Preventing multiplayer cheating

CMSC425.01 Spring 2019
Administrivia

• Final homework (Hw3)
  • Errata: Question 2(a) has a bad heuristic.
  • Corrected homework posted

• Final midterm
  • Prep material on web site
  • Monday will do review
  • Thursday May 8th
Today’s question

How to prevent cheating
Cheating ...

• Frustrates players
• Reduces in game purchases
• Causes players to quit
• Impacts paid player leagues

• Your experiences?
Consider more generally ...

- Always good to consider security

- How are games like
  - Online banking
  - Online retail
  - Social media
  - Anything networked
Cheating ...

• Frustrates players
• Reduces in game purchases
• Causes players to quit
• Impacts paid player leagues

• Your experiences?

• How prevent?
Cheating ...

• Frustrates players
• Reduces in game purchases
• Causes players to quit
• Impacts paid player leagues

• How prevent?
• Your experiences?

Pritchard on Gamasutra

• If you build it, they will come to hack and cheat
• Hacking attempts increase as a game becomes more successful
• Cheaters actively try to control knowledge of their cheats
• Your game, along with everything on the cheater’s computer, is not secure—not memory, not files, not devices and networks
• Obscurity is not security
• Any communication over an open line is subject to interception, analysis and modification
• There is no such thing as a harmless cheat
• Trust in the server is everything in client-server games
• Honest players would like the game to tip them off to cheaters
Network game models

- Client server
  - Client software
  - Server software
- Peer to peer
  - Identical vs. different clients
  - With or without lockstep
- Question
  - How is computation shared?

Cheating attacks

- **Reflex Augmentation**: Improve physical performance, such as the firing rate or aiming
- **Information Exposure**: Clients obtain/modify information that should be hidden.
- **Protocol cheats**: Clients capture and modify or delay packets
- **Authoritative Clients**: Although the server should have full authority, some online games grant clients authority over game execution for the sake of efficiency. Cheaters then modify the client software.
- **Compromised servers**: A hacked server that biases game-play towards the group that knows of the hacks.
- **Bugs and Design Loopholes**: Bugs and design flaws in the game are exploited. Infrastructure Weaknesses: Differences or problems with the operating system or network environment are exploited.

Reflex Augmentation

• Enhance client software
  • Externally capture packets
  • Modify client code

• Aimbot

• Reflex enhancer

• How detect?
Reflex Augmentation

- Enhance client software
  - Externally capture packets
  - Modify client code
- Aimbot
- Reflex enhancer
- How detect?
  - Check client executable
  - Analyze packet traffic
Information Exposure

• Access and use information not entitled to
  • Infrastructure-level cheats

• What might a cheat use?
Information Exposure

• Access and use information not entitled to

• What might a cheat use?
  • Key variables
  • Graphics hacks
  • Item properties
  • Physics behavior
  • Time (speed hack)
  • Ghosting
Key variables

• Lives, strength, health, etc.
• You have the executable – how find these variables?
Key variables

• Lives, strength, health, etc.
• You have the executable – how find these variables?

• Run program.
  • If num_lives=5, search for locations with 5
  • Get killed, find one that just decreased

• How prevent?
Key variables

- Lives, strength, health, etc.
- You have the executable – how find these variables?

- Run program.
  - If num_lives=5, search for locations with 5
  - Get killed, find one that just decreased

- How prevent?
  - Encrypt or protect key variables
Graphics hacks (removal of game elements)

• Change local graphics rendering
• Turn off rendering elements
• Eliminate fog, walls, etc
Item hacks

• Modify item properties
• Firing rate, accuracy, portion action, item frequency
Physics and speed hacks

• Change collider behavior
• Change physics engine behavior

• Control game clock
Ghosting

• Cooperation between two players
• One runs as spectator
• Informs actual player of location of other players, etc
Protocol cheats

• Interfere with network packets
  • Packets may be inserted, destroyed, duplicated, or modified

• Suppressed update:
  • Accept opponent packets, suppress own, to get advantage
  • Uses server policy of interpolating and backwinding
  • Prevent by monitoring patterns of packet delay
Protocol cheats

• Interfere with network packets
  • Packets may be inserted, destroyed, duplicated, or modified

• Suppressed update:
  • Accept opponent packets, suppress own, to get advantage
  • Uses server policy of interpolating and backwinding
  • Prevent by monitoring patterns of packet delay

• Fixed delay:
  • Delay all packets
  • Local players see info faster
  • Prevent by lockstep, analysis, sliding pipelines
Protocol cheats

- Interfere with network packets
  - Packets may be inserted, destroyed, duplicated, or modified
- Inconsistency: corrupt packets
- Time-stamp: post date packets
- Spoofing: sent out packets masquerading as other player
Detecting Cheating

- Signature detection - Detecting of certain patterns of bytes in memory, checked against a database (whitelist)
- Heuristic analysis - Statistical analysis of behavior
- User reports - Information provided by other users
• 1. Metrics for best path on map
• 2. Navmesh process (R_D_P algorithm, triangulation)
• 3. Walkable terrain
• 4. Find paths on triangulated space
• 5. Configuration spaces
• 6. Quality of path
• 7. C-obstacles
• 8. Minkowski sums
• 9. Navmesh–grid, multiresolution grid
• 10. Visibility graph
• 11. Medial axis
• 12. Randomized placement
• 13. Rapidly expanded Random Trees (RRTs)
• 14. L-system plus turtle
• 15. Fractal dimension
• 16. Randomized and 3D L-systems
• 17. Particle systems
• 18. Flocking
• 19. Mandelbrot sets
• 20. Constructive solid geometry
• 21. Shading equation
• 22. Bump mapping
• 23. Polygonal meshes - basics, Euler's formula
• 24. DECL data structures
• 25. Perlin noise
• 26. A*
• 27. Admissible heuristic
• 28. Multiplayer cheating attacks
• 29. Forbidden velocities for crowd motion
• 30. Fitting cubic curves to data points
Problem 3. (20 points) Consider the collection of shaded rectangular obstacles shown in the figure below, all contained within a large enclosing rectangle. Also, consider the triangular robot, whose reference point is located at a point \( s \). (You may take \( s \) to be the origin.)

![Diagram of obstacles and robot](image)

Figure 2: Problem 3.

(a) Draw the C-obstacles for the three rectangular obstacles, including the C-obstacle from region lying outside the large enclosing rectangle.

(b) Either draw an obstacle-avoiding path for the robot from \( s \) to \( t \), or explain why it doesn’t exist.