Sprint 1 Plan

Product Name: Intrusion Detection Tool
Team Name: SlugShield
Sprint Completion Date: 10/21/25
Revision #: 1.0 **Revision Date:** 10/21/25

Goal

Deliver a minimal, privacy-preserving intrusion detection prototype that runs locally and logs suspicious network activity, with foundational project infrastructure in place.

Task Listing, Organized by User Story

User Story 1

“As a user, I want the system to run locally on my laptop or computer so my data stays private and never leaves my device.”[8 story points]

Tasks:

- Design local-only architecture (no cloud dependencies) - 4 hours
- Create initial FastAPI app skeleton for local run – 5 hours
- Implement local config loader (.env integration) – 3 hours
- Test run locally on Linux and macOS – 4 hours
- **Total for User Story 1:** 16 hours

User Story 2

“As a user, I want the IDS to monitor network traffic in real time so it can detect suspicious activities as the system is running.”[13 story points]

Tasks:

- Research and prototype packet capture with pcapy-ng - 6 hours
- Implement basic capture loop (packets → console) – 4 hours
- Develop simple rule-based detection placeholder – 5 hours
- Integrate capture with FastAPI backend – 4 hours
- **Total for User Story 2:** 19 hours

User Story 3

“As a user, I want to see the IDS printing to the console log or writing to a log output when suspicious activity is detected so I know the IDS is working.”[3 story points]

Tasks based off User Stories:

- Implement logging module (console + file) – 4 hours
- Format alerts with timestamps and metadata – 3 hours
- Test log output under mock traffic – 3 hours
- **Total for User Story 3:** 10 hours

Task Breakdown:

- Set up backend environment for project
- ICMP Flooding detection
- Implement port scanning detection
- Implement ssh login brute force detection
- Test with attack simulations
- Adjust threshold as needed
- ARP Spoofing

Spikes (Exploratory Work)

- Research pcapy-ng for lightweight packet capture – 4 hours
- Test libpcap permission requirements on Linux/macOS – 3 hours
- Explore FastAPI WebSocket patterns for real-time alerts – 4 hours
- **Total for Spikes:** 11 hours

Infrastructure Tasks

- Set up GitHub repo, README, and project structure – 3 h
- Configure Docker Compose (edge-agent + web-UI) – 5 h
- Establish .env configuration and base variables – 3 h
- Draft architecture and data flow diagrams – 4 h
- **Total for Infrastructure:** 15 hours

Team Roles

Team Member	Roles
Asaveri	Scrum Master/Developer

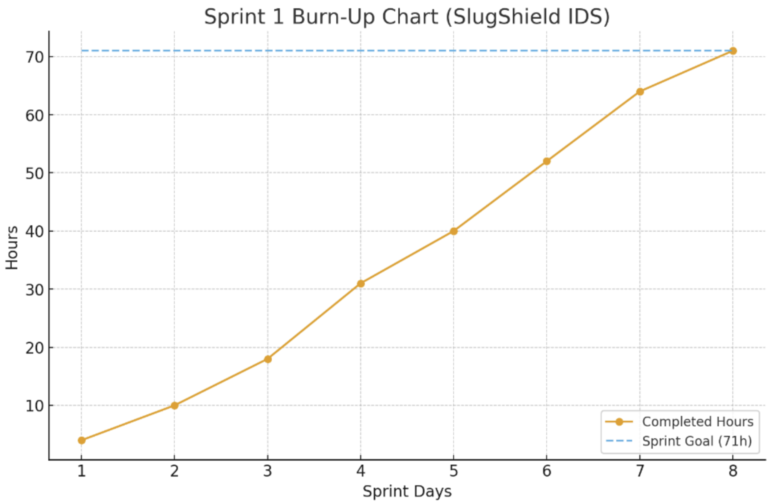
Andy	Researcher/Developer
Anish	Product Owner/Developer
Jace	Researcher/Developer

Initial Task Assignment

Team Member	User Story	Initial Task
Andy	Infrastructure Setup	Docker Compose, ICMP flooding detection, Packet capture & integration
Anish	User Story 2	Github + ARP Spoofing
Jace	User Story 3	Logging output & UI console
Asaveri	Spikes	pcapy-ng research & permissions testing + ssh login brute force

Initial Burnup Chart

Sprint 1 – Local IDS Project
Plot Total Ideal Hours (71 h) vs. Completed Hours per Day



Initial Scrum Board

Physical board labeled with sprint number and project name.
Columns: User Stories, Tasks Not Started, Tasks In Progress, Tasks Completed.
Each task should be in the same row as its corresponding user story.

Scrum Times

Day	Time	TA/Tutor Visit
Tuesday	11:30-12:30	Yes
Friday	6:30-7:30	No
Tuesday	11:30-12:30	Yes

Updated Release Plan

None yet – first release target defined after Sprint 2.

Sprint 2 Plan

Product Name: Intrusion Detection Tool
Team Name: SlugShield
Sprint Completion Date:10/23/25
Revision #: 2.0 **Revision Date:** 10/25/25

Goal

Deliver a simple dashboard connected to the backend we built in sprint 1 that will show possible attacks that were detected.

Roles + Initial Task Assignment

- Andy: Scrum Master, Developer -> User story: 2, 3, 4; Initial Tasks: 2.1, 2.2, 2.3, 2.4, 3.1, 4.1, 4.4
- Asaveri: Developer -> User story: 2, 3, 4; Initial Tasks: 2.41, 3.2, 4.2, 4.5
- Anish: Product Owner, Developer -> User story: 3, 4; Initial Tasks: 3.3, 4.3, 4.6
- Jace: Developer -> User story: 1, 4; Initial Tasks: 1.1,1.2,1.3, 1.4, 4.7

Task Listing, Organized by User Story

User Stories/Tasks:

#1 "As a user, I want to access the dashboard through my browser locally so there isn't a need for third party softwares" [8 story points]

- [1.1] Set up environment and make the files and directories necessary [1 hour]
- [1.2] Implement a simple frontend(just a blank page for testing) [1 hour]
- [1.3] Connects frontend to backend [1 hour]
- [1.4] Dashboard is reachable(host and port configuration) [1 hour]

#2 "As a user, I want a main overview that shows overall systems health so I can get status at a glance" [5 story points]

- [2.1] Implement a little summary on top of the page showing overall status– OK and ALERT; if nothing is detected then status should be OK, if something is detected then it should change dynamically without any user input [1 hour]
- [2.2] Implement timestamp for last time it was checked [1 hour]
- [2.3] Implement function that shows active alert == 0 if overall system is OK [1 hour]
- [2.4] Implement function that shows what was detected alongside alert == x, where x is the number of times it was detected [2 hours]
  - [2.41] Adding to this, if multiple attacks happened, then it should be a drop down menu of the multiple attacks with the alert [1 hour]

#3 "As a user, I want to view alerts on a simple dashboard so I understand my system is safe." [8 story points]

- [3.1] If icmp flood is detected, then dashboard should update that icmp was detected [2 hours]
- [3.2] If ssh bruteforce login is detected, then dashboard should update that ssh bruteforce was detected [2 hours]
- [3.3] If arp spoofing is detected, then dashboard should update that arp spoofing was detected [2 hours]

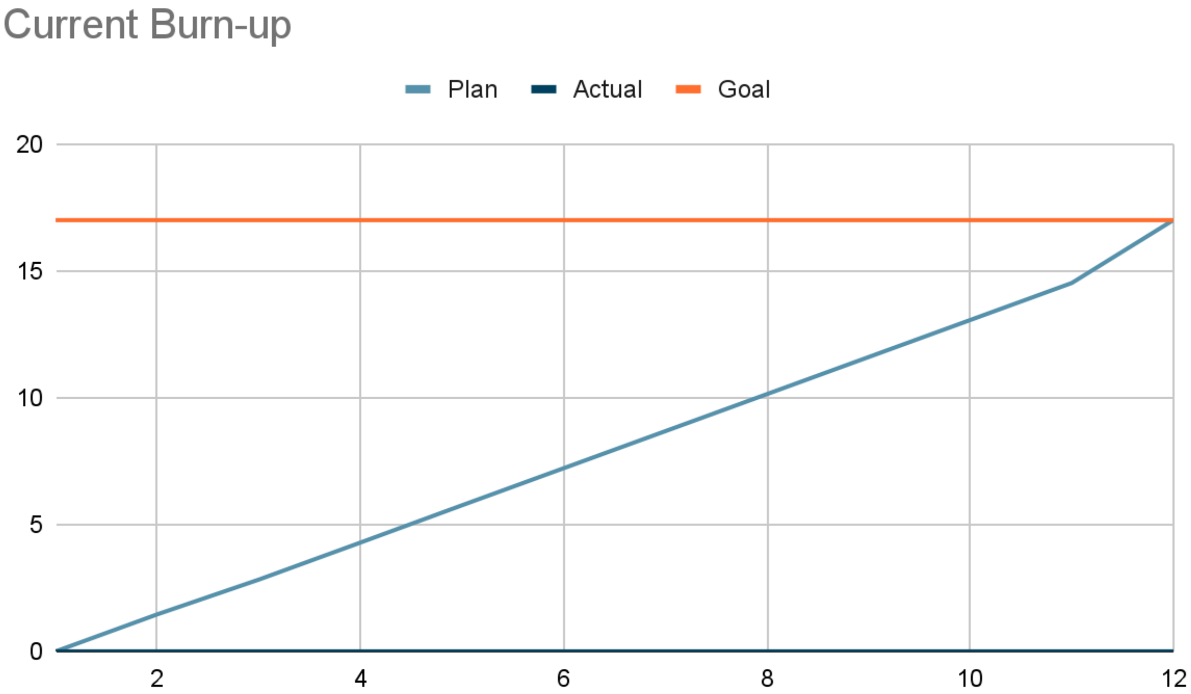
#4 "As a user, I want to see real time alerts that were detected along with simple charts comparing the real time alerts that were detected to normal behavior traffic so I understand what is happening and how severe it is."**(metrics can be adjusted, doesn't have to be per minute or per second– whatever you think will work best)** [8 story points]

- [4.1] A chart displaying real time traffic metrics for icmp flooding(icmp packets per second) [1 hours]

- [4.2] A chart displaying real time traffic metrics for ssh brute force login(login attempts per minute) [1 hours]
- [4.3] A chart displaying real time traffic metrics for arp spoofing(changes per minute) [1.5 hours]
- [4.4] A baseline curve representing normal behavior for icmp traffic(average icmp packets per second) [1 hours]
- [4.5] A baseline curve representing normal behavior ssh login traffic(average login attempts per minute) [1 hours]
- [4.6] A baseline curve representing normal behavior for MAC mapping over time(average MAC addresses changes per minute) [1 hours]
- [4.7] Charts should be updating live as new packets and such are analyzed [1 hours]

Initial Burnup Chart

Sprint 2 – Local IDS Project
Plot Total Ideal Hours (17.5 h) vs. Completed Hours per Day.
Chart posted in lab or Scrum area.



Initial Scrum Board

User Story	To-Do	In-Progress	Done
As a user, I want to access the dashboard through my browser locally so there isn't a need for third party softwares	Set up environment and make the files and directories necessary Implement a simple frontend(just a blank page for testing) Connects frontend to backend Dashboard is reachable(host and port configuration)		
As a user, I want a main overview that shows overall systems health so I can get status at a glance	Implement a little summary on top of the page showing overall status– OK and ALERT; if nothing is detected then status should be OK, if something is detected then it should change dynamically without any user input Implement timestamp for last time it was checked Implement function that shows active alert == 0 if overall system is OK Implement function that shows what was		

	detected alongside alert == x, where x is the number of times it was detected Adding to this, if multiple attacks happened, then it should be a drop down menu of the multiple attacks with the alert		
As a user, I want to view alerts on a simple dashboard so I understand my system is safe.	If icmp flood is detected, then dashboard should update that icmp was detected If ssh bruteforce login is detected, then dashboard should update that ssh bruteforce was detected If arp spoofing is detected, then dashboard should update that arp spoofing was detected		
As a user, I want to see real time alerts that were detected along with simple charts comparing the real time alerts that were detected to normal behavior traffic so I understand what is happening and how severe it is.**(metrics**	A chart displaying real time traffic metrics for icmp flooding(icmp packets per second) A chart displaying real time traffic metrics for ssh brute force login(login attempts per minute)		

can be adjusted, doesn’t have to be per minute or per second– whatever you think will work best)	A chart displaying real time traffic metrics for arp spoofing(changes per minute) A baseline curve representing normal behavior for icmp traffic(average icmp packets per second) A baseline curve representing normal behavior ssh login traffic(average login attempts per minute) A baseline curve representing normal behavior for MAC mapping over time(average MAC addresses changes per minute) Charts should be updating live as new packets and such are analyzed		

Scrum Times

Day	Time	TA/Tutor Visit
Tuesday	11:30-12:30	Yes
Friday	6:30-7:30	No

Saturday	11:30-12:30	No

Sprint 3 Plan

Product Name: Intrusion Detection Tool
Team Name: SlugShield
Sprint Completion Date:11/11/25
Revision #: 2.0 **Revision Date:** 11/11/25

Goal

Deliver a more comprehensive dashboard with user features and more detailed descriptions for each security feature as well as a fully functional email notification feature.

Roles + Initial Task Assignment

- Andy: Developer -> User story: 1, 2, 3, 4 ; Initial Tasks: 1.2, 2.2, 3.1, 4.1
- Asaveri: Developer -> User story: 1, 2, 4; Initial Tasks: 1.5, 2.5, 4.4
- Anish: Product Owner, Developer -> User story: 1, 2, 4 ; Initial Tasks: 1.3, 2.2, 4.2
- Jace: Scrum Master, Developer -> User story: 1, 2, 4; Initial Tasks: 1.4, 2.4, 4.3

Task Listing, Organized by User Story

User Stories/Tasks:

#1 "As a user, I want clear explanations for each alert so I can understand what triggered it and what it means." [8 story points]

- [1.1] Implement the descriptions as a drop down menu under "Recent alerts"[2 hours]
- [1.2] Write description for icmp flooding[1 hour]
- [1.3] Write description for arp spoofing[1 hour]
- [1.4] Write description for port scanning[1 hour]
- [1.5] Write description for ssh bruteforce detection[1 hour]

#2 "As a user, I want to adjust the threshold for how frequently I receive suspicious alerts so I can manage notification frequency.**"** [5 story points]

- [2.1] Create threshold adjustment to the side allowing user to control threshold for each detector mechanism[2 hour]
- [2.2] Allow users to control threshold for icmp flooding detection[1 hour]

- [2.3] Allow users to control threshold for arp spoofing detection[1 hour]
- [2.4] Allow users to control threshold for port scanning detection[1 hour]
- [2.5] Allow users to control threshold for ssh bruteforce detection [1 hour]

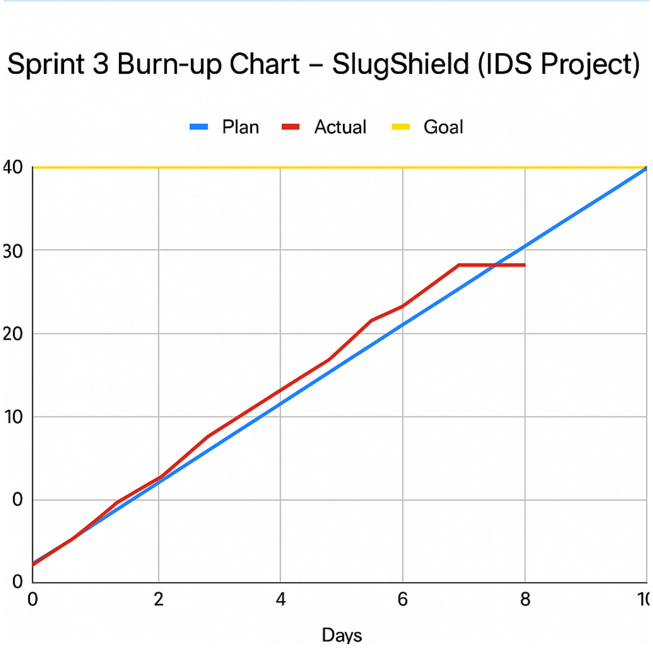
#3 “As a user, I want to pause and resume monitoring from the dashboard so I can control when analysis runs”[6 story points]

- [3.1] Create a button on the side allowing the user to pause monitoring of system whenever they choose[1 hour]
- [3.2] Create a button on the side allowing the user to resume monitoring of system whenever they choose[1 hour]

#4“As a user, I want to receive email notifications whenever an alert is detected so I can be notified when I am not near the computer**”**[8 story points]

- [4.1] Implement email notifications for when icmp flooding is detected alongside a summary of the attack[2 hours]
- [4.2] Implement email notifications for when arp flooding is detected alongside a summary of the attack[2 hours]
- [4.3] Implement email notifications for when port scanning is detected alongside a summary of the attack[2 hours]
- [4.4] Implement email notifications for when ssh bruteforce detection is detected alongside a summary of the attack[2 hours]

Initial Burnup Chart



Initial Scrum Board

User Story	To-Do	In-Progress	Done
As a user, I want clear explanations for each alert so I can understand what triggered it and what it means.	Implement the descriptions as a drop down menu under “Recent alerts” Write description for icmp flooding Write description for arp spoofing Write description for port scanning Write description for ssh bruteforce detection		
As a user, I want to adjust the threshold for how frequently I receive suspicious alerts so I can manage notification frequency.	Create threshold adjustment to the side allowing user to control threshold for each detector mechanism Allow users to control threshold for icmp flooding detection Allow users to control threshold for arp spoofing detection Allow users to control threshold for port scanning detection		

	Allow users to control threshold for ssh bruteforce detection		
As a user, I want to pause and resume monitoring from the dashboard so I can control when analysis runs	Create a button on the side allowing the user to pause monitoring of system whenever they choose Create a button on the side allowing the user to resume monitoring of system whenever they choose		
As a user, I want to receive email notifications whenever an alert is detected so I can be notified when I am not near the computer	Implement email notifications for when icmp flooding is detected alongside a summary of the attack Implement email notifications for when arp flooding is detected alongside a summary of the attack Implement email notifications for when port scanning is detected alongside a summary of the attack Implement email notifications for when ssh bruteforce detection is detected alongside a summary of the attack		

Scrum Times

Day	Time	TA/Tutor Visit
Tuesday	11:30-12:30	Yes
Friday	6:30-7:30	No
Saturday	11:30-12:30	No

Sprint 4 Plan

Product Name: Intrusion Detection Tool
Team Name: SlugShield
Sprint Completion Date:11/23/25
Revision #: 2.0 **Revision Date:** 11/23/25

Goal

Finalize UI improvements and ensure accurate, stable reporting for all detection modules.

Roles + Initial Task Assignment

- Andy: Developer -> User story: 2 ; Initial Tasks: 2.1, 2.3
- Asaveri: Developer -> User story: 1; Initial Tasks: 1.1, 1.2, 1.3, 1.4
- Anish: Scrum Master, Product Owner, Developer -> User story: 2, 3 ; Initial Tasks: 2.2, 3.1, 3.2
- Jace: Developer -> User story: 3 ; Initial Tasks: 3.2

Task Listing, Organized by User Story

User Stories/Tasks:

#1 “As a user, I want a more aesthetic webpage so that it will be easy to follow.” [5 story points]

- [1.1] Change background color[1 hours]

- [1.2] Add labels that clearly depicts [1 hour]
- [1.3] Add labels to webpage[30 mins]
- [1.4] Make the email notification(entering information) button easier to follow[1 hour]

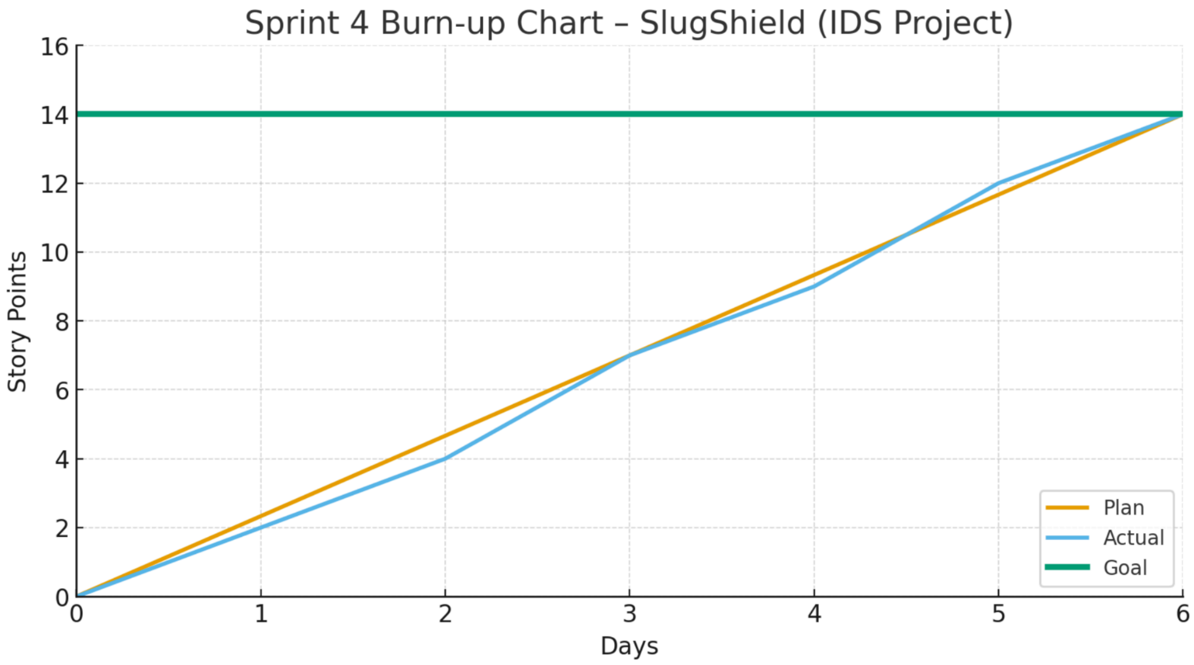
#2 “As a user, I would like to see accurate metrics being reported so I understand the graphs.**”** [5 story points]

- [2.1] Create threshold adjustment to the side allowing user to control threshold for each detector mechanism [2 hours]
- [2.2] Fixing “recent alerts” so that they are more time accurate [1 hour]
- [2.3] Give brief explanations of x and y-axis entail, and why the baseline is not 0. [1 hour]

#3 “As a user, I want to receive email notifications whenever an alert is detected so I can be notified when I am not near the computer**”**[4 story points]

- [3.1] Implement email notifications for when arp flooding is detected alongside a summary of the attack[2 hours]
- [3.2] Implement email notifications for when port scanning is detected alongside a summary of the attack[2 hours]

Initial Burnup Chart



Initial Scrum Board

User Story	To-Do	In-Progress	Done

As a user, I want a more aesthetic webpage so that it will be easy to follow.	Change background color Add labels that clearly depicts Add labels to webpage Make the email notification(entering information) button easier to follow		
As a user, I would like to see accurate metrics being reported so I understand the graphs	Create threshold adjustment to the side allowing user to control threshold for each detector mechanism Fixing “recent alerts” so that they are more time accurate Give brief explanations of x and y-axis entail, and why the baseline is not 0		
As a user, I want to receive email notifications whenever an alert is detected so I can be notified when I am not near the computer	Implement email notifications for when arp flooding is detected alongside a summary of the attack Implement email notifications for when port scanning is detected alongside a summary of the attack		

Scrum Times

Day	Time	TA/Tutor Visit
Monday	9-10 PM	No
Tuesday	11:30-12:30 PM	Yes
Saturday	11:30-12:30 PM	No