Lecture Notes on Critique of 1998 and 1999 DARPA IDS Evaluations

Prateek Saxena

March 3 2008

1 The Problems

Today's lecture is on the discussion of the critique on 1998 and 1999 DARPA IDS evaluations conducted by MIT Lincoln Laboratory. The lecture highlights some of the problems with carrying out such evaluations for IDSs, why this is made hard by real-world problems and finally, concludes with a discussion of our many common IDS evaluation pitfalls which apply more broadly to systems research than just to IDS evaluation.

There are 3 deep problems, which emerge after reading the paper.

- **"Authentic" Traffic/Activity** :

  There is no such thing as "authentic" or "typical" traffic. Studies of network activity shows that there are a lot of variations from site-to-site, across locations, etc. The large inherent diversity in operations across the Internet implies that artificial synthesis of so called "authentic" traffic must be treated skeptically.

- **Desirability of black-box testing (and ranking)**

  Funding agencies often define aggressive goals in terms of metrics to achieve. Metric based numbers are needed to justify the research expenditure. To explain the crucial benefits of a system to someone outside the area, tends to be hard. One has to sometimes report "highlights" that are easier to understand and sell, rather than technical intricacies that may be really meaningful. These problems arise due to the desire to rank systems based on black-box view of their operation, rather than based on its model of operation.

- **Great need for illumination/insight into the IDS's mechanism.**

  If you want to rank, you can not do it using black-box views. You need to have model based understanding of the capabilities. There isn't a single good benchmark that is conclusively best and can be used for evaluation. Therefore, there needs to be insight based studies into the IDS being evaluated to decide its effectiveness.

2 Problems with evaluation studies in systems CS

- Little emphasis on reproducibility - Unlike in other matured sciences, credit is seldom given to works that reproduce the result previously reported. Several works do not have rigor/discipline built in the evaluation process – sometimes even the original authors can not reproduce the work done by them previously.

- Evaluation Data is unpublished - This is largely due to tension between need to reveal full structure of data and maintaining privacy and anonymity.
- Often are expensive undertakings.

3 Follow Up work for supporting such large-scale system evaluations

- **LARIAT** : Follow-Up work on 1999 IDS evaluation by MIT Lincoln Laboratory. Describes machinery to synthesize sites and has components that try to generate realistic background user traffic and network attacks. It was published at IEEE Aerospace Conference, March 9-16, 2002.

It is useful to know about some other efforts to make such evaluations possible :

- **DETER** - Testbed for network security technology.
  - Public facility for medium-scale repeatable experiments in computer security
  - Located at USC ISI and UC Berkeley.
  - 300 PC systems running Utah's Emulab software.
  - Experimenter can access DETER remotely to develop, configure, and manipulate collections of nodes and links with arbitrary network topologies.
  - **Problem** with this is currently that there isn't realistic attack module or background noise generator plugin for the framework. Attack distribution is a problem.
- **PREDICT** - Its a huge trace repository. It is not public and there are several legal issues in working with it.
- **KDD Cup** - Its goal is to provide data-sets from real world problems to demonstrate the applicability of different knowledge discovery and machine learning techniques. The 1999 KDD intrusion detection contest uses a labelled version of this 1998 DARPA dataset, annotated with connection features.

  There are several problems [1] with KDD Cup. Recently, people have found average TCP packet sizes as best correlation metrics for attacks, which is clearly points out the inefficacy of using this dataset.

A question was raised in class: Why isn't Bro there in the study?

Primary reason: as a research system, it lacks a comprehensive set of detectors (signatures, application protoocls) because the emphasis is on building a framework and understanding the underlying issues, rather than coverage. (McHugh mentions this concern in his paper.) Its detection approach also does not fall into the 2 categories, though the LL folks were fine with this (and in fact encouraged adding it as a way to illuminate what other approaches can do). Finally, there was the perceived risk of getting classified for life based on a contrived benchmark result.

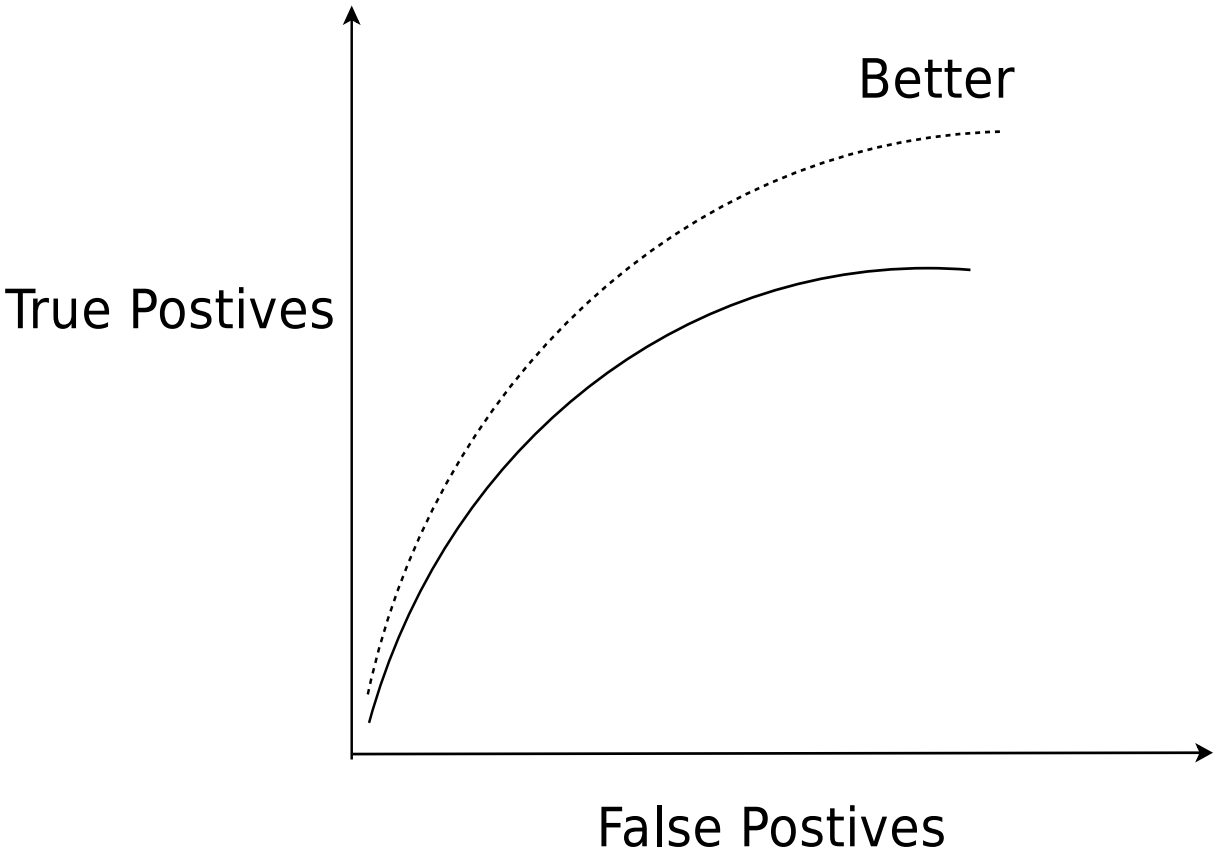

Figure 1: The Basic idea of ROC curve : A plot of the True Positives against the False Positives

4 Problems with Taxonomy Descriptions in the Evaluation

The taxonomy used to classify attacks used in the 1998 evaluation : DOS , R2L, U2R , Surveillance. The problems with using such a taxonomy is :

- Surveillance is not a well-defined term.
- This is an attacker-centric taxonomy. Its based on how costly it is to defend against the classes of attacks. It is deficient in giving any insight into what class of attacks is the system targeting to detect. The critique proposes to use an alternaive, insight based taxonomy. For example, one such taxonomy could be based on how the IDS treats the incoming traffic. Does the IDS look at network layer headers or only application data? If it does not deal with IP header data, and works only for application-specific attacks, then it is obvious that it does not deal with IP layer attacks. This is perhaps a more principled way to analyze different IDS than the strategy used in the evaluation paper.

5 Problems with using ROC

"ROC" stands Receiver Operating Curve.

The basic idea - Every IDS has "knobs" which are parameters the user can use to control IDS behavior. Tuning the knobs yields a curve on the TP vs FP graph as shown in the Fig 1. If a curve for an IDS A, strictly dominates the curve for an IDS B, then intuitively A is better.

5.1 Core Problems with the ROC used in 1998 evaluation

Denominator and Unit of analysis - Use of *log Scale* false alarms *per day* on y-axis, and the number of attacks detected on the x-axis.

The measurement of false positives should be based on a per-event basis, not on per-day basis. What events could we use? It could be based on a unit of analysis that the specif IDS uses. If the IDS uses a packet-based analysis (such as Snort) then packets should be the unit of analysis. Similarly, if sessions are the unit of analysis (such as in Bro), or time-windows (as in anomaly based IDS), then corresponding units should be used as a basis for analysis.

There are other problems with using such an evaluation technique :

- The view that IDS returns a boolean answer to the question "Is instance A an attack?" is true only for some systems. How do we accomodate these in the analysis?
- How do the results of an evaluation transfer to those on another environment ?
- It is hard to compare IDSes with different knobs.
- It is important to characterize specifics of IDS using models of the detection, so that one can take away sound conclusions from the paper.
- Its hard to get IDS models for commercial IDS

To illustrate an example of reporting comparisons with IDSes, consider the performance comparison between Bro, Snort, Snort' (after improvements in Snort). Due to diversity across measurement platforms, the performance figures reported differed considerably from a Pentium IV based test platform, to a Pentium III based test platform. The relative ordering of performance was same on the two test platform – $Snort' > Bro > Snort$, but Bro got slower on Pentium III than on Pentium IV, while Snort became faster, thereby narrowing the gap between the two systems. See [2] for details.

6 Common IDS research pitfalls

- Machine learning based IDS often fail to explore *why* their system classifies a certain metric. It is often useful to report the internal co-efficients of the resulting support vector machines or resulting neural network, to explain which factors contribute to the result heavily.
- Use of data : The *evaluation data traces* used must be different from the *development traces* used. This is analogous to using different test data from training data in machine learning based techniques. Ideally data used in evaluation should be from multiple sites accessed multiple times. This is often violated and data is not aggregated into one before evaluation.
- Data used does not often meet obvious requirements
  - How diverse is the data? This implies how well do the results scale.

  - Quality of data? Details such as the correctness of timestamps in traces are often important.
  - Is the metadata accompanying also stored. Determines reproducibility – for instance, if you don't store the IP to hosts mappings at the time of the experiments, then the data may not be reproducible later.
  - Other artifacts - Does the innocuous data have malicious activity?

- *False positives* are a major concern, and a system needs to investigate these if they arise. Techniques for how to deal with false positives, specially if they are in large number :
  - Randomly sample and pick one – often a good strategy.
  - Prioritize the false positive list, and proceed in highest priority first order.

  False Negatives can be measured by taking a large labelled dataset.

References

[1] M. Mahoney and P. Chan. An analysis of the 1999 darpa/lincoln laboratory evaluation data for network anomaly detection. In *Proceeding of Recent Advances in Intrusion Detection (RAID)-2003*, volume 2820 of *Lecture Notes in Computer Science*, pages 220–237. Springer Verlag, September 8-10 2003.

[2] Robin Sommer and Vern Paxson. Enhancing byte-level network intrusion detection signatures with context. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 262–271. ACM Press, 2003.