Swayam Salunke (225)

**✅ PoC: Homography-Based Scene Manipulation Detection Tool**

**🎯 Tactic: Collection (TA0009)**

*Goal:* Gather data from a compromised visual source and detect spoofed or manipulated imagery using homography matching.

**🔧 Technique 1: T1113 – Screen Capture / Visual Data Collection**

**Procedure:**

1. Attacker captures images from surveillance feed using compromised CCTV camera.
2. The attacker injects fake static frames to hide activity (e.g., theft).
3. Our tool loads two sequential frames:
   * image1.jpg (original)
   * image2.jpg (tampered)
4. Using homography, the tool compares feature correspondences between both images.
5. Mismatches or unaligned transformations suggest manipulation or spoofing.

**🔧 Technique 2: T1203 – Exploitation for Client Execution**

**Procedure:**

1. Attacker tricks the system into executing image injection code through a backdoor (e.g., uploading frames via an unsecured web panel).
2. Injected image slightly differs from original but attempts to maintain spatial consistency.
3. The homography tool detects inconsistencies in perspective, scale, and rotation – revealing the frame injection attempt.

**🔧 Technique 3: T0887 – Sensor Manipulation (ICS/IoT matrix)**

**Procedure:**

1. Adversary uses rogue IoT device to spoof video input.
2. Homography tool compares static camera feed vs. live feed.
3. When artificial frame is detected (no proper keypoint match or warped perspective fails), it alerts for tampering.

**✅ Tool Behavior: Homography Detection Flow**

| **Step** | **Description** |
| --- | --- |
| 1 | Load image1.jpg and image2.jpg |
| 2 | Extract features using ORB |
| 3 | Match keypoints using Hamming BF matcher |
| 4 | Compute homography matrix |
| 5 | Perform perspective transform |
| 6 | Detect visual manipulation if error in transformation > threshold |

**🛡 Detection Indicators**

* Mismatch in homography alignment
* Drastically different keypoint matches across frames
* Failure to compute homography due to lack of consensus (RANSAC breakdown)

**🔐 Mitigation & Response**

| **Issue** | **Mitigation** |
| --- | --- |
| Image injection in feed | Use signed frames or watermarking |
| Lack of consistency in frames | Deploy real-time homography checks |
| Visual spoofing | Compare multi-angle feeds using same tool |

**✅ Why This PoC is Effective**

* Visual data manipulation is increasingly used in **surveillance bypasses**, **deepfake injection**, or **AR misdirection**.
* Homography allows detection of **frame drift**, **visual cloning**, or **injected overlays**.
* Effective in both static analysis and real-time video analytics environments.

**🧪 Optional Extensions**

* Add real-time webcam feed support.
* Include deepfake detection via AI model + homography consistency.
* Integrate alerts on excessive perspective error.