Covert Channel in Smart Phones

Wade C. Gasior, Li Yang
University of Tennessee at Chattanooga
Implementing CC’s on Android

Timing Covert Channel Design

• Timing-based
  – Need: large amount of legitimate traffic
  – Implemented IP camera application
  – Message encoded in delays between video frames (binary)

– Server displays streamed video
– Server decodes and displays messages
Implementing CC’s on Android

Storage Covert Channel Design

• Storage-based
  – Small advertisement banner shown at bottom of app
  – App requests new banners “randomly” (http)
    • Specific ads represent specific input symbols
    • Encode messages in hex
  – Server decodes and records messages
Implementing CC’s on Android

**Challenges**

- Access to sensitive data
  - Fine-grained application permissions (suspicious if many)
  - Sandboxed runtime environment (no inter-application communication)
- No low-level packet access
  - Dalvik VM (Java)
- Cellular network
  - High jitter, high latency
  - UDP packets often dropped
Implementing CC’s on Android

Solutions

• Access to sensitive data
  – On-device covert storage channels (Schlegel)

• No low-level packet access
  – Use timings between events (TCP/UDP messages) rather than packets (custom protocols)
  – Disable Nagle’s algorithm

• Cellular network
  – Use larger delays
  – Use TCP only