1 ROLE RESPONSIBILITIES

Standing up and tuning the SIEM. Calling out alerts. Confirming scope across every host, not just the one under attack.

2 ORDER OF OPERATIONS

Round start: Confirm Security Onion (and Wazuh, if deployed) is up, sensors capturing, agents reporting.
Tuning pass: Reduce known-noisy alerts before the round gets busy.
Ongoing: Work the alert queue continuously.
On a reported foothold: Search every host for the same account, IOC, or timestamp. Report scope to the Captain.
After eviction: Keep watching for it to return.

3 SECURITY ONION: SETUP CHECKS

- `so-status`: every component running.
- `so-allow`: whitelist your IP if the web UI refuses connection.
- Monitor interface (not management) is the one seeing traffic; packet counts climbing.
- `so-user`: manage additional analyst accounts on the box itself.

4 HUNT: KEY QUERIES

- *Lateral movement:* `event.dataset:conn AND source.ip:"[host]" AND destination.ip:10.0.0.0/8`
- *Exfiltration:* `event.dataset:conn AND source.ip:"[host]"`, sorted by `network.bytes` descending.
- *Other hosts:* `event.module:suricata AND rule.name:"[signature]"`, all sensors.
- *DNS tunneling suspicion:* `event.dataset:dns` sorted by query length or entropy, looking for unusually long subdomains.
- *Cleartext creds:* `event.dataset:http AND http.request.method:POST`, review bodies for login forms over plain HTTP.
- Save a confirmed query as a saved search.

5 HUNT: RED TEAM TRADECRAFT

- *Automated push (Ansible or similar):* a burst of SSH (22/tcp) or WinRM (5985/5986/tcp) from one source to many hosts within seconds is a playbook run. `event.dataset:conn AND destination.port:(22 OR 5985 OR 5986)`, grouped by source, sorted by distinct destination count.
- *Impacket-style tooling:* short SMB sessions opening the `svcctl` or `winreg` named pipe then closing within seconds, across multiple hosts, one source. Needs Zeek's DCE-RPC analyzer: `event.dataset:dce_rpc AND rpc.endpoint:(svcctl OR winreg)`.
- *Fan-out to admin shares:* SMB connections to `C$/ADMIN$` from a source that is not a known management host.
- A source hitting both the fan-out query and the admin-share query narrows red team's launch point fast.

6 DATA SOURCES

- *Suricata:* signature-based alerts.
- *Zeek conn.log:* every connection seen, matched or not. Where movement/exfil show up first.
- *Zeek dns.log:* every DNS query and response; good for tunneling and beaconing.
- *Zeek http.log / ssl.log:* request/response metadata and certificate details, even without full payload capture.
- *Zeek files.log:* files extracted from traffic, useful for catching a dropped payload in transit.
- *Forwarded host logs:* account/process detail the network alone cannot see.

7 WAZUH, IF DEPLOYED

- File Integrity Monitoring: fastest confirmation a specific file changed.
- Vulnerability detection: known-CVE software already installed.
- Scope query: `rule.groups:"syscheck" AND agent.name:"[host]"`.
- Custom rule example, extending a built-in FIM rule:

```
<rule id="100010" level="7">
<if_sid>550</if_sid>
<field name="file">nginx.conf</field>
<description>config
modified</description>
</rule>
```

8 WRITING A SURICATA RULE

```
alert tcp any any -> $HOME_NET 4444
(msg:"Possible
reverse shell"; sid:1000001; rev:1;)
```

Test against known-good and known-bad traffic before enabling live. A rule too broad fires constantly; too narrow misses variants.

9 CLI HELPERS

- `zeek-cut`: pull specific fields out of raw Zeek TSV logs on the command line.
- `jq`: parse and filter JSON-formatted logs quickly if exporting from the SIEM.
- `tcpdump -nr [pcap] 'host [ip]'`: spot-check a specific host's traffic straight from a capture, no SIEM needed.

10 ALERT TUNING

- Suppress or threshold a rule that fires on expected, benign traffic rather than disabling it outright.
- Keep a short list of intentionally suppressed rules, with the reason, so nobody re-enables noise blind.
- Alert fatigue: tune, do not just silence. A dropped real alert costs more than a noisy one.

11 WAZUH ACTIVE RESPONSE

- Active response scripts can auto-react to a rule firing (e.g., blocking an IP), configured in `ossec.conf` on the manager.
- Start with logging-only responses during a competition; an automated action that misfires can do more damage than the thing it was meant to stop.
- `/var/ossec/logs/active-responses.log`: audit what active response has actually done.

12 WRITING A WAZUH DECODER

- A decoder tells Wazuh how to parse a log line into fields a rule can then match against.
- Needed when a custom or unusual log format is not recognized out of the box.
- Test with `/var/ossec/bin/wazuh-logtest`, feeding it a sample line before trusting the decoder live.

13 BUILDING A DASHBOARD PANEL

- Run and save the search first, then build a visualization from it (bar chart of matches over time is usually the most useful default).
- Add the visualization to a dashboard, and set a sensible default time range: last 24 hours for a daily check, last hour during an active incident.
- Name panels descriptively; "Panel 3" is useless to a teammate six hours into a round.

14 LOG RETENTION

- Full packet capture fills disk fastest; decide the retention window before the round, not when the disk is already full.
- Elasticsearch index lifecycle settings (or manual index deletion) control how long searchable data sticks around.
- If disk fills mid-round, older indices are usually safer to drop than losing new ingestion; confirm this policy with the team ahead of time.

15 CROSS-SOURCE CORRELATION

- A single alert is a clue. The same timestamp appearing in Suricata, Zeek, and a host log together is a much stronger case.
- When reporting scope to the Captain, cite which sources agree, not just the first alert that fired.
- Keep a short list of IOCs (IPs, hashes, filenames) confirmed bad this round; check new alerts against it fast.

16 COMMON PITFALLS

- Storage vs. retention: decide what to keep before the disk fills.
- Agent drift: a disconnected Wazuh agent stops reporting silently; check the Agents page, not just alert vol-

ume.

- Treating "no alerts" as "nothing happening." Confirm sensors are actually capturing, not just quiet.

17 PATTERN LIBRARY

- A sudden spike in DNS queries to one domain, especially with long or random-looking subdomains.
- Beaconing: a connection recurring at a suspiciously regular interval, to the same destination.
- A host generating alerts it has never generated before, even low-severity ones.
- An agent that stops checking in right around the same time a host starts behaving oddly elsewhere.