Flaws and Frauds in IDPS evaluation

Dr. Stefano Zanero, PhD

Post-Doc Researcher, Politecnico di Milano
CTO, Secure Network

Outline

- Establishing a need for testing methodologies
  - Testing for researchers
  - Testing for customers
- IDS testing vs. IPS testing and why both badly suck
- State of the art
  - Academic test methodologies
  - Industry test methodologies (?)
- Recommendations and proposals

The need for testing

- Two basic types of questions
  - Does it work ?
    - If you didn't test it, it **doesn't** work (but it may be pretending to)
  - How well does it work ?
    - Objective criteria
    - Subjective criteria

Researchers vs. Customers

- What is testing for researchers ?
  - Answers to the “how well” question in an objective way
  - Scientific = repeatable (Galileo, ~1650AD)
- What is testing for customers ?
  - Answers to the “how well” question in a subjective way
  - Generally, very custom and not repeatable, esp. if done on your own network

Relative vs. absolute

- Absolute, objective, standardized evaluation
  - Repeatable
  - Based on rational, open, disclosed, unbiased standards
  - Scientifically sound
- Relative evaluation
  - “What is better among these two ?”
  - Not necessarily repeatable, but should be open and unbiased as much as possible
  - Good for buy decisions

Requirements and metrics

- A good test needs a definition of **requirements** and **metrics**
  - Requirements: “does it work ?”
  - Metrics: “how well ?”
  - I know software engineers could kill me for this simplification, but who cares about them anyway? :)
- Requirements and metrics are not very well defined in literature & on the market, but we will try to draw up some in the following
- But first let's get rid of a myth...

To be, or not to be…

- IPS ARE IDS: because you need to detect attacks in order to block them... **true!**
- IPS aren't IDS: because they fit a different role in the security ecosystem... **true!**
- **Therefore:**
  - A (simplified) does it work test can be the same...
  - A how well test cannot!
- And the “how well” test is what we really want anyway

Just to be clearer: difference in goals

- IDS can afford (limited) FPs
- Performance measured on throughput
- Try as much as you can to get DR higher

- Every FP is a customer lost
- Performance measured on latency
- Try to have some DR with (almost) no FP

Anomaly vs. Misuse

- Find out normal behaviour, block deviations
- Can recognize any attack (also 0-days)
- Depends on the metrics and the thresholds
- = you don't know why it's blocking stuff

- Uses a knowledge base to recognize the attacks
- Can recognize only attacks for which a “signature” exists
- Depends on the quality of the rules
- = you know way too well what it is blocking

Misuse Detection Caveats

- It's all in the rules
  - Are we benchmarking the *engine* or the *ruleset ?*
    - Badly written rule causes positives, FP?
    - Missing rule does not fire, FN ?
      - How do we measure coverage ?
    - Correct rule matches attack traffic out-of-context (e.g. IIS rule on a LAMP machine), FP ?
      - This form of tuning can change everything !
    - Which rules are activated ?! (more on this later)
- A misuse detector alone will **never** catch a zero-day attack, with a few exceptions

Anomaly Detection Caveats

- No rules, but this means...
  - Training
    - How long do we train the IDS ? How realistic is the training traffic ?
  - Testing
    - How similar to the training traffic is the test traffic ? How are the attacks embedded in ?
  - Tuning of threshold
- Anomaly detectors:
  - If you send a sufficiently strange, non attack packet, it will be blocked. Is that a “false positive” for an anomaly detector ?
- And, did I mention there is none on the market ?

An issue of polimorphism

- Computer attacks are polimorph
  - So what ? Viruses are polimorph too !
    - Viruses are as polimorph as a **program** can be, attacks are as polimorph as a **human** can be
  - Good signatures capture the vulnerability, bad signatures the exploit
- Plus there's a wide range of:
  - evasion techniques
    - [Ptacek and Newsham 1998] or [Handley and Paxson 2001]
  - mutations
    - see ADMmutate by K-2, UTF encoding, etc.

Evaluating polimorphism resistance

- Open source KB and engines
  - Good signatures should catch key steps in exploiting a vulnerability
    - Not key steps of a particular exploit
  - Engine should canonicalize where needed
- Proprietary engine and/or KB
  - Signature reverse engineering (signature shaping)
  - **Mutant exploit generation**

Signature Testing Using Mutant Exploits

- **Sploit** implements this form of testing
  - Developed at UCSB (G.Vigna, W.Robertson) and Politecnico (D. Balzarotti - kudos)
    - Generates mutants of an exploit by applying a number of mutant operators
    - Executes the mutant exploits against target
    - Uses an oracle to verify the effectiveness
    - Analyzes IDS results
- Could be used for IPS as well
- No one wants to do that :-)

But it's simpler than that, really

- Use an old exploit
  - oc192’s to MS03-026
- Obfuscate NOP/NULL Sled
  - s/0x90,0x90/0x42,0x4a/g
- Change exploit specific data
  - Netbios server name in RPC stub data
- Implement application layer features
  - RPC fragmentation and pipelining
- Change shell connection port
  - This 666 stuff … move it to 22 would you ?
- Done
  - Credits go to Renaud Bidou (Radware)

Measuring Coverage

- If ICSA Labs measure coverage of anti virus programs ("100% detection rate") why can't we measure coverage of IPS ?
  - Well, in fact ICSA is trying :)
  - Problem:
    - we have rather good zoo virus lists
    - we do not have good vulnerability lists,let alone a reliable wild exploit list
- We cannot **absolutely** measure coverage, but we can perform **relative** coverage analysis (but beware of biases)

How to Measure Coverage

- Offline coverage testing
  - Pick signature list, count it, and normalize it on a standard list
    - Signatures are not always disclosed
    - Cannot cross compare anomaly and misuse based IDS
- Online coverage testing
  - We do not have all the issues but
  - How we generate the attack traffic could somehow influence the test accuracy
- But more importantly... **ask yourselves: do we actually care ?**
  - Depends on what you want an IPS for

False positives and negatives

- Let's get back to our first idea of “false positives and false negatives”
  - All the issues with the definition of false positives and negatives stand
- Naïve approach:
  - Generate realistic traffic
  - Superimpose a set of attacks
  - See if the IPS can block the attacks
- We are all set, aren't we ?

Background traffic

- Too easy to say “background traffic”
  - Use real data ?
    - Realism 100% but not repeatable
    - Privacy issues
    - Good for relative, not for absolute
  - Use sanitized data ?
    - Sanitization may introduce statistical biases
    - Peculiarities may induce higher DR
    - The more we preserve, the more we risk
  - In either case:
    - Attacks or anomalous packets could be present!

Background traffic (cont)

- So, let's really **generate** it
  - Use “noise generation” ?
    - Algorithms depend heavily on content, concurrent session impact, etc.
  - Use artificially generated data ?
    - Approach taken by DARPA, USAF...
    - Create testbed network and use traffic generators to “simulate” user interaction
    - This is a good way to create a **repeatable**, scientific test on solid ground
  - Use no background.... yeah, right
  - What about broken packets ?
    - http://lcamtuf.coredump.cx/mobp/

Attack generation

- Collecting scripts and running them is not enough
  - How many do you use ?
  - How do you choose them ?
  - **… do you choose them to match the rules or not ?!?**
  - Do you use evasion ?
  - You need to run them against vulnerable machines to prove your I **P** S point
  - They need to blend in perfectly with the background traffic
- Again: most of these issues are easier to solve on a testbed

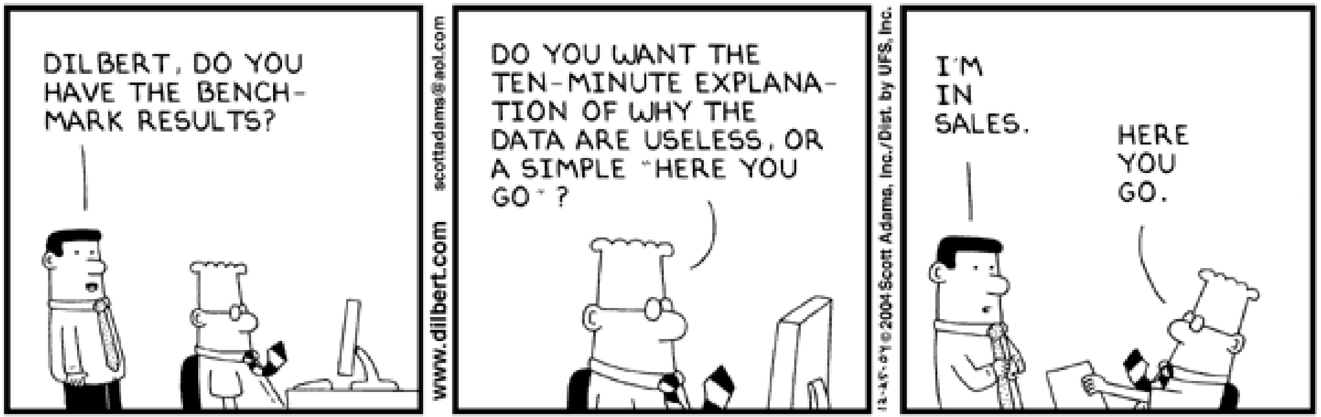
Datasets or testbed tools ?

- Diffusion of datasets has well-known shortcomings
  - Datasets for high speed networks are huge
  - Replaying datasets, mixing them, superimposing attacks creates artefacts that are easy to detect
    - E.g. TTLs and TOS in IDEVAL
  - Tcpreplay timestamps may not be accurate enough
    - Good TCP anomaly engines will detect it's not a true stateful communication
- Easier to describe a testbed (once again)

Generating a testbed

- We need a realistic network...
  - Scriptable clients
    - We are producing a suite of suitable, GPL'ed traffic generators (just ask if you want the alpha)
      - Scriptable and allowing for modular expansion
      - Statistically sound generation of intervals
      - Distributed load on multiple slave clients
  - Scriptable or real servers
    - real ones are needed for running the attacks
    - For the rest, Honeyd can create stubs
  - If everything is FOSS, you can just describe the setup and it will be repeatable !
    - Kudos to Puketza et al, 1996

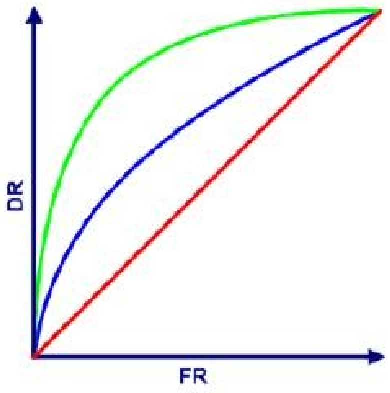
Do raw numbers really matter?

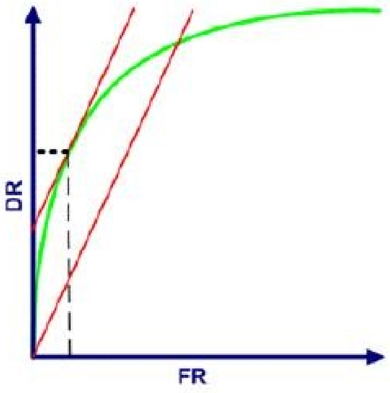

DILBERT, DO YOU HAVE THE BENCHMARK RESULTS?
DO YOU WANT THE TEN-MINUTE EXPLANATION OF WHY THE DATA ARE USELESS, OR A SIMPLE "HERE YOU GO"?
I'M IN SALES.
HERE YOU GO.
www.dilbert.com
scottadams@aol.com
© 2004 Scott Adams, Inc./Dist. by UFS, Inc.

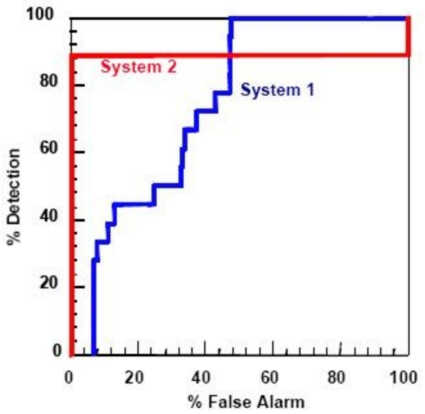
© UFS, Inc.

- If Dilbert is not a source reliable enough for you, cfr. Hennessy and Patterson
  - Personally, I prefer to trust Dilbert... kudos to Scott Adams :-)
- Raw numbers seldom matter in performance, and even less in IDS

ROC curves, then !







- Great concept from signal detection, but:
  - they are painful to trace in real world
  - they are more meaningful for anomaly detectors than misuse detectors
    - Depends, again, on definition of false positive

It is written, "performance"…

- But it reads like "speed"
  - If you want to measure "how fast" an IDS is, you once again need to define your question
    - Packets per second or bytes per second (impacts NIC capacity, CPU, and memory bus speed)
    - Number of hosts, protocols and concurrent connections (memory size and memory bus speed, CPU speed)
    - New connections per second (memory bus speed, CPU speed)
    - Alarms per second (memory size, CPU speed, mass storage, network, whatever...)
  - Each metric "measures" different things !

Metrics, metrics

- Throughput ? Delay ? Discarded packets ?
  - On an IPS you want to measure **delay** and eventually discarded packets
  - On an IDS you want to measure **throughput** and discarded packets

Models, models...

- In theory, this thing acts like an M/M/1/c finite capacity queue...
  - Arrival process is Poisson (simplification, it actually isn't)
  - Service time is exponential (simplification, it is load-dependent and depends on the number of open connections)
  - There is a finite buffer c (this is realistic)
- Delay, rejection, throughput can be statistically computed with simple tests

Queues quirks

- The queueing model also says...
  - That traffic distribution matters !
  - That packets/connections/open connections ratios matter !
  - Packets/bytes ratio matters !
  - We have also verified, as others showed before, that types of packets, rules and checks impact on the service times
- So, all these things should be **carefully documented** in tests... and you should read them when evaluating other people tests
- And if they don't **write down** them, just assume the worse

Existing IDS tests

- A bit outdated
  - Puzetzka at UC Davis (oldies but goldies)
  - IBM Zurich labs (God knows)
  - IDEVAL (more on this later)
  - AFRL evaluations (cool, but not open)
- Current tests (2002-2003...)
  - NSS group tests
    http://www.nss.co.uk
  - Neohapsis OSEC
    http://osec.neohapsis.com/
  - Miercom Labs/Network World
    http://www.networkworld.com/reviews/2002/1104rev.html

MIT/LL and IDEVAL

- IDEVAL is the dataset created at MIT/LL
  - Only available resource with synthetic traffic and full dumps + system audit files
  - Outdated systems and attacks
  - Very few attack types, in particular host-based IDS have just basic overflows...
  - Well known weaknesses in NIDS data:
    - TTLs, TOS, source IP, ... all detectable
  - IDEVAL has been used by **each** and every researcher in the field (including me), i.e. it has biased all the research efforts since 1998

NSS Tests

- NSS Group tests are perhaps the most famous industry testing ground
- On the whole, not bad, but:
  - They are non repeatable (since attacks and other parameters are unspecified)
    - Being not really scientific and not really based on a specific scenario, what's their aim
  - Include lots of qualitative evaluations
  - Use either noise or HTTP traffic for stress testing
  - Unspecified distribution characters of traffic
  - Aging attacks and evasions (for what we know)

Neohapsis / OSEC

- A new pretender on the block
- Good idea, an open, repeatalbe methodology, but:
  - Not addressing breadth of KB
  - Use either noise or HTTP traffic for stress testing
  - Unspecified distribution characters of traffic
  - Not really suitable for anomaly based products

Miercom/Network World

- Less known than the others
- More journalistic than scientific
- Yet, a very good description of the setup, the attacks, and the testing conditions
  - Still not addressing breadth of KB
  - Still HTTP traffic for stress testing
  - Still unspecified distribution characters of traffic
  - But a very very good testing methodology indeed

Existing tests for IPS

- Even less than the ones for IDS!
  - NSS tests
    http://www.nss.co.uk
  - E-week
    http://www.eweek.com/article2/0,1895,1759490,00.asp
  - Network World
    http://www.networkworld.com/reviews/2004/0216ips.html
    http://www.networkworld.com/reviews/2006/091106-ips-test.html
  - Network Computing
    http://www.networkcomputing.com/showArticle.jhtml?articleID=163700046&pgno=1&queryText=IPS+review

NSS Tests

- NSS Group tests are perhaps the most famous industry testing ground
- On the whole, not bad, but:
  - They are non repeatable (since attacks and other parameters are unspecified)
  - Include lots of qualitative evaluations
  - Use either noise or HTTP traffic for stress testing
  - Unspecified distribution characters of traffic
  - “resistance to FP” using neutered exploits?! Puh-lease...
  - Evasion techniques one at a time

Network World

- A very good description of the setup, the attacks, and the testing conditions
  - They already did a good job on IDS
  - No performance test for very good reasons: the vendors cannot even agree on what an IPS is, let alone how to test it for speed
  - A very good testing methodology indeed, very well described
  - Unluckily, just qualitative results... but what can be really expected ?

Network Computing

- A not-so-good description of the setup, the attacks, and the testing conditions
- Still they have performed interesting testing
  - No performance test for very good reasons: the vendors cannot even agree on what an IPS is, let alone how to test it for speed
  - Quantitative results but no good indication of how they were computed

E-week

- Quoting directly:
  *eWEEK Labs' testbed for <censored> combined an artificial, lab-created Internet connection with **traffic** carried by our ISP.*
  *To get **repeatable**, comparable **results**, we also ran **attack tools** such as the open-source **Nessus** on **network devices** … Using **predictable attack traffic significantly speeds up proof-of-concept testing.***
  *Whether you run IPSes in front of or behind firewalls **depends on many factors.***
- My comments will not be written down in order to avoid lawsuits :) but you may guess them by comparing with the previous slides

Conclusions

- Testing IPS is a real, huge mess
  - But still, we must do something
- We are still far away from designing a complete, scientific testing methodology
  - But we can say a lot of things on wrong methodologies
- You can and should design customer-need driven tests in house
  - Difficult, but the only thing you can do
- In general, beware of those who claim "My IPS is better than yours"

QUESTIONS ?

Thanks for your attention !!!

Feedback/Followup/Insults welcome
zanero@elet.polimi.it

Have a look at our website
www.securenetwork.it

securenetwork

Bibliography

- **Traffic measurements, internet traffic mixes**
  - K. Claffy, G. Miller, K. Thompson: The Nature of the Beast: Recent Traffic Measurements from an Internet Backbone http://www.caida.org/outreach/-papers/1998/Inet98/ (1998)
  - S. McCreary, K. Claffy: Trends in Wide Area IP Traffic Patterns: A View from Ames Internet Exchange. http://www.caida.org/outreach/papers/2000/-AIX0005/ (2000)
- **Polimorphism resistance testing**
  - G. Vigna, W. Robertson, D. Balzarotti: Testing Network-based Intrusion Detection Signatures Using Mutant Exploits, ACM CCS 2004
- **General performance literature**
  - D. Patterson, J. Hennessy: Computer Organization and Design: the Hardware/Software interface, 3rd ed., Morgan-Kauffman

Bibliography (2)

- **General IDS testing literature (no IPS literature exists... sorry ;)**
  - M. Hall, K. Wiley: Capacity Verification for High Speed Network Intrusion Detection Systems http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/prod_technical_reference09186a0080124525.html
  - M. J. Ranum: Experiences benchmarking Intrusion Detection Systems, http://www.snort.org/docs/Benchmarking-IDS-NFR.pdf
  - N. Athanasiades, R. Abler, J. Levine, H. Owen, G. Riley: Intrusion Detection Testing and Benchmarking Methodologies, 1st IEEE International Information Assurance Workshop, 2003
  - P. Mell,V. Hu, R. Lippmann, J. Haines, M. Zissman: An Overview of Issues in Testing Intrusion Detection Systems, NIST – LL/MIT, 2003
  - N. J. Puketza, K. Zhang, M. Chung, B. Mukherjee, R. A. Olsson: A Methodology for Testing Intrusion Detection Systems, IEEE Transactions on Software Engineering, 1996