SWINBURNE UNIVERSITY OF TECHNOLOGY

Covert Channels in Multiplayer First Person Shooter Online Games

Sebastian Zander,
Grenville Armitage, Philip Branch

{szander, garmitage, pbranch}@swin.edu.au

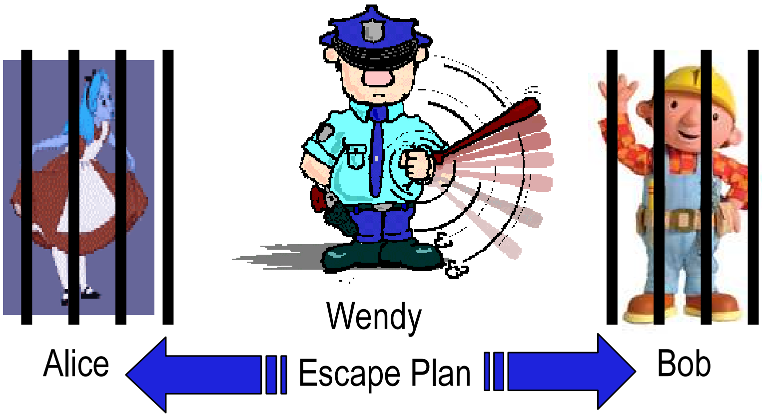
Centre for Advanced Internet Architectures (CAIA)
Swinburne University of Technology

Overview

- Covert channels overview
- Covert channels in First Person Shooter (FPS) games
- Empirical evaluation
- Countermeasures
- Conclusions

Often encryption alone is not sufficient

- Encryption protects communication from being read
- Existence of communication is enough to take actions
- Covert channels hide the existence of communication
- Usually covert channels use means of communication not intended for communication
- Huge amount of overt network traffic makes Internet ideal for covert communication



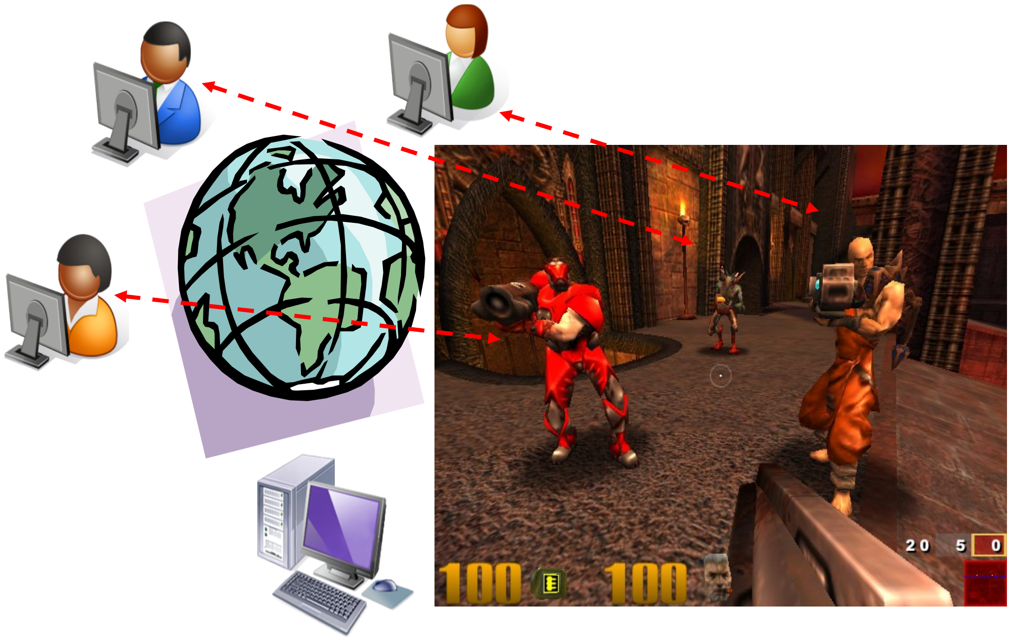
Covert channels have different users

- Government agencies vs. criminals and terrorists hiding communication and coordination
- Hackers ex-filtrating data or controlling systems vs. sysadmins hiding management traffic
- Ordinary users circumventing censorship or encryption laws (or bypassing firewalls)
- Distribution and control of viruses, worms, bots

- Many network protocol-based covert channels have been proposed
- Very limited work on covert channels in network games, focused on board games

Hide covert channels in game traffic

- Hide covert data as slight variations of player character movements in First Person Shooter games
- Channel remains covert so long as variations are visually imperceptible to human players

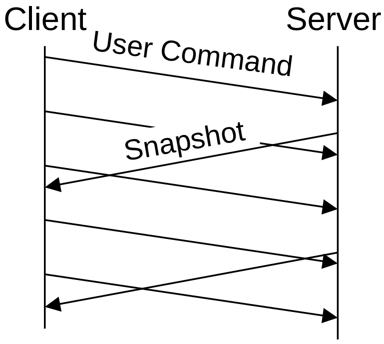


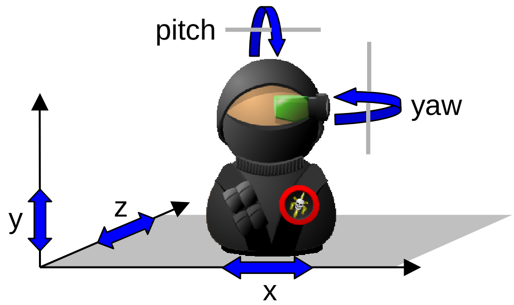
Advantages of FPS covert channels

- FPS games are very common and their traffic is not suspicious
- Channel cannot be eliminated because it is tied to player movement (intrinsic function)
- Sufficient noise in player movement to hide channel
- Covert sender/receiver use game server as intermediary, rather than directly exchanging data
- Tens of thousands of game servers active on the Internet at any time for popular games
- Player movements are not logged or filtered by the servers, unlike in-game chat messages
- Not limited to FPS games

FPS network protocol overview

- Focus on Quake III Arena (Q3), but other FPS games have similar protocols
- Asynchronous message exchange over IP/UDP
- Client sends user commands to server
  - Movement (x,y,z) and view angles (pitch, yaw)
- Server sends game state to client in snapshots
  - State of player character
  - State of other entities (players, bots and objects)
- Compression: delta encoding + adapt. Huffman coding





Encoding and decoding of covert bits

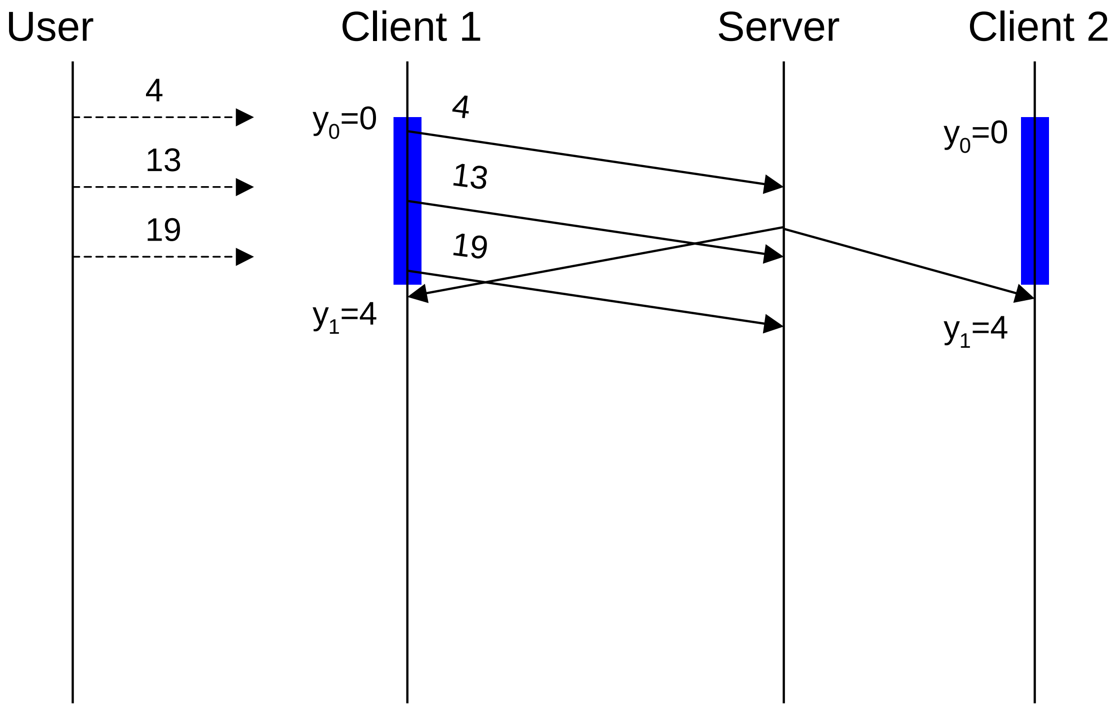
- Encode covert data as slight, yet continuous, variations of player character movements
- Encode N covert bits with integer value b in changes of (modified) parameter values $\tilde{y}$ between snapshots:

  $$b = |\tilde{y}_j - \tilde{y}_{j-1}| \; mod \; 2^N$$

- Covert sender can only manipulate $\tilde{y}$ indirectly through user commands
  - Character's position perturbed by various 'forces'
  - But view angles mostly depend on player input only
  - Encode only when player changes view angles so that channel is effectively masked
- Covert bits can be encoded simultaneously in pitch and yaw

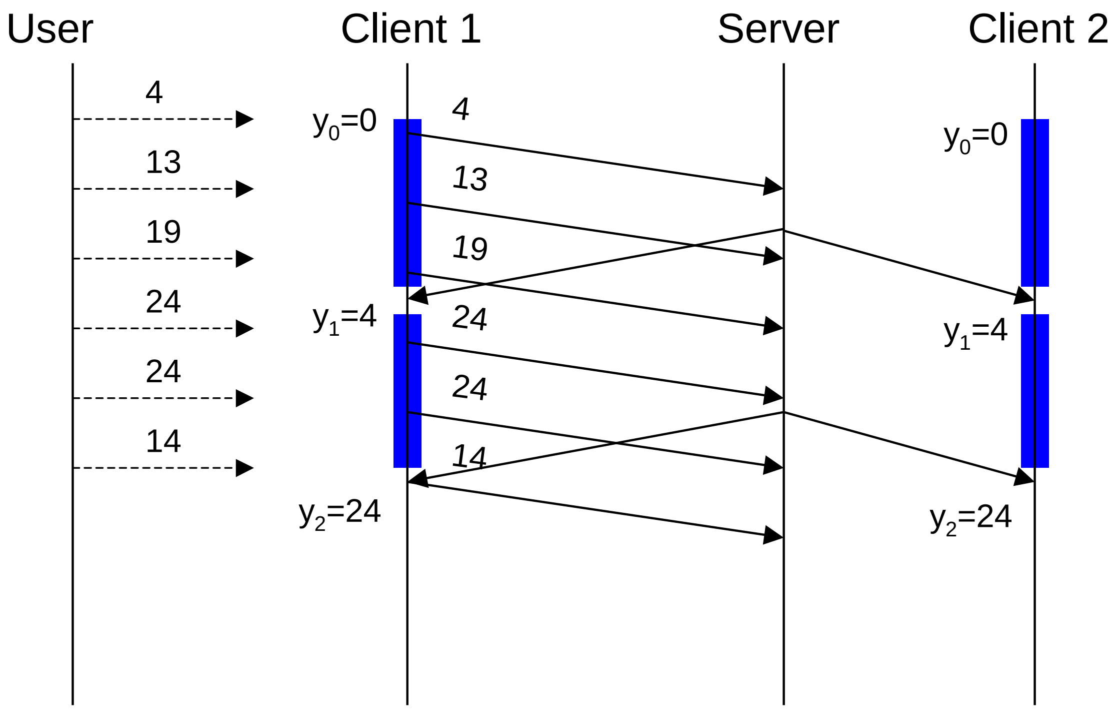
Encoding and decoding example

- Without covert channel



Encoding and decoding example

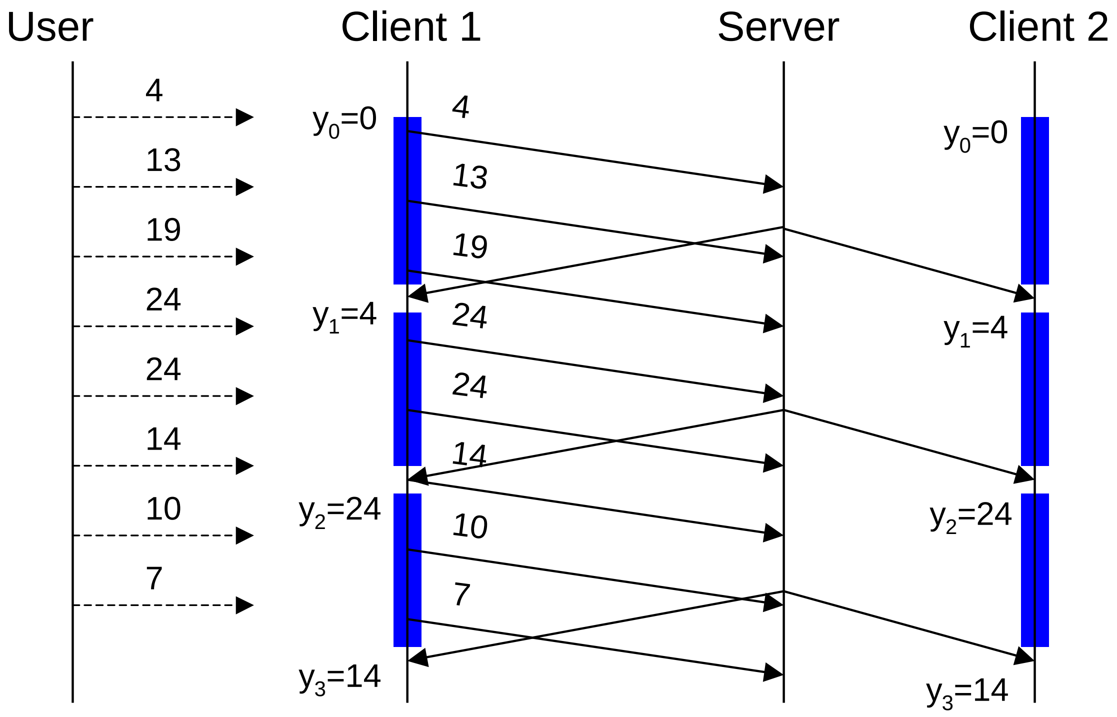
- Without covert channel



LCN'08 http://www.caia.swin.edu.au {szander, garmitage, pbranch}@swin.edu.au October 2008 9

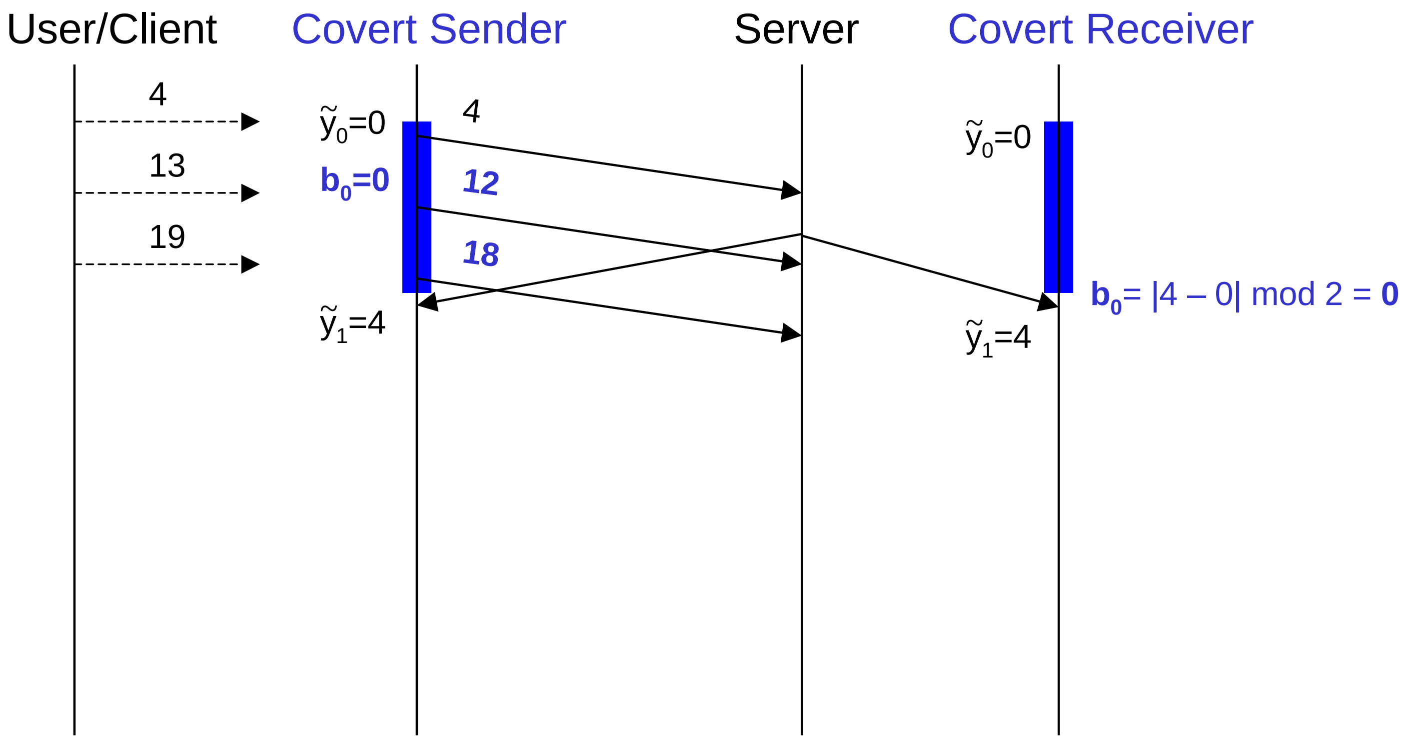
Encoding and decoding example

- Without covert channel



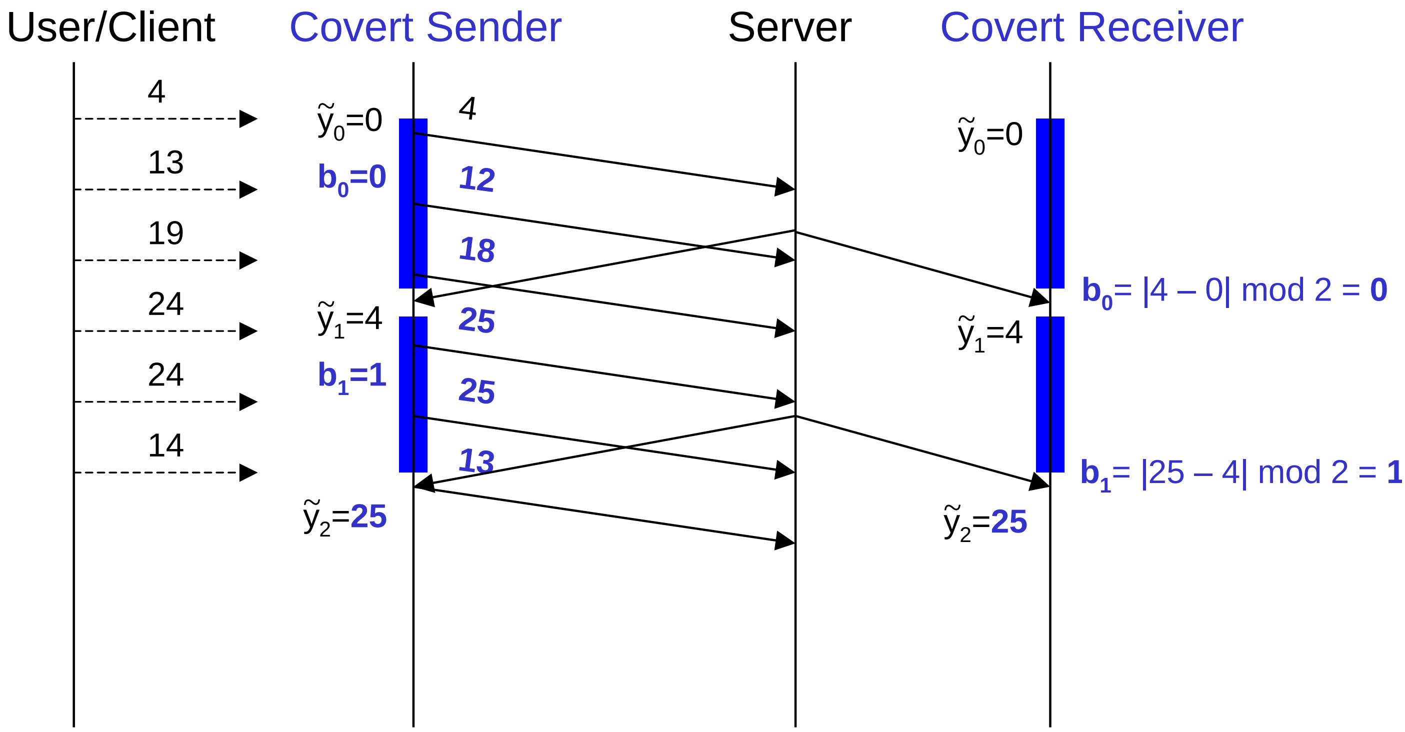
Encoding and decoding example

- With covert channel



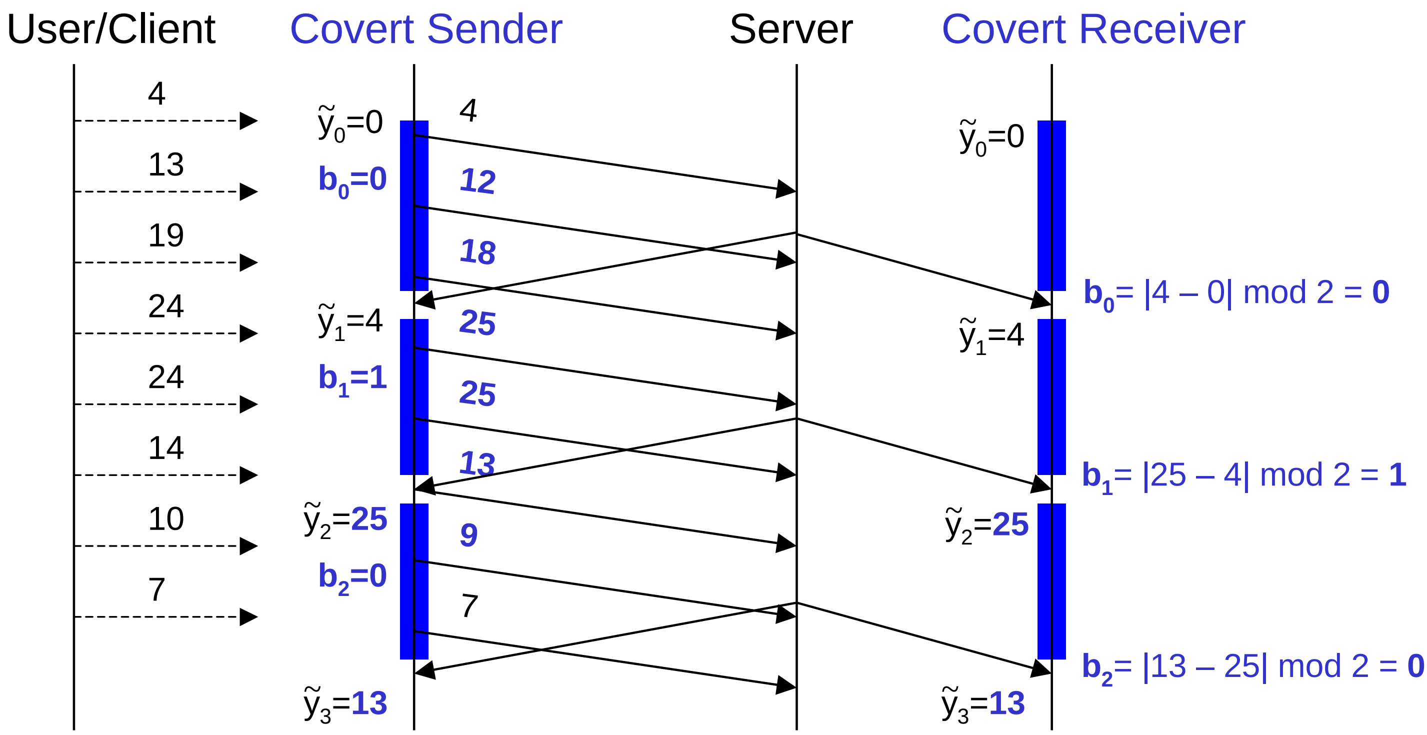
Encoding and decoding example

- With covert channel



Encoding and decoding example

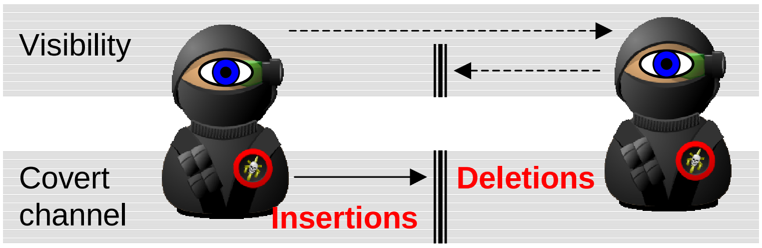
- With covert channel



Synchronisation errors

- Synchronisation errors
  - Bits lost on channel (bit deletions)
  - Bits inserted on channel (bit insertions)
- Exchange of player state based on potential visibility
  - Players only receive state updates for pot. visible players
  - Depends on static information (map) and current positions
  - In Q3 potential visibility is asymmetric
- IP/UDP is unreliable so snapshots can be lost

⇒ Bit synchronisation mechanism



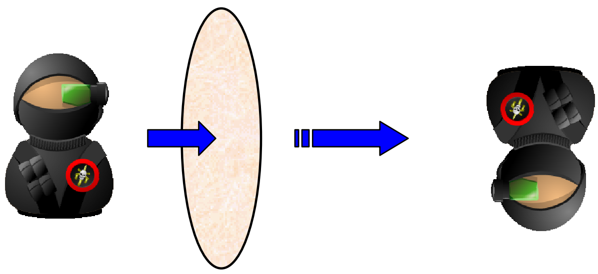
Substitution errors

- Substitution errors (= flipped bits)
- Game mechanics change player's view angles
  - Respawning after death
  - Pitch is clamped between -87 and +87 degrees

  ⇒ Pause encoding and decoding
- Map elements change player's view angles
  - Teleportation portals

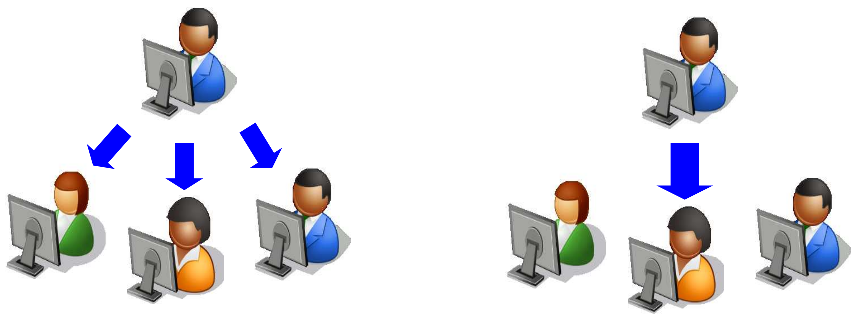
  ⇒ Same as respawning
  - Moving platforms players can step on

  ⇒ Rare on multiplayer maps, can be avoided



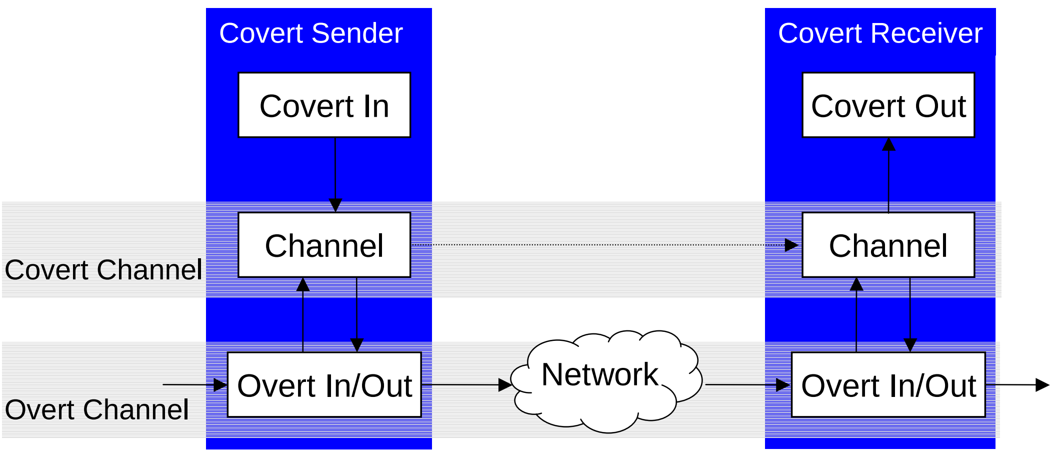
Broadcast vs. unicast communication

- In broadcast mode covert sender continuously sends, but covert receiver(s) receive only when sender visible
  - No insertions but many deletions
- In unicast mode covert sender only sends when receiver visible
  - Minimises deletions but introduces insertions
  - Covert sender must know covert receiver's in-game identity
- Covert receiver either knows covert sender's in-game identity or uncovers it (meaningful bit sequence)



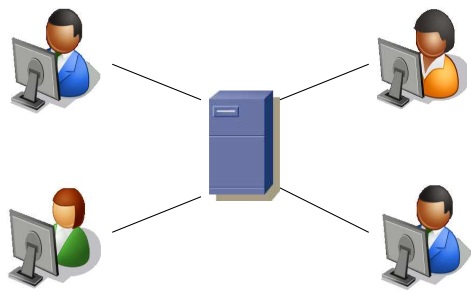
Implementation and deployment

- Implemented for Q3 as transparent proxy using the Covert Channels Evaluation Framework (CCHEF)
  - Cheat detection mechanisms cannot detect proxies
  - But knowledge of protocol's 'encryption' is needed
- Could be implemented as client modification (mod), but many Q3 servers do not allow modified clients
- Could be implemented like other client-side cheats



Evaluation in local testbed

- One Q3 server and 2–4 Q3 clients (human players)
- One covert sender and 1–3 covert receivers
- 10 minute games on the standard map q3dm1
- Measure average bit rates and bit error rates
  - Broadcast and unicast mode
  - Encode 1 bit per angle change
  - Encode in pitch and yaw
- Analyse how different covert channel looks from normal traffic (fingerprint)



Bit rates and bit errors

- Broadcast mode

Sender Rate [bits/s]	Deletions [%]	Insertions [%]	Substitutions [%]
14.6–18.0	42.9–54.9	0.0	0.0

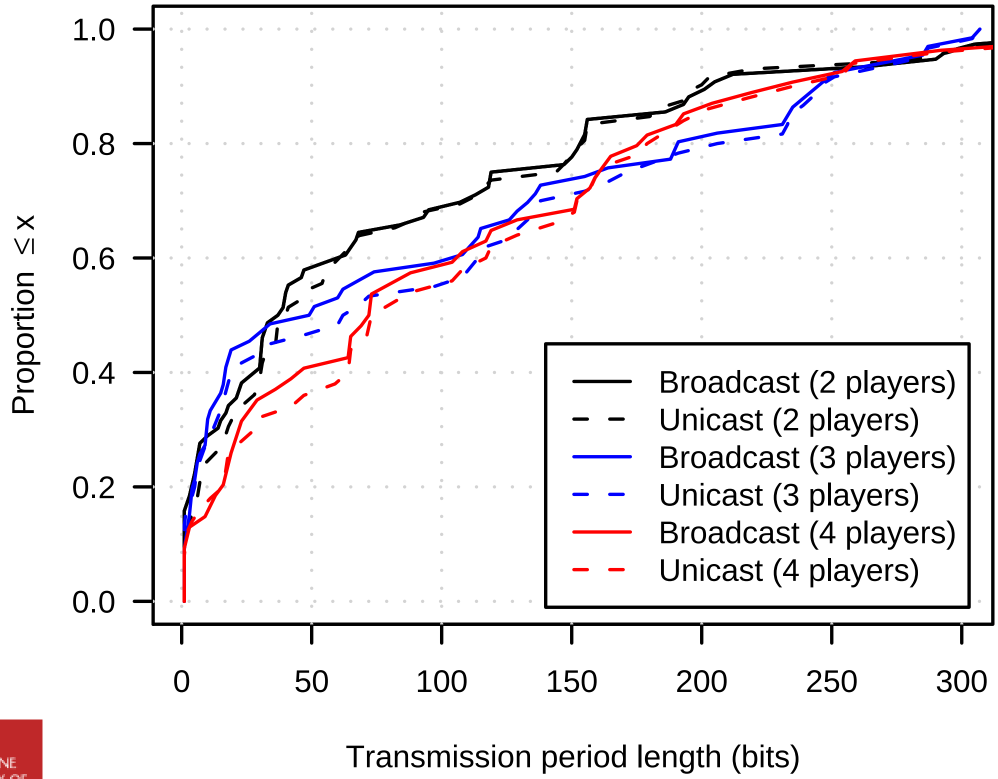
- Unicast mode

Sender Rate [bits/s]	Deletions [%]	Insertions [%]	Substitutions [%]
8.0–10.4	3.0–3.6	1.0–2.5	0.0

$\Rightarrow$ Net bit rates of 7.7–9.8 bits/s

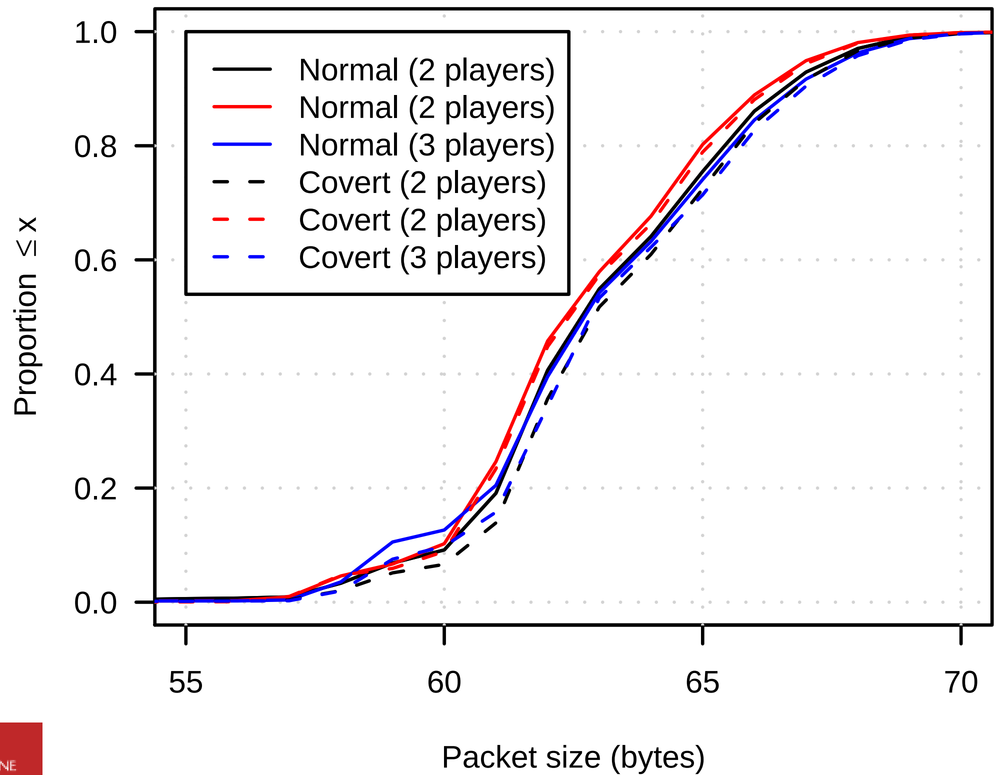
Length of transmission periods

- Bit deletions/insertions occur in bursts (between errors → transmission periods)



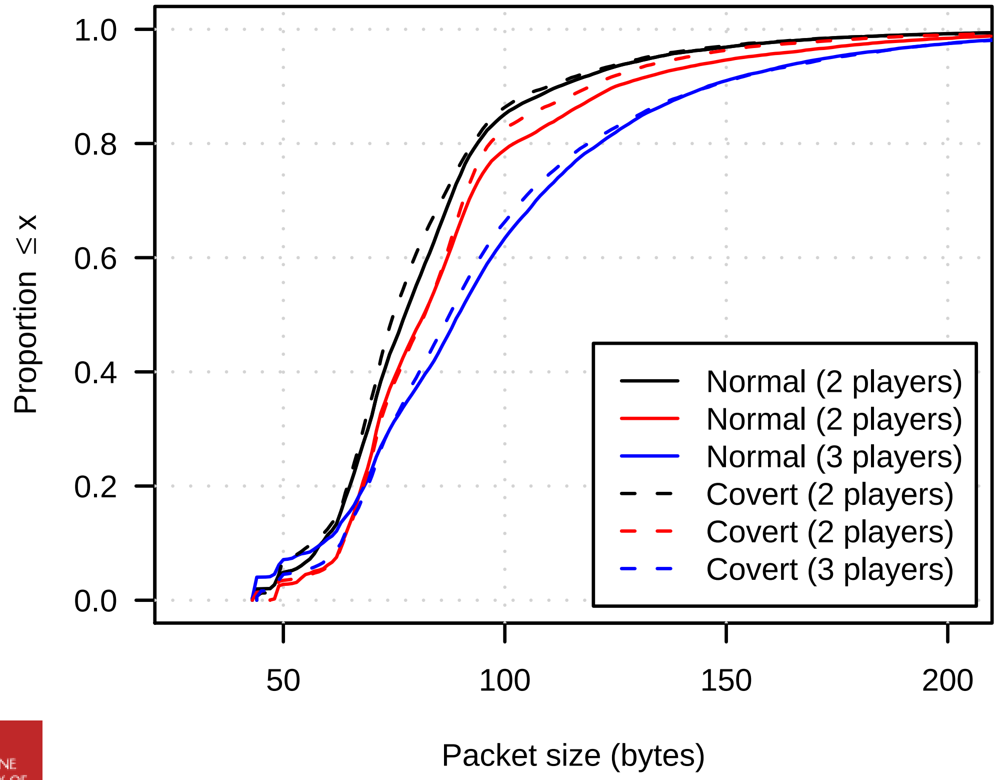
Investigating the fingerprint: packet sizes

- Compare client-to-server packet size distribution of covert sender with normal players



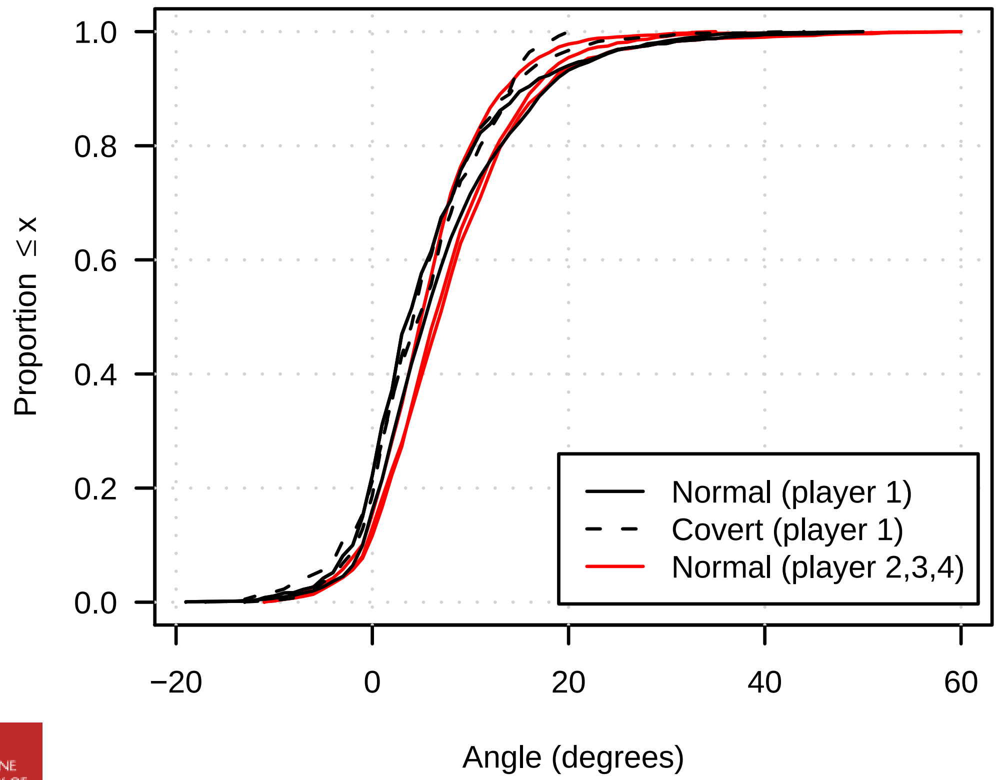
Investigating the fingerprint: packet sizes

- Compare server-to-client packet size distribution of covert sender with normal players



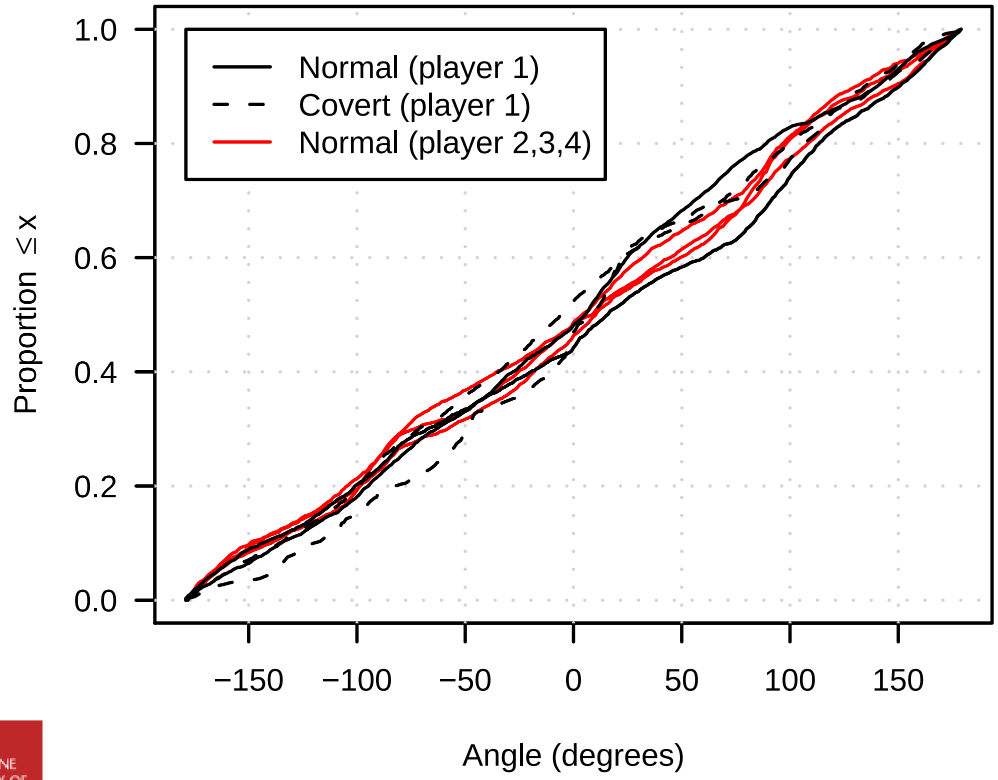
Investigating the fingerprint: view angles

- Compare pitch angle distribution of covert sender with normal players



Investigating the fingerprint: view angles

- Compare yaw angle distribution of covert sender with normal players



Countermeasures

- Generally available countermeasures
  - Elimination
  - Capacity limitation
  - Detection
- Channel cannot be eliminated because player movement is intrinsic function of FPS games
- Capacity of channel could be reduced
  - Warden introduces noise (random angle fluctuations)
  - Noise in own view angles more easily visible than in other player's character's view angles
  - Covert sender could always send with higher 'power'
  - Warden could target specific players if channel is detected
- Detection is non-trivial, but probably possible

Ongoing and future work

- Reliable message transport: bit synchronisation, framing, error correction etc.
  - Developed bit synchronisation and framing mechanism (unicast)
  - Throughput of 2–13 bits/s
  - No synchronisation errors, even with packet loss ($\leq$ 1%)
  - No substitution errors
- More and longer trials to better understand performance and limitations (bots and humans)
- Develop efficient detection mechanism
  - Statistical tests
  - Anomaly detection or machine learning methods

Conclusions

- Developed novel covert channel in first person shooter online game traffic
- Channel is not limited to FPS games, but could be used for other game types
- Throughput is low, but sufficient for short text messages or chatting
- Covert channel cannot be eliminated
- Detection is non-trivial as covert channel looks similar to normal traffic (but probably possible)

- CCHEF is available, but Q3 module is not public yet (http://caia.swin.edu.au/cv/szander/cc/cchef/)

Acknowledgements

- Many thanks to Lucas Parry, Lawrence Stewart and Kewin Stoeckigt for being tough opponents in the Q3 tests