A New Hope for DARPA OpTC

Frédéric MAJORCZYK, Barbara PILASTRE and Fanny DIJOUD

December 8^{th}, 2025

INRIA Teams: PIRAT\'); and Malt

Introduction

Intrusion Detection System (IDS)

- IDS offer a way to detect attacks and let operators react according to the alerts.
- To design and evaluate new IDS, we need data.
- Network and host datasets exist, a few datasets contain both types of data[1]. The DARPA OpTC dataset is one of them.

[1] Goldschmidt and Chudá. "Network intrusion datasets: a survey, limitations, and recommendations", 2025.

DARPA OpTC

- DARPA OpTC is interesting:
  - Long dataset (with a long period of time with no attack).
  - Large IT system.
  - APT attacks.
  - Both types of data.
- However it has some issues:
  - Errors.
  - "Official" labels are imprecise.
  - Lack of documentation.
  - Making unclear the performances of IDS.

Contributions

- We identify some patterns of errors and fix them.
- We label both network and host data and compare the host labels to a recent work.
- We evaluate the impacts of the errors and the labels on three IDS.
- We discuss how to build good datasets.

Summary

Errors and Corrections

DARPA OpTC Details

DARPA OpTC Description

- 9 days of activity (from the 16th to the 25th September 2019).
- 3 APTs attacks on the last 3 days.
- Some benign activity (but no documentation).
- 1000 Windows 10 clients + some servers (notably a DC).
- 2 types of logs:
  - Zeek logs.
  - eCAR logs[2].

[2] Anjum et al. "Analyzing the Usefulness of the DARPA OpTC Dataset in Cyber Threat Detection Research", 2021.

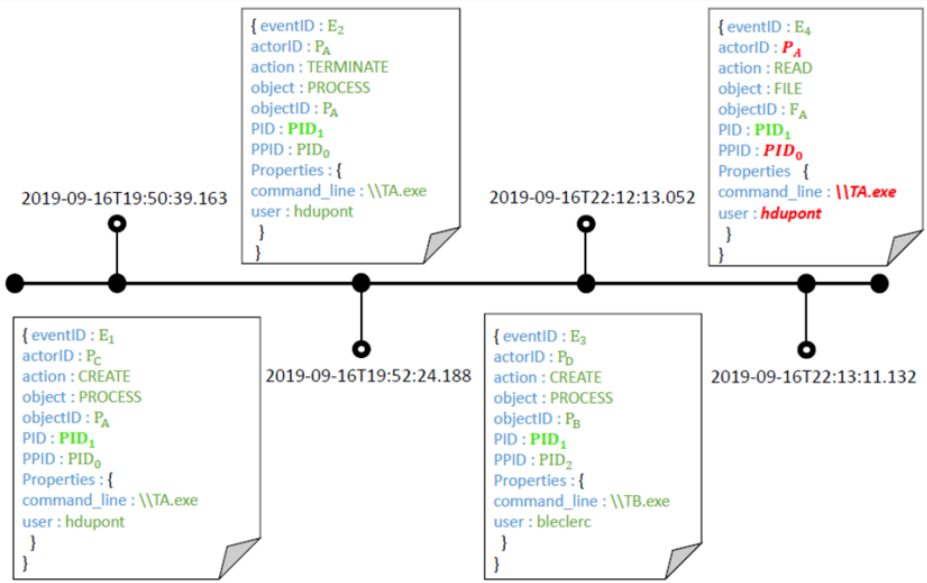

```
{
 "action":"TERMINATE",
 "actorID":"5689402e-9c41-4fe1-aff4-a0b43a8417cf",
 "hostname":"SysClient0063.systemia.com",
 "id":"71ac43d3-3bb8-44b9-9bad-127ddd0c57fe",
 "object":"PROCESS",
 "objectID":"5689402e-9c41-4fe1-aff4-a0b43a8417cf",
 "pid":4772,
 "ppid":776,
 "principal":"SYSTEMIACOM\\agilbard",
 "properties":{
  "command_line":"\\??\\C:\\Windows\\system32\\conhost.
       exe 0xffffffff -ForceV1",
  "image_path":"\\Device\\HarddiskVolume1\\Windows\\
       system32\\conhost.exe",
  "parent_image_path":"\\Device\\HarddiskVolume1\\
       Windows\\system32\\conhost.exe",
  "sid":"S-1-5-21-4190936083-3304963419-
1584388968-3208"
,
  "user":"SYSTEMIACOM\\agilbard"
 },
 "timestamp":"2019-09-20T23:15:42.194-04:00"
}
```

An Example of Error

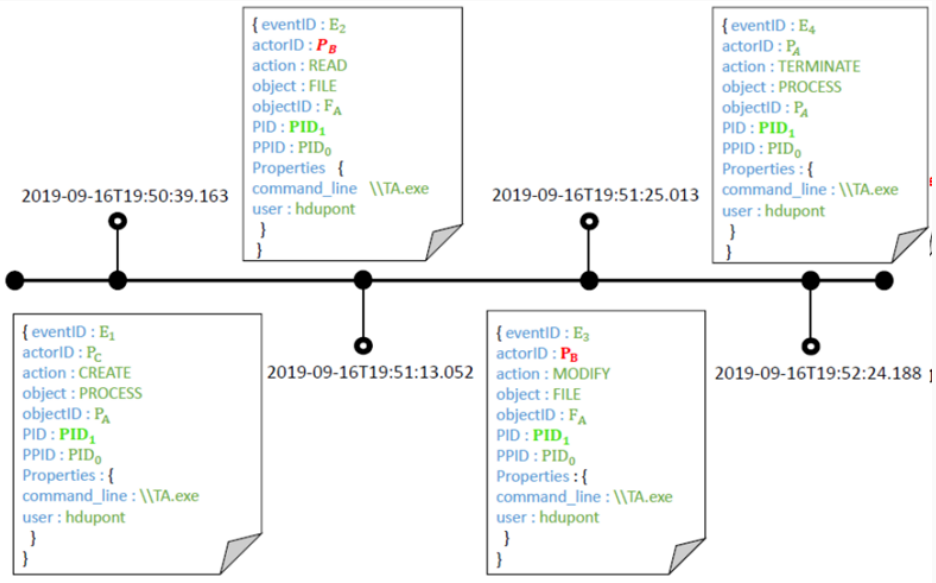

```
{
 "action":"CREATE",
 "actorID":"ecb11af3-3262-4fae-bb3c-083fc9475443",
 "hostname":"SysClient0063.systemia.com",
 "id":"f946815a-d428-41c5-8a5f-d808d0c9bd45",
 "object":"PROCESS",
 "objectID":"b8d71d4a-2184-4217-8383-123f66043e39",
 "pid":4772,
 "ppid":3812,
 "principal":"NT AUTHORITY\\SYSTEM",
 "properties":{
  "command_line":"C:\\Windows\\system32\\cmd.exe /c "
       schtasks /delete /tn "KickoffAutoLoginRandom" /F
       "",
  "image_path":"\\Device\\HarddiskVolume1\\Windows\\
       system32\\cmd.exe",
  "parent_image_path":"\\Device\\HarddiskVolume1\\
       Python27\\python.EXE",
  "sid":"S-1-5-18",
  "user":"NT AUTHORITY\\SYSTEM"
 },
 "timestamp":"2019-09-21T03:03:07.087-04:00"
}
```

```
{
 "action":"TERMINATE",
 "actorID":"5689402e-9c41-4fe1-aff4-a0b43a8417cf",
 "hostname":"SysClient0063.systemia.com",
 "id":"15aff6c7-a663-4244-a97a-258fc66625ed",
 "object":"PROCESS",
 "objectID":"5689402e-9c41-4fe1-aff4-a0b43a8417cf",
 "pid":4772,
 "ppid":776,
 "principal":"SYSTEMIACOM\\agilbard",
 "properties":{
  "command_line":"\\??\\C:\\Windows\\system32\\conhost.
       exe 0xffffffff -ForceV1",
  "image_path":"\\Device\\HarddiskVolume1\\Windows\\
       system32\\conhost.exe",
  "parent_image_path":"\\Device\\HarddiskVolume1\\
       Windows\\system32\\conhost.exe",
  "sid":"S-1-5-18",
  "user":"NT AUTHORITY\\SYSTEM"
 },
 "timestamp":"2019-09-21T03:03:07.132-04:00"
}
```

Timeline Error 1



Timeline Error 2



Correction Analysis (1)

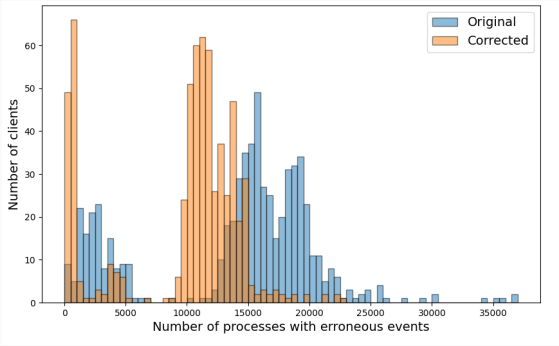
- Error detection step.
- Error correction step.

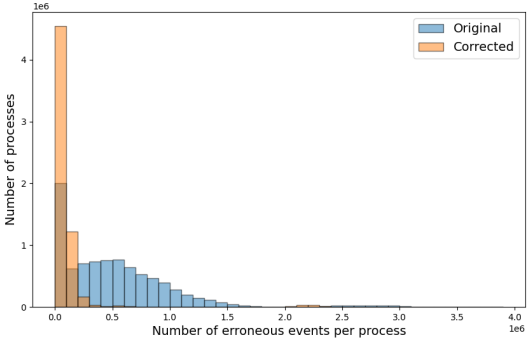
Number of errors (proportion in parenthesis)

Correction	Processes with errors	Erroneous events
Original	8 645 600 (40%)	701 039 318 (4.22%)
First pass	6 169 386 (28.83%)	313 063 564 (1.89%)
Second pass	6 035 310 (28.32%)	289 904 664 (1.75%)
Third pass	6 014 764 (28.24%)	261 603 670 (1.6%)

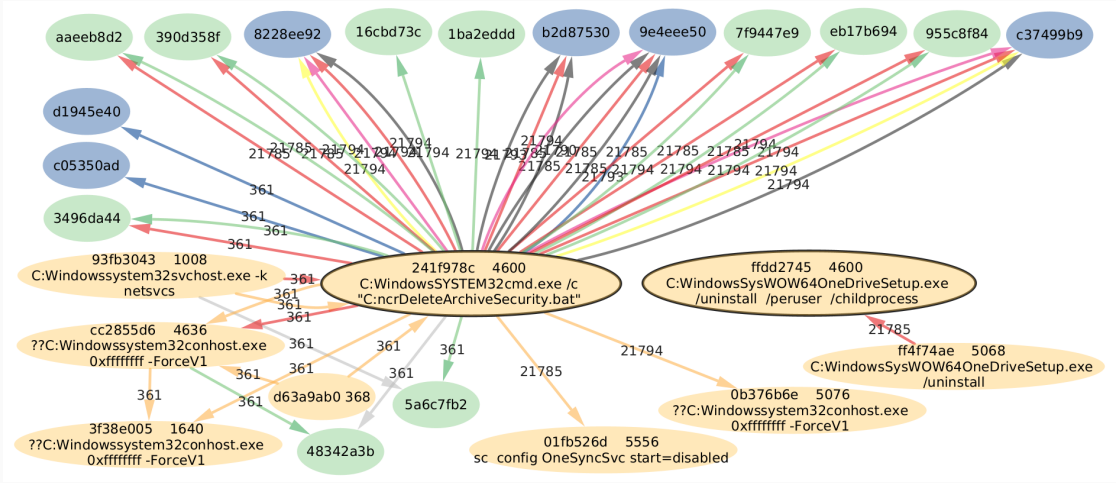
- Decrease of the number of erroneous events.
- Smaller decrease of the number of erroneous processes but many processes have only a few erroneous events.
- One month to compute the three correction passes.

Correction Analysis (2)

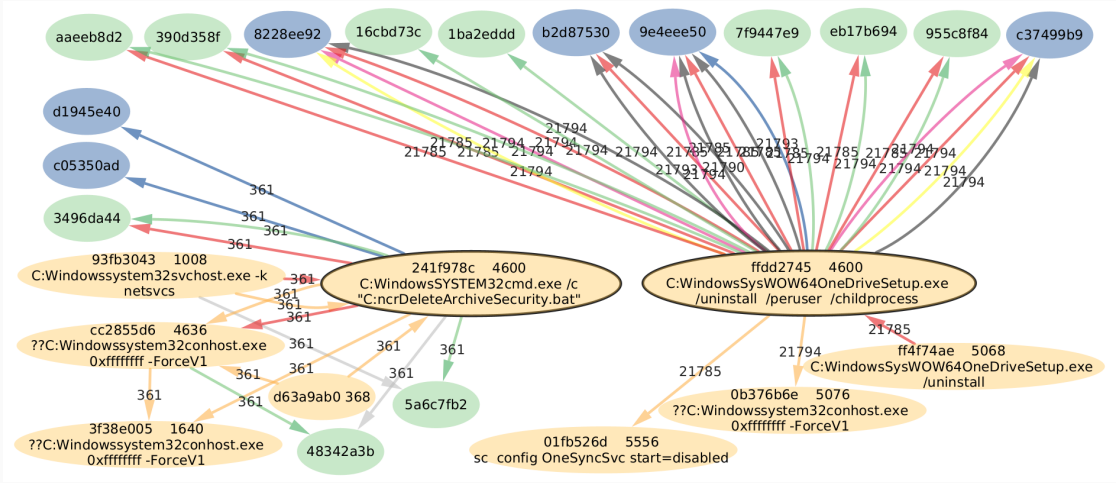




Correction Analysis (3) : Provenance Graphs



Correction Analysis (4) : Provenance Graphs



Labeling

Description of Attacks (1)

Day 1 "Plain PowerShell Empire"

Summary:
Manually connected to Sysclient0201 and downloaded malicious PowerShell Empire stager stored in a batch file. Used priv escalation methods to obtain elevated agent. Used Mimikatz to collect credentials. Attempted to inject into LSASS but it failed. Used registry edits to establish persistence. Collected screenshot of desktop. Attempted network scanning methods. Created port scan script, imported into memory and ran against /24 network. Pivoted to Sysclient0402 with WMI.

On Sysclient0402, conducted ping sweep of /24 network. Pivoted to Sysclient0660 via WMI.

On Sysclient0660, used ipconfig through powershell to obtain ip. Used mimikatz in attempt to gather logged in user's password. Attempted process migration, failed. Attempted to inject shellcode, failed. Ran scripts to get information on Domain Controller. Downloaded file from C:/. Used WMI to pivot to DC1.

Killed all other agents.

On DC1, ran mimikatz. Obtained user hashes with lsadump. Pivoted to 14 stations. Killed all agents. End of day1.

C2:
server -- news.com:80
ip -- 132.197.158.98
delay -- 5 seconds
profile -- /admin/get.php,/news.php,/login/process.php|Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko

Client,AgentID, ReportedIP, PID
Sysclient0201 VL8B5T3U 142.20.56.202 5452
Sysclient0201 LUAVR71T 142.20.56.202 2952
Sysclient0402 NEK5H8GX 142.20.57.147 3168
Sysclient0660 DS29HY41 142.20.58.149 880

Description of Attacks (2)

Log:
09/23/19 11:23:29 -- Manually accessed console on Sysclient0201 and navigated to news.com:8000 to download runme.bat, a malicous Powershell empire stager
09/23/19 11:24:19 -- Sysclient0201, closed firefox tab and deleted runme.bat
09/23/19 11:24:54 -- Successfull checkin for Agent VL8B5T3U on Sysclient0201 ip 142.20.56.202
09/23/19 11:26:02 -- On Sysclient0201 agent VL8B5T3U, Used PowershellEmpire Module Bypasses UAC, which performs a registry modification of the "windir" value in "Environment" to bypass UAC.
09/23/19 11:26:38 -- On sysclient0201, successful checkin of elevated agent LUAVR71T
09/23/19 11:26:56 -- On Sysclient0201, killing agent VL8B5T3U
09/23/19 11:33:14 -- On Sysclient0201 agent LUAVR71T, ran mimikatz to collect clear text passwords in memory
09/23/19 11:35:26 -- On Sysclient0201 agent LUAVR71T, obtained password for user systemia.com\zleazer via mimikatz
09/23/19 11:37:15 -- On Sysclient0201 agent LUAVR71T, used psinject to inject into the LSASS process
09/23/19 11:38:09 -- On Sysclient0201 agent LUAVR71T, Process injection seems to have failed.
09/23/19 11:39:25 -- On Sysclient0201 agent LUAVR71T, established persistence by modifying HKCU:Software\Microsoft\Windows\CurrentVersion\Debug registry entry
09/23/19 11:40:41 -- On Sysclient0201 agent LUAVR71T, obtained process listing by running ps through powershell
09/23/19 11:41:50 -- On Sysclient0201 agent LUAVR71T, retried psinject into lsass
09/23/19 12:51:59 -- On Sysclient0201 agent LUAVR71T, collected data collection by obtaining a screenshot of the desktop.
09/23/19 12:58:20 -- On Sysclient0201 agent LUAVR71T, conducted ARP scan on /22 of 142.20.56.202
09/23/19 13:07:11 -- On Sysclient0201 agent LUAVR71T, ARP scan failed. Attempting to use SMB on 142.20.56.204 with credentials from Sysclient0201
09/23/19 13:08:28 -- On Sysclient0201 agent LUAVR71T, re-attempted ARP scan without setting any values.

Network Methodology

- Use the information about the C2.
- Use the information about the compromised clients.
- Use some more precise information in the details (browsing, RDP connections, ...).

  $\rightarrow$ to filter the Zeek logs.
- Manually verify the logs.
- Extend the research to other clients.

Results Examples

```
1569261902.996342 C2rmiIdN7rpSVZUc8 142.20.61.130 51653
132.197.158.98 80 tcp http 0.051639 203 1443 SF F F 0 ShADadfF 5 415
6 1695 (empty)

1569261903.000729 C2rmiIdN7rpSVZUc8 142.20.61.130 51653
132.197.158.98 80 1 GET news.com /login/process.php - 1.0 Mozilla/5.0
(Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko 0 1216 200
OK - - (empty) - - - - - - FpYgq92O7Q5jQgbNm8 - text/html

1569261903.053730 FpYgq92O7Q5jQgbNm8 132.197.158.98 142.20.61.130
C2rmiIdN7rpSVZUc8 HTTP 0 NEMAEXTRACT text/html - 0.000000 F F 1216
1216 0 0 F - - - - - - -
```

Strange Findings

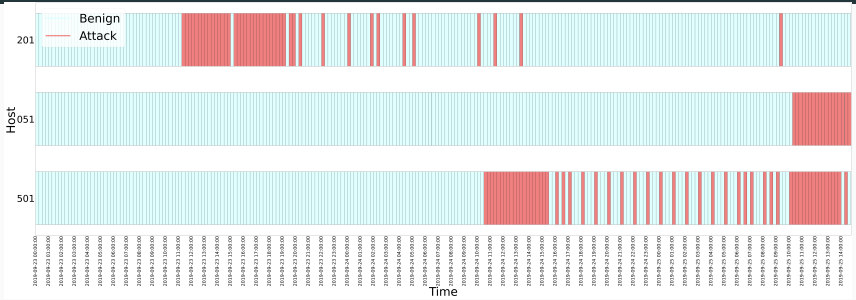
- For APT Scenario 1, 4 clients (not described in the documentation) connect to the C2 with exactly the same patterns.
- For APT Scenario 2, 1 client (not described in the documentation) connects to the C2 with a quite similar pattern.
- For APT Scenario 3, 1 client (not described in the documentation) also connects to the C2 with a quite similar pattern.

  $\rightarrow$ We label these connections as malicious.
- The eCAR logs are not provided for these clients.

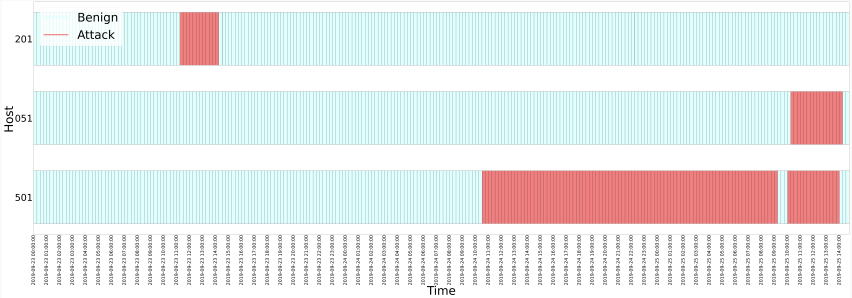
Host Methodology

- Use the information about the pid of the malware agents.
- Use some more precise information in the details (RDP connections, process injections, ...).
- Reconstruct the process trees from this information.
- Check if the parent of each process is malicious.
- Label as malicious all the activities of the processes.

Some results and comparison with Nikulshin and Tahli



Nikulshin and Tahli



Our labels

Experiments

Experimentation setups

Goals

- Evaluate the impacts of the **labels**.
- Evaluate the impacts of the **correction**.

The purpose of these experiments is not to promote a particular label or OpTC version that would improve IDS performance, but rather to focus on **performance variations**.

Methodology

- Analyze performances of **3 IDS**: Kairos [1] (S&P 2024), Flash [5](S&P 2024) and a Graph Auto-Encoder (GAE) [2] [3](NEURIPS 2022).
- Focus on **3 critical clients**: 051, 201 and 501. Each involved in a **distinct attack scenario and key stages** of the intrusion.
- Apply **3 labels**: our labels, labels from Nikulshin and Tahli [4] and labels from Flash [5].

Experimentation results

IDS performances on the original eCAR logs depending on the **labels**.

Approach	Client	Labels	Precision	Recall	F1-score
Kairos [1]	051,201,501	Nikulshin and Tahli	0.81	0.11	0.20
		Ours	0.81	0.10	0.18
Flash [5]	051	Flash	0.14	0.85	0.24
		Nikulshin and Tahli	0.98	0.23	0.38
		Ours	0.99	0.25	0.39
	201	Flash	0.13	0.95	0.23
		Nikulshin and Tahli	0.98	0.73	0.84
		Ours	0.98	0.73	0.84
	501	Flash	0.79	0.92	0.85
		Nikulshin and Tahli	1.00	0.72	0.83
		Ours	1.00	0.72	0.84
GAE [2] [3]	051	Nikulshin and Tahli	0.07	1.0	0.14
		Ours	0.07	0.88	0.13
	201	Nikulshin and Tahli	0.46	0.59	0.52
		Ours	0.14	0.67	0.24
	501	Nikulshin and Tahli	0.23	1.0	0.38
		Ours	0.43	0.99	0.60

The Nikulshin and Tahli's labels and our labels **converge**, suggesting a strong consensus in identifying anomalous events of the dataset.

The influence of labeling is **significant**.

Experimentation results

IDS performances on the original eCAR logs depending on the **labels**.

Approach	Client	Labels	Precision	Recall	F1-score
Kairos [1]	051,201,501	Nikulshin and Tahli	0.81	0.11	**0.20**
		Ours	0.81	0.10	**0.18**
Flash [5]	051	Flash	0.14	0.85	0.24
		Nikulshin and Tahli	0.98	0.23	**0.38**
		Ours	0.99	0.25	**0.39**
	201	Flash	0.13	0.95	0.23
		Nikulshin and Tahli	0.98	0.73	**0.84**
		Ours	0.98	0.73	**0.84**
	501	Flash	0.79	0.92	0.85
		Nikulshin and Tahli	1.00	0.72	**0.83**
		Ours	1.00	0.72	**0.84**
GAE [2] [3]	051	Nikulshin and Tahli	0.07	1.0	**0.14**
		Ours	0.07	0.88	**0.13**
	201	Nikulshin and Tahli	0.46	0.59	0.52
		Ours	0.14	0.67	0.24
	501	Nikulshin and Tahli	0.23	1.0	0.38
		Ours	0.43	0.99	0.60

The Nikulshin and Tahli's labels and our labels **converge**, suggesting a strong consensus in identifying anomalous events of the dataset.

The influence of labeling is **significant**.

Experimentation results

IDS performances on the original eCAR logs depending on the **labels**.

Approach	Client	Labels	Precision	Recall	F1-score
Kairos [1]	051,201,501	Nikulshin and Tahli	0.81	0.11	0.20
		Ours	0.81	0.10	0.18
Flash [5]	051	Flash	0.14	0.85	0.24
		Nikulshin and Tahli	0.98	0.23	0.38
		Ours	0.99	0.25	0.39
	201	Flash	0.13	0.95	0.23
		Nikulshin and Tahli	0.98	0.73	0.84
		Ours	0.98	0.73	0.84
	501	Flash	0.79	0.92	0.85
		Nikulshin and Tahli	1.00	0.72	0.83
		Ours	1.00	0.72	0.84
GAE [2] [3]	051	Nikulshin and Tahli	0.07	1.0	0.14
		Ours	0.07	0.88	0.13
	201	Nikulshin and Tahli	0.46	0.59	**0.52**
		Ours	0.14	0.67	**0.24**
	501	Nikulshin and Tahli	0.23	1.0	**0.38**
		Ours	0.43	0.99	**0.60**

The Nikulshin and Tahli's labels and our labels **converge**, suggesting a strong consensus in identifying anomalous events of the dataset.

The influence of labeling is **significant**.

Experimentation results

IDS performances on the original eCAR logs depending on the **labels**.

Approach	Client	Labels	Precision	Recall	F1-score
Kairos [1]	051,201,501	Nikulshin and Tahli	0.81	0.11	0.20
		Ours	0.81	0.10	0.18
Flash [5]	051	Flash	0.14	0.85	**0.24**
		Nikulshin and Tahli	0.98	0.23	0.38
		Ours	0.99	0.25	0.39
	201	Flash	0.13	0.95	**0.23**
		Nikulshin and Tahli	0.98	0.73	0.84
		Ours	0.98	0.73	0.84
	501	Flash	0.79	0.92	0.85
		Nikulshin and Tahli	1.00	0.72	0.83
		Ours	1.00	0.72	0.84
GAE [2] [3]	051	Nikulshin and Tahli	0.07	1.0	0.14
		Ours	0.07	0.88	0.13
	201	Nikulshin and Tahli	0.46	0.59	0.52
		Ours	0.14	0.67	0.24
	501	Nikulshin and Tahli	0.23	1.0	0.38
		Ours	0.43	0.99	0.60

The Nikulshin and Tahli's labels and our labels **converge**, suggesting a strong consensus in identifying anomalous events of the dataset.

The influence of labeling is **significant**.

Experimentation results

IDS performances depending on **eCAR logs version**, with our labels.

Approach	Client	OpTC Version	Precision	Recall	F1-score
Kairos [1]	051,201,501	original	0.81	0.10	0.18
		corrected	0.79	0.09	0.16
Flash [5]	051	original	0.99	0.25	0.39
		corrected	1.0	0.25	0.39
	201	original	0.98	0.73	0.84
		corrected	0.99	0.73	0.84
	501	original	1.0	0.72	0.84
		corrected	1.0	0.78	0.87
GAE [2] [3]	051	original	0.07	0.88	0.13
		corrected	0.07	1.0	0.13
	201	original	0.14	0.67	0.24
		corrected	0.12	0.58	0.19
	501	original	0.43	0.99	0.60
		corrected	0.43	0.99	0.60

The corrections **did not significantly** affect IDS behavior.

The corrected version provides a consistent dataset structure, making the **analysis** of IDS's results easier for security experts.

Discussion

How to build a dataset

We encourage researchers to provide:

- A detailed network configuration.
- The entire IT system configuration: firewall rules, routing, authentication processes and services.
- Attack tool configurations, command lines and monitoring setup.
- Clear description and evaluation of the normal activity generation process.
- Artifacts.

Provide as much information as possible.

One solution is to use testbed generation tools.

We did it with DEDALE: `https://dedale.inria.fr`

Conclusion

Conclusion

Summary

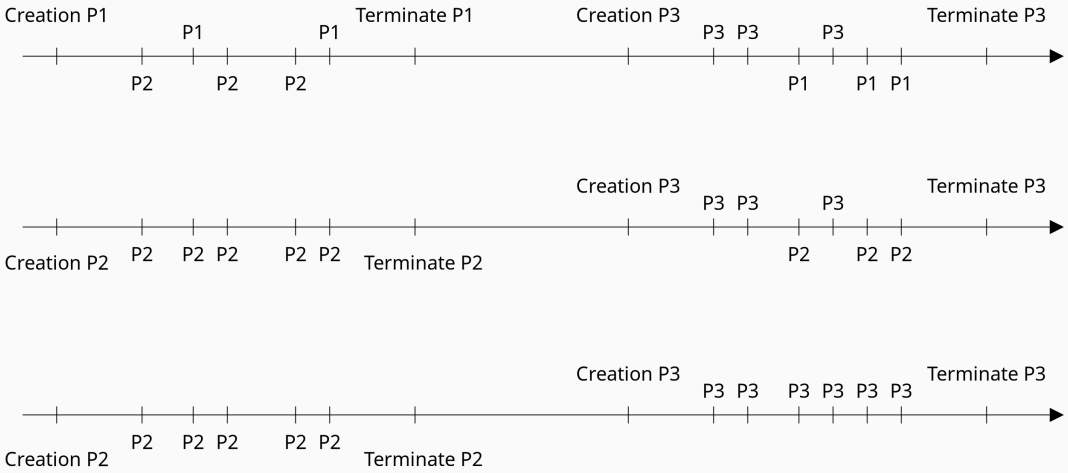
- We identify and fix `actorIDs` mismatches.
  $\rightarrow$ better provenance graphs.
- We label network and host data.
  $\rightarrow$ better labels to evaluate and compare IDS.
- We run experiments with three IDS. $\rightarrow$ more impact of the labels.
- All the scripts are available on
  `https://gitlab.inria.fr/fmajorcz/a_new_hope_for_darpa_optc`
- More information on `https://correctedoptc.inria.fr`

Future Work

- Merge the labels of Nikulshin and Tahli with our labels
- Introduce (at least) three levels of labels : malicious, related to malicious activity and benign

Additional materials

Multiple Corrections Passes



Zijun Cheng, Qiujian Lv, Jinyuan Liang, Yan Wang, Degang Sun, Thomas Pasquier, and Xueyuan Han.
Kairos: Practical intrusion detection and investigation using whole-system provenance.
In Proceedings of the 2024 IEEE Symposium on Security and Privacy (S&P), pages 3533–3551. IEEE, 2024.

Kay Liu, Yingtong Dou, Xueying Ding, Xiyang Hu, Ruitong Zhang, Hao Peng, Lichao Sun, and Philip S. Yu.
PyGOD: A Python library for graph outlier detection.
Journal of Machine Learning Research, 25(141):1–9, 2024.

Kay Liu, Yingtong Dou, Yue Zhao, Xueying Ding, Xiyang Hu, Ruitong Zhang, Kaize Ding, Canyu Chen, Hao Peng, Kai Shu, Lichao Sun, Jundong Li, George H Chen, Zhihao Jia, and Philip S Yu.
BOND: Benchmarking unsupervised outlier node detection on static attributed graphs.
In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, Proceedings of the Advances in Neural Information Processing Systems, volume 35, pages 27021–27035. Curran Associates, Inc., 2022.

Victor Nikulshin and Chamseddine Talhi.
Effective IDS under constraints of modern enterprise networks: revisiting the OpTC dataset.
In Proceedings of the 2024 7th Conference on Cloud and Internet of Things (CIoT), pages 1–8. IEEE, 2024.


Mati Ur Rehman, Hadi Ahmadi, and Wajih Ul Hassan.
FLASH: A comprehensive approach to intrusion detection via provenance graph representation learning.
In Proceedings of the 2024 IEEE Symposium on Security and Privacy (S&P), pages 139–139. IEEE Computer Society, 2024.