Using Contextual Information for IDS Alarm Classification

François Gagnon

Ph.D. Student

Carleton University, Canada

fgagnon@sce.carleton.ca

www.nmai.ca

Frédéric Massicotte

Communications Research

Centre Canada

Babak Esfandiari

Carleton University

Canada

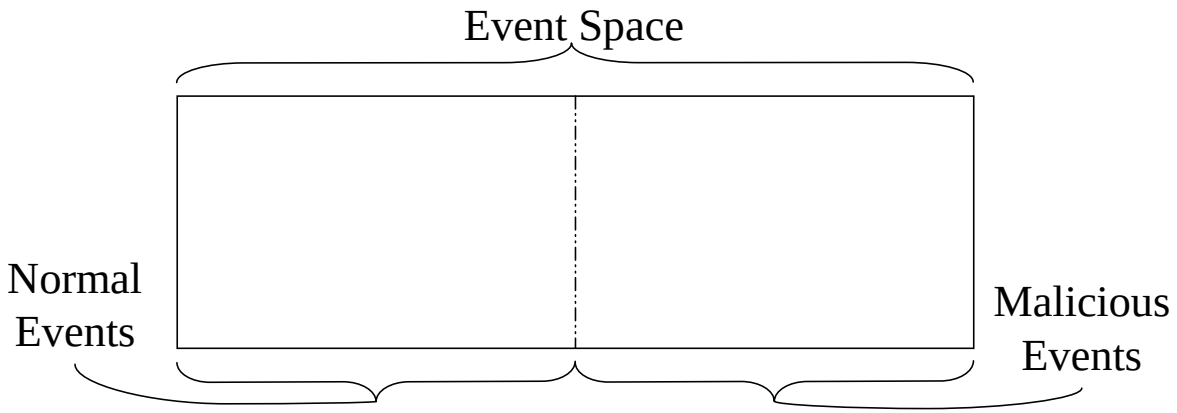
- IDSes produce a lot of alarms.
- Administrators are overwhelmed with *non-critical* alarms.

Outline

- **Introduction**
- Experiment Setup
- Results
- Conclusion

Alarm Classes

Non-critical alarms are not indicative of a plausible threat.

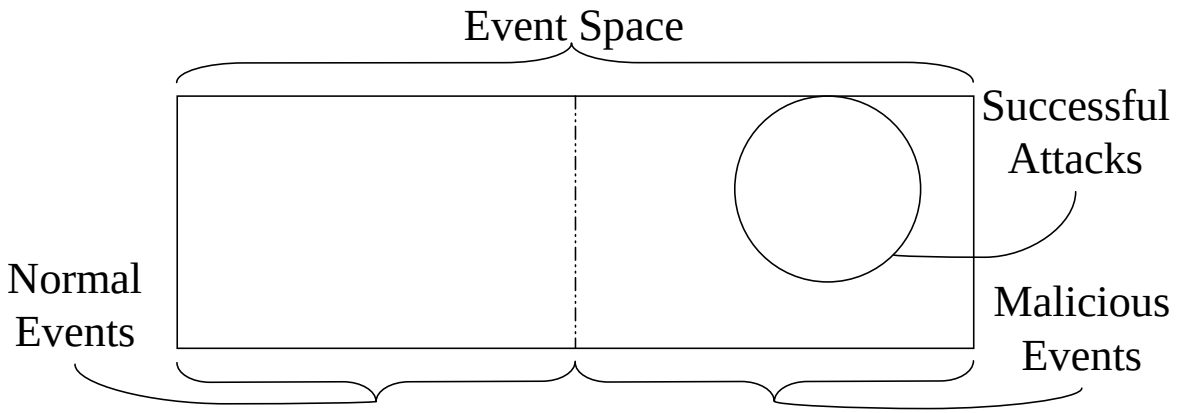


Non-Critical Alarm: A false positive (alarm related to normal background traffic) or a non-relevant positive (alarm related to an unsuccessful attack attempt).

- They pose two problems:
  - Distract security officers from real threats.
  - Prevent automatically blocking attacks.

Alarm Classes

Non-critical alarms are not indicative of a plausible threat.

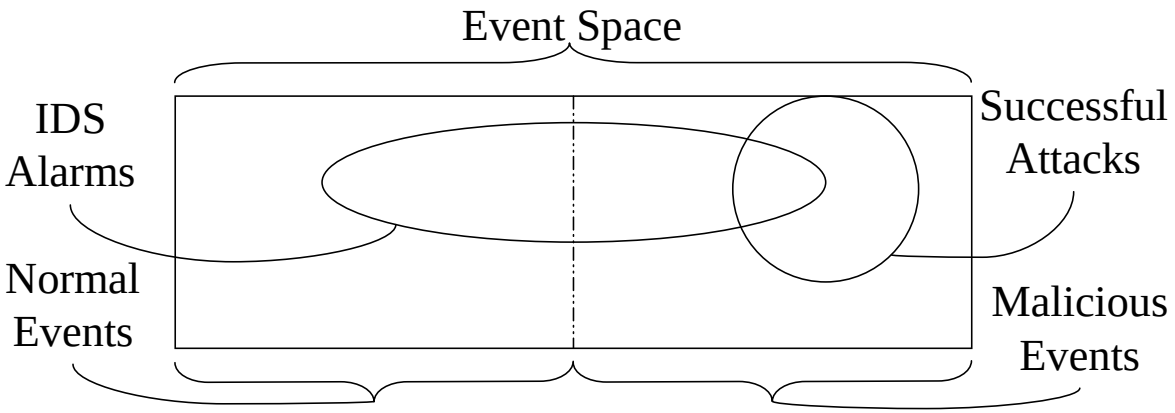


Non-Critical Alarm: A false positive (alarm related to normal background traffic) or a non-relevant positive (alarm related to an unsuccessful attack attempt).

- They pose two problems:
  - Distract security officers from real threats.
  - Prevent automatically blocking attacks.

Alarm Classes

Non-critical alarms are not indicative of a plausible threat.

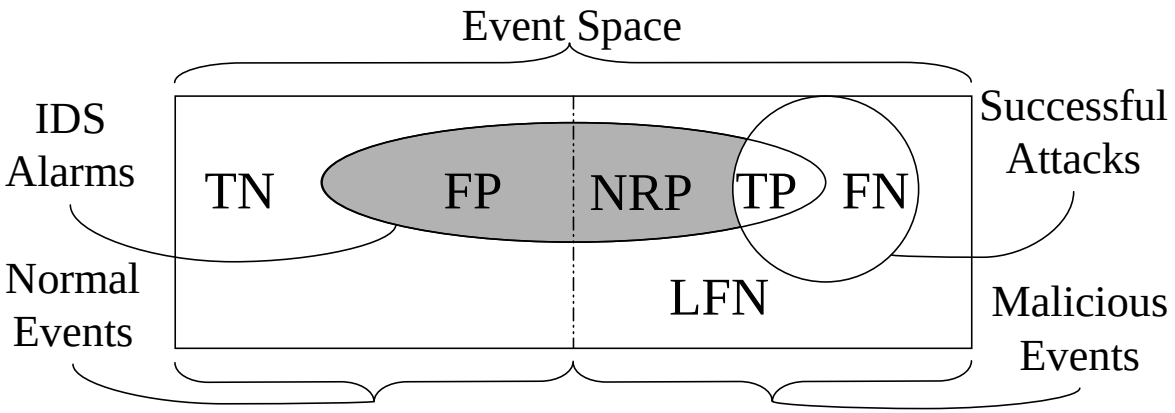


Non-Critical Alarm: A false positive (alarm related to normal background traffic) or a non-relevant positive (alarm related to an unsuccessful attack attempt).

- They pose two problems:
  - Distract security officers from real threats.
  - Prevent automatically blocking attacks.

Alarm Classes

Non-critical alarms are not indicative of a plausible threat.



Non-Critical Alarm: A false positive (alarm related to normal background traffic) or a non-relevant positive (alarm related to an unsuccessful attack attempt).

- They pose two problems:
  - Distract security officers from real threats.
  - Prevent automatically blocking attacks.

Introduction

Using Contextual Information

- An attack succeeds only when several conditions are met.
- As soon as 1 condition is not respected, the attack fails.
- Using the attack context, we can identify some of those that will fail.
- Several types of contextual information:
  - Network (topology and protocols)
  - Attack side effect (returned messages and log files)
  - Vulnerability assessment
  - **Target configuration** (operating system and applications)

Introduction

Objectives

- Potential:

  Is target configuration an effective piece of contextual information to classify IDS alarms ?

- Current:

  Are the existing tools good enough to gather this context automatically ?

Outline

- Introduction
- **Experiment Setup**
- Results
- Conclusion

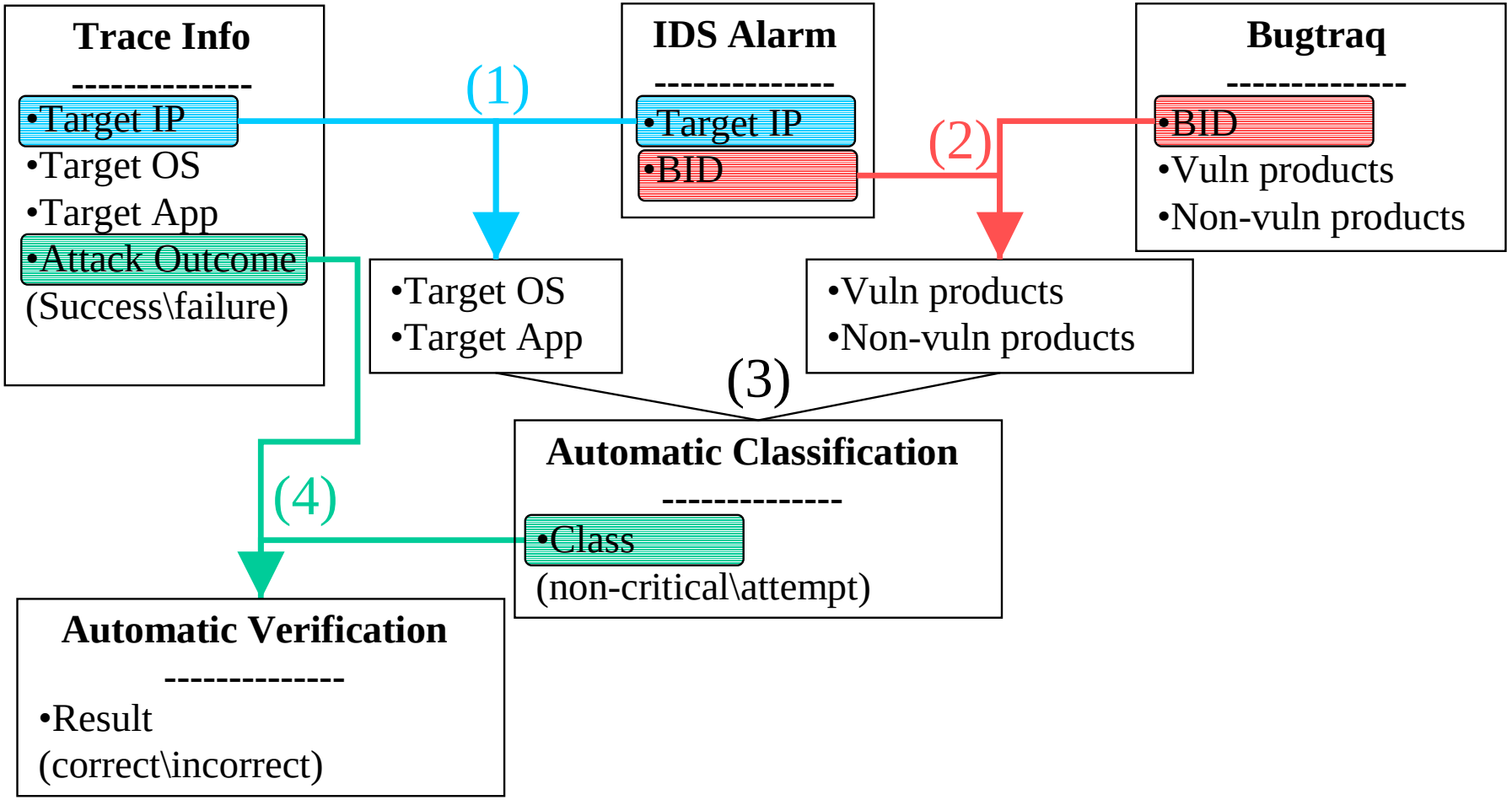
Experiment Setup

Dataset

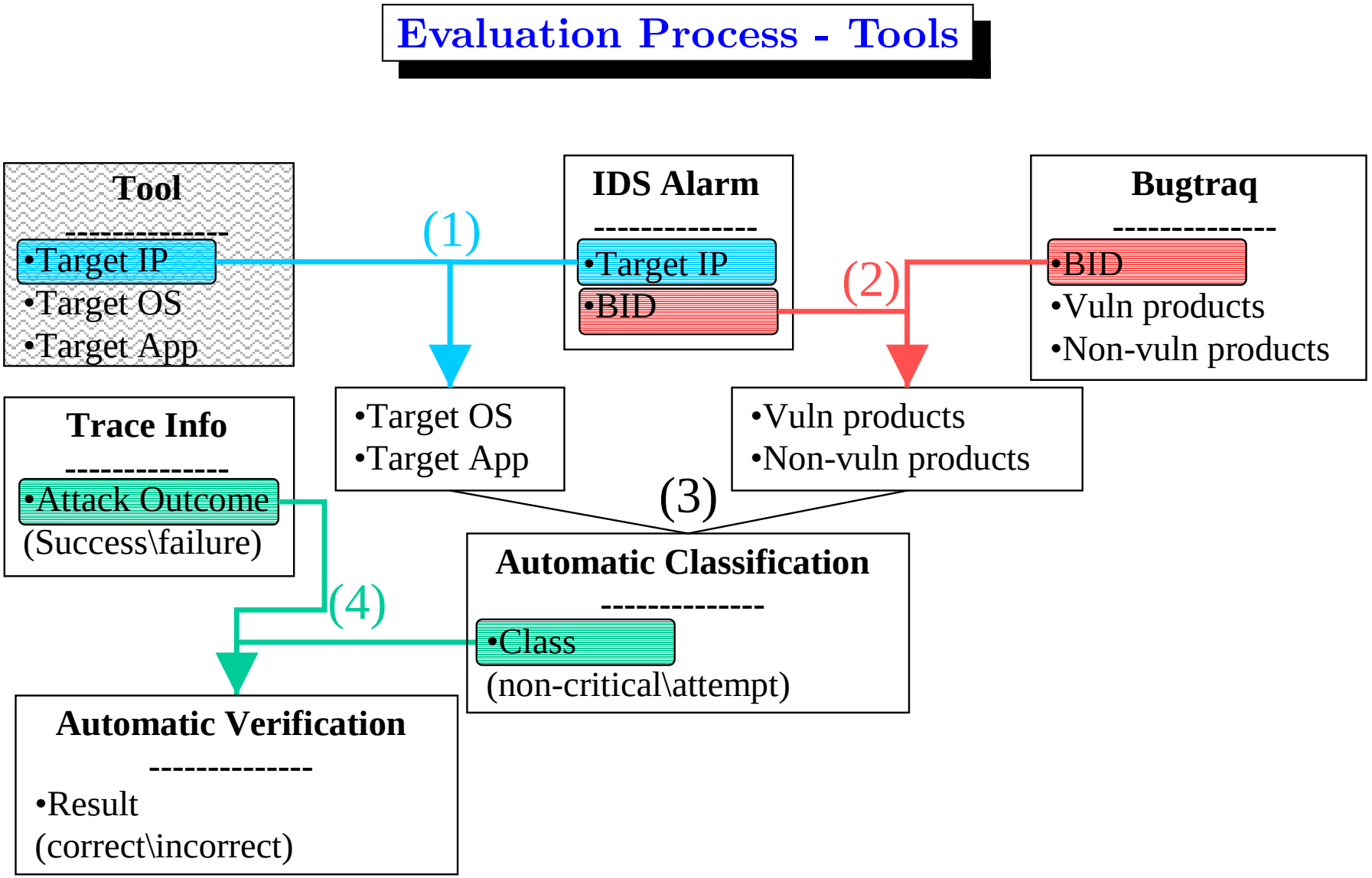
- Using freely available attack dataset from CRC [2]
- 5,761 traces (1 trace ⇒ 1 attack attempt ⇒ 1 alarm)
- No background traffic
- 92 exploits
  - Covering 47 vulnerabilities (BIDs)
  - Targeting 18 ports (TCP and UDP)
- 95 targets (34 BSD, 25 Linux, 36 Windows)
- Well-documented
  - Target OS and App
  - Attack result (success/failure)
  - Snort alarms

Experiment Setup

Evaluation Process - Potential



Experiment Setup



Experiment Setup

Classification Algorithms

ContextOS:

(1) if the target OS is listed as non-vulnerable for this exploit, return NC

(2) if the target OS is not listed as vulnerable for the BID and

(2.1) if all the products listed as vulnerable are OSes, return NC

(3) return A

ContextApp: considers only application
ContextOSApp: considers both OS and App
ContextOSDeduction: considers only OS and deduce some App info from OS
(e.g., Microsoft IIS cannot run on a Linux computer)

Outline

- Introduction
- Experiment Setup
- **Results**
- Conclusion

Performance Measures

$$Recall = \frac{\text{\# of non-critical alarms classified as NC}}{\text{\# of non-critical alarms}} = \frac{\alpha}{\alpha + \gamma}$$

$$Precision = \frac{\text{\# of non-critical alarms classified as NC}}{\text{\# of alarms classified as NC}} = \frac{\alpha}{\alpha + \beta}$$

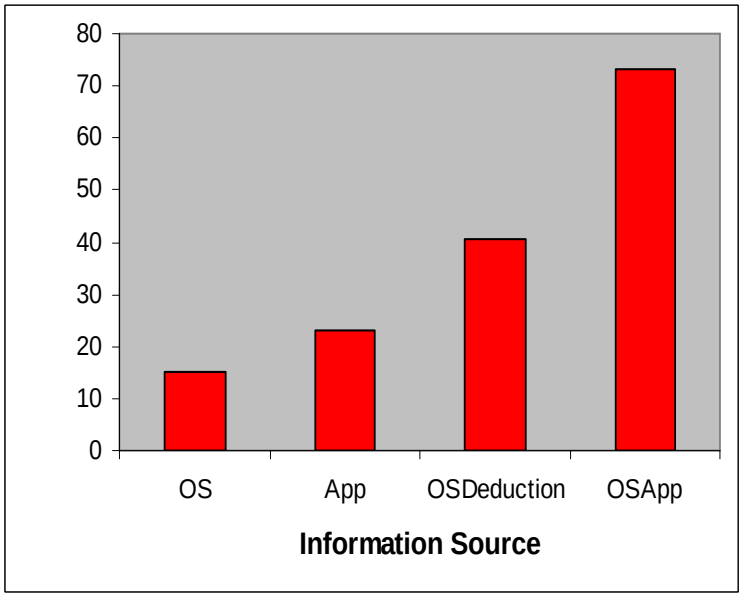
		Alarm	
		Non-critical	Critical
Classification	NC	α	β
	A	γ	δ

Results

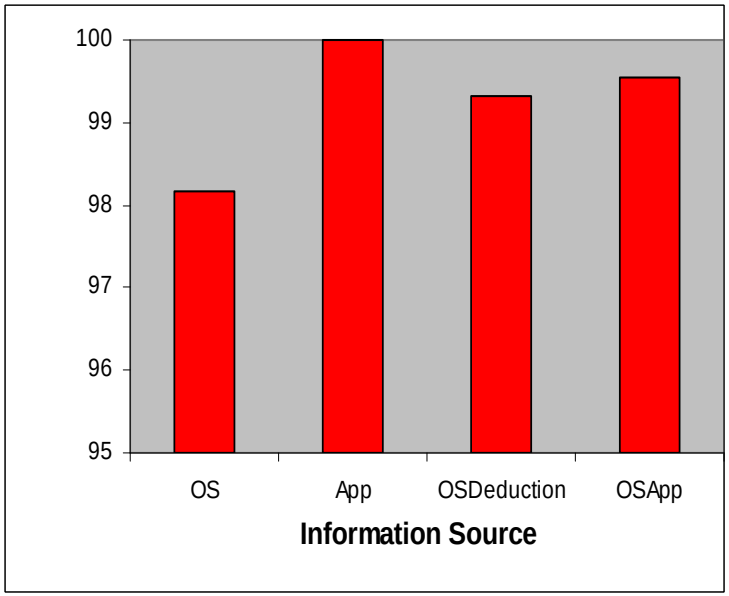
Target Configuration Potential

Assuming we know the exact target configuration (OS and App)

Recall %



Precision %

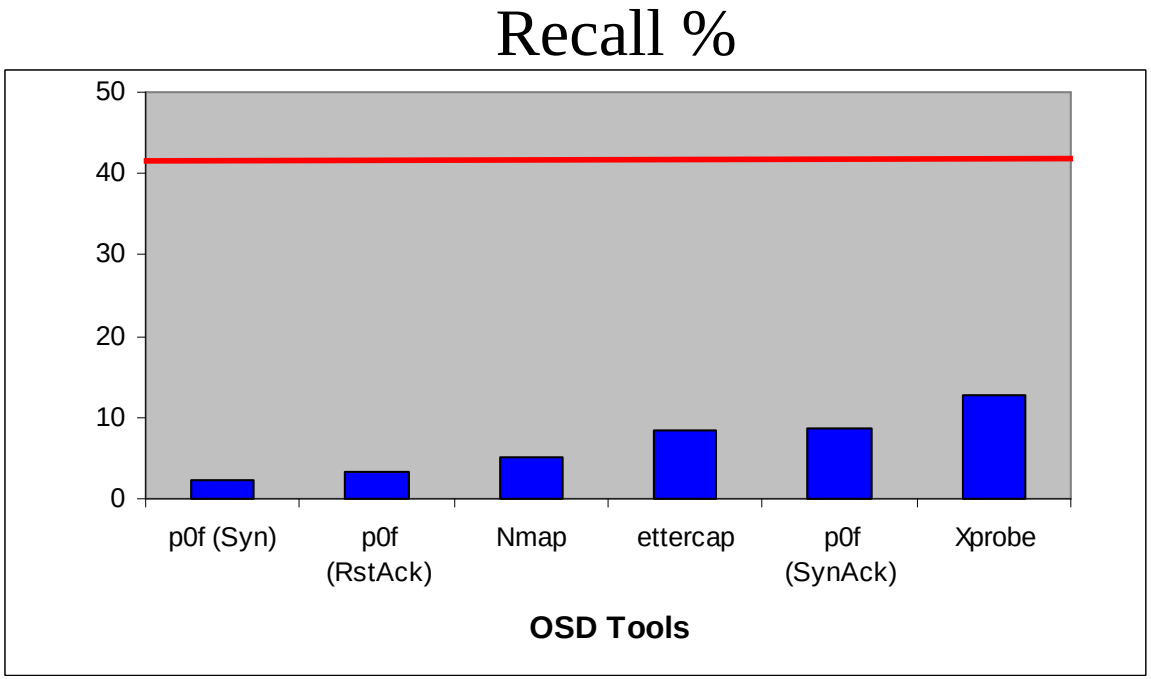


- Errors (precision decrease) are due to missing entries on securityfocus.

Results

OSD Tools

Using the ContextOSDeduction algorithm

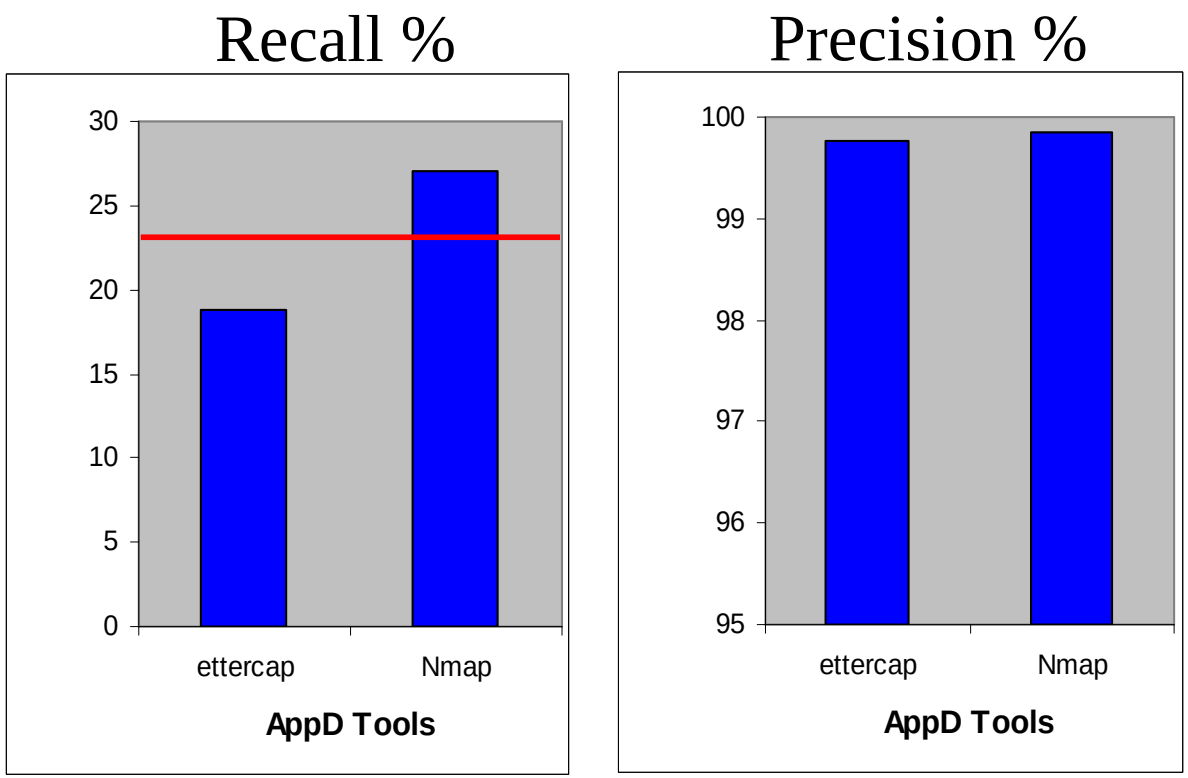


- Current OSD tools are not nearly good enough (13% vs 40%).

Results

AppD Tools

Using the ContextApp algorithm



- How can Nmap be better than the ideal case (27% vs 23%)?

Results

Weird Results

Suppose the target application (Microsoft IIS FTP) is vulnerable to the attack, but the attack fails anyway (thus it is non-critical):

- The alarm is classified A by ContextApp with exact knowledge.
- This means 0/1 for recall.

Suppose Nmap thinks the target application is wuftpd (not vulnerable):

- The alarm is now classified NC by ContextApp with Nmap.
- This means 1/1 for recall.

Those mistakes should result in a decrease of precision for Nmap (successful attack misclassified as NC).

The dataset does not have enough successful attacks.

Outline

- Introduction
- Experiment Setup
- Results
- **Conclusion**

Conclusion

Discussion

- Target configuration is very useful for IDS context:
  - Filtering 73% of non-critical alarms.
  - Not filtering critical alarms.
- OSD tools are not adequate to gather the required contextual information (they achieve only 1/3 of potential).
- There is a possibility for an attacker to manipulate the context, by injecting traffic.

Conclusion

Future Work

- Compare the effectiveness of the different IDS context elements (e.g., vulnerability assessment with Nessus vs target configuration vs attack side effect).
- Develop a new OS discovery approach (HOSD[a]) [1].
  - Detect manipulation attempts on the context.
- Re-run the experiment on another dataset.

[a] `http://hosd.sourceforge.net`

Questions

fgagnon@sce.carleton.ca

References

[1] François Gagnon, Babak Esfandiari, and Leopoldo Bertossi. A Hybrid Approach to Operating System Discovery Using Answer Set Programming. *Proceedings of the 10th IFIP/IEEE Symposium on Integrated Management (IM'07)*, pages 391–400, 2007.

[2] Frédéric Massicotte, François Gagnon, Mathieu Couture, Yvan Labiche, and Lionel Briand. Automatic Evaluation of Intrusion Detection Systems. *Proceedings of the 2006 Annual Computer Security Applications Conference (ACSAC'06)*, 2006.

Extra

Comparing HOSD