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
  - Fixed inter-packet gap
  - Fixed sized packets

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

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**

**Realization on middlebox**

---

**Attack scenarios**

- Co-located VMs
- Intranet service outsourced to public Cloud

---

**Workload-partitioned shaping**

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

**Per-request shaping**

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