Leveraging other data sources with flow to identify anomalous network behavior

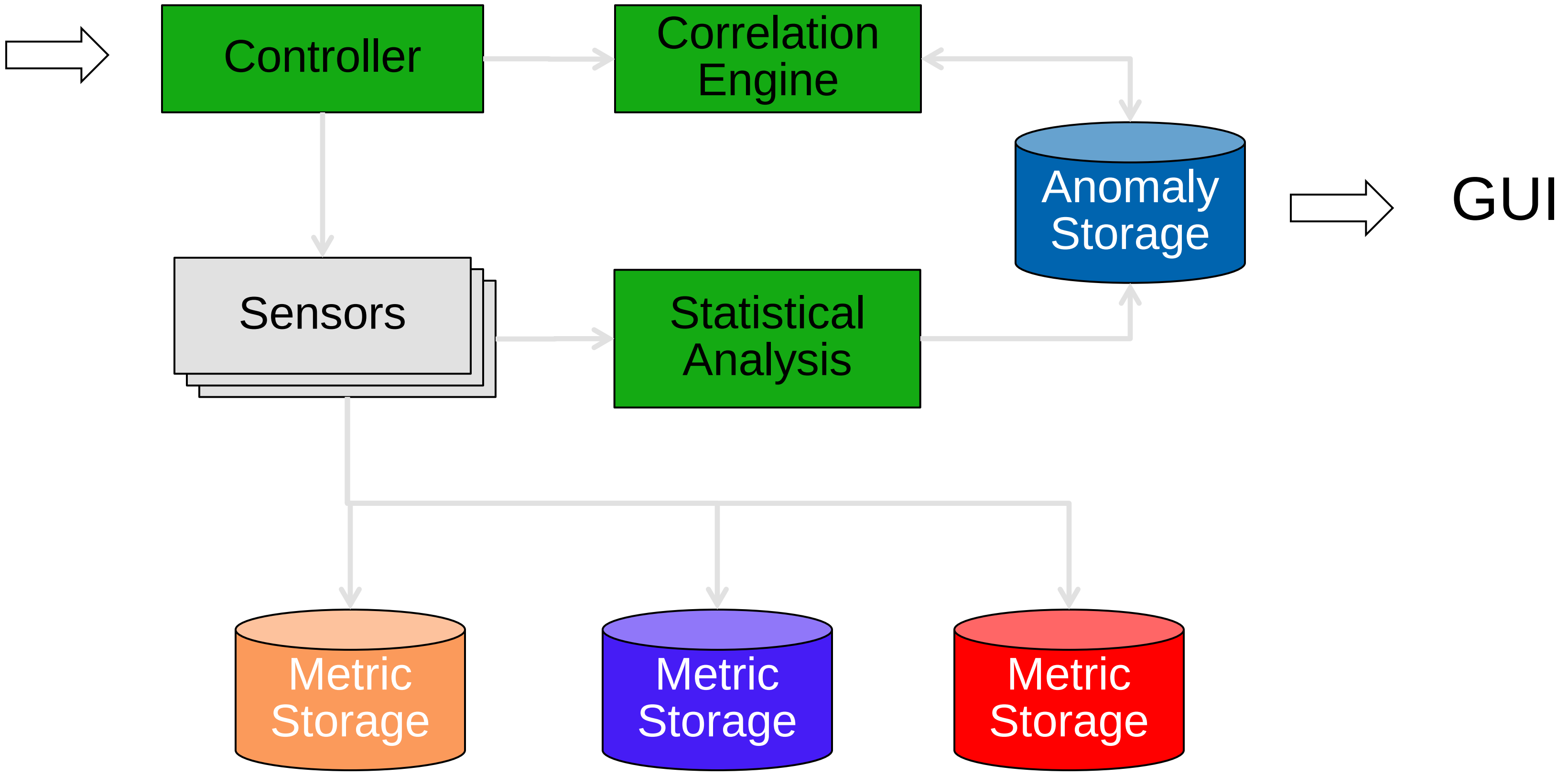
Peter Mullarkey, Peter.Mullarkey@ca.com
Mike Johns, Mike.Johns@ca.com
Ben Haley, Ben.Haley@ca.com

FloCon 2011

Goal and Approach

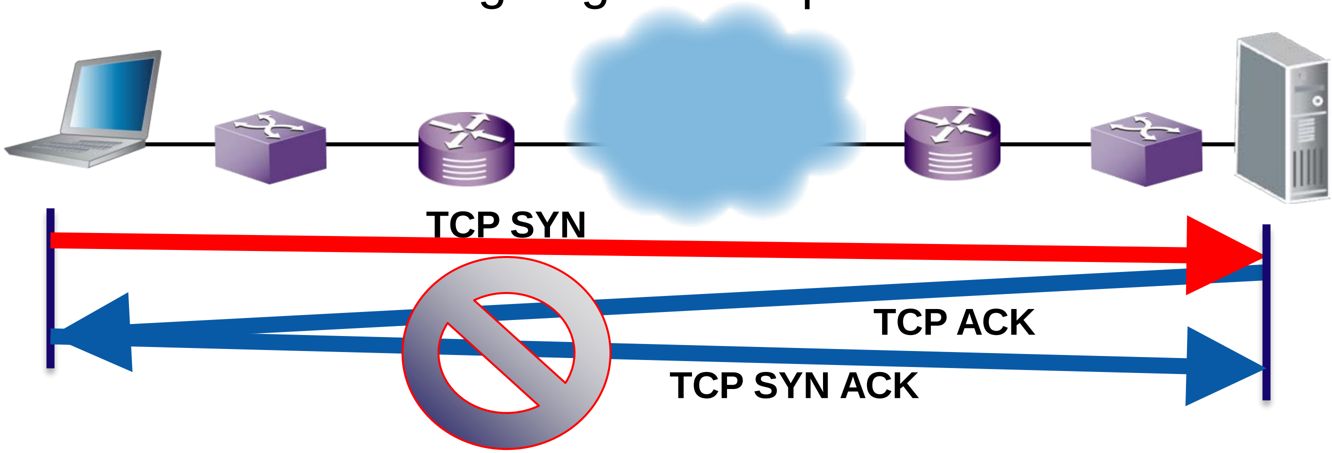
- Goal: Create high quality events without sacrificing scalability
- Approach: Create a system that
  - Is more abstract than a signature-based approach
  - Leverages domain knowledge more than a pure statistical approach
  - Makes use of all available data to increase event quality
  - Relies only on 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

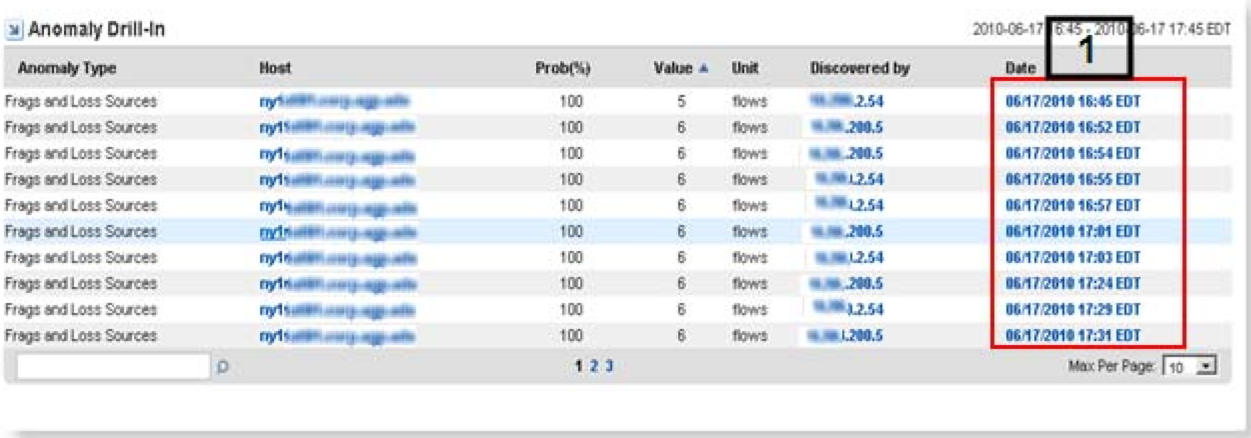
Types of Problems Found

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

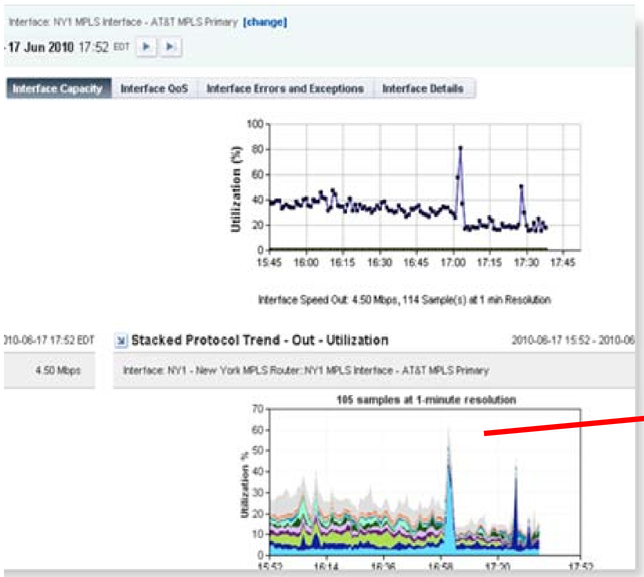
Ny1-x.x.100.52



Anomaly Drill-In 2010-06-17 16:45 - 2010-06-17 17:45 EDT

Anomaly Type	Host	Prob(%)	Value ▲	Unit	Discovered by	Date
Frags and Loss Sources	ny1[redacted]	100	5	flows	[redacted].2.54	06/17/2010 16:45 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].200.5	06/17/2010 16:52 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].200.5	06/17/2010 16:54 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].2.54	06/17/2010 16:55 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].2.54	06/17/2010 16:57 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].200.5	06/17/2010 17:01 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].2.54	06/17/2010 17:03 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].200.5	06/17/2010 17:24 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].2.54	06/17/2010 17:29 EDT
Frags and Loss Sources	ny1[redacted]	100	6	flows	[redacted].200.5	06/17/2010 17:31 EDT

1 2 3 Max Per Page: 10



Interface: NY1 MPLS Interface - AT&T MPLS Primary **[change]**

-17 Jun 2010 17:52 EDT

Interface Capacity | Interface QoS | Interface Errors and Exceptions | Interface Details

Interface Speed Out: 4.50 Mbps, 114 Sample(s) at 1 min Resolution

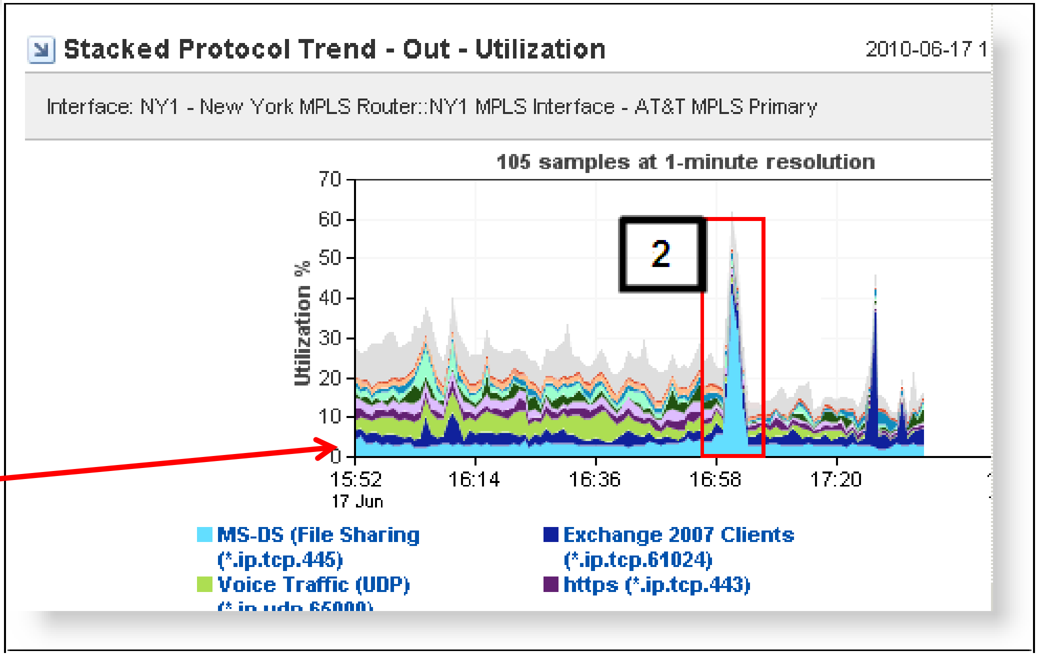
010-06-17 17:52 EDT

4.50 Mbps

Stacked Protocol Trend - Out - Utilization 2010-06-17 15:52 - 2010-06

Interface: NY1 - New York MPLS Router::NY1 MPLS Interface - AT&T MPLS Primary

105 samples at 1-minute resolution
Utilization %
70
60
50
40
30
20
10
0
15:52
16:14
16:36
16:58
17:20
17:52

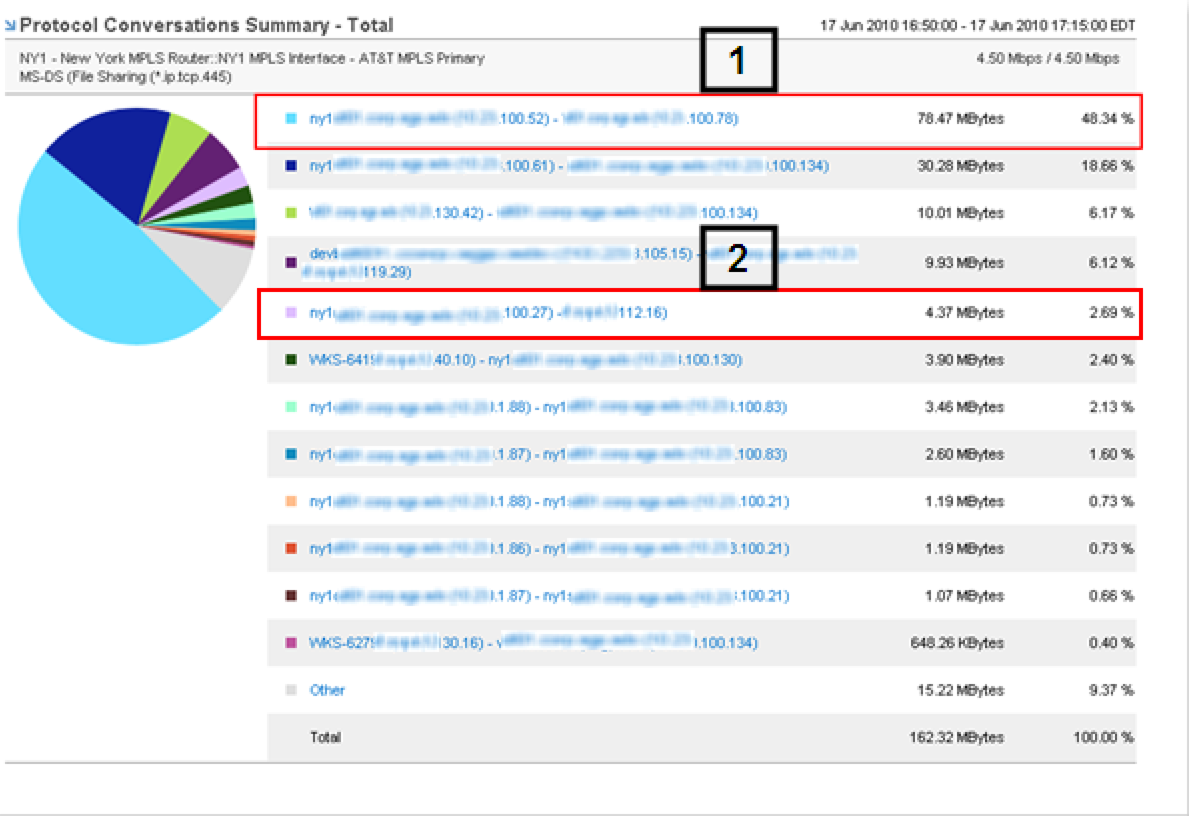


Stacked Protocol Trend - Out - Utilization 2010-06-17 1

Interface: NY1 - New York MPLS Router::NY1 MPLS Interface - AT&T MPLS Primary

105 samples at 1-minute resolution
Utilization %
70
60
50
40
30
20
10
0
15:52
17 Jun
16:14
16:36
16:58
17:20
2
🟦 MS-DS (File Sharing (*.ip.tcp.445)
🟦 Exchange 2007 Clients (*.ip.tcp.61024)
🟩 Voice Traffic (UDP) (*.ip.udp.65000)
🟪 https (*.ip.tcp.443)

Customer Example 1: Unexpected Performance Degradation

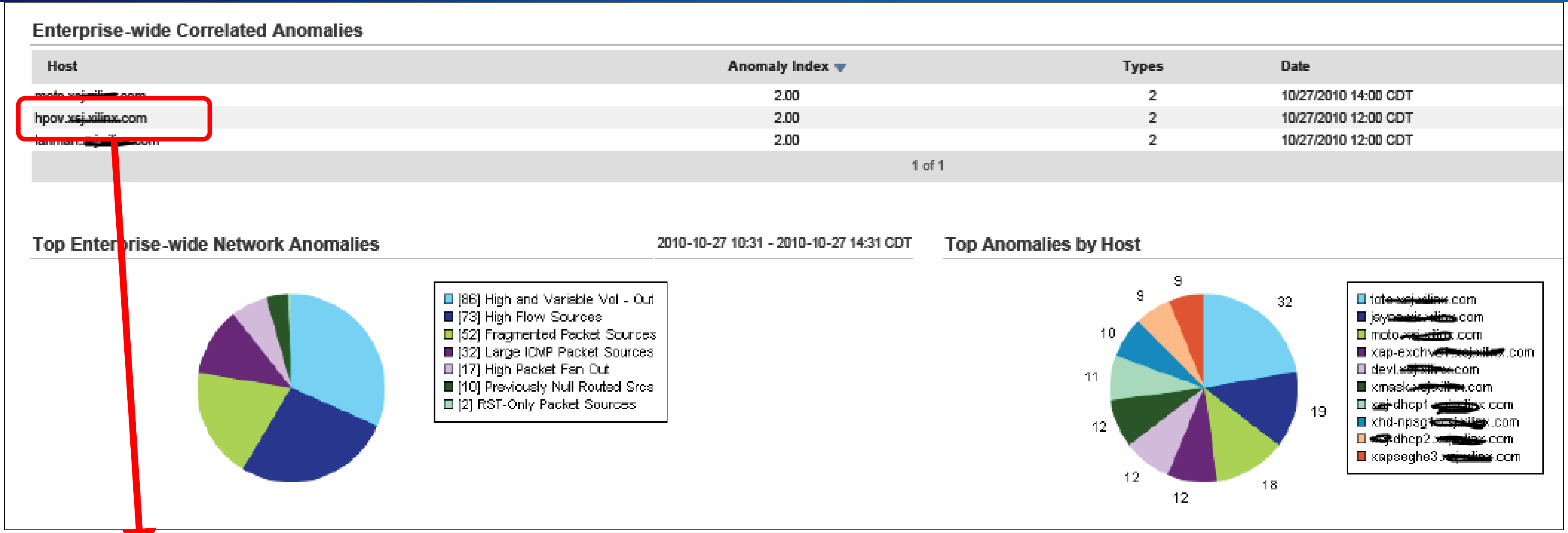


Protocol Conversations Summary - Total 17 Jun 2010 16:50:00 - 17 Jun 2010 17:15:00 EDT

NY1 - New York MPLS Router::NY1 MPLS Interface - AT&T MPLS Primary 4.50 Mbps / 4.50 Mbps
MS-DS (File Sharing (*.ip.tcp.445)

■	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 %
■	devl[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].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 %

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



Enterprise-wide Correlated Anomalies

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

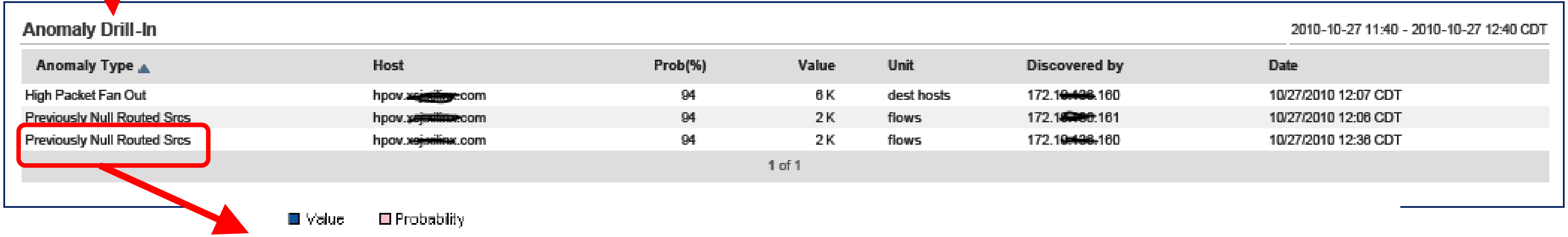
1 of 1

Top Enterprise-wide Network Anomalies

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

[86] High and Variable Vol - Out
[73] High Flow Sources
[52] Fragmented Packet Sources
[32] Large ICMP Packet Sources
[17] High Packet Fan Out
[10] Previously Null Routed Srcs
[2] RST-Only Packet Sources

Top Anomalies by Host

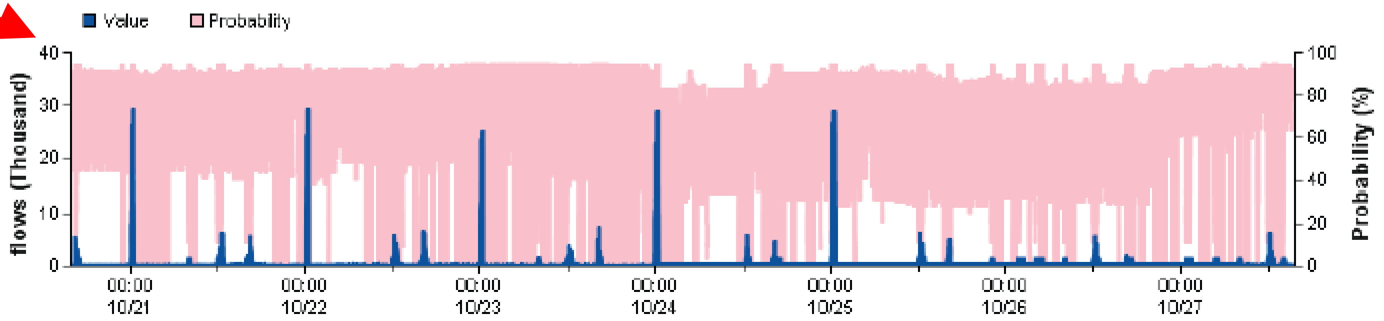


Anomaly Drill-In

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	172.1[redacted].160	10/27/2010 12:07 CDT
Previously Null Routed Srcs	hpov.[redacted].com	94	2 K	flows	172.1[redacted].161	10/27/2010 12:06 CDT
Previously Null Routed Srcs	hpov.[redacted].com	94	2 K	flows	172.1[redacted].160	10/27/2010 12:36 CDT

1 of 1



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

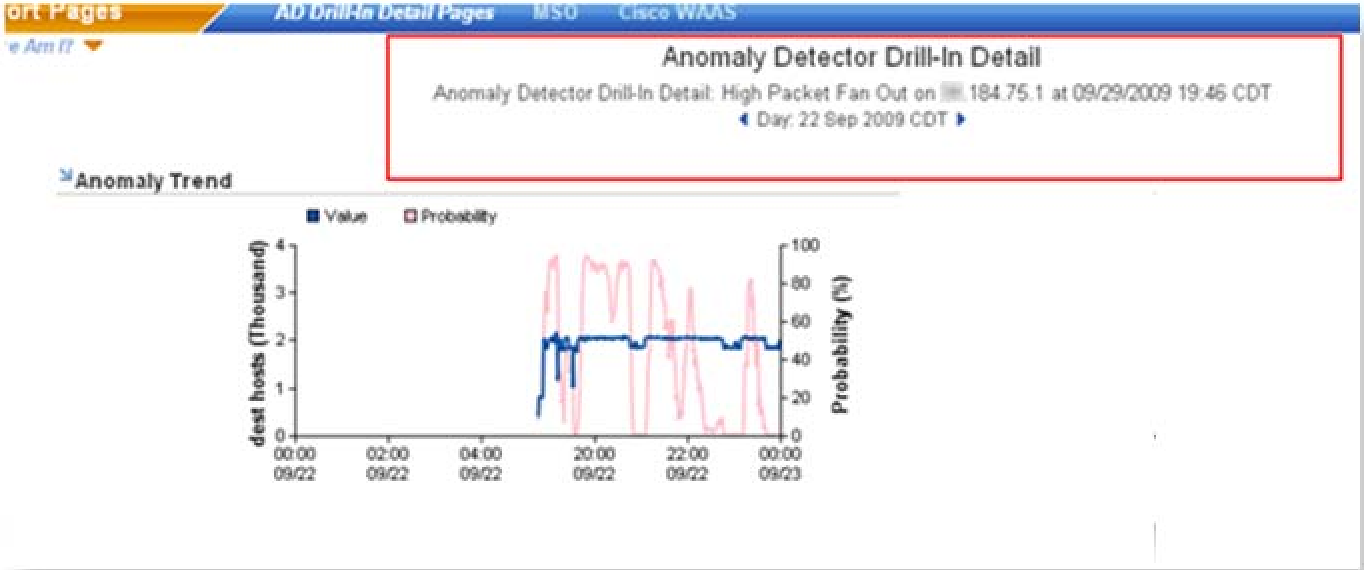
Effectiveness of approach has been shown by solving customer problems on real networks

Questions?

Backup Slides

- Extra info slides

Customer Example 3: Malware Outbreak

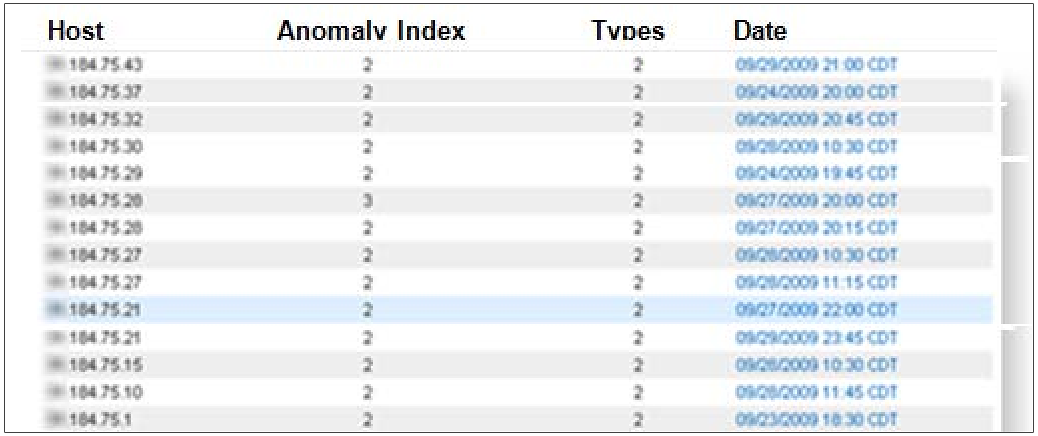


Anomaly Detector Drill-In Detail

Anomaly Detector Drill-In Detail: High Packet Fan Out on [redacted].184.75.1 at 09/29/2009 19:46 CDT

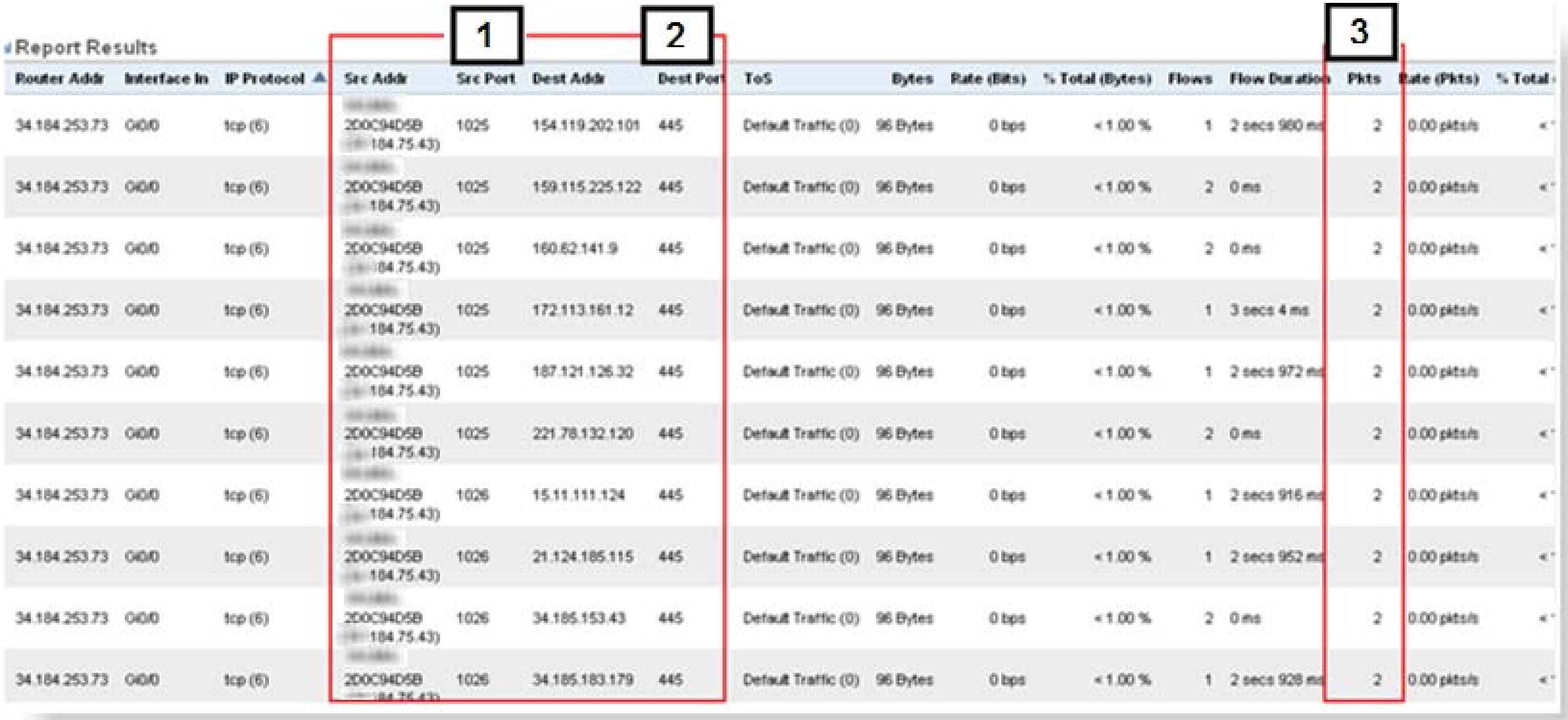
◀ Day: 22 Sep 2009 CDT ▶

Anomaly Trend



Host	Anomaly Index	Types	Date
[redacted].184.75.43	2	2	09/29/2009 21:00 CDT
[redacted].184.75.37	2	2	09/24/2009 20:00 CDT
[redacted].184.75.32	2	2	09/29/2009 20:45 CDT
[redacted].184.75.30	2	2	09/28/2009 10:30 CDT
[redacted].184.75.29	2	2	09/24/2009 19:45 CDT
[redacted].184.75.28	3	2	09/27/2009 20:00 CDT
[redacted].184.75.28	2	2	09/27/2009 20:15 CDT
[redacted].184.75.27	2	2	09/28/2009 10:30 CDT
[redacted].184.75.27	2	2	09/28/2009 11:15 CDT
[redacted].184.75.21	2	2	09/27/2009 22:00 CDT
[redacted].184.75.21	2	2	09/29/2009 23:45 CDT
[redacted].184.75.15	2	2	09/28/2009 10:30 CDT
[redacted].184.75.10	2	2	09/28/2009 11:45 CDT
[redacted].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 Por	ToS	Bytes	Rate (Bits)	% Total (Bytes)	Flows	Flow Duratio	Pkts	ate (Pkts)	% Total
34.184.253.73	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 184.75.43)	1025	154.119.202.101	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 980 m	2	0.00 pkts/s	<
34.184.253.73	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 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	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 04.75.43)	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	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 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	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 184.75.43)	1025	187.121.126.32	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 972 m	2	0.00 pkts/s	<
34.184.253.73	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 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	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 184.75.43)	1026	15.11.111.124	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 916 m	2	0.00 pkts/s	<
34.184.253.73	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 184.75.43)	1026	21.124.185.115	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 952 m	2	0.00 pkts/s	<
34.184.253.73	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 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	Gi0/0	tcp (6)	[redacted] 2D0C94D5B [redacted] 84.75.43)	1026	34.185.183.179	445	Default Traffic (0)	96 Bytes	0 bps	< 1.00 %	1	2 secs 928 m	2	0.00 pkts/s	<

Customer Example 4: Retransmissions traced back

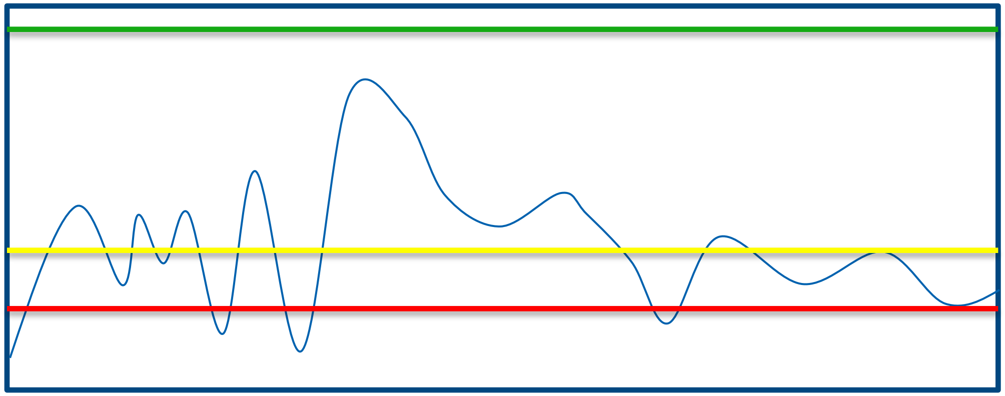
Statistical Analysis Methodology

- 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

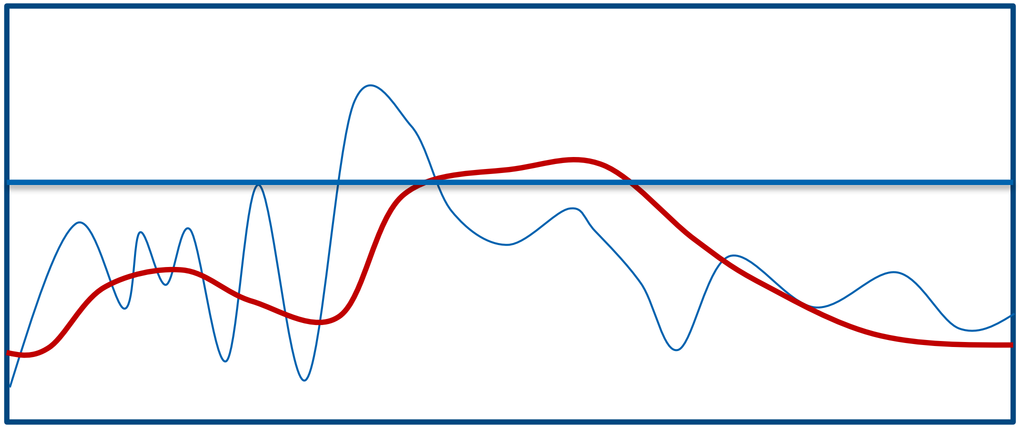
$$p(anomaly \mid point) = \frac{p(point \mid anomaly) * p(anomaly)}{p(point)}$$

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

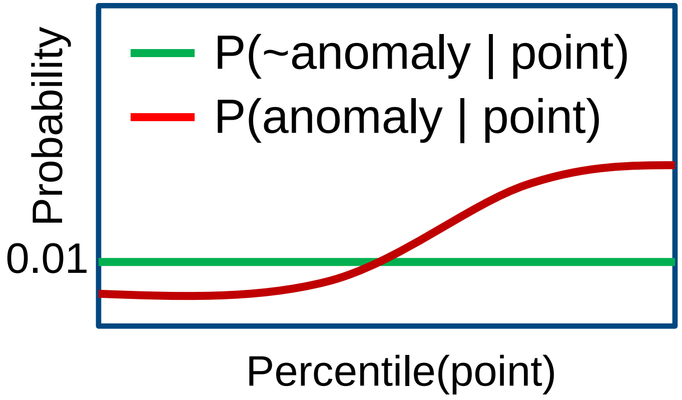


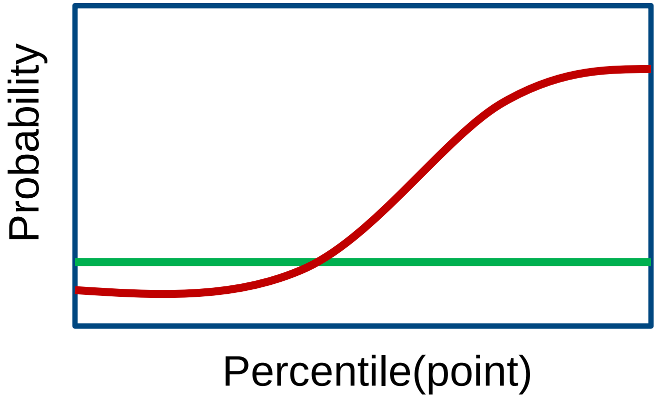
Customizable, pluggable Engines

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

***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*

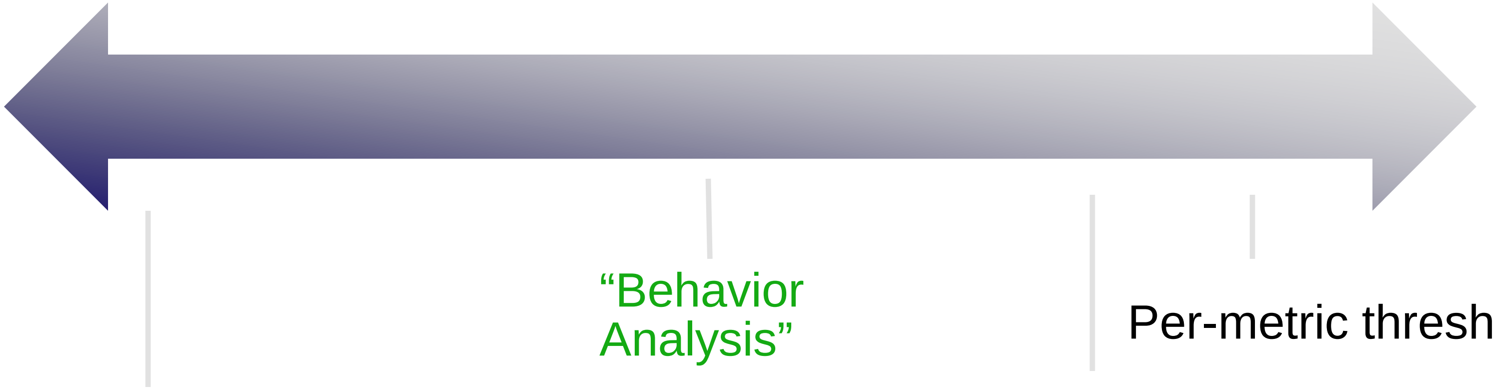




Motivation

Less Scalable
Higher Quality Events

More Scalable
Lower Quality Events



"Behavior
Analysis"

Per-metric thresholds

Baselining

Intrusion Detection Systems

Virus Scanners

Packet Inspection

Signature-Based

Statistical Methods