Side Channel Sensors

Daniel Everson
Side Channel Attacks – How Do They Work?

- Idea is to gather sensitive information passively
- Doesn't require direct access
- The less direct the access the better
  - Obvious channels are more suspected, and thus better secured
- Usually requires filtering through data that the target didn't realize was sensitive
Methods of Side-Channel Sensing

- Physical keylogging can be intrusive and obvious
- Are there other ways to log keystrokes? (Preferably ones that don't have a glowing arrow pointing them out?)
Logging Keystrokes Like a Boss

- Acoustic Attacks
  - One of the more famous passive keylogging techniques [1].
- Power Monitoring
- Lasers (!)
- EM Attacks
  - Van Eck Phreaking [2].
More on EM

- 1960s, government was aware of possibility of information gathering from EM emission
  - 75% of plaintext recovered at a distance of 80ft.

- 1980s, Van Eck read plaintext from CRT displays with $15 of equipment at hundreds of meters [2].
Other Ways to Gather Information?

- These methods use sensors to grab unprotected information, and then filter it.
- We don't even have to spend $15, because people equip themselves with sensors 24/7.
  - Which ones can we get at?
Soundcomber – What it is

- Designed as “a trojan with few and innocuous permissions”
- Sophisticated trojan on Android platform, exploiting voice capture
- An example of a known risky sensor being used illicitly
Soundcomber – How it Works

- Intelligently extracts sensitive information using microphone
  - Filters out recognizable number sequences (PINs, CC numbers)
  - Knows about automated call centers and can direct you to input sensitive information
- Has access to a restricted sensor, but doesn't pair it with internet access, to seem legitimate
Soundcomber – How it Works

- How can it transfer the sensitive data?
  - As a standalone trojan, it could transmit small amounts by opening an “ad”
  - Really a method of transmitting information via query string

- Even better – pair it with another app
  - Soundcomber covertly communicates with a “Deliverer”
Covert Android Communication

- It turns out there are a ton of fun ways to leverage the Android OS to send nefarious messages
  - Setting Vibration Settings – 87bps
  - Setting Volume – 150bps
  - Wake-lock – 5bps
  - Orchestrated system of file-locking – 685bps
The File-Locking Method
Soundcomber - Mitigation

- Android only performs static permission checks
- Authors created a controller which can be told that a voice call is “exclusive” or “non-exclusive”
  - On exclusive calls, even apps with audio data permissions only get silence
Less Obvious Sensor Attacks

- While permissions to microphone and camera are explicitly tightly controlled in Android and iOS, some sensors are not.
- Accelerometer data can be used in some tricky ways to infer keystrokes.
  - Apps with accelerometer access are not generally flagged as dangerous.
  - Simultaneous internet access permissions are not a red flag.
Inferring Keystrokes from Accelerometer Data
Inferring Keystrokes from Accelerometer Data

- This sensor method is not as fine as some others
  - Sampling rate for audio ~ 44.1kHz with standard equipment
  - Sampling Rate for accelerometers ~ 100 Hz

- Nyquist Rate describes minimum sample rate given duration of activity to be captured
  - Keypress ~0.1sec
  - 100Hz = 0.01sec sample rate
Inferring Keystrokes from Accelerometer Data

- Prior to attack phase, a training phase is conducted
- A trained NN along with a dictionary is deployed during the attack phase
Inferring Keystrokes from Accelerometer Data

- Training data averaged into “Feature Vectors”
  - Using FFT, cepstral features
- Prediction accuracy at first is sub-par
  - 25% compared to 78% with acoustic methods
Inferring Keystrokes from Accelerometer Data

- Pairing of letters allows for better recovery
- Keypresses modeled on left/right and near/far attributes
- Word of length $n$ broken to $n-1$ pairs
  - Canoe $\rightarrow$ ca – an – no – oe
Inferring Keystrokes from Accelerometer Data

- The example word, canoe, turns to this:
  - LLN.LRF.RRF.RLF
- Intermediary format is fed to dictionary for matching
- Training set is carefully balanced
Inferring Keystrokes from Accelerometer Data

- Using these 2 Neural Nets raises accuracy significantly
  - 56% of the time, top 50 word choices contained correct word
  - compared to 72% for acoustic methods in similar test conditions

- Targeted dictionary further improves recovery
  - 80% in top 5 word predictions
Inferring Keystrokes from Accelerometer Data

- This method relies on proximity to keyboard
- Strict training conditions might decrease viability in real-world scenarios
Inferring Keystrokes from Accelerometer Data - Mitigation

- The publishers suggest not setting the phone next to your keyboard...
- Certain surfaces were less likely to conduct vibrations effectively
- Leveraging the idea of Nyquist Sampling Rate
  - Allowing < 30 Hz resolution to accelerometers by default would make sampling much more difficult
TouchLogger

- This problem is easier using the on-screen keys
  - Similar extraction of feature vectors, but using motion of the phone in space
- On 449 numeric keypresses, 71% correctly inferred
TouchLogger - Mitigation

- No mitigation strategies proposed by authors
- They suspect larger screens would lead to even better results
  - Bad news for tablets
Let's reflect...

- If it's a sensor, someone can probably use it to steal stuff
- With enough filtering and cleverness, even garbage signals can contain useful information
- Basically cover your life in aluminum foil