Application Identification in Information-poor Environments

Charalampos (Haris) Rotsos
Computer Laboratory
University of Cambridge
charalampos.rotsos@cl.cam.ac.uk

UNIVERSITY OF CAMBRIDGE

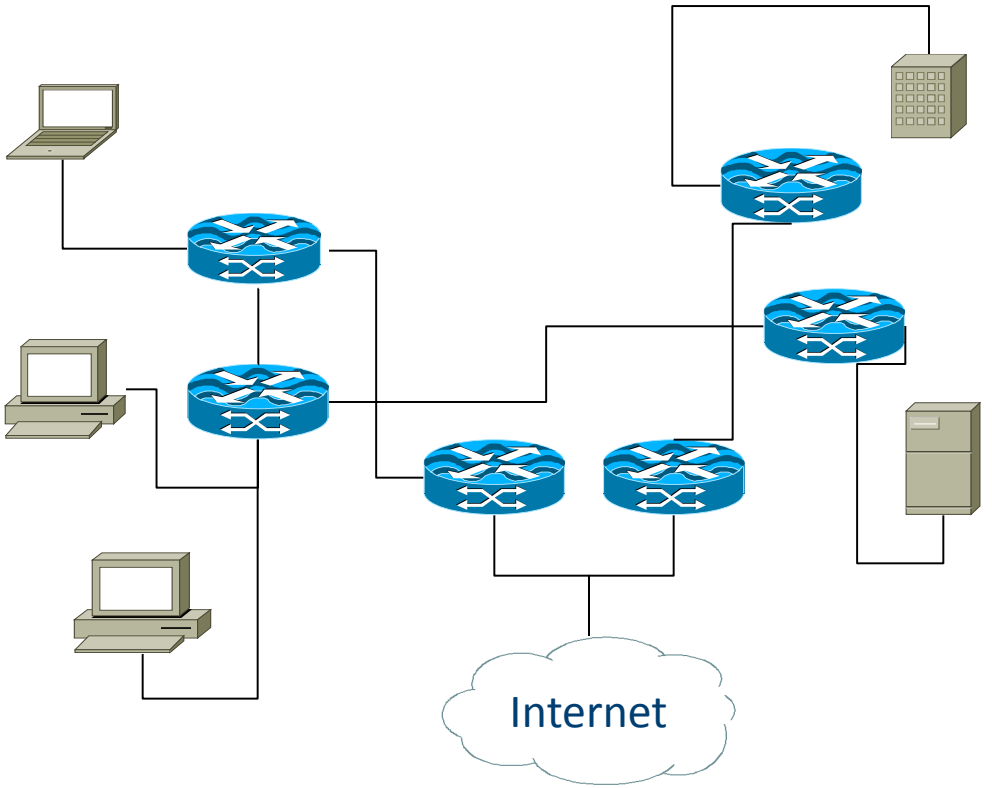
Overview

Application Identification allows:

- New Services (QoS/QoE)
- Administration (SLA)
- Understanding

But it is difficult in a large network because:

- VPN / Multihoming
  - where can I monitor your data?
- Data (2+TB/day/University)
- Sophisticated Users and Complex Networks
  - Encrypted Applications & Overlay Networks



What is the problem

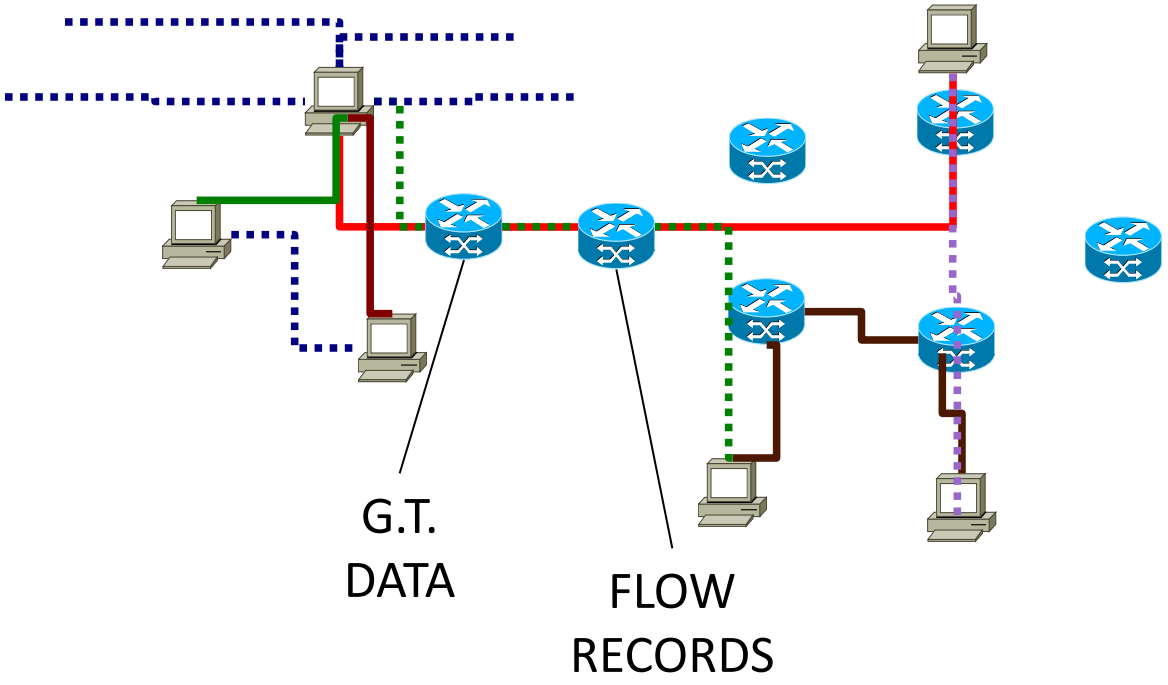
- How can I Identify the application class from a flow of packets?
- Can I do this with sampled and summarised flow records(Netflow)?
  - Available in most routers
  - ISPs collect this as standard and often have been for many years
    - 25Gb per day for a 1st layer ISP (x000's of routers)

Current technologies

IDS / Anomaly detection	fast	depends on protocol implementations specifications, task specific
Deep packet inspection	accurate results	Full payload, fails on encryption & protocol changes
Statistical analysis	Flow granularity, can run online on fast link	Requires diverse ground truth data for training
Connection Pattern / BLINC	low information requirement	host granularity, fails to adjust on small protocol changes, complex design

Can we fuse these different approaches to achieve better performance by reducing the effect of the disadvantages and keeping the advantages?

First Approach



- Using ground truth flow records and machine learning discover patterns from :
  - ✓ Flow statistics
  - ✓ Connection Pattern
  - ✓ Host behavior (roles)

First Problems

- Netflow records have 20 fields. Some of them have no value for the identification.
- Flow records are unclear about client - server role and simplex
- **Hints**:
  - Extract more information from the context of the network.
  - Infer extra fields by analyzing ground truth data. What extra statistics can make a difference?

Time and space variance

An example of temporal decay in accuracy

- A model with 92% accuracy decays to 62-81% accuracy 18 months later

A naïve example of spatial decay

- A model with near 100% accuracy for one site might achieve 87-99%

Long-term fragility comes from changes in IP addresses

coding as AS numbers and subnets help a little (but not much)

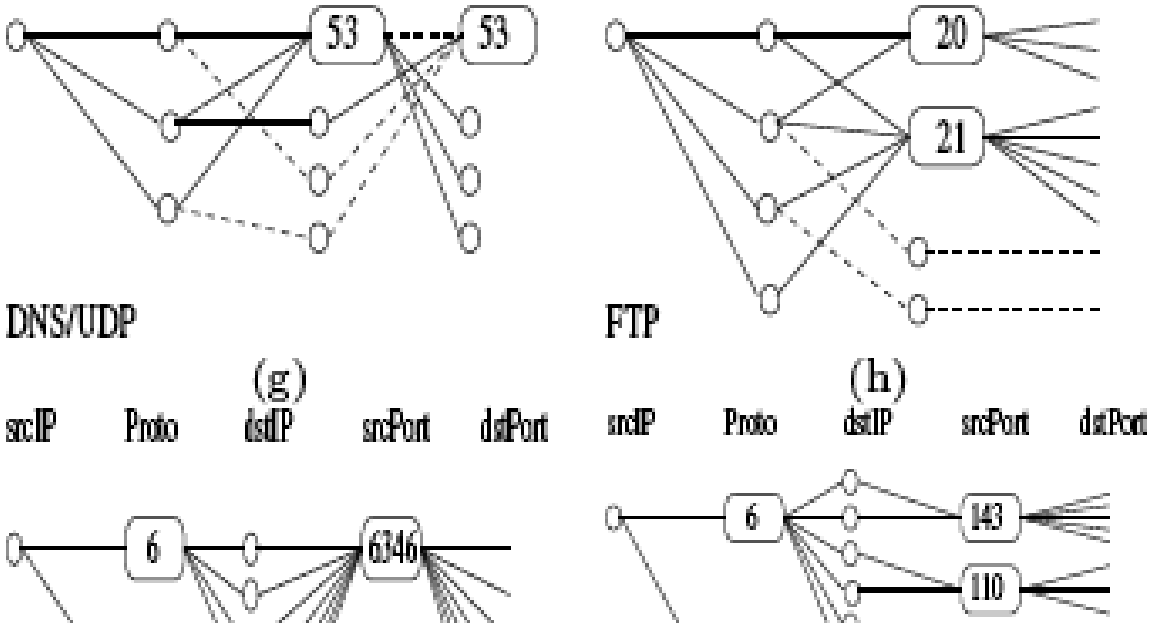
More Issues

- Netflow data tend to be able to describe the situation for short time
- While many servers are stabile for long periods, the heavy-tail is not... (p2p, keyloggers, botnets).

Solutions:

- Mix in prior knowledge; diverse datasets
- Capture behavior with better Mach.-Learn.
- Semi supervised learning to automatic-update

Behavioural models



- What is important for a behavioural model?
- Can we describe it in a compact way?
- Difficult to build automatically

Summary

- NetFlow (flow summary) records are a rich source of data, fused with other network data we can build a useful Application Identification System

- Machine-learning works
  - at least in the short-term
- Stabile/useful models need continuous update
- Behavioural model hold promise too…

THANK YOU