Meeting Notes for our Windows Process Protector Senior Design Project – Fall Semester 2024

Meeting 1: September 7, 2024

Summary:

We kicked off our project today. We spent most of the meeting figuring out exactly what we want to do and who's doing what. Bartosz is diving into the backend stuff—think anti-cheat mechanisms and processes protection mechanism development. Harshil is taking on the frontend, building out the web app for real-time monitoring.

We realized it's super important to understand existing anti-cheat systems, so we're both going to dig into some research on current process protection techniques. To keep things smooth, we decided to set up a shared GitHub repo and use some collaboration tools for task management. We also agreed to stay in close touch, responding to each other's messages quickly and keeping updates flowing.

Action Items:

- Set up our shared GitHub repository.
- Start individual research on anti-cheat and process protection methods.
- Plan our next meeting for September 14, 2023.\

Hours Spent Each:

- Harshil Patel: 0.5
- Bartosz Kawalkowski: 0.5

Meeting 2: September 14, 2024

Summary:

We shared what we've found in our research so far. Bartosz talked about the challenges with Windows API & kernel-level driver development and Windows security hurdles. Harshil brought up some cool web technologies for the monitoring interface and is leaning towards using React because it's flexible and robust.

We both brainstormed how the backend and frontend are going to talk to each other. Key points were figuring out the data flow between the process protector and the web interface, and what communication protocols we'll use. We agreed that security and performance are key, so we need to pick technologies that give us both.

Action Items:

- Bartosz will sketch out a preliminary design for the process protection mechanisms.
- Harshil will whip up some wireframes for the web interface.
- Both of us will look into secure communication protocols for the backend and frontend.

Hours Spent Each:

- Harshil Patel: 3
- Bartosz Kawalkowski: 3

Meeting 3: September 21, 2024

Summary:

Bartosz showed off a draft of the process protection module. It was all about how we can prevent unauthorized access and spot attacks like DLL injection and code patching. Harshil shared some wireframes for the web interface, focusing on making it user-friendly and good-looking.

We talked about the challenges of getting the backend to play nice with the web app, especially when it comes to data serialization and real-time updates. We looked into secure communication options like using WebSockets over TLS to keep our data encrypted.

Action Items:

- Start building a proof-of-concept (POC) for the web interface.
- Figure out how to get real-time data flowing between the driver and the web app.
- Set up a development environment to test the low-level backend code.

Hours Spent Each:

- Harshil Patel: 6
- Bartosz Kawalkowski: 6

Meeting 4: September 28, 2024

Summary:

Harshil demoed the initial POC of the web UI today. It can show all the running processes on a user's system with details like PID, status, image path, and more. You can search processes, terminate them, suspend/resume them, and even check out their modules.

Bartosz talked about how we can inject and unload our protection DLL into processes. We realized there are some security risks with DLL injection, so we brainstormed ways to make it safer. We also agreed that adding user authentication to the web interface is a future must-have to prevent unauthorized access.

Action Items:

- Harshil will note down that we need to add user authentication to the web app.
- Bartosz will work on refining the DLL injection method to beef up security.
- We'll both document what we've done so far for future reference.

Hours Spent Each:

- Harshil Patel: 10
- Bartosz Kawalkowski: 8

Meeting 5: October 5, 2024

Summary:

We tested how the backend and frontend are working together. The web interface can now successfully inject and unload the protection DLL from selected processes. We tried it out on a sample process and watched how it behaved through the web UI.

We also discussed some of the research we've been doing on attack and protection mechanisms. Bartosz shared insights on advanced attacks like function hooking and return address manipulation. Harshil talked about cool ways to visualize process security statuses.

We noticed we've strayed a bit from our original milestones, but we think the hands-on approach has been far more beneficial. We're focusing more on building and testing stuff rather than sticking strictly to the initial timeline.

Action Items:

- Keep researching advanced attack methods to improve our protection strategies.
- Enhance the web UI with features like real-time alerts.
- Plan for some user testing to get feedback on how usable everything is.

Hours Spent Each:

- Harshil Patel: 7
- Bartosz Kawalkowski: 7

Meeting 6: October 12, 2024

Summary:

Harshil showed updates on the web interface. We've got better data visualization and user authentication should be coming soon. The interface feels more intuitive now, with clearer indicators for process statuses and smoother controls.

Bartosz talked about improvements to the process protection mechanisms, especially around detecting and stopping code patching and function hooking. We tested these features by simulating common attacks and watched how the system reacted. It's looking good, but we need to work on reducing false positives. We will focus more on refining the UI & IPC systems before focusing more on protection mechanisms, this is something we will do more heavily in the spring semester.

We also started prepping for the upcoming design review, figuring out what key points we need to cover and any gaps we need to fill before then.

Action Items:

- Bartosz will tweak the detection algorithms to cut down on false positives.
- Harshil will prep presentation materials to showcase the web interface.
- We'll work together on the design review documentation.

Hours Spent Each:

- Harshil Patel: 8
- Bartosz Kawalkowski: 10

Meeting 7: October 19, 2024

Summary:

We focused on getting ready for the design review. Harshil shared a draft of the presentation slides, including an overview of our project, and explanations of what we've been doing so far. Bartosz added some technical details about the backend and the challenges we've faced with development.

We ran through the presentation to make sure we're communicating our progress effectively. We gave each other feedback to improve clarity and flow. We also tried to anticipate any questions we might get during the review and prepared some answers.

Action Items:

- Finalize the presentation slides and practice individually.
- Finish up the design review documentation.

Hours Spent Each:

- Harshil Patel: 8
- Bartosz Kawalkowski: 8

Meeting 8: October 26, 2024

Summary:

After tweaking our presentation based on each-other’s feedback, we did a final run-through today. We focused on timing and making sure we hit all the key points.

We also talked about what we're going to tackle after the design review. Since we've made good progress on the POC and web UI, we're planning to get back to our milestones, especially working on the behavior analysis subsystem and network monitoring features. We split up the tasks and adjusted our schedule to fit these goals.

Action Items:

- Present at the design review and gather any feedback.
- Bartosz will start developing the behavior analysis subsystem.
- Harshil will look into network monitoring tools and add relevant features to the web UI.

Hours Spent Each:

- Harshil Patel: 6
- Bartosz Kawalkowski: 6

Meeting 9: November 2, 2024

Summary:

The design review went well. We got some feedback highlighting what we're doing right and what we can improve. One big suggestion was to beef up our network monitoring capabilities and tighten up the security measures in our process protection mechanisms.

Harshil shared some initial thoughts on network monitoring tools and how we can integrate network activity data into the web interface. Bartosz laid out plans for the behavior analysis subsystem, which will help us spot anomalies in process behavior.

We also talked about how important thorough testing is, especially for the security features. We're going to design test cases that cover various attack scenarios to make sure our system can handle them.

Action Items:

- Harshil will start adding network monitoring features.
- Bartosz will develop algorithms for behavior analysis.
- We'll both work on creating a testing framework for the security features.

Hours Spent Each:

- Harshil Patel: 7
- Bartosz Kawalkowski: 6

Meeting 10: November 9, 2024

Summary:

Making good progress. Harshil started on a POC of basic network activity tracking, working on adding the ability for users to see what network connections each process has. Bartosz developed some initial algorithms to detect weird behavior based on CPU and memory usage patterns.

We tested these new features by simulating common attacks like unauthorized remote thread creation and suspicious network activity. The system caught these actions, but we realized we need to fine-tune things to balance catching real threats without too many false alarms.

We also started talking about the final report and presentation due at the end of the semester. We outlined what needs to be included and divvied up who's writing what.

Action Items:

- Keep refining the network monitoring and behavior analysis features.
- Start drafting the final report.
- Plan the final presentation, including demos of the new features.

Hours Spent Each:

- Harshil Patel: 6
- Bartosz Kawalkowski: 6

Meeting 11: November 16, 2024

Summary:

With the semester wrapping up, we're focusing on pulling everything together. Harshil continued work on the web interface to include alerts for suspicious network activity, which should help users respond to potential threats quickly. This is not yet implemented in the web UI. Bartosz improved the behavior analysis algorithms, even bringing in some machine learning techniques to better detect anomalies.

We reviewed the draft of the final report to make sure it accurately reflects what we've done, the challenges we faced, and how we solved them. We also outlined the final presentation and decided to include some recorded demos to show off our current progress.

Action Items:

- Finalize the final report and get it ready for submission.
- Make sure all our code and documentation are up to date in the repo.

Hours Spent Each:

- Harshil Patel: 3
- Bartosz Kawalkowski: 4

Meeting 12: November 23, 2024

Summary:

Last official meeting of the fall semester! We tied up some loose ends and ran through our current work as few times. We thoroughly tested the whole system to make sure everything is stable and ready for the demo. We also went over the feedback on our final report and made the necessary tweaks. We decided to scrap/remove some incomplete features or features that were having a lot of bugs, we realized we need to spend more time on some of the more sophisticated mechanisms and will revisit these challenges more thoroughly during the spring semester.

Looking back, we realized that even though we didn't stick strictly to our initial milestones, taking a more practical approach really helped us build a functional and robust prototype. We've also got some ideas for future improvements and features we'd like to add down the line.

Action Items:

- Submit the final report.
- Plan a meeting after the project wraps up to discuss future work & plans

Hours Spent Each:

- Harshil Patel: 2
- Bartosz Kawalkowski: 2

Meeting 13: January 11, 2025

Summary:

The spring semester began with a comprehensive review of our progress to date and updates on feature completion. Harshil spent the break redesigning parts of the web dashboard to better support real-time interactions and alert handling. He shared mockups that introduced new alert banners and dynamic elements tied to backend triggers. Bartosz returned to his development on behavior-based detections, focusing on modularizing the backend for better scalability. We re-established our weekly meeting schedule and finalized plans to test real-time detection of process manipulation.

Action Items:

- Continue behavior engine logic.
- Sync UI alert triggers.
- Test module stability.

Hours Spent Each:

- Harshil Patel: 7
- Bartosz Kawalkowski: 7

Meeting 14: January 18, 2025

Summary:

Work this week centered on backend performance and UI enhancements. Bartosz restructured the logging system to include more granular data for each monitored process, which will aid both detection accuracy and post-analysis. Harshil implemented visual indicators on the dashboard to reflect risk levels, allowing users to distinguish low, medium, and high-priority alerts. We also discussed IPC refinements to reduce communication overhead between backend and frontend. Tests showed slight delays under heavy load, prompting the need for optimization.

Action Items:

- Implement alert tiers.
- Refactor IPC calls.
- Review JSON data handling.

Hours Spent Each:

- Harshil Patel: 8
- Bartosz Kawalkowski: 8

Meeting 15: February 1, 2025

Summary:

This meeting focused on connecting memory monitoring outputs to frontend components. Harshil linked backend detection of DLL injections and remote memory access to UI popups and system logs. We simulated common attack scenarios using controlled test cases and observed real-time flagging of suspicious actions. However, a few bugs surfaced related to inconsistent JSON formatting, which were traced to serialization conflicts between modules. Bartosz fixed the memory leak in the logger while Harshil worked on input sanitation.

Action Items:

- Expand test cases.
- Harden JSON validation.
- Monitor memory usage.

Hours Spent Each:

- Harshil Patel: 6
- Bartosz Kawalkowski: 6

Meeting 16: February 15, 2025

Summary:

Progress was made on integrating NT API-based thread injection detection. Bartosz added logic to capture unusual thread creation patterns, and we tested the feature against several sandboxed exploits. Harshil finalized the UI component to show historical logs, giving users a searchable view of all process behavior over time. We reviewed integration performance and noted response times were within acceptable limits. The team debugged race conditions in the alert system and validated fixes through parallel session testing.

Action Items:

- Validate remote thread detector.
- Finalize logging table design.
- Automate response timing tests.

Hours Spent Each:

- Harshil Patel: 4
- Bartosz Kawalkowski: 6

Meeting 17: February 29, 2025

Summary:

We conducted a full system integration test combining memory detection, behavior scoring, and network awareness modules. Harshil created a streamlined export function to package logs as downloadable JSON reports. Bartosz ran into issues with signature detection inconsistencies caused by slight memory variations across environments. We added normalization logic to the hashing process and re-tested. The output was much more stable after patching. Screenshots and evidence were collected to document this milestone

Action Items:

- Refactor memory check logic.
- Log test coverage.
- Start performance report.

Hours Spent Each:

- Harshil Patel: 10
- Bartosz Kawalkowski: 10

Meeting 18: March 7, 2025

Summary:

We focused this session on optimizing our detection model and UI layout. Bartosz tuned the weights in the behavior scoring system and analyzed results from false positives flagged during prior tests. Harshil optimized dashboard responsiveness, especially during bursts of alert activity. We simulated 5 concurrent cheating attempts and observed excellent performance with minimal UI lag. Overall, both backend and frontend operated in sync with reduced CPU overhead.

Action Items:

- Cross-test across VMs.
- Review all IPC points.

Hours Spent Each:

- Harshil Patel: 10
- Bartosz Kawalkowski: 10

Meeting 19: March 15, 2025

Summary:

Following the mid-semester review, we aligned our priorities for the final sprint. Feedback emphasized showcasing real-time capabilities, so Harshil pushed forward the integration of per-process network event feeds into the dashboard. Bartosz refined rule definitions and adjusted thresholds for flagging behavior-based anomalies. We reviewed milestone documentation and ensured we were on track to deliver a stable, complete version by Expo.

Action Items:

- Document revised timeline.
- Confirm live network event capture.
- Plan final benchmarks.

Hours Spent Each:

- Harshil Patel: 5
- Bartosz Kawalkowski: 4

Meeting 20: March 28, 2025

Summary:

This was our most comprehensive system test to date. We enabled all subsystems (memory, code integrity, behavior, network) and validated their interoperation under stress. The dashboard successfully logged, visualized, and allowed user response to live threats. Harshil finalized the alert color scheme and added CSV export. Bartosz analyzed memory and CPU metrics, confirming we stayed under the performance ceiling. All logs, screenshots, and videos were saved for the final report and Expo display.

Action Items:

- Archive all demo data.
- Update Expo poster.
- Final code review.

- Plan final benchmarks.

Hours Spent Each:

- Harshil Patel: 3
- Bartosz Kawalkowski: 3

Bartosz Kawalkowski (kawalkba@mail.uc.edu)

Harshil Patel (patel3hs@mail.uc.edu)