Video Game Security

Carter Jones

Overview

- Industry overview
- Risks
  - Business
  - Technical
- Attack & Defense
  - Thick client
  - Network
- Industry comparisons
- Conclusion

whoami

- Senior security consultant @ Cigital
- Previously a security researcher
- Student of life (always learning)
- Husband
- Music lover
- Video game player/hacker

How I got here

- Video games are fun, but really hard to win
- So how do you win?
  1. Practice, practice, practice
  2. Cheat
- Too lazy/impatient to practice, so I learned how to hack

Hacking side scrollers

- Played an MMORPG
- Saw some player's character flying around the screen, which is not normally possible
- Googled how to fly in the game
- Followed tutorials
  - Had elements of reverse engineering
- Flew around the screen

Hacking games → security industry

- Skillset crossover between game hacking and security consulting/research
- Activities:
  - Threat modeling
  - Reverse engineering
  - Network protocol analysis
  - And more

Assessment activities

- Threat modeling
  - Identify the components of a system, various threat actors that can attack those components, and the possible ways that the components can be attacked
  - Identify key technical risks and existing controls (protections)
- Reverse engineering
  - Take apart the client (and server if available) to find weaknesses
- Network protocol analysis
  - Reverse engineer the protocols used by the client and server

Overview

- ~~Industry overview~~
- Risks
  - Business
  - Technical
- Attack & Defense
  - Thick client
  - Network
- Industry comparisons
- Conclusion

Business risks

- Various risks exist for business in the gaming industry
- Examples of cyclical risks
  - Profit loss → tarnished brand
  - Lots of hacks → customer disloyalty
  - Tarnishing the brand → customer disloyalty
  - Loss of customer data → customer disloyalty
  - Intellectual property theft → loss of revenue
- Risk prioritization can differ between business models
  - Freemium: gaining paid-only in-game benefits for free
  - Subscription-based: playing the game for free
  - One-time payment: DRM bypass/piracy

Overview

- ~~Industry overview~~
- Risks
  - ~~Business~~
  - Technical
- Attack & Defense
  - Thick client
  - Network
- Industry comparisons
- Conclusion

Technical risks for video games

- Account and asset hijacking/theft (account credential theft, in game item theft, etc.)
- Cheating, automation, botting, etc.
- Denial of service
- Fraud in a virtual economy
- Piracy of game titles and game content

Risks vary by game genre

- Different game genres may have different risks
- Examples:
  - FPS
    - Statpadding user scores (attacker goal: bragging rights)
    - Aimbots/cheating (attacker goal: competitive advantage)
  - MMORPG
    - Account theft (attacker goal: financial gain)
    - Private servers (attacker goal: avoiding paying subscriptions)
  - RTS
    - Map hacking (attacker goal: competitive advantage)

Risks vary by gaming platform

- Game consoles
- Mobile devices
- PCs (Windows, Linux, Mac)
- Web browsers
- Cloud hosted PC’s
- Examples:
  - Game consoles tend to be focused on client-side validation
  - Web-based games tend to be focused on server-side validation

Example video game assets

- Game content and patches
- Player account information
- Payment/billing information
- In-game assets (inventory, points, virtual currency, etc.)
- Fraud and cheat detection data
- Customer service representatives (or their accounts)

Example video game controls

- Encrypted protocols
- VPN tunnels
- Anti-tamper security on the game client
- Security event monitoring
- Cheat/fraud analysis
- IP address white-listing

Overview

- ~~Industry overview~~
- ~~Risks~~
  - ~~Business~~
  - ~~Technical~~
- Attack & Defense
  - Thick client
  - Network
- Industry comparisons
- Conclusion

Defense: protect assets using a layered approach

- Relying on a single control for protecting assets is not enough
- Example asset: premium items that must be purchased
- Example layering of controls:
  - Anti-tamper/anti-reversing of client – make it difficult for players to trick their client into thinking they've paid for a premium item
  - Network encryption – make it difficult to send spoofed messages over the wire to the server
  - Server-side checks – when a game client says to use a premium item and gain whatever perks are normally given, make sure the player actually has purchased that item

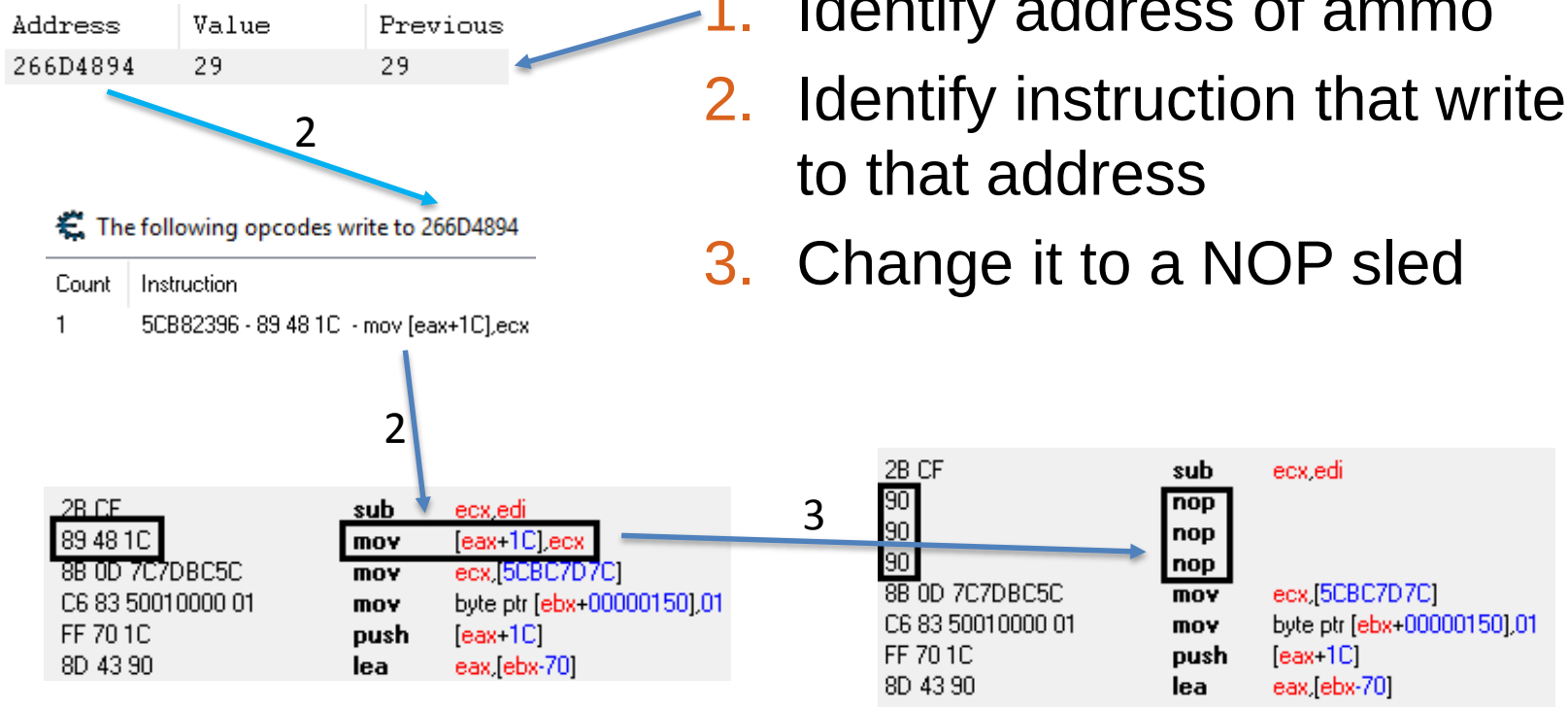
Offense: general hacking process

- Think of a goal
  - Get unlimited health
  - Gain access to restricted parts of the game
    - See hidden parts of a map
    - Access powers that are above current level
- Plan the attack: think of a way to reach that goal (example: unlimited ammo)
  - Find out how to modify the game client (at runtime or on disk)
  - Find out how to send spoofed network packets to the game server
  - Look for server logic flaws
- Execute the attack

Example: unlimited ammo in Pwnie Island

- Find out how to modify the game client
  - Game trainers! (no anti-tampering mechanism exists)
- Find out how to send spoofed network packets to the game server
  - Possible when proxying network traffic or by changing the behavior of the game client itself
- Look for server logic flaws
  - Server side checks look for discrepancies in ammo count between client and the server
  - Server itself can be controlled by the attacker!

Example: unlimited ammo in Pwnie Island



1. Identify address of ammo
2. Identify instruction that writes to that address
3. Change it to a NOP sled

Cigital

Demo

Example: shields/invisibility

1. Identify key addresses
2. Identify patterns in memory, which can reveal structures
3. Identify other key values within the structure
4. Identify reliable pointers to an instance of the structure
5. Repeatedly change values at offsets in the structure
   - Example: every 50 milliseconds, enable invisibility and large shields

- This is useful for when instructions that would be NOP'd would give advantage to enemy players or NPCs.

Offset	Value
0x100	XAxisPosition
0x104	YAxisPosition
0x108	ZAxisPosition
0x200	XAxisVelocity
0x204	YAxisVelocity
0x300	DirectionFacing
0x400	Health
0x404	**Shields**
0x408	**Invisibility**

Client-side protections

- Obfuscation
  - Makes static analysis more difficult
  - Can be applied to all of the code or just to portions
  - Generally is just some encryption applied to most of the binary
- Anti-debugging
  - Checks to see if a debugger is enabled
  - Either stops execution or behaves differently (common with malware)
- Runtime integrity checks
  - Checks to see if portions of the code have been changed after the program was launched
  - Useful for identifying non-debugging-based runtime hacks (WriteProcessMemory, VirtualAlloc, etc.)

Overview

- ~~Industry overview~~
- ~~Risks~~
  - ~~Business~~
  - ~~Technical~~
- Attack & Defense
  - ~~Thick client~~
  - Network
- Industry comparisons
- Conclusion

Network-based hacks

- Useful when client-side protections prevent easily tampering with client
- Do everything from a network level (no interaction with the game client's process)
- Usually done by proxying client→server traffic, but can also be done by completely replicating the client's behavior

Example network hacks

- Map viewers
  - View all mob locations on the map, rather than what is normally available through the game client mini-map
  - Reveal locations of hidden items that are hidden from the game client's view
- Speed/teleport hacks
  - Intercept XYZ coordinates & velocity in network packets and modify them slightly to give a speed advantage or to teleport the player
- Server-side exploits
  - Identify and exploit flaws (logic vulnerabilities) or bugs (implementation vulnerabilities) in the server to execute arbitrary code

Server side protections

- Only send data to the client on a need to know basis
- Consider all data from the client is potentially malicious
- Compare data received from the client to an acceptable range of expected data
- Performance improvement: perform sampling of data received from the clients

Overview

- ~~Industry overview~~
- ~~Risks~~
  - ~~Business~~
  - ~~Technical~~
- ~~Attack & Defense~~
  - ~~Thick client~~
  - ~~Network~~
- Industry comparisons
- Conclusion

Industry comparisons

- These types of hacks are not specific to the video gaming industry
- Examples:
  - Financial institutions
    - bank websites
    - stock trading
    - ATM transactions
  - Media & entertainment
    - Video streaming restriction bypass
    - Copyright protection for physical media

Overview

- ~~Industry overview~~
- ~~Risks~~
  - ~~Business~~
  - ~~Technical~~
- ~~Attack & Defense~~
  - ~~Thick client~~
  - ~~Network~~
- ~~Industry comparisons~~
- Conclusion

Conclusion

- Video game security: not just about preventing cheating
- Business & technical risks:
  - Differs by type of game (genre, platform, etc.)
  - Some risks are unique to gaming industry
  - Some risks are shared with other industries
- Layered defenses are very important
- Both attack and defense processes apply to more than just the video game industry

Questions?

Thank You