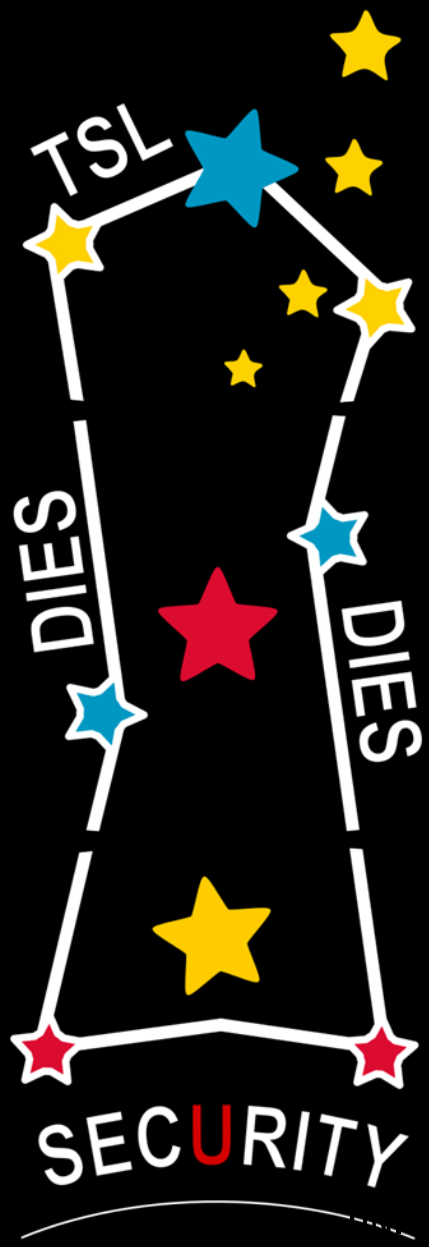
UNIVERSITY OF TWENTE.



PANACEA: AUTOMATING ATTACK CLASSIFICATION FOR ANOMALY-BASED NETWORK INTRUSION DETECTION SYSTEMS

DAMIANO BOLZONI, SANDRO ETALLE AND PIETER HARTEL

DISTRIBUTED AND EMBEDDED SECURITY GROUP

TWENTE SECURITY LAB

10+ YEARS OF RESEARCH OVER ANOMALY DETECTION…

- Sadly though, few commercial implementations
  - most of them use “behavioral-based” anomaly detection → catchy words to say they detect portscans and DDoS…
  - others promise “protocol-based” anomaly detection → only a few HTTP attacks will use “Content-Length: -1”…
- What went wrong? Where is the anomaly-based Snort ?

IT'S A HARD LIFE IN THE REAL WORLD FOR AN ANOMALY-BASED IDS…

- Training sets are not "clean by default"
- Threshed values must be manually set
- Monitored systems "tend" to change over time

next presentations in this session

- Alerts must be manually classified

lack of usability → nobody will deploy such an IDS

WHY ALERT CLASSIFICATION SHOULD BE AUTOMATED?

- Use alert correlation/verification and attack trees techniques
  - so far, only available for signature-based IDSs
- Automatic countermeasures activated based on attack classification/impact
  - block the source IP in case of a buffer overflow
  - wait the next action in case of a path traversal
- Reduce the required user knowledge and workload
  - less knowledge and workload → less €€€

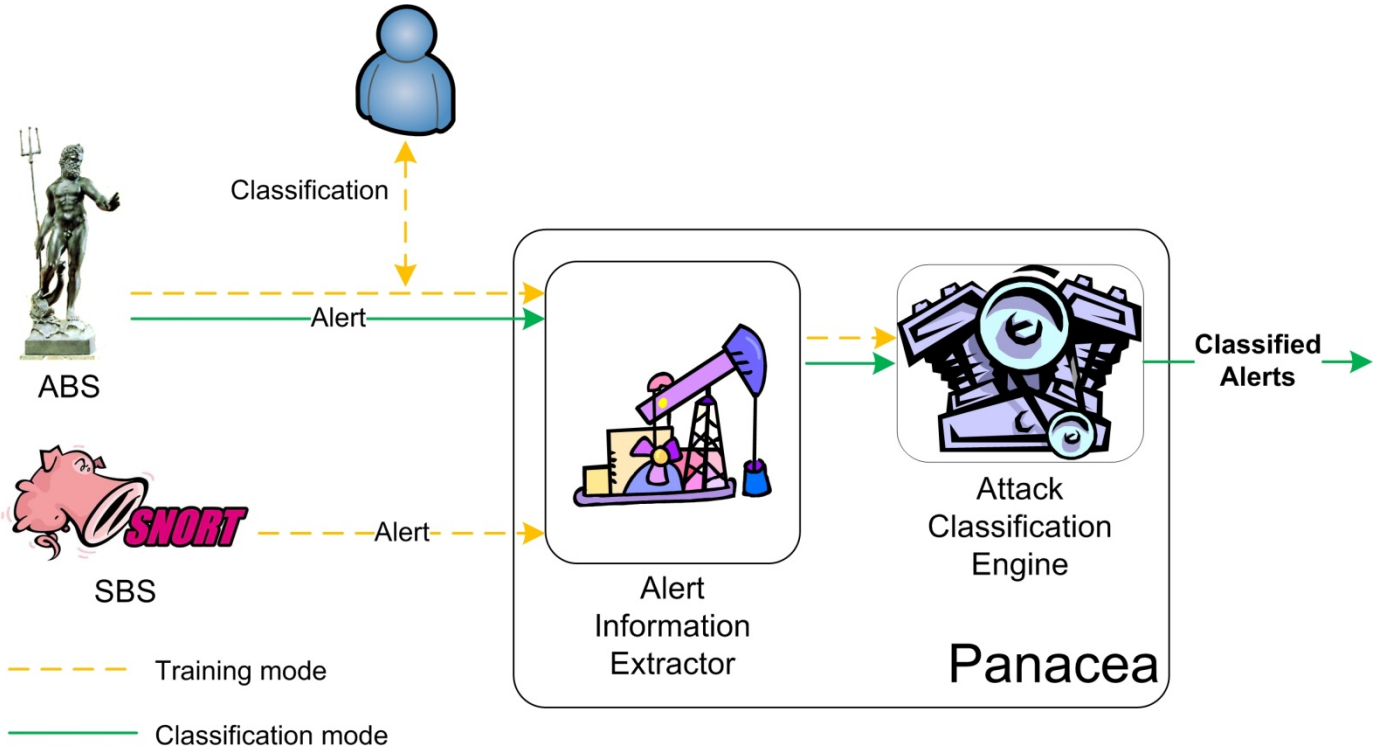
PANACEA

AUTOMATIC ATTACK CLASSIFICATION

- Idea:
  - attacks in the same class share some common content
- Goals:
  - effective
    - ✓ > 75% of correct classifications, with no human intervention
  - flexible
    - ✓ allow both automatic and manual alert classification in training mode
    - ✓ allow pre- and user-defined attack classes
    - ✓ allow users to tweak the alert classification model

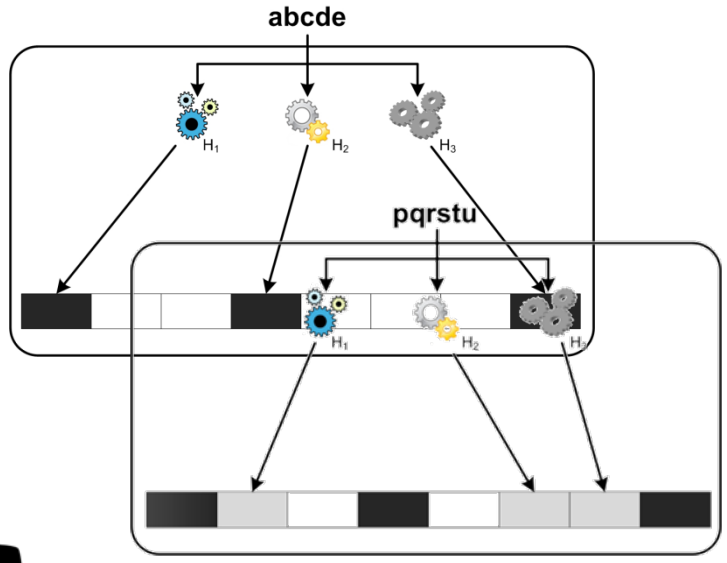
PANACEA

INTERNALS



ALERT INFORMATION EXTRACTOR

- Uses a Bloom filter to store occurrences of n-grams
  - data are sparse, few collisions
  - can handle N-grams (N >> 3)
- Stores thousands of alerts, for “batch training”



+ ALERT CLASSIFICATION (manually or automatically provided)

ATTACK CLASSIFICATION ENGINE

- Two different classification algorithms
  - non-incremental learning, more accurate than incremental ones
    - ✓ incremental learning is “simulated” by using batch training
  - process 3000 alerts in less than 40s
  - each bit of the BF is an analysis dimension
- Support Vector Machine (SVM)
  - black box, users have a few “tweak” points
- RIPPER
  - generates human-readable rules

BENCHMARKS

AUTOMATIC MODE - DATASET A

- 3000+ Snort alerts
  - pre-defined alert classes (10)
  - alerts generated by Nessus and a proprietary VA tool
  - no manual classification
  - cross-folding validation

	SVM # of samples			RIPPER # of samples		
Attack Class	1000	2000	3000	1000	2000	3000
attempted-recon	**90.9%**	90.5%	90.7%	90.4%	93.9%	**94.0%**
web-application-attack	79.8%	**89.0%**	88.8%	97.4%	98.8%	**99.1%**
web-application-activity	80.8%	**81.2%**	80.9%	93.7%	**96.1%**	95.8%

BENCHMARKS

MANUAL MODE - DATASET B

- 1500+ Snort web alerts
  - alerts generated by Nessus, Nikto and Milw0rm attacks
  - attacks are manually classified (WASC taxonomy)
  - cross-folding validation

Attack Class	SVM	RIPPER
Path Traversal	98.6%	**99.1%**
Cross-site Scripting	97.5%	**98.4%**
SQL Injection	**97.6%**	96.2%
Buffer Overflow	37.5%	37.5%
Percentage of total attacks correctly classified	**98.0%**	97.7%

BENCHMARKS

MANUAL MODE - DATASET C

- Training set:
  - Dataset B
- Testing set: 100 anomaly-based alerts
  - alerts have been captured in the wild by our POSEIDON (analyzes packet payloads) and Sphinx (analyzes web requests)

Attack Class	SVM	RIPPER
Path Traversal	**98.1%**	94.4%
Cross-site Scripting	**92.6%**	88.9%
SQL Injection	**100.0%**	87.5%
Buffer Overflow	**50.0%** (75.0%)	25.0% (50.0%)
Percentage of total attacks correctly classified	**92.0%** (93.0%)	89.0% (91.0%)

BENCHMARKS

SUMMARY

- SVM performs better than RIPPER on a class with few samples (~50)
- RIPPER performs better than SVM on a class with a sufficient number of samples (~70)
- SVM performs better than RIPPER on a class with a high intra-class diversity and when attack payloads have not been observed during training

CONCLUSION & FUTURE WORK

- Panacea fulfills our goals
  - however, it works only in combination with payload-based NIDSs

- Panacea 2.0
  - improved classification
    - ✓ a 2nd order polynomial for SVM increases accuracy to 99% but is x50 slower!
    - ✓ combining SVM and RIPPER when training samples are scarce
  - apply to alert verification
    - ✓ non-relevant true positives and false positives

QUESTIONS

?