Mitigating Network Side Channels using Secure and Efficient Traffic Shaping

MAX PLANCK INSTITUTE FOR SOFTWARE SYSTEMS

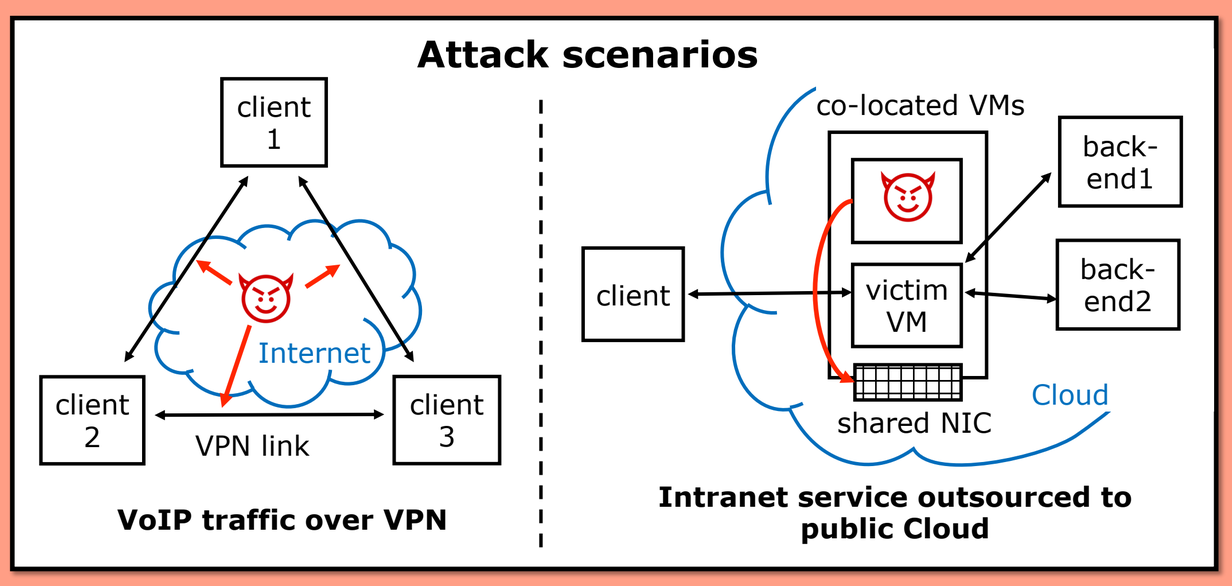
Aastha Mehta, Mohamed Alzayat, Roberta De Viti, Björn Brandenburg, Peter Druschel, Deepak Garg

1. **Problem:** Network side channels can reveal application secrets

Network side channels
- traffic observation in the network (e.g. by ISP)
- traffic sensing via contention at bottleneck links (in a Cloud)

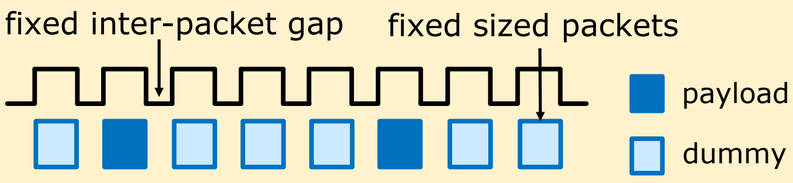
Traffic shape
- packet size, timing, bandwidth, etc.

Secrets
- traffic content (e.g. VoIP, Web)
- application secrets (e.g. crypto keys, medical records)



2. **Solution:** Make traffic shape independent of secrets

Strawman: uniform packet stream



✗ high latency or bandwidth overhead for bursty traffic

Our approach: allow variations in traffic shape based on public information

Workload-partitioned shaping
- Partition workloads by public inputs
- Select different shape for each partition

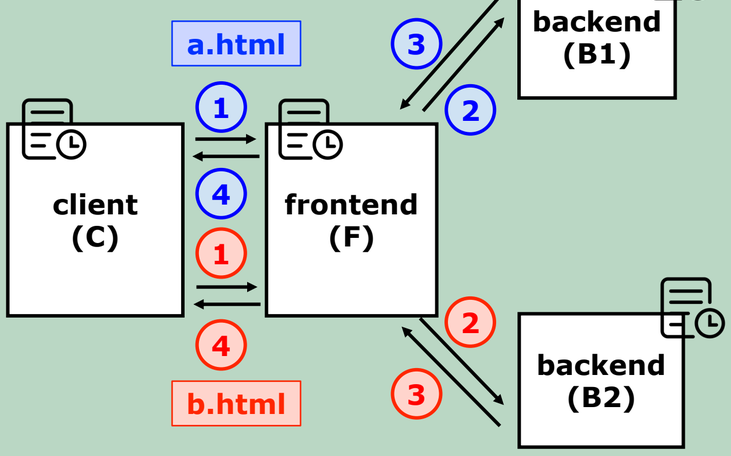
HTML video

Per-request shaping
- Shape only in response to client requests
- (Assume: time of client requests does not reveal secrets)

a.html b.html

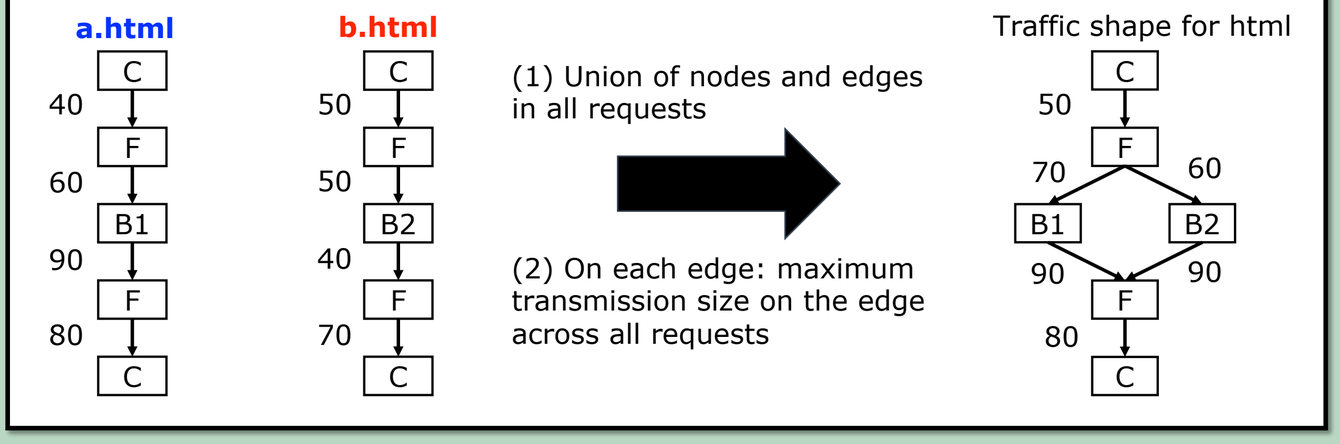
3. Compute traffic shape using **distributed profiling**

Step 1: Distributed tracing
- capture message causalities, network packet trace



Step 2: Traffic shape as a directed acyclic graph
- Subsumes communication in all requests

Example: size shaping (edge labels denote message sizes)



4. Enforcement using **traffic-shaping tunnel between each node pair**

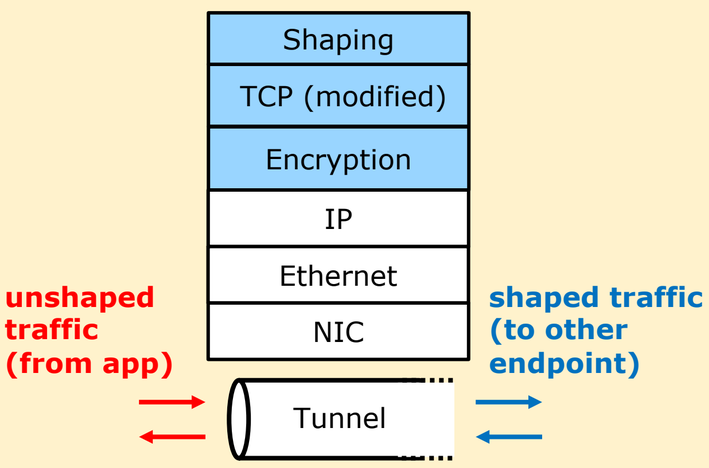
Tunnel requirements

Payload obfuscation
- Hide flow control
- Pad packet size at/above TCP
- Encrypt packets after padding

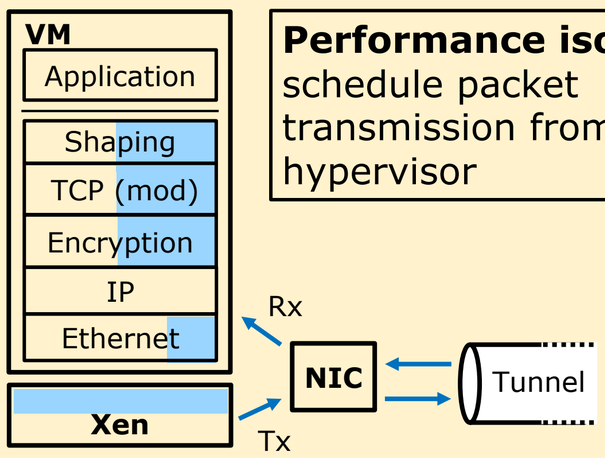
Secret-independent transmission
- Transmit only at scheduled times
- Performance-isolate transmission from app, secrets

Congestion control
- Only to ensure network stability (no implications for confidentiality)

Conceptual endpoint design



Realization on end host



Performance isolation: schedule packet transmission from hypervisor

Realization on middlebox