CSE 115A – Introduction to Software Engineering
Release Summary

Product Name: Intrusion Detection Tool
Team name: SlugShield
Date: December 1, 2025

Key user stories & acceptance criteria:

US1 — Local System Execution

As a user, I want the IDS to run locally so data stays private.

Sprint 1 - Report

Acceptance Criteria

- User can run the backend locally with no external dependencies.
- All required environments and dependencies are documented in README.
- System captures real traffic on the local machine.
- System does not upload data to any external server.

US2 — Real-Time Traffic Monitoring

As a user, I want the IDS to monitor network traffic in real time.

Sprint 1 - Report

Acceptance Criteria

- IDS continuously captures packets (ICMP, ARP, SSH, port scan related).
- Alerts are triggered when thresholds are exceeded.
- Monitoring loop runs without crashing for at least 30 minutes.
- Real-time detection supports ICMP flood, ARP spoofing, SSH brute-force, and port scans.

US3 — Suspicious Activity Logging & Alerts

As a user, I want the IDS to log or print alerts when suspicious activity is detected.

Acceptance Criteria

- Alerts include timestamp, type of attack, and summary.
- Alerts appear in logs and the dashboard "Recent Alerts" section.
- Logs persist between restarts.

US4 — Access Local Dashboard

As a user, I want to access the dashboard through my browser locally.

Acceptance Criteria

- Dashboard is reachable at configured host/port (default: localhost).
- Frontend loads without errors.
- Backend and frontend communication confirmed via test endpoint.
- No third-party software is required for visualization.

US5 — Overview of System Health

As a user, I want an overview showing overall system health at a glance.

Acceptance Criteria

- System displays status: **OK** if no active alerts, **ALERT** if ≥1 alert is active.
- Overview dynamically updates without refresh.
- Timestamp of last check shown.
- Active alert count updates in real time.

- Alerts drop-down appears if multiple attacks occur.

US6 — View Alerts on Dashboard

As a user, I want to view alerts so I understand my system is safe.

Acceptance Criteria

- Dashboard shows when ICMP flood, SSH brute-force, ARP spoofing, or port scanning occur.
- Alerts update live.
- Recent alerts include explanations (from Sprint 3).
- Partially completed ARP alerts must fully connect backend → frontend.

US7 — Real-Time Traffic Charts & Baselines

As a user, I want charts that compare real-time alerts to normal behavior traffic.

Acceptance Criteria

- Real-time ICMP packets-per-second graph updates continuously.
- Real-time SSH login attempts-per-minute graph updates.
- ARP mapping change graph (changes/min) updates live.
- Baseline curves are displayed for all attack types.
- Graphs update without manual refresh.
- Normal vs suspicious traffic visualized clearly.

US8 — Clear Explanations for Alerts

As a user, I want clear explanations for each alert type.

Acceptance Criteria

- Dropdown under “Recent Alerts” expands to show explanation text.
- All four explanations exist: ICMP flood, ARP spoofing, port scanning, SSH brute-force.
- Explanations are concise (<150 words) and non-technical.
- Explanations render correctly on desktop browser.

US9 — Adjust Alert Thresholds

As a user, I want to adjust thresholds for suspicious alerts.

Acceptance Criteria

- Threshold sliders/panels exist for all detectors.
- Changing thresholds updates backend logic instantly.
- Thresholds persist across sessions.
- Sliders have valid ranges and prevent invalid input.
- UI polishing for sliders remains incomplete (Sprint 4).

US10 — Pause & Resume Monitoring

As a user, I want to pause and resume monitoring from the dashboard.

Acceptance Criteria

- “Pause” button halts real-time monitoring within 1 second.
- “Resume” restarts monitoring without restart or crash.
- UI clearly shows paused vs active state.
- No alerts are generated while paused.

US11 — Receive Email Notifications

As a user, I want to receive email notifications whenever an alert is detected.

Acceptance Criteria

- User can input email address in dashboard.
- Email form is intuitive (Sprint 4 improvement request).
- Emails are sent for all four detector types.
- Email includes summary and timestamp.
- Formatting for some alerts remains incomplete (Sprint 4).
- User can toggle notifications (future planned feature per Sprint 4).

US12 — Accurate & Polished UI Components

(A cross-cutting story from recurring issues in Sprint 3–4.)

Acceptance Criteria

- Dashboard background, labels, and typography are clear and consistent.
- Threshold sliders are fully styled and easy to use.
- Recent alerts show accurate timestamps.
- No overlapping UI elements on different screen sizes.
- All dropdowns animate smoothly.

Known Problems:

- Bugs
  - The alerts in recent alert are not in order
  - Packets analyzed section does not update for arp spoof and port scanning detectors
- Missing functionality
  - Missing pause and resume option of detector
- Design Shortcuts
  - Baseline values in the simulation attack itself is hardcoded, no actual data
  - Values within config.yaml are also hardcoded when started

- Edge Cases
    - Ipv6 not supported
    - Packets can drop during high traffic(5000+) leading to inaccurate graphs
    - User opening dashboard in multiple tabs can flood backend with redundant connections

Product Backlog: (High-Priority Items for Follow-On Project)

1.1 Intrusion Prevention Automation (NEW)

As a user, I want the system to automatically block suspicious IPs so attacks are not only detected but prevented.
(Proposed in Sprint 4 as a next-phase direction)

Why High Priority

- Extends IDS into IPS (major functional upgrade).
- Uses existing detection data.

1.2 Machine-Learning–Based Anomaly Detection (NEW)

As a user, I want the system to detect unusual network behavior using ML so I can catch new or evolving threats.
(Listed as "future release" candidate)

Why High Priority

- Moves system beyond rule-based methods.
- Complements threshold adjustments already implemented.

1.3 Multi-Device Monitoring & Alert Sharing (NEW)

As a user, I want to monitor multiple devices and share alerts across my network so I can protect an entire home or lab environment.

Why High Priority

- Major expansion of system scope.
- Ideal for a Version 2.0 release.

1.4 Improved Real-Time Visualizations (NEXT ITERATION)

As a user, I want richer real-time visualization tools (live packet flow graphs, heatmaps, interactive charts) so I can interpret attack patterns more effectively.
(Proposed as future enhancements in Sprint 4)

Why High Priority

- Addresses repeated dashboard clarity issues across multiple sprints.

1.5 Modular Notification Settings

As a user, I want to choose which alerts trigger email notifications so I only receive important information.

Sprint 4 Report

Why High Priority

- Builds on existing, fully implemented email notifications.
- Adds meaningful user control.

1.6 Deployment-Ready Packaging (Docker + Edge Device Support)

As a user, I want a deployment-ready version of the IDS that runs on Docker or a Raspberry Pi so the system is portable.

Why High Priority

- Requested as a next-phase architectural improvement.
- Enables real-world deployment.

1.7 External Logging Integration (ELK, Splunk, Syslog)

As a user, I want alerts forwarded to external monitoring tools so I can maintain long-term logs and perform deeper analysis.

Why High Priority

- Common security operations requirement.
- Natural extension beyond dashboard-only logging.

1.8 Complete UI/UX Overhaul and Modernization

As a user, I want a more modern, minimalistic dashboard theme so the system is easier to use.

Why High Priority

- UI/UX weaknesses appear across *all* sprint reports (backlogged or unfinished).
- Several UI enhancements were started but not fully completed.

1.9 Enhanced Time Accuracy & Historical Views

As a user, I want precise timestamps and a way to review historical alert trends so I can understand long-term patterns.

Why High Priority

- Time inaccuracies were explicitly flagged as problematic.
- Historical views are a natural extension of “recent alerts.”

2. High-Priority Bug Fixes & Technical Debt

These items are compiled from:

- Repeated sprint issues
- Incomplete work
- User-story features marked "Not Completed"
- Known gaps in UI/UX and backend linking

2.1 Fix inaccurate timestamp reporting in “Recent Alerts”
Problem: Sprint 4 specifically identifies inaccurate times.
Impact: Misleads users when reviewing attack events.

2.2 Fully polish threshold slider UI

Problem: Sliders technically work but UI/UX incomplete.
Impact: Reduces user control clarity.

2.3 Improve email formatting and reliability

Problem: Formatting is inconsistent across alert types.
Impact: Unprofessional and unclear notifications.

2.4 Fix incomplete ARP → Frontend integration

Problem: Backend ARP detection works, but UI integration is unfinished.
Impact: User may not realize ARP spoofing is happening.

2.5 UI labeling inconsistencies (Sprint 4 aesthetic improvements unfinished)

Problem: Missing or unclear labels.
Impact: Users struggle to interpret graphs and metrics.

2.6 Clean up long, unclear backend files

Problem: The team noted that some files were “super long and time-consuming to understand.”
Impact: Reduces developer velocity.

2.7 Optimize performance for low-resource systems

Matches Sprint 4 high-priority story US12: keep CPU usage low.

Problem: System may slow down under long monitoring sessions.
Impact: Limits deployment on older or mobile hardware.

2.8 Reduce false positives (Sprint 4 Issue)

Matches Sprint 4 high-priority story US13 about improving accuracy.

Problem: Detectors are rule-based with fixed thresholds.
Impact: Users may distrust warnings if too frequent.

3. Optional but Valuable Medium-Priority Backlog Items

These are not critical but add quality, stability, and usability.

- Improve documentation structure and completeness (noted repeatedly across sprints).
- Create demo mode with fake traffic for easier testing.
- Add dark mode/light mode toggle.
- Add profile/settings page for user preferences.
- Create automated end-to-end test suite.

Summary: High-Priority Backlog for Follow-On Project

Top User Stories to Implement Next

1. Intrusion prevention (auto IP blocking)
2. ML-based anomaly detection
3. Multi-device monitoring
4. Advanced visualization tools
5. Modular alert notifications
6. Containerized / Pi deployment
7. External logging integration

8. Complete UI/UX redesign

9. Accurate timestamps + historical view