- Capture deterministic UAT scenarios and outputs for:
  - IOC flow
  - VPN outside-TH flow
  - IRIS case creation/update
  - PagerDuty Stub escalation path

4. Production hardening items

- Rotate default/local secrets used in lab config
- Lock down internal API keys and access boundaries

6) Risks / Notes

- Current escalation target is **PagerDuty Stub** by design for MVP. Real PagerDuty production integration is the next stage.
- Some Wazuh config certificate directories are root-owned in the local lab clone, which may affect local git add operations if not excluded/fixed.

7) Next Milestone (Proposed)

Next milestone: **MVP UAT Completion**

Target outputs:

- Approved UAT checklist execution
- Tuned policy thresholds for customer environment
- Signed-off incident lifecycle flow: Wazuh event -> soc-integrator decision -> IRIS case -> PagerDuty Stub escalation

---

Date: February 26, 2026 Project: FoodProject SOC Platform (Wazuh + Shuffle + IRIS-web + SOC Integrator)

Incremental Progress Since February 13, 2026

1) IOC Enrichment and Evaluation

- Added IOC APIs in `soc-integrator`:
  - `POST /ioc/enrich`
  - `POST /ioc/evaluate`
  - `GET /ioc/history`
  - `POST /ioc/upload-file`
  - `POST /ioc/evaluate-file`
  - `GET /ioc/analysis/{analysis_id}`
- Integrated VirusTotal adapter for domain/hash/file intelligence and analysis lookups.
- Integrated AbuseIPDB adapter for IP reputation checks.
- Added IOC trace persistence (`ioc_trace`) and repository methods for audit/history.

2) IRIS Integration Enhancements

- Added IRIS ticket APIs in `soc-integrator`:
  - `POST /iris/tickets`
  - `GET /iris/tickets`
- Updated IRIS API key in environment and verified ticket creation path via API.
- Added demo data seeding script:
  - `scripts/seed-iris-demo-data.sh`

3) Shuffle Workflow Automation

- Created and updated sample Shuffle workflow assets for webhook-driven IRIS ticket creation:
  - `shuffle-workflows/sample-webhook-soc-integrator-iris-workflow.json`
  - `shuffle-workflows/sample-webhook-soc-integrator-iris-workflow.md`
- Added workflow update helper script:
  - `scripts/update-shuffle-workflow-from-template.sh`
- Updated target workflow (`07ecad05-ff68-41cb-888d-96d1a8e8db4b`) with:
  - webhook trigger
  - HTTP action (`http 1.4.0`) to call `soc-integrator` ticket API
  - tested webhook execution path to successful completion

4) Networking and Runtime Fixes

- Resolved Shuffle action DNS failure to `soc-integrator` by attaching `soc-integrator` service to Shuffle execution network(s) in:
  - `compose-overrides/soc-integrator.yml`
- Verified connectivity from Shuffle execution context to:
  - `http://soc-integrator:8080/health`

5) Security and Repository Hygiene

- Added `.env` and `.env.*` to root `.gitignore` (kept `.env.example` tracked).
- Removed tracked env files from git cache to prevent secret leakage.
- Updated operational API keys in `soc-integrator/.env` for Shuffle, IRIS, VirusTotal, and AbuseIPDB.

6) Current Status (Lab)

- `soc-integrator` health endpoint: reachable.
- IOC enrich/evaluate flows: operational for domain/hash and file submission paths.
- Shuffle webhook-to-IRIS automation: operational after network fix.
- Core stack components remain available for continued UAT and tuning.

7) Simulation Logs Workstream

Completed

- Added FortiGate simulation coverage for multiple models:
  - 40F
  - 60F
  - 80F
  - 501E
- Added endpoint agent simulation coverage for:
  - Windows clients
  - macOS clients
  - Linux clients
- Added continuous run mode (`--forever`) to simulation scripts for long-running lab traffic generation.
- Extended script set to support realistic event streams for Wazuh ingestion and rule validation.

Operational scripts

- `scripts/send-wazuh-test-events.sh`

Date: March 4, 2026 Project: FoodProject SOC Platform (Wazuh + Shuffle + IRIS-web + SOC Integrator)

Incremental Progress Since Previous Update (March 4, 2026)

1) Production-Profile Simulator Payloads

- Added production payload mode to proposal simulators:
  - `scripts/send-wazuh-proposal-required-events.sh`
  - `scripts/send-wazuh-proposal-appendix-b-events.sh`
- New argument:
  - `--profile=simulation|production` (default remains `simulation`)
- In `production` profile, simulator messages omit `section/usecase_id/usecase` markers and emit production-like key/value fields to support real parser/decoder testing.

2) Wazuh Normalize API Improvements

- Enhanced `POST /ingest/wazuh-alert` in `soc-integrator`:
  - returns both legacy normalized shape and SOC normalized event shape.
- Added `GET /ingest/wazuh-alert/samples` with practical sample request/response cases for:
  - DNS IOC
  - VMware auth
  - Windows Sysmon
  - C1 impossible travel

3) C1 Normalization (Production-First)

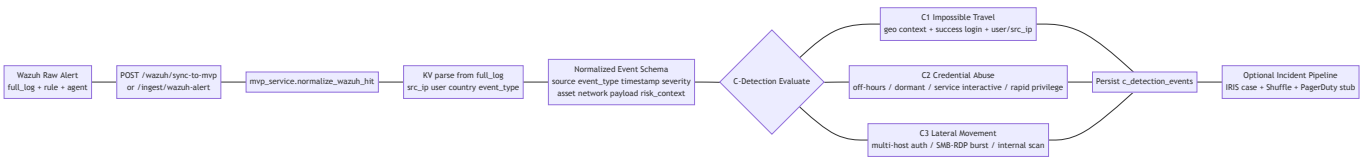
- Updated C1 normalization logic in:
  - `soc-integrator/app/services/mvp_service.py`
- C1 now maps from production characteristics (not only simulator markers):
  - identity/vpn source context
  - successful login/auth indicator
  - user + source IP present

    - geo context present (`country` and/or `src_lat/src_lon`)
- Legacy `section/usecase_id` C1 markers are kept as fallback for backward compatibility.

4) Current Validation Status

- Production-profile simulator events are being sent to Wazuh successfully.
- Some runs still show only base-rule matching during verification due to current Wazuh manager runtime instability in lab (intermittent restart/init issues), which affects deterministic decoder/rule validation windows.
- Next validation step after stable manager window:
    - re-run production-profile A1/B1/B2/B3 batches
    - confirm `110xxx` production rules with consistent hit evidence.

5) Mermaid Diagram: C1-C3 Normalization Flow (SOC Integrator)



- `scripts/send-wazuh-endpoint-agent-test-events.sh`
- additional simulation scripts under `scripts/` for firewall and endpoint scenarios with continuous mode enabled

Detection alignment status

- Simulation work has been aligned to the detection objectives documented in:
    - `Security Detection & Threat Intelligence Enhancement Proposal-2.md`
- Proposal use-case mapping explicitly covered in simulation:
    - **A1. DNS / Firewall (IOC)**:
        - DNS network communication to malicious domain
        - DNS/Firewall malicious domain IOC detection events

    - **A2. FortiGate IPS/IDS & Firewall**:
        - allowed RDP from public IP
        - admin password change
        - create/add admin account
        - disable email notification
        - config download
        - multiple critical/high IDS alerts
        - port scanning (public/private source variants)
        - IOC detection and communication to malicious IP
    - **A3. FortiGate VPN**:
        - authentication success from guest account
        - authentication success from multiple countries
        - brute-force success pattern
        - multiple fail patterns (many accounts from one source)
        - authentication success from outside Thailand
    - **A4. Windows / Active Directory**:
        - privileged/service account authentication failures
        - password spray and multi-source fail patterns
        - success from public IP / guest account
        - pass-the-hash style success indicators
        - account/group privilege change and account lifecycle events (create/re-enable)
        - AD enumeration behavior indicators
- Endpoint client simulations were added to complement proposal scope for heterogeneous environments:
    - Windows agent events
    - macOS agent events
    - Linux agent events
- Current use is suitable for pipeline and workflow validation (ingest -> detect -> automate -> case creation).
- Remaining work is focused on fine-grained scenario calibration:
    - event frequency tuning
    - field/value realism per source
    - expected alert volume by use case for cleaner UAT evidence

8) API Request/Response Samples

IOC Enrich

Request:

```
curl -sS -X POST http://localhost:8088/ioc/enrich \
  -H 'Content-Type: application/json' \
  -d '{
    "ioc_type": "domain",
    "ioc_value": "google.com",
    "sources": ["virustotal"]
  }'
```

Sample response:


```
{
  "success": true,
  "ioc_type": "domain",
  "ioc_value": "google.com",
  "enrichment": {
    "virustotal": {
      "reputation": 120,
      "last_analysis_stats": {
        "malicious": 0,
        "suspicious": 0,
        "harmless": 90
      }
    }
  }
}
```


IOC Evaluate

Request:

```
curl -sS -X POST http://localhost:8088/ioc/evaluate \
  -H 'Content-Type: application/json' \
  -d '{
    "ioc_type": "hash",
    "ioc_value": "44d88612fea8a8f36de82e1278abb02f",
    "sources": ["virustotal"]
  }'
```

Sample response:

```
{
  "success": true,
  "matched": true,
  "severity": "high",
  "reason": "VirusTotal marked IOC as malicious",
  "ioc_type": "hash",
  "ioc_value": "44d88612fea8a8f36de82e1278abb02f"
}
```

Create IRIS Ticket (via soc-integrator)

Request:

```
curl -sS -X POST http://localhost:8088/iris/tickets \
  -H 'Content-Type: application/json' \
  -d '{
    "title": "Suspicious domain detected",
    "description": "Automated ticket from IOC evaluation pipeline",
    "severity": "medium",
    "source_ref": "shuffle-webhook-demo"
  }'
```

Sample response:

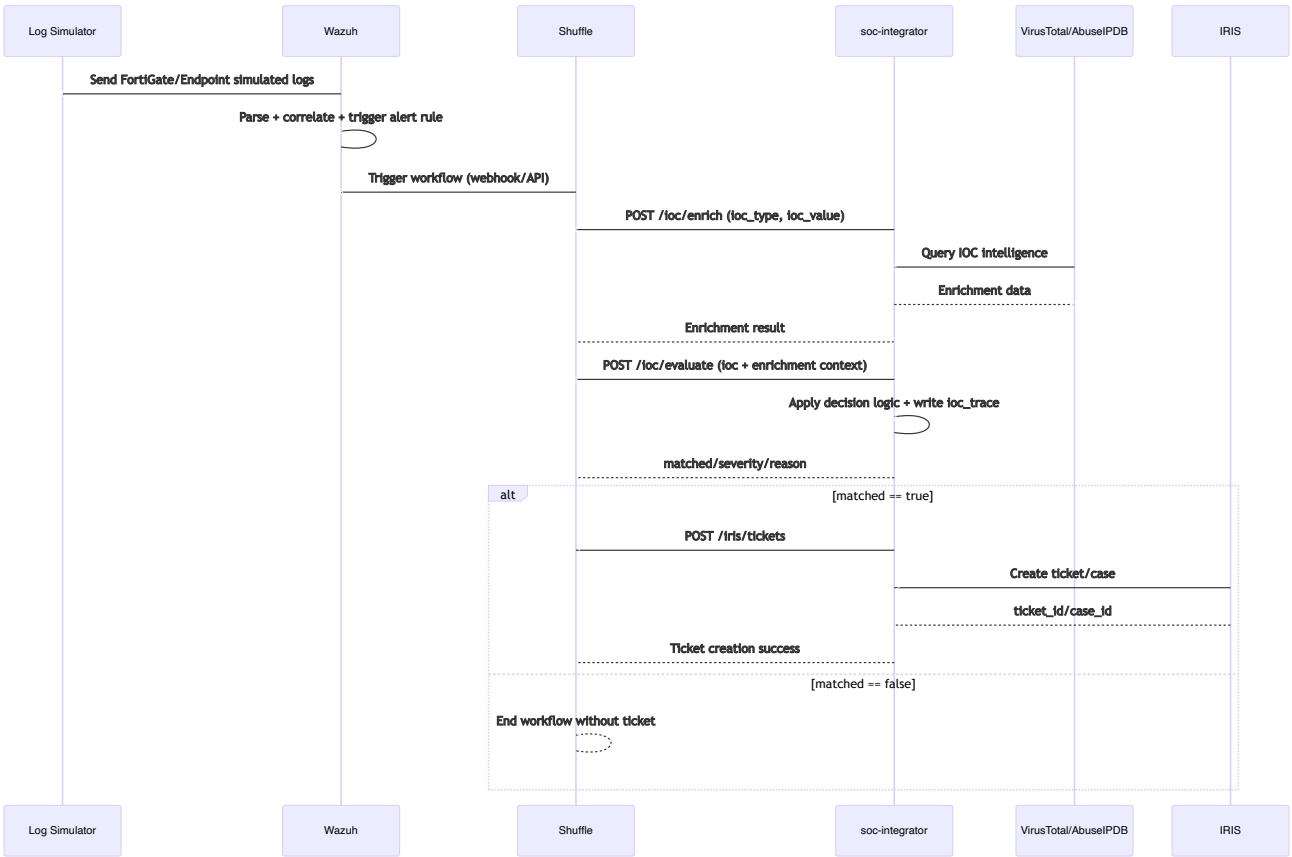
```
{
  "success": true,
  "ticket_id": 53,
  "case_id": 53,
  "status": "open"
}
```

9) Why IOC Was Added to SOC Integrator

- To centralize threat-intelligence logic in one API layer instead of duplicating enrichment/evaluation rules across Shuffle workflows and other services.
- To provide a consistent decision contract (`enrich` for context, `evaluate` for action/verdict) that downstream automation can trust.
- To improve traceability by storing IOC checks and decisions in `soc-integrator` history for audit, tuning, and UAT evidence.
- To simplify integrations with multiple intelligence providers (VirusTotal, AbuseIPDB, and future sources) behind one internal interface.

- To reduce workflow complexity in Shuffle so playbooks focus on orchestration (branching, ticketing, notifications) while IOC decisioning stays in backend logic.

10) Sequence Diagram (MermaidJS)



11) SOC Integrator API Inventory

Group	Method	Endpoint	Notes
Core	GET	`/health`	Service health and target configuration
Core	POST	`/ingest/wazuh-alert`	Normalize inbound Wazuh alert payload
Core	POST	`/action/create-incident`	Create PagerDuty incident
Core	POST	`/action/trigger-shuffle`	Trigger Shuffle workflow execution
Core	POST	`/action/create-iris-case`	Create IRIS case (legacy action endpoint)
IRIS	POST	`/iris/tickets`	Create IRIS ticket/case via soc-integrator
IRIS	GET	`/iris/tickets`	List/query IRIS tickets/cases
IOC	POST	`/ioc/enrich`	IOC enrichment from configured intel sources
IOC	POST	`/ioc/evaluate`	IOC decisioning/verdict
IOC	POST	`/ioc/upload-file`	Upload file to IOC backend (VirusTotal flow)
IOC	GET	`/ioc/analysis/{analysis_id}`	Retrieve IOC analysis status/result

Group	Method	Endpoint	Notes
IOC	POST	`/ioc/evaluate-file`	Evaluate file indicator or uploaded sample
IOC	GET	`/ioc/history`	Retrieve stored IOC trace history
Shuffle	GET	`/shuffle/health`	Shuffle service reachability check
Shuffle	GET	`/shuffle/auth-test`	Validate Shuffle API key access
Shuffle	POST	`/shuffle/login`	Login against Shuffle API
Shuffle	POST	`/shuffle/generate-apikey`	Generate Shuffle API key from credentials
Shuffle	GET	`/shuffle/workflows`	List workflows
Shuffle	GET	`/shuffle/workflows/{workflow_id}`	Get workflow detail
Shuffle	POST	`/shuffle/workflows/{workflow_id}/execute`	Execute specific workflow
Shuffle	GET	`/shuffle/apps`	List installed/available Shuffle apps
Shuffle	POST	`/shuffle/proxy`	Generic proxy request to Shuffle API
Wazuh	GET	`/sync/wazuh-version`	Fetch Wazuh version information
Wazuh	GET	`/wazuh/auth-test`	Validate Wazuh API authentication

Group	Method	Endpoint	Notes
Wazuh	GET	`/wazuh/manager-info`	Manager information
Wazuh	GET	`/wazuh/agents`	List Wazuh agents
Wazuh	GET	`/wazuh/alerts`	Query recent Wazuh alerts
Wazuh	GET	`/wazuh/manager-logs`	Read manager logs
Wazuh	POST	`/wazuh/sync-to-mvp`	Sync Wazuh alerts into MVP pipeline
Wazuh	GET	`/wazuh/auto-sync/status`	Auto-sync loop status
MVP	POST	`/mvp/incidents/ingest`	Ingest incident into MVP flow
MVP	POST	`/mvp/ioc/evaluate`	Evaluate IOC under MVP policy
MVP	POST	`/mvp/vpn/evaluate`	Evaluate VPN event under MVP policy
MVP	GET	`/mvp/config/policies`	Read MVP policy configuration
MVP	PUT	`/mvp/config/policies`	Update MVP policy configuration
MVP	GET	`/mvp/health/dependencies`	Dependency health snapshot

Additional FastAPI-generated endpoints:

- `GET /docs`
- `GET /openapi.json`

Date: March 4, 2026 Project: FoodProject SOC Platform (Wazuh + Shuffle + IRIS-web + SOC Integrator)

Appendix C (C1-C3) Production Log Mapping Update

This update documents production log sources and required fields for Appendix C detections implemented in `soc-integrator`.

C1. Impossible Travel Detection

- Use case:
  - `C1-01` Impossible Travel
- Primary production log sources:
  - VPN authentication success logs
  - Active Directory / Windows authentication success logs (`event_id=4624`)
  - Cloud IdP login success logs (Entra/Okta/Google Workspace)
- Required normalized fields:
  - `asset.user`
  - `network.src_ip`
  - `timestamp`
  - login success indicator (`payload.success=true` or equivalent)
  - geo context (`network.country` and `network.src_lat/src_lon`) or GeoIP enrichment from source IP
- Detection logic summary:
  - Compare consecutive successful logins for same user
  - Calculate distance and travel time
  - Trigger when computed travel speed exceeds threshold (`c1_max_travel_speed_kmph`)

C2. Advanced Credential Abuse & Privilege Misuse

- Use cases:
  - `C2-01` Privileged off-hours login
  - `C2-02` Dormant account activation
  - `C2-03` Service account interactive logon
  - `C2-04` Rapid privilege escalation followed by sensitive access
- Primary production log sources:
  - Windows Security logs (`4624` , `4672` , `4728` , `4732` , `5145`)
  - Linux auth/sudo/PAM logs

    - VPN/IdP authentication logs
- Required normalized fields:
    - `asset.user`, `asset.is_admin`, `asset.is_service`
    - `payload.logon_type`, `payload.event_id`, `payload.action`, `payload.success`
    - `network.src_ip`, `network.dst_host`, `network.dst_port`
    - `timestamp`

C3. Lateral Movement & Internal Reconnaissance

- Use cases:
    - `C3-01` Multi-host authentication success burst
    - `C3-02` SMB/RDP lateral movement burst pattern
    - `C3-03` Admin account accessing many servers rapidly
    - `C3-04` Internal scanning/enumeration burst
- Primary production log sources:
    - Windows authentication and share access logs
    - East-west firewall telemetry
    - IDS/NDR internal movement/scanning alerts
    - Endpoint network telemetry (e.g., Sysmon network events)
- Required normalized fields:
    - `asset.user`, `asset.is_admin`
    - `network.src_ip`, `network.dst_host`, `network.dst_port`
    - login success indicator where applicable
    - `timestamp`

Minimum Windows Event IDs for Initial Rollout

- `4624` Successful logon
- `4672` Special privileges assigned to new logon
- `4728`, `4732` Privileged group membership changes
- `5145` Detailed file share access

Implementation Note

- Simulation scripts exist for Appendix C validation and UAT replay:
    - `scripts/send-wazuh-proposal-appendix-c-events.sh`
- In production, these simulated events are replaced by actual VPN/AD/cloud/endpoint/network telemetry sources listed above.

Appendix C Production Data Onboarding Checklist

Source	Log Path / Channel	Must-Have Fields	Use Cases	Veri (Wa
VPN Gateway (FortiGate/SSL-VPN)	Syslog export from firewall/VPN device	`timestamp`, `user`, `src_ip`, `action/result`, `event_id` (if mapped), `country` (optional)	C1, C2	`full_log:` `full_log:*`
Active Directory / Windows DC	Windows Security Event Log (agent/forwarder)	`event_id`, `timestamp`, `user/account`, `src_ip` (where present), `logon_type`, `success/failure`	C1, C2, C3	`rule.id:*` `(data.win.` `OR full_lo`
Cloud IdP (Entra/Okta/Google)	API export / SIEM connector -> syslog/json	`user`, `src_ip`, `event_time`, `outcome`, `geo.country` (if available), `app/service`	C1, C2	`full_log:` `full_log:*` `full_log:*`
Windows Endpoints/Servers	Wazuh agent + Sysmon/Security logs	`event_id`, `user`, `src_ip`, `dst_host`, `dst_port`, `process/action`	C2, C3	`full_log:` `AND rule.i`
Linux Servers	auth.log / secure / sudo / sshd	`timestamp`, `user`, `src_ip`, `action`, `success`	C2, C3	`full_log:` `full_log:*`
East-West Firewall	Internal traffic logs (allow/deny/flow)	`src_ip`, `dst_ip/dst_host`, `dst_port`, `action`, `timestamp`	C3	`full_log:` `full_log:*`

Source	Log Path / Channel	Must-Have Fields	Use Cases	Verif (Wa
IDS/NDR	IDS alerts / network detection logs	`src_ip`, `dst_ip/dst_host`, `dst_port`, `signature/category`, `timestamp`	C3	`full_log:` `full_log:*` `full_log:*`

Acceptance Checklist (Per Source)

- Parsing/decoder is stable (no malformed key fields in sampled logs)
- Required fields are present and normalized into event model used by `soc-integrator`
- Timestamp format is valid ISO-8601 after normalization
- Sample events can be found in `wazuh-alerts-*` within expected ingestion latency
- At least one C-use-case evaluation run confirms source contributes to detection context

---

Date: March 4, 2026 Project: FoodProject SOC Platform (Wazuh + Shuffle + IRIS-web + SOC Integrator)

Git Diff Progress Summary (Base: `0de071e` -> Head: `5e215c0`)

Diff Snapshot

- Base commit: `0de071e7c9327c8c9135f0c15cf80c31c9b2e59a`
- Head commit: `5e215c0`
- Net change: **40 files changed, 6083 insertions(+), 108 deletions(-)**

Major Progress Areas

1. SOC Integrator Expansion

- Added full admin UI stack:
  - `soc-integrator/app/ui/index.html`
  - `soc-integrator/app/ui/assets/app.js`
  - `soc-integrator/app/ui/assets/styles.css`
- Added Appendix C correlation/detection service:

    - `soc-integrator/app/services/c_detection_service.py`
- Extended API/data layers for monitoring, simulation control, IOC, and detection history:
    - `soc-integrator/app/main.py`
    - `soc-integrator/app/models.py`
    - `soc-integrator/app/repositories/mvp_repo.py`
- Added GeoIP adapter integration:
    - `soc-integrator/app/adapters/geoip.py`

2. Wazuh Simulation and Dashboard Delivery

- Added Appendix-specific event generators:
    - `scripts/send-wazuh-proposal-appendix-b-events.sh`
    - `scripts/send-wazuh-proposal-appendix-c-events.sh`
- Added dashboard artifacts/import pipeline:
    - `scripts/events/*.ndjson`
    - `scripts/import-wazuh-dashboard.sh`
- Added Wazuh custom decoder/rules artifacts for proposal scenarios:
    - `wazuh-docker/single-node/config/wazuh_cluster/local_decoder.xml`
    - `wazuh-docker/single-node/config/wazuh_cluster/local_rules.xml`
    - `wazuh-docker/single-node/config/wazuh_cluster/rules/soc-*.xml`

Wazuh Custom Rules Added (Current Active Set)

Active custom rules are currently defined in:

- `wazuh-docker/single-node/config/wazuh_cluster/local_rules.xml`

Rule groups/ranges implemented:

- Base and appendix classifiers:
    - `100200`: base marker for synthetic SOC events (`soc_mvp_test=true`)
    - `100210`: Appendix A classifier
    - `100220`: Appendix B classifier
    - `100230`: Appendix C classifier
- Appendix A:
    - `A1` IOC/DNS: `100301-100302`
    - `A2` FortiGate firewall/IPS/IDS: `100311-100320`
    - `A3` VPN anomalies: `100331-100335`
    - `A4` Windows/AD behaviors: `100341-100364`

- Appendix B:
  - `B1` VMware/vCenter/ESXi: `100401–100403`
  - `B2` Log-loss monitor signal: `100411`
  - `B3` Sysmon-focused detections: `100421–100426`
- Appendix C (implemented scope C1-C3):
  - `C1` Impossible travel: `100501`
  - `C2` Credential abuse/privilege misuse: `100511–100514`
  - `C3` Lateral movement/internal recon: `100521–100524`

Operational note:

- Split rule files under `wazuh_cluster/rules/soc-*.xml` exist as staging artifacts in this workspace; active detection content is loaded from `local_rules.xml`.

3. Operations and Runtime Hardening

- Updated orchestration and runtime configuration:
  - `run-combined-stack.sh`
  - `compose-overrides/soc-integrator.yml`
  - `soc-integrator/Dockerfile`
  - `soc-integrator/.env.example`

Documentation Progress Included in This Range

- Added/updated proposal revision document:
  - `Security Detection & Threat Intelligence Enhancement Proposal-revise.md`
- Expanded progress log coverage in this file (`progress-update.md`) including:
  - Appendix C (C1-C3) production log mapping
  - Production data onboarding checklist
  - Acceptance criteria for source onboarding and validation