Talk Overview

• Talk Goals
  – IDS as detector or design constraints?
• OAS and Detection Surfaces
• Attack and Payoff Models
  – Acquisition
  – Reconnaissance
• Evaluation and Comparison
• Conclusions
IDS as Detector

• Standard method for evaluation: ROC curves
  – True Positive Vs. False Negative
  – Hidden variable used to determine sensitivity, specificity
A Lack of Context

• The ROC based approach evaluates IDS as an alarm
• The world has changed
  – Many attacks are failures
  – Many attacks are
• Base-Rate Fallacy: Continuous tests and low FPR yields high objective false alarm count
There are also many True Positives
Our alternative

- View IDS as a design spec, with the attacker as an engineer
  - Attacker’s goal – succeed without getting caught
  - Attacker is rational

Anyone who talks to more than 15 hosts/minute is suspicious

I’ll talk to 14 hosts every minute
Observable Attack Space

• What attacks look like is a function of your logging software
  – We’re using NetFlow – no payload, can crunch a lot of data
• OAS: parameterized representation of varying types of attacks
  – Our OAS is simple:
    • Aggressiveness: # of hosts contacted in a 30s period
    • Success: Probability that they contact a host
Methodology

- IDS are state variables from a log file $\Lambda$
  - $g(\Lambda)$: total number of nodes [Collins + Reiter, 2007]
  - $c(\Lambda)$: largest component size [Collins + Reiter, 2007]
  - $h(\Lambda)$: entropy of server addresses [Lakhina et al., 2005]
  - $r(\Lambda)$: longest number of failed connections from an IP [Jung et al., 2004]
  - $d(\Lambda)$: maximum client degree [Everybody]

- Model variables using historical ssh data

- All the IDS (except d) have parameterized models
  - Control FPR (threshold of aggravation)
Detection Surface

• Generate an attack log from \((a,s)\)
  – calculate \(x(\Lambda \cup \Lambda_{\text{atk}})\), estimate
  – Multiple Monte-Carlo runs, build up a model for the probability of detection: \(P_{\text{det}}^x(a,s)\)

• Limited number of runs for this test
  – More runs, smoother lines
  – These show the behavior
Estimate a Detection Surface
This surface

• Aggregate behavior
  – FPR calibrated to 0.1% for all sensors combined
  • Control FPR by increasing thresholds, again statistical models
  – 0.1% = 1 False Alarm/8 hour shift w 30s observations
• d limits the size of the space
  – Upper limit: d = 150; anything greater automatically detected
Individual IDS results

![Diagram showing individual IDS results with contour lines for success percentage against aggressiveness. The diagrams indicate varying success rates at different levels of aggressiveness.]
Observations

• Why no $d$?
  – Max degree is treated as an upper limit. > 150 targets and you’re tagged.

• All of these things have the same FPR – 1/1000 tests.
  – Very different behavior
  – Different tests complement each other

• Some tests are very low detectors
  – Because we’re dealing with very smart attackers
  – You can get a good FPR/TPR because many attackers aren’t
Acquisition Scenario

- Attacker opens communication with $a$ hosts, $s$ % of which are on a hit list
  - Communication is assumed to be takeover
- Takes place in $k$ rounds, each of which is an $(a,s)$ attack
- If the defender detects the attacking IP:
  - Blocks the attacking IP
  - Restores all hosts the attacking IP talked with
Acquisition Payoff

• Payoff Function:

\[ H_{acq}^x(a, s, k) = (1 - P_{det}^x(a, s))^k(ask) \]

• Continue attacking to point of diminishing returns

\[ k < \frac{1 - P_{det}^x(a, s)}{P_{det}^x(a, s)} \]
Acquisition Payoff

Payoff

Aggressiveness

Success
Observations

• Aggression doesn’t pay
  – Too high an $a$, and the attack is detected and the damage undone

• The attacker can earn a very high profit through patience and a good hit list
Reconnaissance Scenario

• Attacker wants to scout out hosts
• Real payoff is failure rate
  – Success means attacker picked from hit list
  – Failures mean that the attacker has communicated with a new address
    • Assume IP space isn’t very dense
Reconnaissance Payoff

• Payoff Function:

\[ H_{\text{rec}}^x (a, s) = a(1 - s) \frac{1}{P_{\text{det}}^x (a, s)} \]

• Scanners continue until they get caught
  – No penalty for being caught, still get intelligence
  – “First round free”
Reconnaissance Payoff

![3D graph showing Payoff, Aggressiveness, and Success relations]
Observations

• Aggressiveness pays
  – We limit $d$, but an attacker could hit thousands in that timeframe
  – Similar hump to attacks – slow and subtle identification can also pay well
Inverting the situation

- Instead of asking how effective attacks are
  - How effective do defenses have to be in order to stop them?
- Invert payoff calculation, solve detection probability for fixed payoff

\[ P_{det}^x (a, s) = 1 - k \sqrt{H_{acq}^x (a, s, k)} \]
Individual IDS results

![Graph g](image)

Success (Percentage) vs. Aggressiveness

- g: 32%
- c: 5%
- h: 1.2%
- r: 3%

![Graph h](image)

![Graph r](image)
Observations

• The values drop
  – But they still aren’t really that good
    • 1% = 10 alerts/shift = 30 alerts/day
  – Again, these are very subtle attacks

• Since there’s a payoff for subtlety, an attacker can continue to hit the system using a very low success rate even if the system is amazingly sensitive (and therefore annoying)
Conclusions

• Developed a payoff-based model for comparing IDS efficacy
  – Shows strength and weaknesses of various anomaly detectors
• Attackers can conduct very subtle and effective attacks which even very sensitive detectors will fail to detect
  – Conversely, any old detector will work soon enough
• Subtlety pays; analyst patience will run out far before the system is an effective detector
• Probably need different detectors for slow attacks than fast ones
Questions?

• Michael Collins, RedJack LLC
  – michael.collins@redjack.com

• Michael Reiter, UNC Chapel Hill
  – reiter@cs.unc.edu