COLLEGE OF COMPUTING KHON KAEN UNIVERSITY
มหาวิทยาลัยขอนแก่น KHON KAEN UNIVERSITY

673380537-1 จรรยวรรธน์ หัสคุณ Sec.3 กลุ่ม 3
673380545-2 ธนวิชญ์ ธนูสา Sec.3 กลุ่ม 3
673380544-4 ธนภัทร ประสานพันธ์ Sec.3 กลุ่ม 3
673380540-2 ชลชยพงษ์ ชวลิตเมธารัตน์ Sec.3 กลุ่ม 3

รายวิชา CP422011 INTRODUCTION TO COMPUTING NETWORKINK
อาจารย์ผู้สอน ศ.ดรจักรชัย โสอินทร์ และ ผศ.ดร.เพรช อิ่มทองคำ

TBAT GUARD: ANTI-CHEAT

วัตถุประสงค์ (Objectives)

1. พัฒนาระบบตรวจจับการโกงในเกมออนไลน์ เช่น speed hack, teleport, FPS/Ping ผิดปกติ และไฟล์เกมไม่ตรงตามมาตรฐาน
2. สร้าง Dashboard แบบเรียลไทม์ แสดงข้อมูลผู้เล่น, flag, กราฟ trajectory และสถิติการโกง
3. ประเมินความรุนแรงของผู้เล่นและจัดการแบนอัตโนมัติเมื่อเกินเกณฑ์
4. สนับสนุนการวิเคราะห์พฤติกรรมผู้เล่นเพื่อปรับปรุงความยุติธรรมในการเล่นเกม
5. สรุปผลและรายงานข้อมูลในรูปแบบตารางและกราฟ เพื่อช่วยการตัดสินใจของผู้ดูแลระบบ

งานที่เกี่ยวข้อง (Related Work)

- Anti-Cheat Systems ในเกมออนไลน์ – ระบบตรวจจับการโกงแบบ client-server เช่น VAC, Easy Anti-Cheat, BattlEye ใช้วิธีตรวจจับ speed hack, aimbot, และ integrity mismatch ของไฟล์เกม
- การวิเคราะห์พฤติกรรมผู้เล่น – ใช้การเก็บข้อมูลการเคลื่อนที่, ความเร็ว, และพฤติกรรมการเล่น เพื่อตรวจสอบความผิดปกติแบบเรียลไทม์
- Dashboard และ Visualization – มีงานวิจัยและซอฟต์แวร์ที่นำข้อมูลผู้เล่นมาสรุปเป็นกราฟ trajectory, Pie Chart ของชนิด flag เพื่อช่วยผู้ดูแลระบบตัดสินใจ

ฟังก์ชันหลัก (Main Features)

1. เก็บข้อมูลผู้เล่นแบบเรียลไทม์ – เก็บตำแหน่ง, ความเร็ว, FPS, ping และเหตุการณ์ต่าง ๆ ของผู้เล่น
2. ตรวจจับพฤติกรรมโกง – ตรวจจับ speed hack, teleport, integrity mismatch, FPS/Ping สูงเกินกำหนด
3. ระบบ Ban/Unban อัตโนมัติ – บันทึกและบังคับใช้โทษต่อผู้เล่นที่มีคะแนนความรุนแรงสูง
4. Dashboard แสดงผลแบบ Visualization – ตารางผู้เล่น, ตาราง flag ล่าสุด, Pie Chart และ Trajectory Chart
5. กรองและเรียงข้อมูล – กรองผู้เล่น flagged/banned, กำหนด threshold severity, เรียงตามค่า severity, events, last seen
6. WebSocket อัปเดตข้อมูลเรียลไทม์ – ข้อมูล dashboard อัปเดตทันทีเมื่อมีเหตุการณ์ใหม่

วิธีการทำงาน (Methodology)

1.ฝั่ง Client (Unity)
- ติดตั้งสคริปต์ ACClient เพื่อตรวจจับเหตุการณ์ในเกม เช่น การเคลื่อนไหว การกระโดด การยิง
- ระบบจะส่งข้อมูลเหตุการณ์ (Event) และค่าประสิทธิภาพ เช่น FPS, Ping ไปยังเซิร์ฟเวอร์ผ่าน HTTP
- มีการตรวจสอบไฟล์สำคัญ (Integrity Check) และสแกน Process ต้องห้าม

2.ฝั่ง Server (Node.js + MongoDB)
- รับข้อมูลจาก Client แล้วบันทึกลงฐานข้อมูล
- วิเคราะห์พฤติกรรมของผู้เล่นตามกฎ (Rule-based Detection) และคำนวณคะแนนความผิดปกติ (Score System)
- หากเกินเกณฑ์ ระบบจะทำการ “Flag” หรือ “Auto-ban” อัตโนมัติ

3.Dashboard (Web Application)
- แสดงข้อมูลผู้เล่นที่ถูกตรวจจับแบบเรียลไทม์ผ่าน WebSocket
- ผู้ดูแลสามารถตรวจสอบเหตุการณ์, สถานะการ Ban, และกราฟสถิติการตรวจจับได้

ความแตกต่างจากงานอื่น

ระบบนี้ตรวจจับและจัดการการโกงแบบเรียลไทม์ผ่าน Dashboard เชื่อมต่อกับเกมโดยตรง มีระบบ Auto-ban และ Scoring อัตโนมัติ ต่างจากงานทั่วไปที่ตรวจสอบย้อนหลังจาก log เท่านั้น.

การต่อยอด

สามารถพัฒนาให้รองรับเกมหลายประเภท เพิ่มระบบ Machine Learning เพื่อเรียนรู้พฤติกรรมผู้เล่นอัตโนมัติ และเปิดให้บุคคลภายนอกใช้งานผ่าน API หรือ Unity Package สำหรับเชื่อมต่อกับเกมอื่นได้โดยตรง.

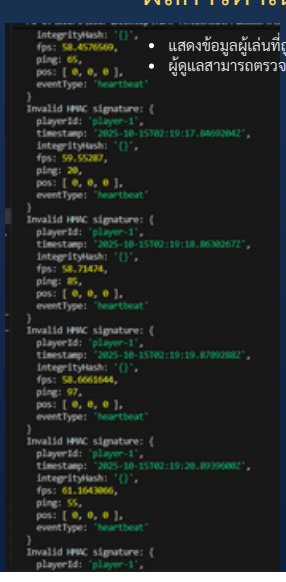
เครื่องมือและเทคโนโลยีที่ใช้ใน Anti-Cheat Framework

- ภาษาโปรแกรม:
- C# สำหรับ Client ในเกม (Unity)
- JavaScript/Node.js สำหรับ Server และ Dashboard
- ฐานข้อมูล:
- MongoDB สำหรับจัดเก็บข้อมูลผู้เล่น, Event, Flag และ Ban
- การสื่อสารแบบเรียลไทม์:
- WebSocket สำหรับส่งข้อมูลจาก Server ไป Dashboard
- Frameworks & Libraries:
- Express.js สำหรับสร้าง API Server
- Chart.js สำหรับสร้างกราฟบน Dashboard
- Mongoose สำหรับจัดการ MongoDB
- เครื่องมือพัฒนา:
- Unity Editor สำหรับสร้าง Client และตรวจจับ metrics
- VS Code / IDE อื่น ๆ สำหรับพัฒนา Server และ Dashboard
- เทคโนโลยีตรวจสอบไฟล์:
- Hashing (เช่น SHA-256) สำหรับตรวจสอบความสมบูรณ์ของไฟล์เก

หลักการและเหตุผล

- เกมออนไลน์มักเผชิญปัญหา ผู้เล่นโกง (Cheating) เช่น การปรับแต่งเกมหรือปลอมข้อมูลระหว่างการส่งข้อมูล ทำให้เสียความยุติธรรมและประสบการณ์ของผู้เล่น โครงงานนี้จึงพัฒนา Anti-Cheat Framework เพื่อตรวจจับพฤติกรรมผิดปกติของผู้เล่นแบบ Real-time ผ่านระบบ Client–Server และ WebSocket โดยมี Dashboard สำหรับผู้ดูแลเพื่อเฝ้าตรวจสอบและดำเนินการ แจ้งเตือนหรือแบนผู้เล่นอัตโนมัติ ระบบนี้ช่วยให้เข้าใจการสื่อสารข้อมูลในเครือข่าย และสามารถต่อยอดใช้ในการเรียนการสอนหรือระบบเกมจริงได้

ผลการดำเนินการ



- แสดงข้อมูลผู้เล่นที่ถูกตรวจจับแบบเรียลไทม์ผ่าน WebSocket
- ผู้ดูแลสามารถตรวจสอบเหตุการณ์, สถานะการ Ban, และกราฟสถิติการตรวจจับได้