Endangered Privacy:
Large-Scale Monitoring of Video Streaming Services

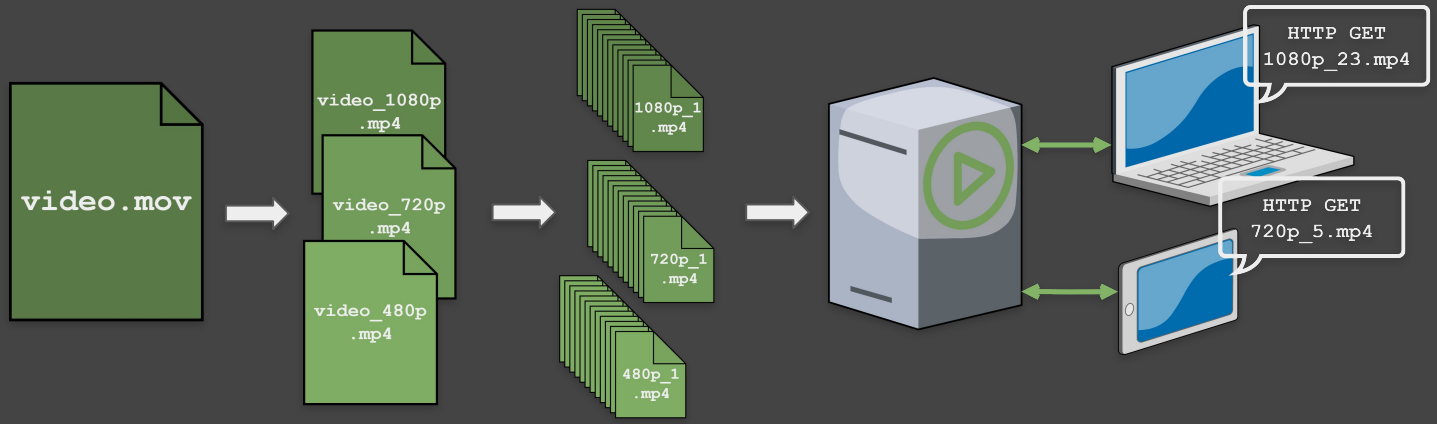
Martin Björklund, Romaric Duvignau


CHALMERS
UNIVERSITY OF TECHNOLOGY
|

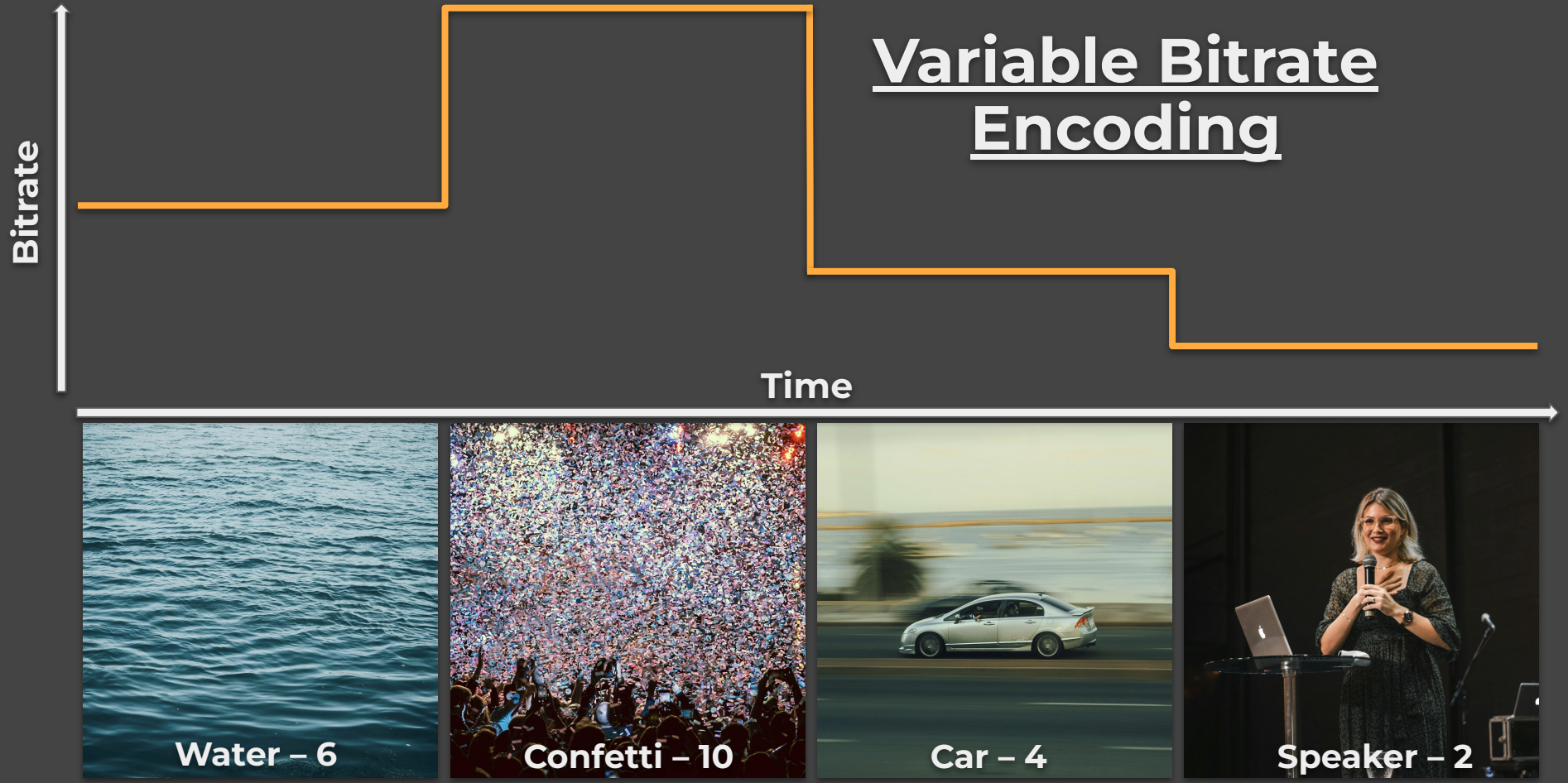
UNIVERSITY OF GOTHENBURG

Adaptive Bitrate (ABR) Streaming

- **HTTP-based** – integrates well with existing infrastructure and content-delivery networks
- Video encoded at **various bitrates** and split into **segments**
- Client fetches **manifest** to facilitate the stream (e.g. find available qualities, locate segments)

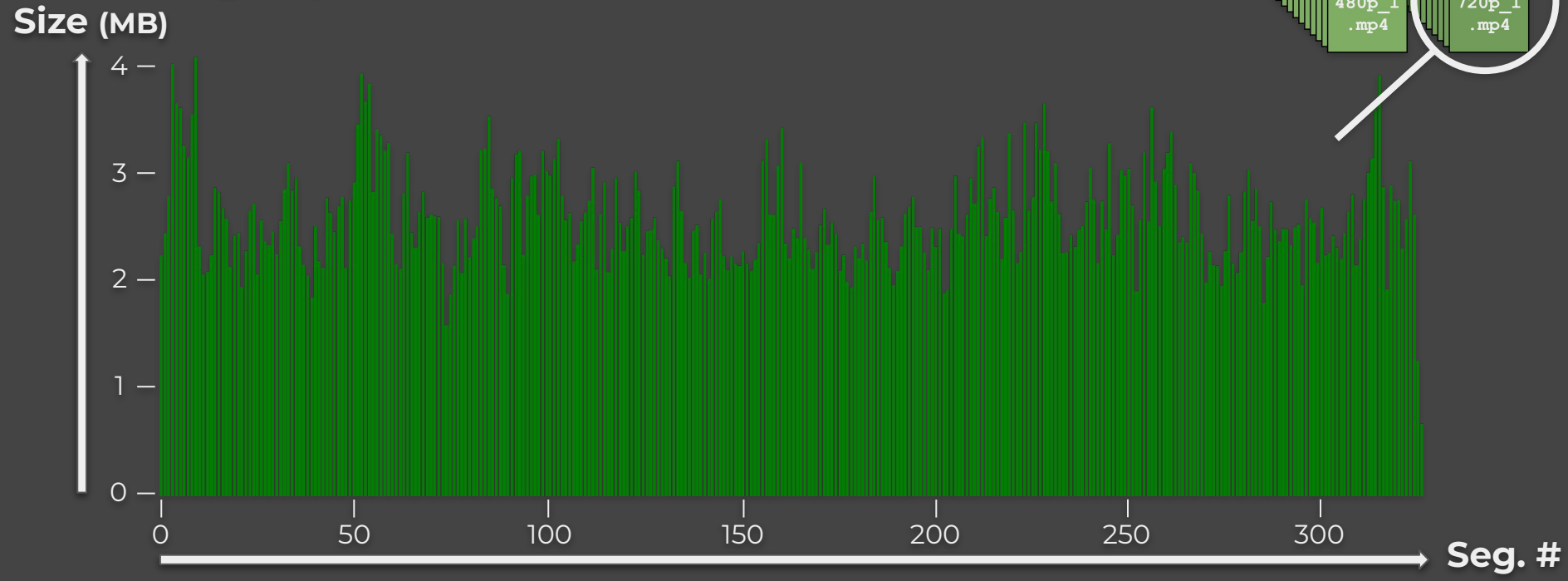


Variable Bitrate Encoding



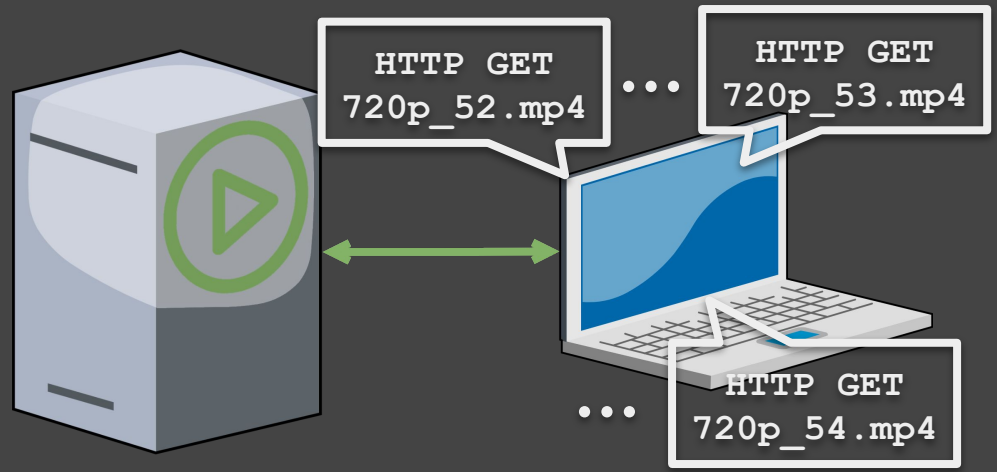
Segment Sizes Become a Fingerprint

- Variable bitrate encoding results in **variable sized segments**
- **Fingerprint** for *Friends S1E1* on Max



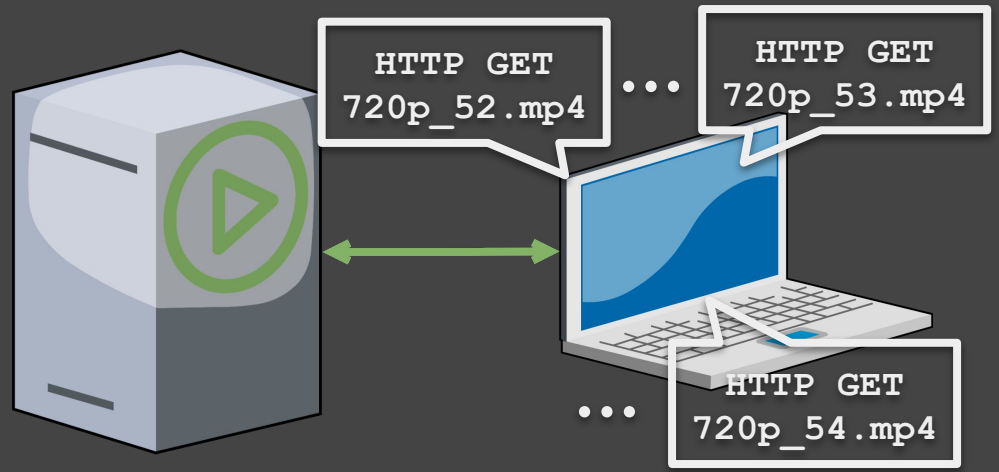
Adaptive Bitrate (ABR) Streaming

- ABR streaming in a nutshell: **downloading** and **playing** segments
- Common strategy: new **segment** requested when **buffer** goes below some threshold
- Consequence: a **bursty** network pattern



ABR Streaming Leaks Information

- Network traffic analysis – packet **sizes** & **times**
- Size of burst ≈ size of segment
- Consecutive bursts can leak the **exact video** being watched



Current Landscape

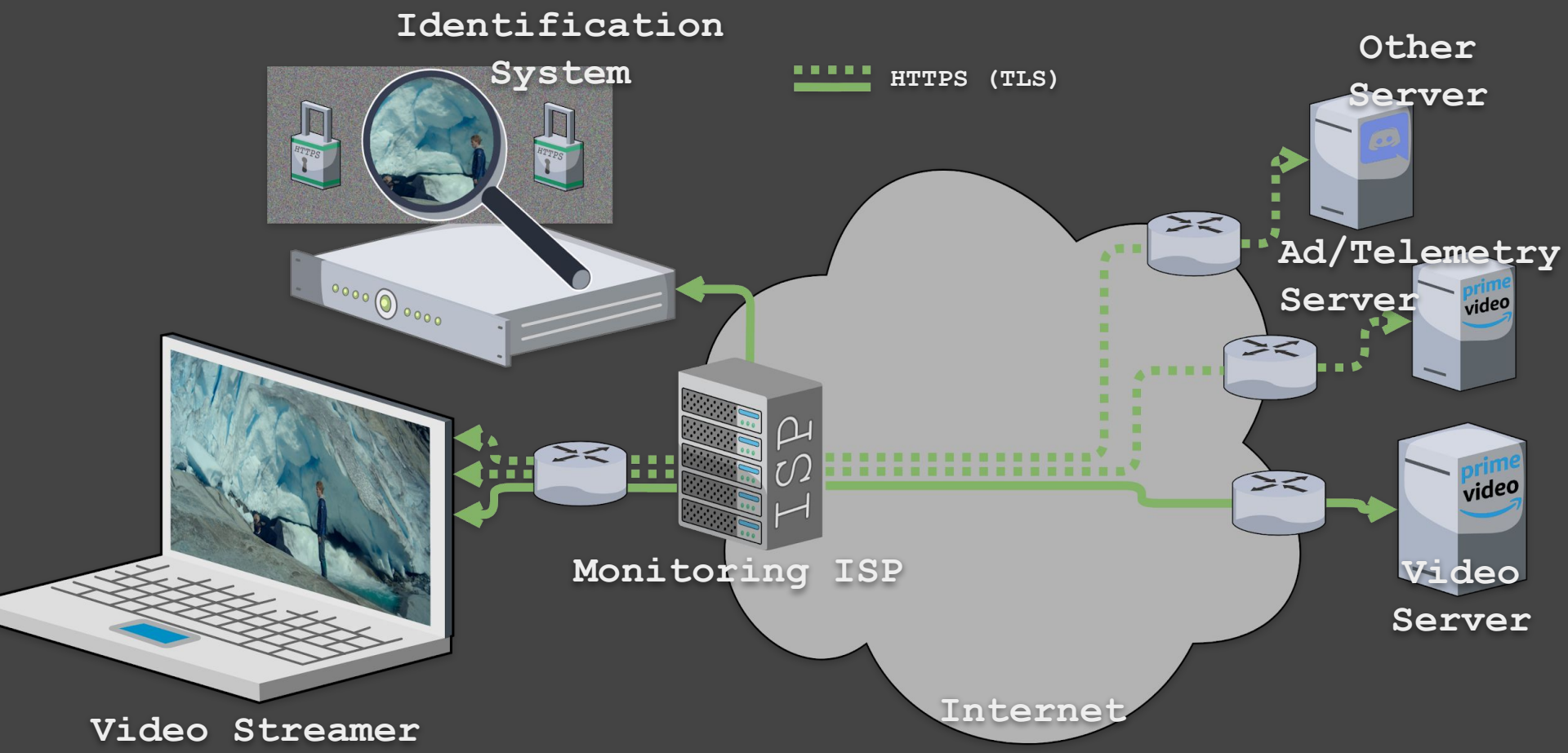
- Attack has been shown to work
- Two schools of thought: **ML** vs. **fingerprinting**
- Impractical due to strong assumptions
  - Small datasets with base rate neglect
  - Costly data collection
  - Requires specific network characteristics

This Work

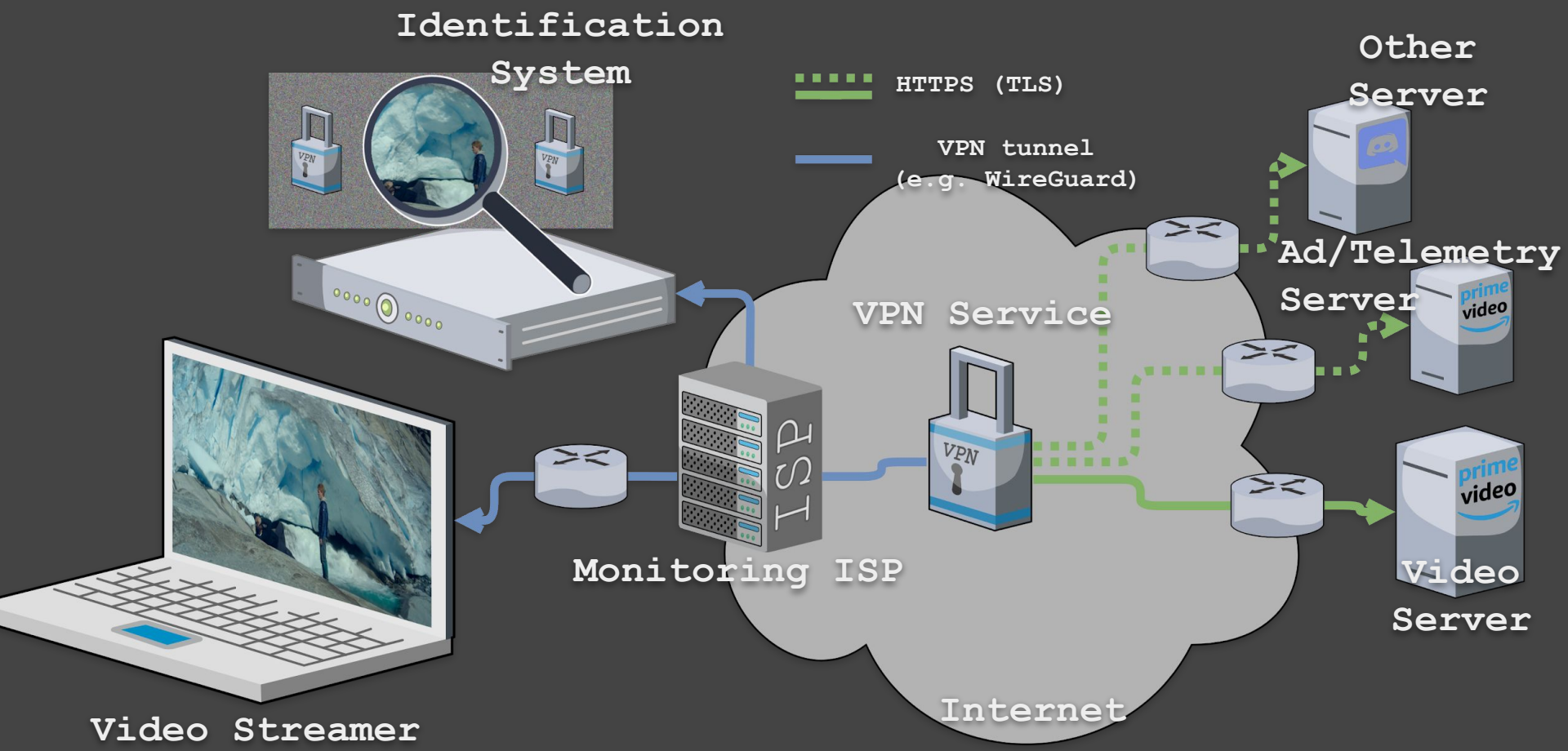
- Aim to convincingly demonstrate that the attack is **practical** to deploy for **monitoring** of **entire streaming services**
- Largest dataset of monitored videos by far
  - **240,000** videos
  - **3,000,000** fingerprints
  - Facilitated by efficient data collection
- Protocol-agnostic – works on **VPN** and **Wi-Fi**

Attack Models

Attack Model – “strong”



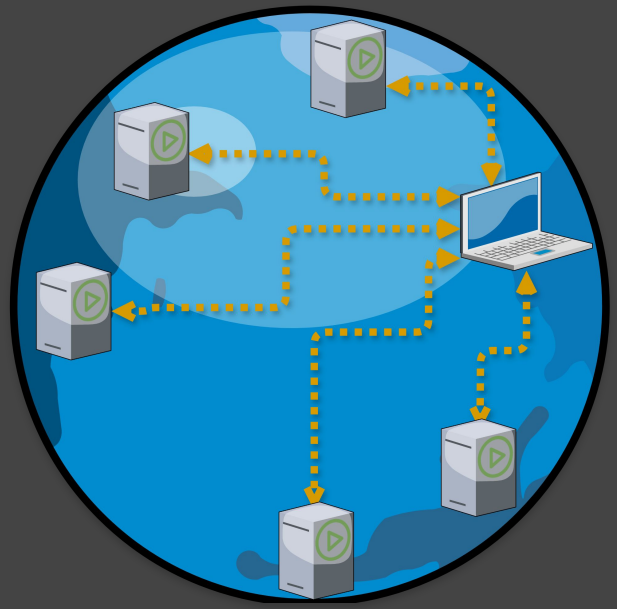
Attack Model – “weak”



Attack Outline

1: Data Collection

- Dedicated program that obtains **fingerprints** for all encodings
- Leverages ABR **manifests**
- **3,000,000** fingerprints extracted in less than **1 day** from a single host
- Shows that data collection is not a bottleneck!



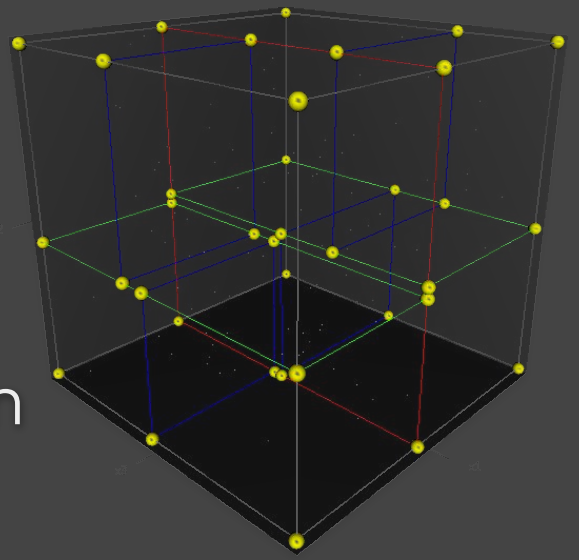
svt

prime video

max

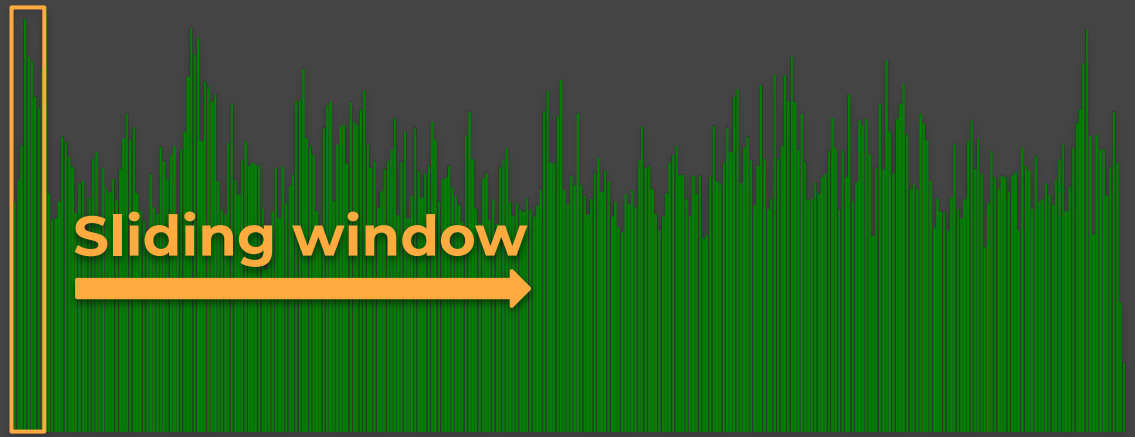
2: Data Organization

- ***k*-d tree** – space partitioning data structure for fast **nearest neighbor** searches
- **8-dimensional** points extracted from each of the 3 million **fingerprints**



Technical details

- **~1.89 billion** points
- **~18 ms** query time
- **~152 GB** in memory



3: Network Capture

- **TShark wrapper** that aggregates data packets into larger **bursts** of ***N*** bytes on multiple network flows in real time or from capture files
- *Remember* – **bursts** analogous to downloaded **segment sizes** by the streaming target
- **Bursts** used to query ***k*-d tree**

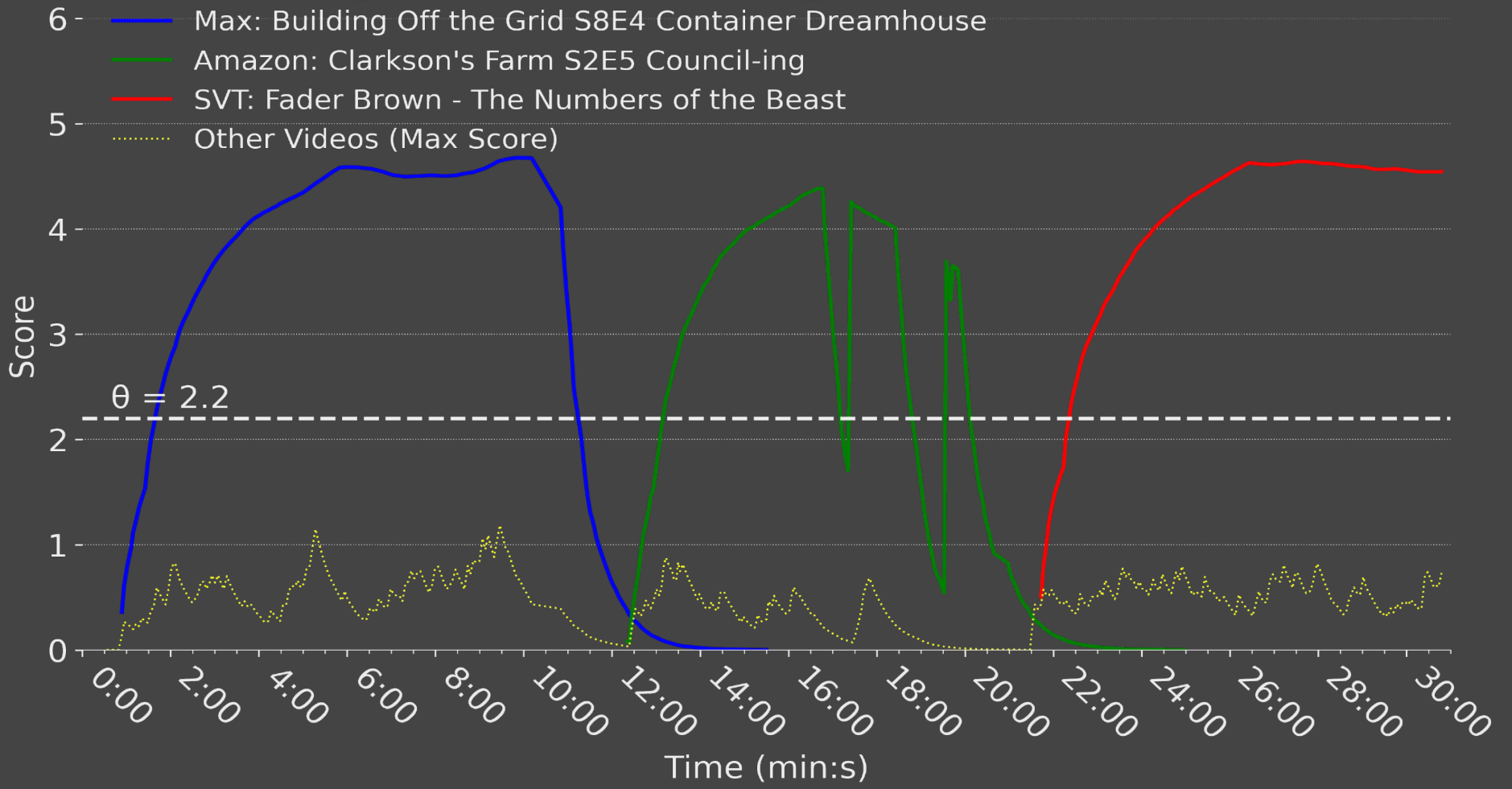
Snapshot of output

96	6.794086700	46.59.50.129	443	192.168.1.143	57776	1716127206.863761663	1716127206.919014454	29	93500
97	6.794214454	185.102.103.6	443	192.168.1.143	48326	1716127205.785032034	1716127206.907913446	752	19896658
98	7.319919905	46.59.50.129	443	192.168.1.143	57792	1716127206.863753796	1716127206.994019508	149	2295572
99	8.399518972	192.168.1.143	48344	185.102.103.6	443	1716127208.020317316	1716127208.020317316	1	505
100	8.399562165	192.168.1.143	48326	185.102.103.6	443	1716127208.019775152	1716127208.019775152	1	519
101	8.403792426	185.102.103.6	443	192.168.1.143	48326	1716127208.030199528	1716127208.089705944	70	1356396
102	8.403883709	185.102.103.6	443	192.168.1.143	48344	1716127208.030192375	1716127208.077608347	31	93721
103	10.450369475	192.168.1.185	43	192.168.1.255	53744	1716127210.284352541	1716127210.284352541	1	35
104	10.450414386	162.159.135.234	443	192.168.1.143	50630	1716127209.978724480	1716127209.978724957	2	98
105	11.005702572	192.168.1.143	57792	46.59.50.129	443	1716127210.857568026	1716127210.857568026	1	526
106	11.005720243	192.168.1.143	57776	46.59.50.129	443	1716127210.858245373	1716127210.858245373	1	518
107	11.006808038	46.59.50.129	443	192.168.1.143	57776	1716127210.866856098	1716127210.885960340	23	93760
108	11.531504969	46.59.50.129	443	192.168.1.143	57792	1716127210.867540598	1716127210.954466343	98	1986579
109	12.217996322	192.168.1.143	48344	185.102.103.6	443	1716127212.022902727	1716127212.022902727	1	505
110	12.218053371	192.168.1.143	48326	185.102.103.6	443	1716127212.021765232	1716127212.021765232	1	519
111	12.219307734	185.102.103.6	443	192.168.1.143	48344	1716127212.041018963	1716127212.066549301	26	93594
112	12.746025682	185.102.103.6	443	192.168.1.143	48326	1716127212.041031122	1716127212.164145708	103	1988000

4: Identification Method

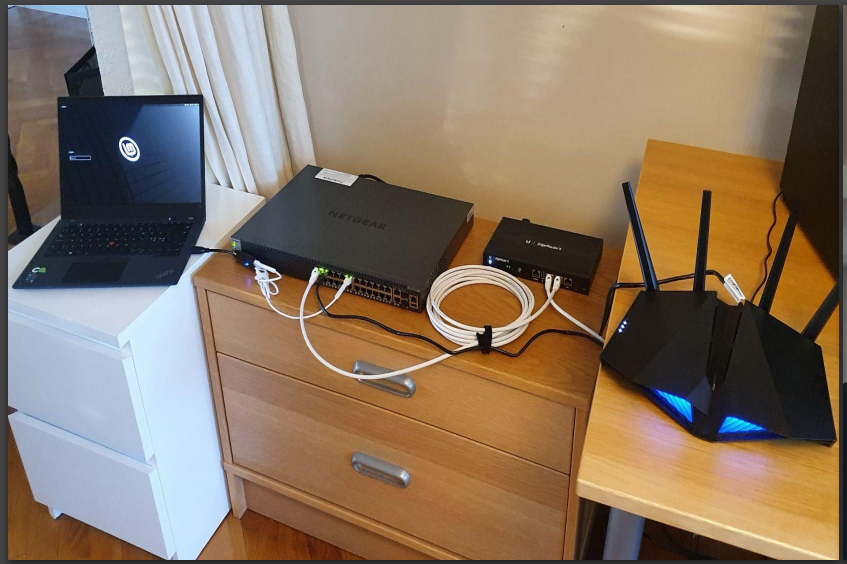
- ***k*-d tree** query generates **candidate** videos
    - Cannot be confident in a single query due to low entropy, noise etc.
- **Heuristic** – the correct video should appear more often with a closer distance than other videos
- System predicts best-scoring video in each time step that is above a set threshold θ

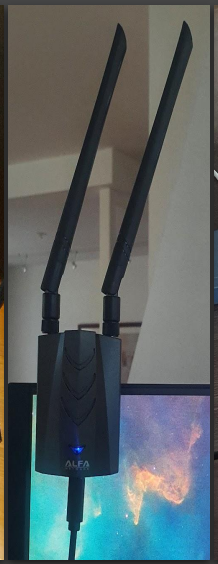
System Scoring Over Time

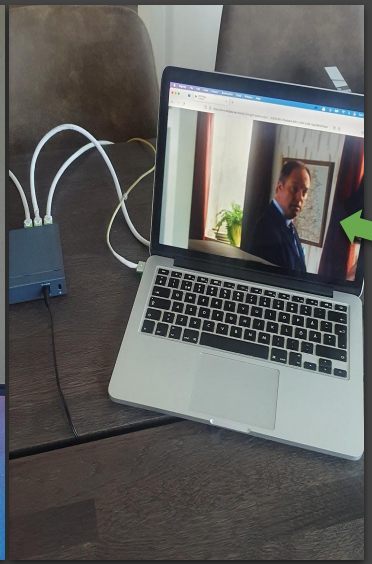


Evaluating the System

- **900** non-VPN and **840** VPN streams in an authentic network environment
    - Random videos played for **10** minutes from a random starting position
- Target devices: **Linux**, **Mac**, **Windows** laptops



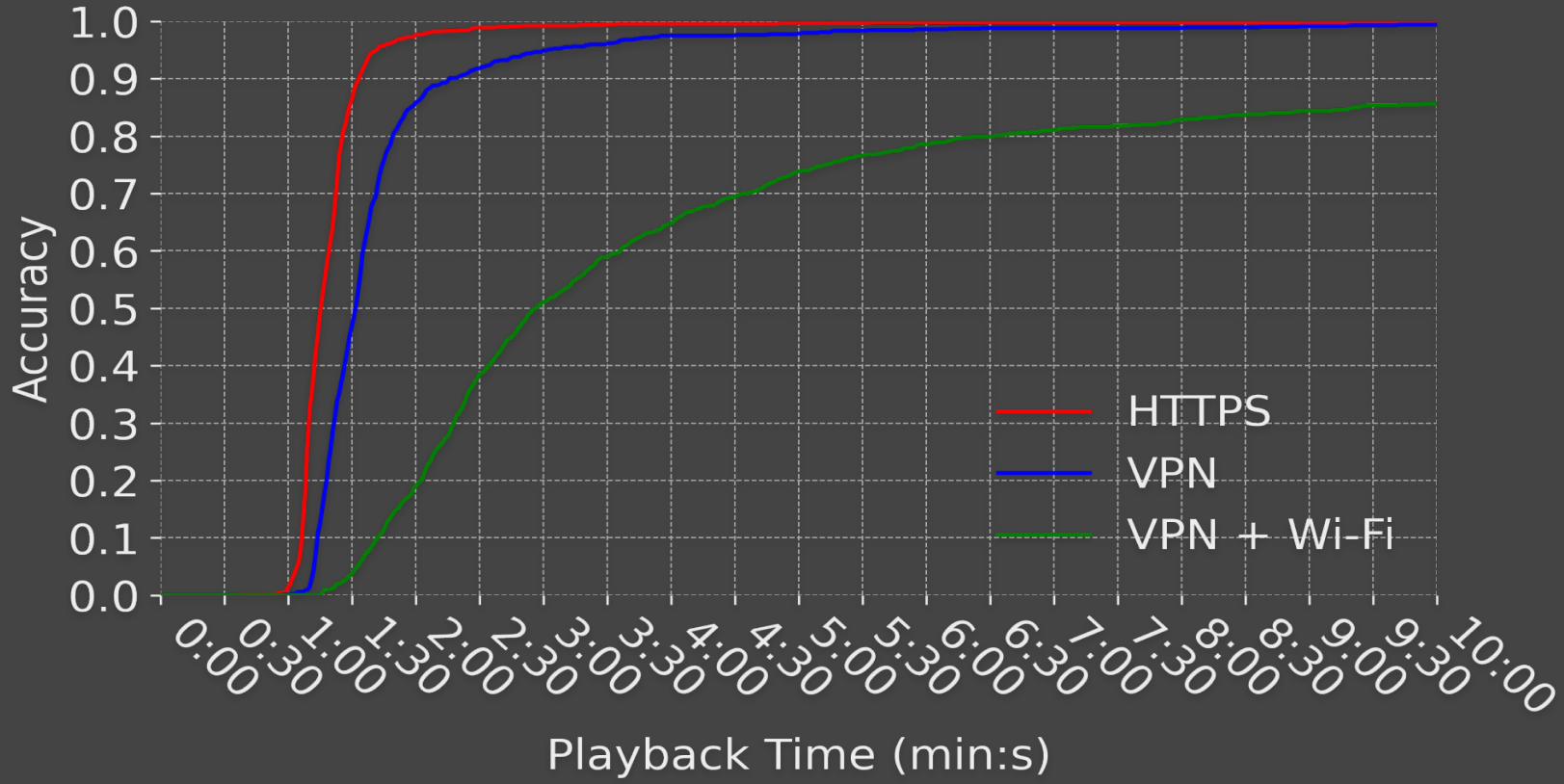




Automated web driver script

Evaluation Results

- **99.5 %** accuracy at **10 minutes** of streaming and eavesdropping. **Zero** false positives.



Mitigating the Attack

- **Traffic Manipulation**
  - The obvious solution?
- **Costly!**
  - **~2.2x** more data use
  - **Q**uality **o**f **E**xperience
  - **Good option** when **privacy** is **paramount!**

- The leak is **inherent** to **ABR streaming** itself
  - Trade-offs: **privacy** vs. **cost** & **QoE**
  - **Mitigation strategy** proposed in paper, revolving around buffer management

Conclusion

- Monitoring videos on **entire video services** is not just possible, but **practical**
- **VPN** does not offer full protection against this attack!
- **Mitigative** actions by streaming services should be taken **promptly**