Empirically Based Analysis: The DDoS Case

Jul 22nd, 2004

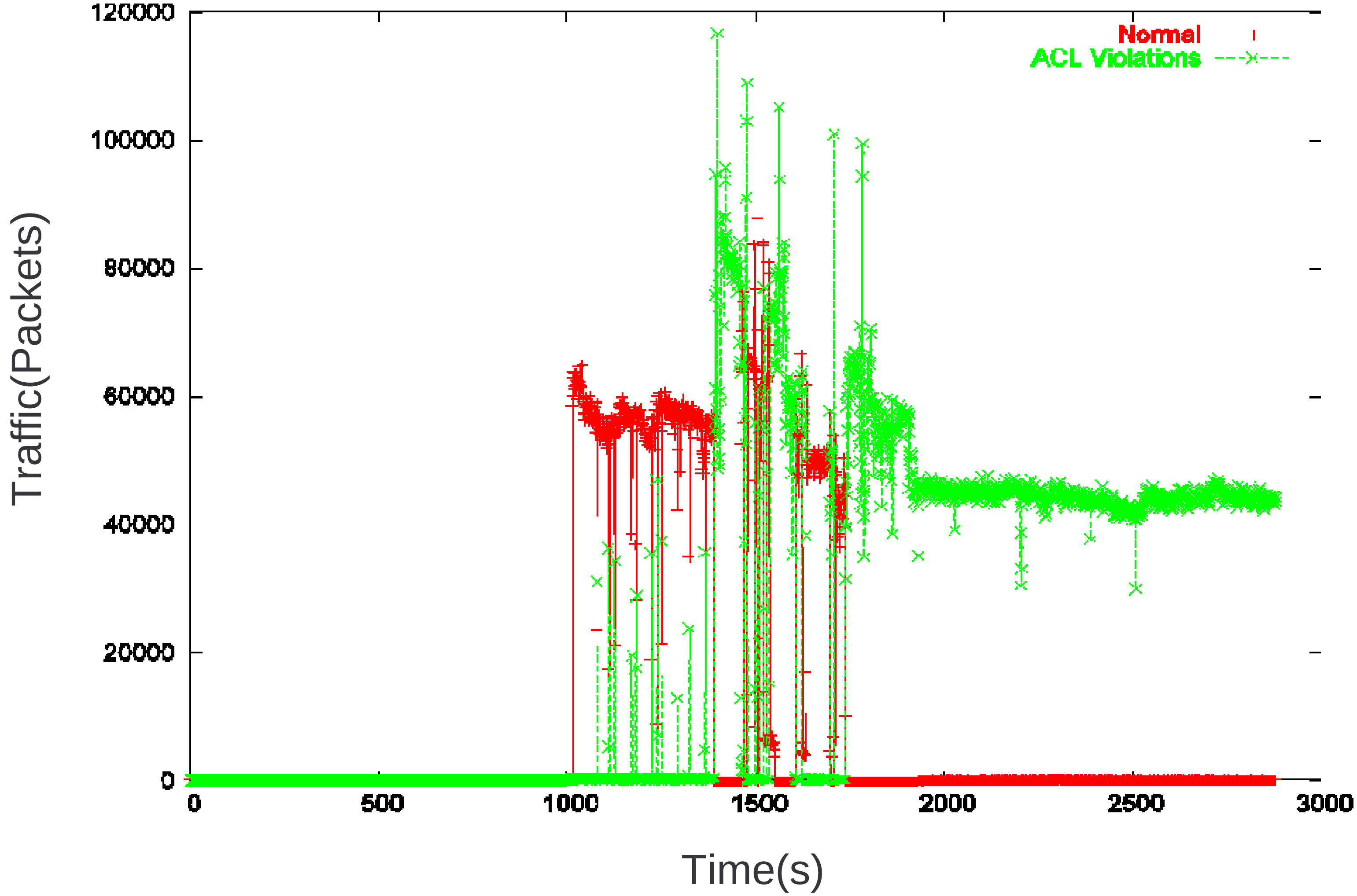
CERT® Analysis Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890

The CERT Analysis Center is part of the Software Engineering Institute. The Software Engineering Institute is sponsored by the U.S. Department of Defense.

Introduction

- Access to the dataset gives us a large enough record of traffic to test hypotheses in network security.
- Given this, we select and evaluate various security measures against real traffic
  - Or a reasonable facsimile thereof
- One example: target resident DDoS Filters
  - Heavily constrain the problem– not considering SYN floods, smurfing, reflection attacks…

Attacks like this



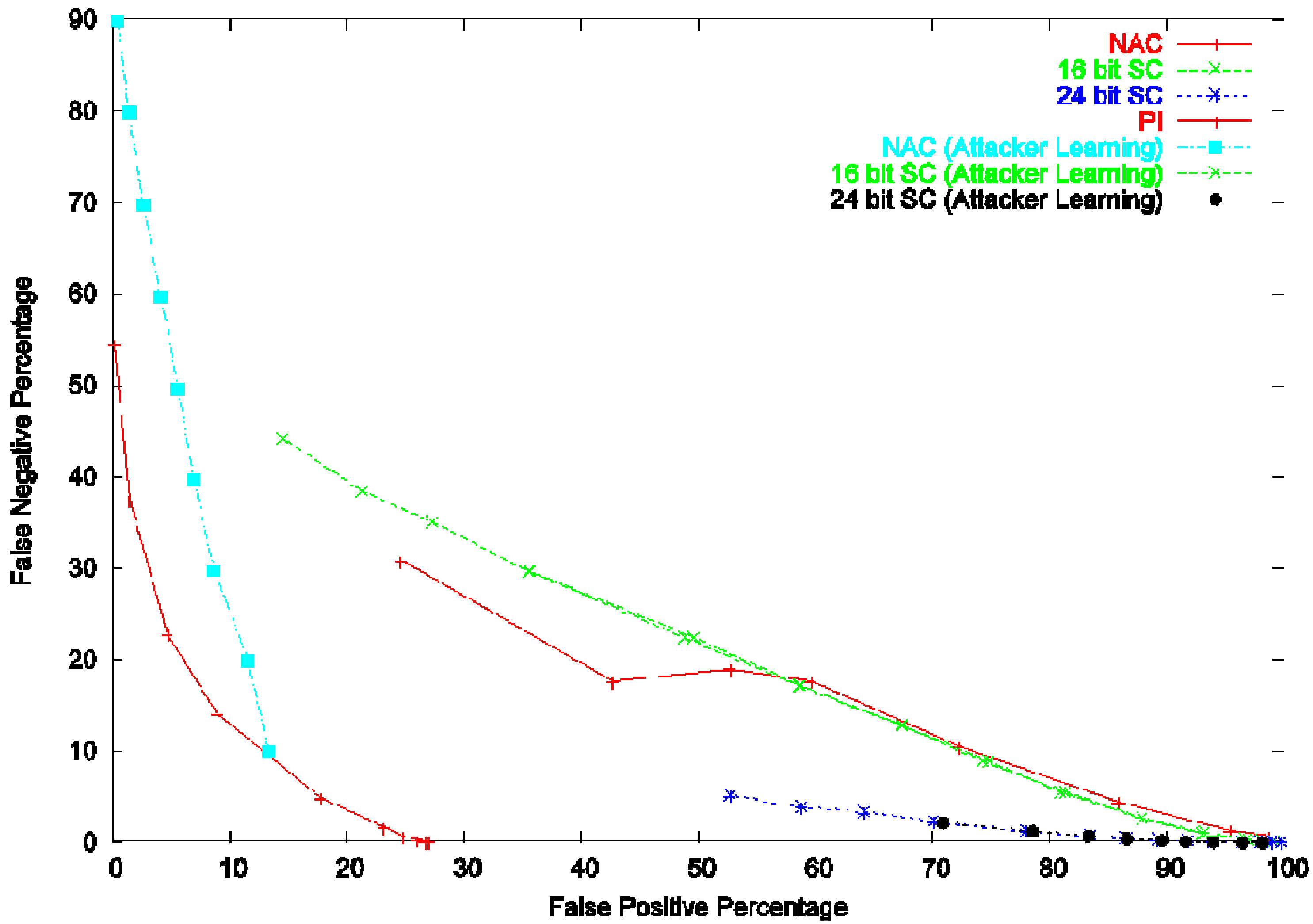
How Do We Test?

- Any analysis opens a can of worms…err, “assumptions”
  - The network constantly changes
  - What is a representative host?
- Rerunning attacks is of debatable value
  - Most of the legitimate traffic is dropped, that’s what a DoS is *for*
- We want our results to be representative
  - Test and summarize over multiple machines
- We want our results to be reproducible
  - Depend heavily on SiLK structures and tools

Evaluation

- Trained filters on 15 days of legitimate traffic
  - Built a representation of IP address: volume relationship (via `rwaddrcount`)
- Then generated a simulated DoS
  - Botnet IPs collected with `rwset`
  - Normal traffic selected from another day
- Resulting traffic was then evaluated for failure rates
- Tested 2 types of filters:
  - Clustering – groups of adjacent IP addresses
  - PI – path marking approach

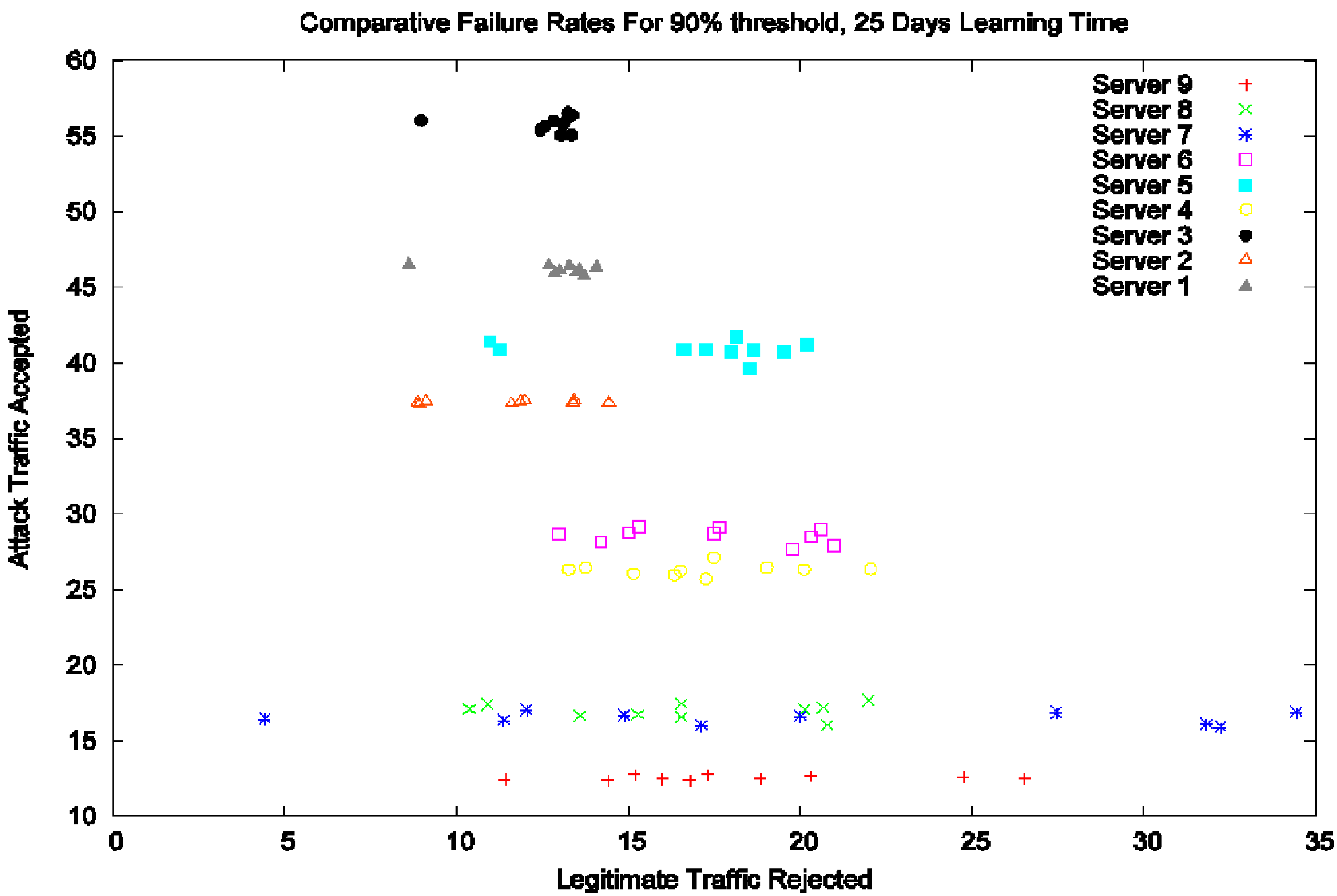
DoS Filters


NAC
16 bit SC
24 bit SC
PI
NAC (Attacker Learning)
16 bit SC (Attacker Learning)
24 bit SC (Attacker Learning)
False Negative Percentage
False Positive Percentage
0
10
20
30
40
50
60
70
80
90
100

Initial Observations

- Two groups
  - One group assumes a magic DoS Detection Oracle
    - That’s the group with better results
- In general, the filters don’t do well
  - Should we compare IP addresses, or packets?
  - Is traffic different for different servers?
- Let’s look at one result in more depth

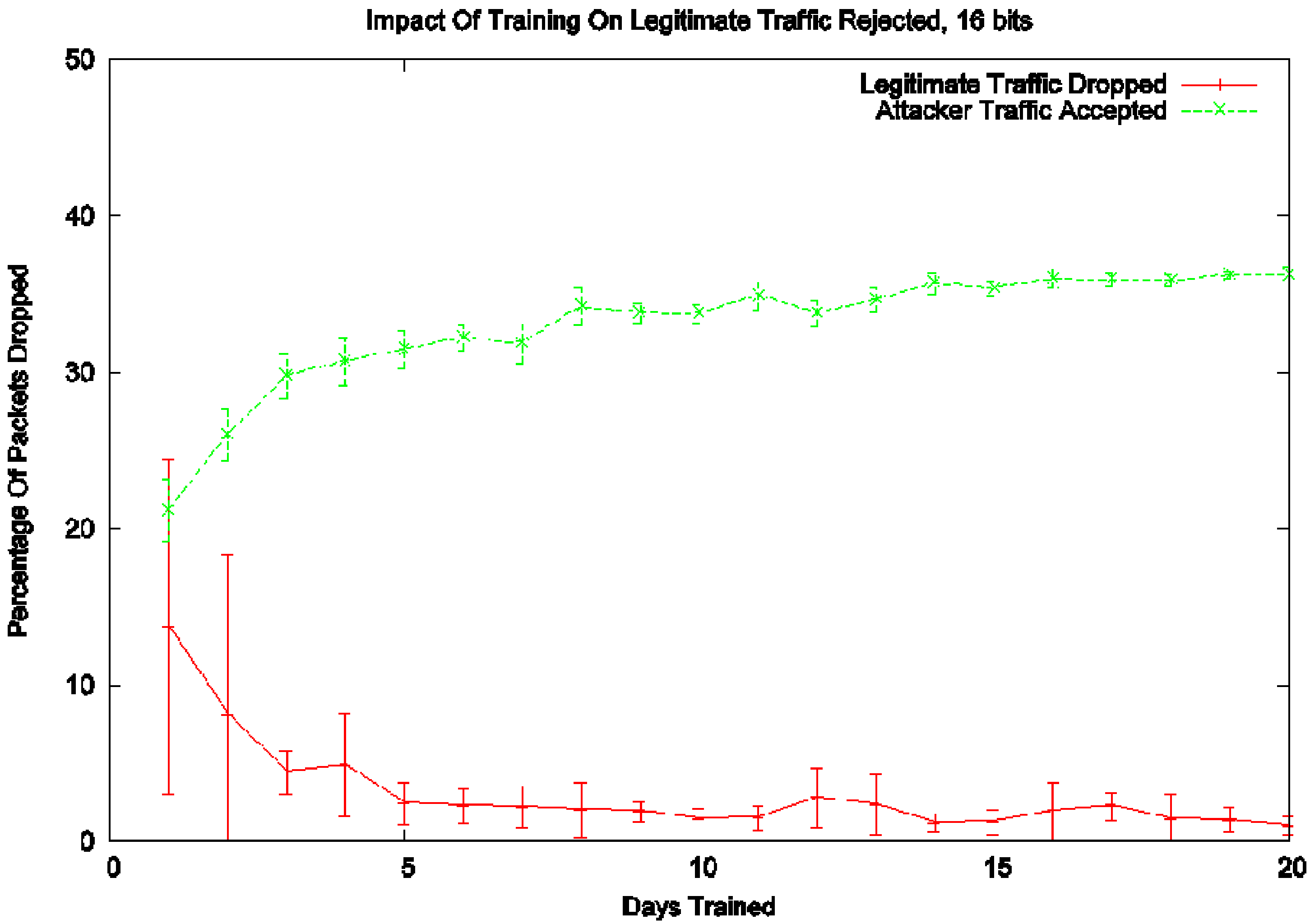
One result in more depth



Observations

- Normal traffic varies extensively
  - Although it seems to vary more with “smaller” servers
  - And it’s better when you look at packet counts
    - Which makes sense, given the absurd number of scanners we see.
- False negative rate (attackers accepted) seems to be related to server activity – the busier the higher.
  - Attackers don’t vary as much

Learning Curves – 95% threshold



Other Observations

- In the majority of cases, packets are dropped because they've never been seen before
  - Short learning curves – effectively no change in false positive rate after a week of learning.
  - Especially true for spoofed traffic
- Entropy is lower than expected
  - Filters that rely on spoof defense (HCF, PI) drop less than 10% of their packets because they detect a spoof

Further Work

- Exploiting our DoS attack traffic records further
  - We know how the network reacts
  - We know how the attack starts and ends
    - Which impacts learning curve for defenses that *only* profile the attack
- Further use of other network maps
  - Skitter (used for PI), &c.
- Formalization of the techniques used
  - Developed a matrix based approach for the final iteration
  - Tools are going to be available publicly

A Final Note

- URL for the SiLK tools: http://silktools.sourceforge.net