It's all in your head(set): Side-channel attacks on AR/VR systems

Yicheng Zhang, Carter Slocum, Jiasi Chen, Nael Abu-Ghazaleh

yzhan846@ucr.edu

University of California, Riverside
AR/VR Systems – An Immersive Lifestyle

• Augmented Reality/Virtual Reality (AR/VR) systems are everywhere.
AR/VR Systems – An Immersive Lifestyle

• Augmented Reality/Virtual Reality (AR/VR) systems are everywhere.
• Over 60 Billion US Dollar Market Size.
Outline

• Background: Software and Hardware Architecture for AR/VR Systems.

• Threat model and Leakage Vectors.

• Three Classes of Side-channel Attacks.
  • User Interaction: Hand gestures, Voice input, and Keystrokes.
  • Concurrent Application: Application Launch.
  • Real-world Environment: Bystander Estimation.

• Mitigation.
AR/VR Systems

• Main Components.
  • Developer Tools.
  • Software Development Kit (SDK).
  • Device Hardware.

• Rendering Performance Counters.
  • Track the performance of AR/VR applications.
  • Normal user permission.
  • Hundreds of counters are available.
Threat model – Software Side-channel Attacks

• A malicious program runs in the background.
  • Standard application-level permissions.
  • No physical access.
  • Periodically probes rendering performance counters.

What leakages can be measured?
Leakage Vectors

• Memory Allocation API.
  • Expose memory usages on AR/VR devices.

• Rendering Performance Counters.
  • Unity & Unreal Engine SDK.
  • Frame Rate.
    • CPU/GPU frame rate, Refresh Rate, etc.
  • Thread Counters.
    • Game/Render thread time, etc.
  • Render Task Counters.
    • Number of draw calls, Number of primitives, Vertex count, etc.
Side-channel Attacks Overview

• We demonstrate three classes of attacks.

Spy application runs in the background.
Side-channel Attacks Overview

• We demonstrate three classes of attacks.
  • Spying on user interactions.

*User interaction*

*Spy application runs in the background.*
Side-channel Attacks Overview

• We demonstrate three classes of attacks.
  • Spying on user interactions.
  • Spying on concurrent applications.

*Concurrent applications*

Spy application runs in the background.
Side-channel Attacks Overview

• We demonstrate three classes of attacks.
  • Spying on user interactions.
  • Spying on concurrent applications.
  • Spying on the real-world environment.

*Real-world environment*

*Spy application runs in the background.*
Experimental Setup

- Two popular headsets.
  - Microsoft Hololens 2 (AR).
  - Meta Quest 2 (VR).
- Ten volunteers.
  - Various ages, heights, weights, and gender.
- Attack workflow.

![Side-channel signal](image)

Feature Engineering

Classifiers/regressors
Attack 1: Hand gestures inference

- **Victim**: Directly interacts with digital artifacts via hand gestures.
- **Spy**: Collects special signal patterns depending on victim's hand gestures.
Performance Counter Trace

- "Vertex count".
  - The number of vertices in existing 2D/3D scenes.
Classification Results

• Five basic system-level hand gestures on both Hololens 2 and Quest 2.
• The classification results for hand gestures inference attack:
  • K Nearest Neighbors (KNN).
  • Decision Tree (DT).
  • Random Forest (RF).
  • Light Gradient Boosting Machine (LightGBM).
  • Weighted majority rule voting (Voting).

<table>
<thead>
<tr>
<th></th>
<th>Hololens 2</th>
<th></th>
<th>Quest 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>Prec</td>
<td>Rec</td>
<td>F1</td>
</tr>
<tr>
<td>KNN</td>
<td>53.6</td>
<td>55.4</td>
<td>54.2</td>
<td>57.9</td>
</tr>
<tr>
<td>DT</td>
<td>80.0</td>
<td>80.5</td>
<td>80.0</td>
<td>91.3</td>
</tr>
<tr>
<td>RF</td>
<td>86.6</td>
<td>86.6</td>
<td>86.7</td>
<td>93.7</td>
</tr>
<tr>
<td>LightGBM</td>
<td>84.7</td>
<td>86.7</td>
<td>85.0</td>
<td>89.0</td>
</tr>
<tr>
<td>Voting</td>
<td><strong>89.2</strong></td>
<td><strong>89.3</strong></td>
<td><strong>89.2</strong></td>
<td><strong>91.3</strong></td>
</tr>
</tbody>
</table>
**Attack 2: Voice Commands inference**

- **Victim**: Communicates with the headset through voice commands.

  - "Take a picture"

  - **AR/VR devices**
  - **Speech recognition**
  - **Open camera**
Attack 2: Voice Commands inference

- **Victim**: Communicates with the headset through voice commands.
- **Spy**: Measures a content-related pattern performed by the victim.

![Diagram showing the process of Attack 2: Voice Commands inference]
Memory Allocation Trace

• “AppMemoryUsage” API.
  • Track the spy app’s current memory usage.

“Go to start”  “Take a picture”  “Start/stop recording”
Classification Results

• Five basic headset-specific voice commands on both Hololens 2 and Quest 2.
• The classification results for voice commands inference attack:
  • K Nearest Neighbors (KNN).
  • Decision Tree (DT).
  • Random Forest (RF).
  • Light Gradient Boosting Machine (LightGBM).
  • Weighted majority rule voting (Voting).

<table>
<thead>
<tr>
<th></th>
<th>Hololens 2</th>
<th>Quest 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>Prec</td>
</tr>
<tr>
<td>KNN</td>
<td>87.5</td>
<td>87.7</td>
</tr>
<tr>
<td>DT</td>
<td>93.7</td>
<td>93.8</td>
</tr>
<tr>
<td>RF</td>
<td>91.2</td>
<td>91.3</td>
</tr>
<tr>
<td>LightGBM</td>
<td>88.9</td>
<td>90.9</td>
</tr>
<tr>
<td>Voting</td>
<td>91.3</td>
<td>92.4</td>
</tr>
</tbody>
</table>
Attack 3: Keystroke Monitoring

- **Victim**: Enters keystrokes through virtual keyboard.

"Foreground application"

"Virtual keyboard"
Attack 3: Keystroke Monitoring

- **Victim**: Enters keystrokes through virtual keyboard.
- **Spy**: Monitors rendering performance counters to infer the digit input of a victim.
Performance Counter Trace

• “Game thread time” & “Render thread time”.
  • Track the execution time of two primary threads in applications.

**The victim presses the digit.**
Classification Results

• Ten digits (0-9) on the virtual keyboard on both Hololens 2 and Quest 2.

• The classification results for keystroke monitoring:
  • K Nearest Neighbors (KNN).
  • Decision Tree (DT).
  • Random Forest (RF).
  • Light Gradient Boosting Machine (LightGBM).
  • Weighted majority rule voting (Voting).

<table>
<thead>
<tr>
<th></th>
<th>Hololens 2</th>
<th></th>
<th></th>
<th></th>
<th>Quest 2</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>F1</td>
<td>Prec</td>
<td>Rec</td>
<td>F1</td>
<td>Prec</td>
<td>Rec</td>
<td>F1</td>
<td>Prec</td>
</tr>
<tr>
<td>KNN</td>
<td>43.3</td>
<td>49.6</td>
<td>44.3</td>
<td>44.1</td>
<td>49.4</td>
<td>44.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DT</td>
<td>88.7</td>
<td>89.8</td>
<td>88.6</td>
<td>92.1</td>
<td>93.7</td>
<td>92.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RF</td>
<td>52.1</td>
<td>54.0</td>
<td>52.9</td>
<td>73.7</td>
<td>75.5</td>
<td>75.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LightGBM</td>
<td>87.5</td>
<td>88.0</td>
<td>88.8</td>
<td>93.8</td>
<td>94.8</td>
<td>94.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Voting</td>
<td><strong>91.4</strong></td>
<td>91.7</td>
<td>91.4</td>
<td>90.1</td>
<td>91.6</td>
<td>90.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Attack 4: Concurrent App Fingerprinting

- **Victim**: Launches a concurrent App on AR/VR devices.
- **Spy**: Track performance counters and identify the victim's application.
Performance Counter Trace

• “Frame time”.
  • Time takes for two consecutive frames are shown.

“Microsoft Edge”

“One Drive”

“Mail”
Classification Results

• Twelve applications are profiled on Hololens 2.

• The classification results for concurrent App fingerprinting:
  • K Nearest Neighbors (KNN).
  • Decision Tree (DT).
  • Random Forest (RF).
  • Light Gradient Boosting Machine (LightGBM).
  • Weighted majority rule voting (Voting).

<table>
<thead>
<tr>
<th></th>
<th>F1</th>
<th>Prec</th>
<th>Rec</th>
</tr>
</thead>
<tbody>
<tr>
<td>KNN</td>
<td>33.7</td>
<td>39.4</td>
<td>35.0</td>
</tr>
<tr>
<td>DT</td>
<td>84.7</td>
<td>86.5</td>
<td>85.0</td>
</tr>
<tr>
<td>RF</td>
<td>51.3</td>
<td>53.0</td>
<td>50.8</td>
</tr>
<tr>
<td>LightGBM</td>
<td>85.8</td>
<td>87.4</td>
<td>86.8</td>
</tr>
<tr>
<td>Voting</td>
<td><strong>89.3</strong></td>
<td><strong>91.0</strong></td>
<td><strong>89.2</strong></td>
</tr>
</tbody>
</table>
Attack 5: Bystander Ranging

- **Victim**: Bystander steps into the field of view of an AR/VR device.
Attack 5: Bystander Ranging

• **Victim**: Bystander steps into the field of view of an AR/VR device.

• **Spy**:
  • Profiles leakage vectors.
  • Generates spatial mesh of the surrounding environment.

*Turn an AR/VR device into a surveillance device!*
Performance Counter Trace

- “CPU frame rate”.
  - CPU frame time between two consecutive frames.
- Distance-dependent fingerprint.

Distance = 0.5 meters  Distance = 2 meters  Distance = 4 meters
Regression Results

• Distance ranging from 0.5 meters to 5 meters (0.5, 1, 2, 3, 4, and 5 meters).

• The regression results (meters) for bystander estimation attack:
  • K Nearest Neighbors (KNN).
  • Decision Tree (DT).
  • Random Forest (RF).
  • Light Gradient Boosting Machine (LightGBM).
  • Weighted majority rule voting (Voting).

<table>
<thead>
<tr>
<th></th>
<th>KNN</th>
<th>DT</th>
<th>RF</th>
<th>LightGBM</th>
<th>Voting</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAE</td>
<td>0.401</td>
<td>0.103</td>
<td>0.257</td>
<td>0.279</td>
<td>0.164</td>
</tr>
</tbody>
</table>

10.3 cm
Mitigation

- Managing access to performance counters.
  - Completely blocking access to leaky APIs and counters.
  - Limiting the precision or rate of performance counters.
Conclusion

• Side-channels on AR/VR systems.
  • Through rendering performance counters (First).

• Three AR/VR-specific attack scenarios.
  • Five end-to-end side-channel attacks.

• Mitigation based on limiting the precision or rate is not effective.

• Future work:
  • Multi-user AR/VR systems; better profiling systems for AR/VR.
Thank you!
Any questions?
Yicheng Zhang
yzhan846@ucr.edu
https://yichez.site