Anomaly Sampling
(bringing diversity to network security)

David Moore

CAIDA
University of California, San Diego

Anomaly Detection Workshop – July, 2006
Basic Idea

• Existing systems focus on accurate counting of packets (or bytes) for large traffic aggregates
  – e.g., Smart Sampling, Traffic Summaries, Adaptive NetFlow, …

• Instead, focus on interesting, new information
Why? – Operational Network Security

• Forensics – “Bad guy did something”
  – When did they do it?
  – How did they do it?
  – What other machines did they get?

• Detection
  – Host ABC unexpectedly responded to a probe
  – Host XYZ used a service it never did previously
Living on the Network Edge

• The problem is *ours*, not our customer’s.
• We care about *all* of the hosts.
• But each as an *individual*.
  – Some hosts are naturally more important.
  – Each host has its own services, risks, users, threats to other resources, ….
• We care about *small* events, not affecting performance.
• The problem remains *after* the “event” is over.
• Monitored network bandwidths are still high.
Basic Idea

• Existing systems focus on accurate counting of packets (or bytes) for large traffic aggregates
  – e.g., Smart Sampling, Traffic Summaries, Adaptive NetFlow, …

• Instead, focus on interesting, new information
What is “interesting” and “new”?

• Imagine you are the poor recipient of collected network data. What do you see?
  – Here’s a record about our web server. Oh, and here’s another record about our web server. And our mail gateway. Oh, here’s another packet about our web server, ….  

• Please, tell me something I don’t know
  – Tell me what is “abnormal”, “unusual” or “new”.
  – Tell me “just enough” about everything.
  – Do not prioritize telling me redundant information.

• These change over time.
But doesn’t hashing solve this?

• Just put some packet (header) fields together and hash them. Then sample against the hash space.
• Use some memory to avoid redundancy in what has already been reported.
• Voila!
• Unfortunately, packet fields are too variable to use on their own.
  – Imagine the standard “5-tuple” (or 13-tuple, etc). All hits to the web server are sampled the same as a malicious probe to my desktop.
Work in Progress

• Currently trying multiple schemes and examining their operational usefulness.

• Initial results promising, but approaches not polished.
Feature Spaces

• Operator chooses sets of fields/etc. over which they want coverage: (e.g.)
  – Source IP address
  – Destination IP address
  – Source & destination IP address pair
  – Protocol, source port, destination port
  – Src. IP addr., protocol, src. port
  – Src. IP addr., protocol, dst. Port
  – ...

• Might chose weights to specify relative importance
Rough Approaches

- Simple novelty of feature combined with weights
- Counting of occurrences of features
- “Bit-vector” distance of features
- Entropy (compressibility) of features
- Hashing, bloom filters, multi-resolution bit maps, sketches, etc are tools
  - Although each has additional advantages for things like garbage-collection, expiry, memory usage, ability to implement in hardware, …
Rough Results

- Using an effective sampling ratio of 1:100 packets
- With “simple novelty” approach, able to get 5-7x coverage of desired feature sets compared to normal 1:100 sampling
- Other approaches do better.
- Current effort is on examining operational usefulness (particularly in forensics).
Conclusions

• The anomaly detection problem appears to be radically different for security at the edge compared with performance inside an ISP.

• Forensic (historical) analysis is a requirement.

• Driven by operational needs, not research goals.

• We are working on a variety of approaches.
  – would love to discuss more about this
What is “interesting” and “new”? 

- Imagine you are the poor person who has to look at all of the collected network data.
  - What do you see? Here’s a record about our web server. Oh, and here’s another record about our web server. And our mail gateway. Oh, here’s another packet about our web server, ....
  - What do you do? Write some code to throw away useless information or aggregate or do anything to make the pain stop.
  - Why is that a problem? The measurement system has not collected what the person wanted. In fact the system is optimized to provide highly redundant information.
“Interesting” and “new” changes!

• Imagine a port/host scan. When it starts, that is definitely interesting. But as it keeps going for minutes, hours and days (all from the same source, walking your entire network), the scanning isn’t interesting. At that point things are interesting only if a host sends a response or something else different from the scanning occurs.