LADEMU: a modular & continuous approach for generating labelled APT datasets from emulations

1st Julie Gjerstad
Norwegian Defence Research Establishment (FFI)
Kjeller, Norway

2nd Fikret Kadiric
FFI
Kjeller, Norway

3rd Gudmund Grov
FFI & University of Oslo
Kjeller, Norway
Gudmund.Grov@ffi.no

4th Espen Hammer Kjellstadli
FFI, Kjeller, Norway
Espen-Hammer.Kjellstadli@ffi.no

5th Markus Leira Asprusten
FFI, Kjeller, Norway
Markus.Asprusten@ffi.no

***Abstract*—Development and evaluation of data-driven capabilities for both threat hunting and intrusion detection require high-quality and up-to-date datasets. The generation of such datasets poses multiple challenges, which has led to a general lack of suitable datasets for this domain.**

One such difficulty is the ability to correctly label each datapoint at a suitable level of granularity. In this paper, we argue that the challenges faced when labelling datasets can to some degree be decoupled from realistic emulations of up-to-date attacks and benign behaviours. We propose a modular labelling approach that can be combined with existing emulation platforms that provide the necessary details used for labelling. A proof-of-concept implementation is provided with our LADEMU (Labelled Apt Datasets from EMUlations) tool, which is integrated with the Mitre CALDERA emulation platform and uses the GHOSTS framework for benign behaviour. LADEMU captures both host and network logs and labels them at a sufficient level of detail to separate the various attack steps. This provides dataset support for the development of data-driven APT, multi-step and kill-chain capabilities. As a case, LADEMU is used to generate a labelled dataset from an intelligence-driven emulation plan of an advanced persistent threat (APT) group.

***Index Terms*—Dataset generation, labelling, APT**

I. Introduction

Realistic and labelled datasets are a necessity when developing data-driven capabilities for both threat hunting and intrusion detection [1], [34], [42]. Datasets used to build such hunting or detection capabilities comes with a large set of requirements from different sources:

R1) datasets must contain modern attack data that is representative of current trends [20], [28];
R2) datasets need to be representative and accurate [20];
R3) datasets must provide all the relevant behavioural patterns for malicious and normal activities, and network traces [8], [29];
R4) datasets must capture the stages and strategies involved in the attacks to defend against *Advanced Persistent Threats* (APTs) [1];
R5) datasets must contain *ground truth*[1] of the datapoints; to develop capabilites to detect APTs, or perform kill-chain detection, the labels must be fine-grained and indicate the different stages of an attack/campaign [8], [20], [20].

Satisfying these requirements is far from easy. Modern attacks are rapidly evolving with new and ”better” malware and attacks being introduced every day [1] – and utilising both the growing complexity of network systems, software, and services, as well as their rising integration and dependencies [35]. They increasingly contain stealthy behaviour, characterised by low and slow movements [1], while APTs often use methods that defenders have never seen before [7] and their complexity makes it challenging to effectively model their capabilities [36].

As a result, there has consistently been a lack of suitable datasets for this domain [9], [20], [30], [36] – and even more so for APTs [36] – causing some to argue that lack of suitable datasets constitutes one of the biggest challenges for developing capabilities to defend against APTs [34].

There are several ways to create datasets which we can roughly put into two categories:

- use of existing data – e.g. logs from real enterprise network;
- emulation[2] of malicious and/or benign behaviour, typically in a controlled environment.

Note that these can also be combined when creating datasets.

Focusing on emulation, we address (R5), labelling of datasets, in this paper and argue that this can to a large extent be decoupled from the other requirements. This is grounded

Both the source code of LADEMU and generated dataset can be found here: https://github.com/FFI-no/Paper-LADEMU

978-1-6654-8045-1/22/$31.00 ©2022 IEEE

[1] We will use the term *label* for such ground-truth and call a dataset containing ground truth for a labelled dataset. Course-grained labels will simply separate between ’malicious’ and ’benign’ while fine-grained will also e.g. be able to seperate on stages or techniques used.

[2] In certain cases it is natural to separate between ‘emulation’ and ‘simulation’, however we will use the term ‘emulation’ for both to simplify presentation.

in our view that the skills needed for – and challenges faced when – generating realistic emulations are separate to those faced when providing high-quality labels. Possible advantages of such decoupling are:

a) Separation of concerns & modularity: there are several tools and approaches for emulating attacks and APTs, which can be directly used.
b) Continous development of new datasets. As new attack emulations becomes available, these can be directly exploited to continously generate new datasets.
c) Reuse of labelling approach and tool across emulation tools.

The advantages of a tight integration of emulation and labelling is that the contextual information from controlling the emulation can be exploited by the labelling. As a result, approaches where emulation and labelling are integrated have provided labels that are more finely grained and of higher quality than more ad-hoc methods for labelling datasets [8], [28]. We are therefore not arguing for full separation, but assume that certain contextual information is provided from the emulation.

Building on the experience of state-of-the-art labelling approaches [8], [28], we develop an approach, with a supporting proof-of-concept prototype called LADEMU, for the Mitre CALDERA[3] emulation platform [2].

CALDERA builds on Mitre ATT&CK[4], and contains emulations of APT behaviour based on ATT&CK and ATT&CK techniques enabling labelling of the different attack steps/phases at technique level (R4), which can be directly used by our work. CALDERA is also actively developed meaning we can directly benefit from new techniques and emulations, possibly based on APTs. Finally, as CALDERA is developed by others, we can make a stronger claim for independence and separation of concerns.

Concretely, the contributions of this papers are:

1) a modular approach for generating high-quality labelled datasets at ATT&CK technique level, which is only weakly coupled with emulation platforms;
2) a proof-of-concept implementation of the approach with the LADEMU tool for CALDERA which can capture and label both host (Sysmon) and network (PCAP) logs;
3) experimental evaluation through the existing CALDERA APT29 emulation plan;
4) generation of a new labelled dataset, containing host and network data labelled at ATT&CK technique level from the APT29 emulation.

The paper is structured as follows:

- Section II describes necessary background on datasets, emulation and labelling approaches.
- Section III provides a high-level and illustrative overview of our approach and the LADEMU tool.
- Section IV provides a more detailed description of LADEMU, including implementation details.
- Section V evaluates the approach through generation of a labelled dataset from an existing CALDERA emulation plan of APT29.
- Section VI concludes the paper and discuss future work.

II. Background

A. Datasets

We will not provide a detailed survey of available datasets for this domain – for that we refer to e.g. [17] – and will instead focus on some more generic observations. The challenges faced when developing suitable datasets, and their scarcity, has already been discussed [9], [20], [30], [36]. A significant number of published works have used old and outdated datasets as benchmarks [8]. Two such outdated datasets that are heavily used are DARPA 99 [13] and KDD Cup 99 [24]. Even though both are from the last century, a study from a few years ago showed that 85% of published papers on anomaly-based IDS systems used the DARPA dataset [23], [36]. Most of the available datasets only contain network-based logs, and it has been argued that the few host-based datasets (e.g. [12]) was insufficient and incomplete [22]. However, newer datasets have been published since then, e.g. OpTC/DARPA [5] [3] and OTRF[6], which we have not analysed.

B. Emulation and cyber exercises

By emulating both malicious and benign behaviour, and therefore being in full control of both the red and blue team, one can capture logs and use the information from the emulation to label the logs. This is used in both [8] and [28].

A variant of this is to emulate only malicious behaviour and combine it captured with logs from real systems and networks, where the latter is either assumed to be benign or unknown/background traffic.

CALDERA is an adversary emulation platform. However, it has been developed to automate adversary behaviour for red-team exercises and not to generate datasets. Thus, it does not contain capabilities for capturing and labelling logs. CALDERA is tighly integrated with Mitre ATT&CK, which is a curated knowledge base that works as a common taxonomy to categorise and model adversary behaviour and attacks. These are structured into tactics and techniques. *Tactics* represent the ”why” of an attack, denoting short-term, tactical adversary goals during an attack, while *techniques* represent the ”how” or ”what” of an attack, describing how adversaries achieve their tactical goals. ATT&CK provides a mapping from APTs to the techniques they have been observed to use, from which emulation (EMU) plans of the APT is generated and then implemented in CALDERA. The EMU plans contain step-by-step procedures that defenders can run from start to finish or as individual tests to test their system [39]. At the time of writing, CALDERA contains six EMU plans, with more plans expected in the future [6], [40]. CALDERA focuses solely on post-compromise activities to test the target network

[3] https://caldera.mitre.org/
[4] https://attack.mitre.org/
[5] https://github.com/FiveDirections/OpTC-data
[6] https://github.com/OTRF/Security-Datasets

Dette er en postprint-versjon/This is a postprint version.
DOI til publisert versjon/DOI to published version: 10.1109/BigData55660.2022.10020549

or system for weaknesses and vulnerabilities [2], but can be extended to e.g. simulate human behaviour or adding encryption to network traffic [11]. It uses an (AI) planning-based approach when deciding which actions to take during red team exercises and emulations [2]. A plan is developed based on the current information in its knowledge base, which is updated as the emulation progresses [11]. Note that CALDERA has recently been extended with *Micro Emulation Plans*[7] to arrange for easier automation of compound behaviours and to bring adversary emulation to a broader audience. This extension has been published after we completed this work presented here, and has thus not been included in our work.

Other relevant emulation platforms include Mnemnomic's *Adversary Emulation Planner* (AEP)[8], which is also closely alligned with ATT&CK, and Splunk's *Attack Range*[9]. The intention behind Splunk's Attack Range is closely aligned with ours in that the goal is to generate synthetic log data.

A *cyber range* offers a training space which can emulate security incidents, in which red teams can train and practice in the cyber domain. While cyber ranges are strictly not emulators, they provide an excellent source of data to train new data-driven capabilities – as seen in e.g. [26]. Examples of cyber ranges are: CCDCOE's cyber range [10] (NATO/Estonia); AIT's cyber range[11] (Austria), CRATE [21] (Sweden); and NCR[12] (Norway). The aforementioned work [26] used logs from the 2019 Locked Shields cyber exercise, hosted by CCDCOE, to label a subset of the generated logs in order to train a ML model for detecting command & control traffic.

C. Labelling log data

The advantage of a highly integrated approach of emulation and labelling is that information from the emulation can be exploited to achieve high-quality labels. One example of this is [8], where Docker containers are used to separate malicious and benign behaviour which is directly exploited for labelling. However, this approach is unlikely to be able to separate different stages of an attack as each step is likely to depend on former stages of the attack, meaning there is no natural containerisation of temporal aspects. Furthermore, the work in [8] was only used for network logs.

Another approach, used in [28], is to use a timing intervals from the attack to label logs. This is rather easily accomplished if we know the timing from each attack step, but just alone it is prone to mislabelling, as benign actions occurring within the attack interval will be incorrectly labelled as malicious [20], [28], and is therefore combined with additional information of the attack in [28].

The advantage of more ad-hoc labelling methods is that they can be applied to existing log data. One example is [26], which used information from the cyber exercise to post-hoc label the data. Another example is found in [18], where all network data coming from the hosts containing malware were labelled as malicious and real traffic captured from the network was not. The problem of these more rudimentary labelling approaches is that they will cause a large degree of mislabelling – i.e. a host with malware may also contain benign traffic that will be mislabelled. Such mislabelling will consequently have a negative impact on data-driven capabilities developed based on them.

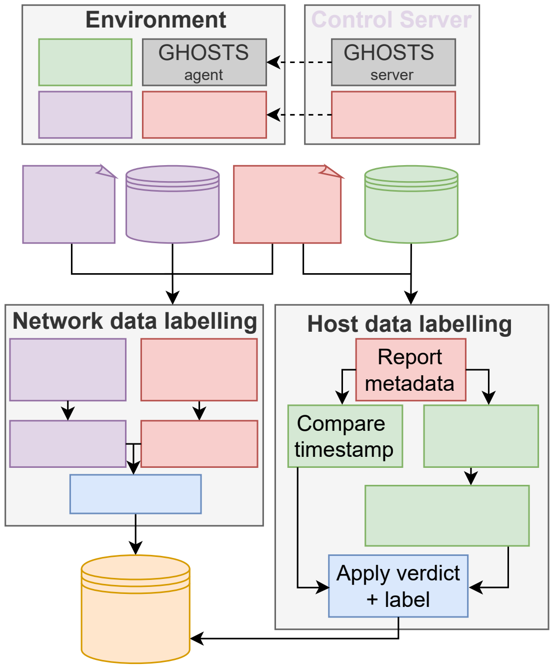

Fig. 1. LADEMU: high-level approach.

Finally, there are several generative approaches, either to label data or to avoid labelling at all by using e.g. *Generative Adversarial Networks* (GANs), [44]. Snorkel [33] is an example of a generative labellling approach. It is a weakly supervised ML method, where *weak*[13] labels are generated on a dataset based on heuristic which a domain expert has programmed. We have previously experienced with such method with mixed results for network logs [15]. Another generative method is Splunk's *Synthetic Adversarial Log Objects* (SALO) [27]. We consider such generative approaches to be beyond the scope of this paper and will not be addressed further.

III. Overall approach

There is a balance to be struck between a labelling process that is highly integrated with the emulation engine and a fully decoupled approach. It if is too tightly coupled then it becomes hard to separate emulation and labelling and the advantages of the decoupling, as explained above, becomes hard to realise; on the other hand, a too loose integration will not benefit from the knowledge gained when running the emulation, resulting in an ad-hoc labelling approach with all the challenges of mislabelling discussed above.

[7] https://ctid.mitre-engenuity.org/our-work/micro-emulation-plans/

[8] https://github.com/mnemonic-no/aep.

[9] https://github.com/splunk/attack_range/.

[10] https://ccdcoe.org/.

[11] https://cyberrange.at/.

[12] https://www.ntnu.no/ncr/.

[13] A weak label can be seen as a label of lower quality.

Figure 1 gives a high-level overview of our approach – both in terms of flow and architecture – which we believe is at a suitably level of modularity to benefit from the decoupling without loosing the benefits of the information from the emulation.

The top left of the figure shows the *environment* where the emulation will play out the attack/campaign. It consists of a network of connected hosts and/or virtual machines (VMs). CALDERA is used to emulate the attack, which only support post-compromise emulation. We can therefore assume that one or more of the hosts in the environment are compromised. This is achieved by placing a CALDERA *agent* on the hosts that are assumed to be infected. This will typically be a single host that will act as the initial attack vector for the emulation. In order to include benign behaviour we have additionally used the GHOSTS framework[14] [43], developed by the Software Engineering Institute at Carnegie Mellon University, in the environment. A GHOSTS agent is included in selected hosts to mimic benign behaviour. Both CALDERA and GHOSTS are controlled by external servers, which are hosted in a separate *control server*.

CALDERA does not perform logging of the behaviour which is required in order to generate the datasets. We therefore enrich the environment with two logging capabilities:

- To capture host logs, we used System Monitor (Sysmon) – a highly configurable tool from the Windows Sysinternals Suite that monitors and logs system activities. Sysmon provides detailed information regarding process creation/termination, driver and library loads, network connections, file creations, registry changes, process injections and more.
- To capture network logs, we used tcpdump[15].

During emulation, CALDERA is responsible for executing attacks and generating metadata while GHOSTS generate bening behaviour. We log these events in host logs and capture the network traffic. The emulation process produces four artefacts that we later use in our labelling:

- network logs in the form of PCAPs;
- host logs in the form of Sysmon entries;
- the CALDERA report, containing metadata about the performed emulation;
- a network configuration file, containing details of how the network is set up, including which IPs CALDERA and GHOSTS are hosted on.

We illustrate the labelling process by a truncated example of one attack entry in the CALDERA report, each individual ATT&CK ability executed by CALDERA is represented as an individual entry in the report:

```
{
 "command": "aXBjb25maWcgL2FsbA==",
 "delegated_timestamp": "2022-09-29T17:42:44Z",
 "finished_timestamp": "2022-09-29T17:43:27Z",
 "platform": "windows",
 "executor": "cmd",
 "pid": 4160,
 "agent_metadata": {
   "username": "WIN10\vagrant",
   "location": "...\sandcat.go-windows.exe",
   "pid": 5476,
   "ppid": 1732,
   "privilege": "Elevated",
   "host": "win10",
   "contact": "HTTP",
 },
 "attack_metadata": {
   "tactic": "discovery",
    "technique_name": "System Network ...",
    "technique_id": "T1016"
 }
}
```

To label the logs, LADEMU uses the following entries from the report: the timestamps, tactic, technique name, technique id and process ID.

Label categories. LADEMU uses four label categories: *benign*, *ATT&CK technique ID*, *C&C* and *background*. A *benign* label implies that a data point is normal or harmless. A label with an *ATT&CK technique ID* indicates a known and defined attack vector to initiate malicious activity while a *C&C* is a malicious data point of unknown origin. Finally, a *background* label indicates uncertainty and/or lack of indicators to imply whether they belong to any of the previous labels.

The process IDs (PIDs) is only used for the host logs, where a list of malicious PIDs is generated from the CALDERA report. This list will contain all PIDS in the report, and if a process ID in the logs is seen to interact with any PID in this list it is also added to the list. This continues in an iterative manner. By matching the process ID's observed in the host logs with the metadata extracted from CALDERA LADEMU is able to find the malicious log events. This is done by matching the PIDs from the CALDERA report with fields containing the PID and the parent PID within each event in the Sysmon logs generated during the attack. In order for these events to be labelled as *malicious* with the related ATT&CK technique and tactic, the timestamp of the event has to be within the time period of the attack entry related to the process ID. Once LADEMU has processed all malicious PIDs the remaining events are labelled as *benign*. LADEMU does not differentiate between OS background activity and events generated by GHOST, all are labelled as benign. However, malicious events occurring outside the defined time period will be labelled as *background* to indicate uncertainty.

We illustrate the Sysmon logs by a truncated example of two log entries related to the previously illustrate CALDERA attack entry. The fields "isMalicious" and "verdict" is added to the logs by LADEMU:

```
@metadata: { ... }
@timestamp: 2022-09-29 17:43:01.600 ...}
isMalicious: true
message: Process Create:
```

[14] https://github.com/cmu-sei/GHOSTS

[15] https://www.tcpdump.org/

```
  UtcTime: 2022-09-29 17:43:01.600
  ProcessId: 4160
  CommandLine: cmd.exe /C ipconfig /all
  CurrentDirectory: C:\Users\vagrant\
  User: WIN10\vagrant
  ...
  ParentProcessId: 5476
  ParentImage: C:\Users\Public\sandcat.go-windows.exe
  ParentCommandLine: "...\sandcat.go-windows.exe"
      -server http://192.168.56.104:8888 -group red
process: { ... }
verdict: Malicious discovery -
     System Network Configuration Discovery - T1016
winlog: { ...   }
```

PID 4160 is initially extracted from the CALDERA report together with the attack metadata, start and finish timestamp. The first example shows that PID 4160 is spawning a new command-line window in order to run the ”ipconfig /all” command. This new process has the PID 4388 and we can see the parent PID 4160 in the logs. Since these two PIDS are related, both of them are labelled with the attack metadata from PID 4160:

```
...
message: Process Create:
  ...
  ProcessId: 4388
  CommandLine: ipconfig /all
  ParentProcessId: 4160
  ParentImage: C:\Windows\System32\cmd.exe
  ParentCommandLine: cmd.exe /C ipconfig /all
verdict: Malicious discovery -
   System Network Configuration Discovery - T1016
...
```

For network data, tcpdump will capture traffic that is traversing the network, both between the hosts/VMs and externally. Timestamps, IP addresses and attack technique IDs are used when labelling these logs. As a first step, all traffic between the infected host and the attacker is labelled as C&C, due to the relation between CALDERA server and agent. We base this on the machine's respective IP addresses. Some of these network packets have timestamps that match the CALDERA attacks, for which we label them with the corresponding attack technique. Traffic between other IP addresses in the environment are labelled as background, and external traffic from the compromised host are labelled as benign as this will have been generated by GHOSTS.

To illustrate, consider the timestamp and IP addresses for the following four packets:

```
29-09-2022 17:42:51, 192.168.56.104 -> 192.168.56.107
29-09-2022 17:44:01, 192.168.56.107 -> 192.168.56.104
29-09-2022 17:45:15, 192.168.56.102 -> 192.168.56.107
29-09-2022 17:45:58, 192.168.56.1 -> 239.255.255.250
```

The first two packets are both labelled as *C&C*. As the first packet is within the time frame of the CALDERA report it is relabelled with the technique specified in the report (`T1016`). The third packet is external communication to the infected host. It is thus related to GHOSTS and labelled as *benign* while the last packet is not related to GHOSTS nor an attack/CALDERA and is thus labelled as *background*.

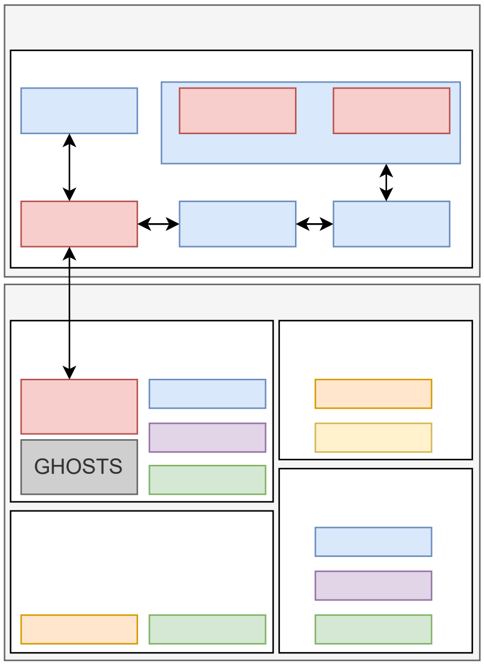

Fig. 2. LADEMU: implementation details.

This process generates a labelled dataset, consisting of both host and network logs, as illustrated at the bottom of figure 1.

IV. LADEMU: Implementation

To build a enterprise environment consisting of hosts and network configurations, we used a pre-arranged enterprise setup by the DetectionLab project [16]. This setup was extended with a separate Ubuntu VM hosting the CALDERA server using the orchestration framework called Vagrant[17]. DetectionLab provides the following four pre-configured hosts:

1) **Domain Controller** (DC): Domain controller on a Windows 2016 server
2) **Windows Event Forwarder** (WEF): A Windows 2016 server that manages Windows Event Collection
3) **Endpoint Host**: A Windows 10 host to simulate a non-server endpoint
4) **Logger**: An Ubuntu 16.04 host running Splunk for collecting logs.

Figure 2 further details figure 1 with implementation details of the enterprise environment.[18] The environment contains three networks, each with different purposes: a NAT network,

[16] https://detectionlab.network/

[17] https://www.vagrantup.com

[18] Our LADEMU implementation uses the following software versions:

- CALDERA version 4.0.0-alpha for adversary emulation;
- DetectionLab: https://github.com/clong/DetectionLab commit: 4318620a4dd279665fd11ae5b88217385047fe9d;
- Winlogbeat version 7.15.0;
- Elastic Common Schema (ECS) version 1.11.0
- GHOSTS version 6.0.0;
- LinuxLite version 5.6 on the attacker VM;
- Windows 10 version 19H2 on the victim VM;
- Firefox version 98.0.1 on Linux and Windows;
- Python 3.10.7

Dette er en postprint-versjon/This is a postprint version.
DOI til publisert versjon/DOI to published version: 10.1109/BigData55660.2022.10020549

a host-only network and an internal network. Only IPv4 was used as IPv6 is not currently supported by the emulation plan. The NAT network is used by DetectionLab for outbound communication over the internet for the *endpoint host*; the host-only network is used for inter-communication between hosts/VMs in the environment; and finally, the internal network enables isolation of GHOSTS API traffic from the traffic capture. Both the CALDERA agent and server are deployed onto the host-only network, to allow capture of C&C communication and required arrangement to get a foothold into the enterprise network.

One reason for this network segregation is that enterprise services typically do not permit outbound communication with public IP addresses, and the separation thus provides a more realistic environment. Another reason is the benefit of isolating management traffic irrelevant to our emulation, enabling finer control when capturing network packets in an emulation.

To capture network data, tcpdump is installed on one VM, and used to capture all traffic in the network. Traffic from the internal network used by GHOST is not captured, and the capture from the other two networks are combined into a large PCAP file which is subsequently labellled as described below. To support the labelling process, all relevant IP addresses of th environment are captured in a network configuration file as seen in figure 1.

Sysmon is used to log host behaviour and, while strictly not necessary, a Sysmon configuration is recommended as it helps to tune and filter the logs generated before processing them. We have used best practices by Microsoft [37] and a well-known Sysmon configuration file developed by Olaf Hartong[19] for our work. Once new hosts are added to the domain they are also added to the Windows workstation group, which in return set their audition configurations. This process is done automatically when adding new hosts to using Vagrant.

Windows logs are by default in the EVTX format, with an underlying XML structure. This format is only used by Microsoft and not supported by other operating systems. Ideally, the logs collected should use a common representation format to account for information sharing amongst different components, which will ensure that individual components (e.g., logging, alert-generation, analysis) from different vendors can work together. In order to both collect the logs and convert these into a standardised format, we used Winlogbeat[20] into the environment. Winlogbeat is commonly used as a "data shipper" that sends collected logs from clients to a centralised log management solution. Event logs are read using Windows APIs, filtered based on user-configured criteria and then sent to the configured outputs. However, in our case, Winlogbeat was used strictly for collecting and converting the host based logs directly on the client. Splunk was used to visualise and search the logs during development.

The following steps were applied to each host in order to prepare the domain:

- install Sysmon and applying the configuration file of Olaf Hartong;
- install and configure Winlogbeat to collect the EVTX log files from Sysmon and convert them into a JSON format;
- deactivate Windows Defender, optionally configure it to notify only;
- install and configure the GHOSTS client agent for simulating benign user activities;
- install and configure tcpdump and libpcap

Once emulation has completed for a given experiment, the resulting Sysmon logs are extracted from Winlogbeat in JSON format, the PCAP files are extract from the two networks, the network configuration file is extracted and the CALDERA report is extracted from the CALDERA server.

A. Labelling network traffic

The generated PCAP file, the CALDERA report and IP addresses formed the basis for the labelling module of LADEMU, which we call LADEMU$_N$[21]. LADEMU$_N$ is implemented in Python and uses the Editcap[22] tool for Wireshark to add the labels as comments in the PCAP file – as well as exporting the dataset in both JSON and CSV format.

The tool first iterates over the PCAP-file and separates the packets into three lists: background; attack; and benign. Both the CALDERA report and the network configuration file contains the IP addresses of the VM with the CALDERA agent and CALDERA server. Any traffic between these are considered part of an attack and initially labelled as *C&C*. These will in a later phase be labelled by the appropriate Mitre ATT&CK technique when possible and are added to the a list called *attack*, which is used in this phase. The IP addresses which GHOSTS communicate to externally are considered as *benign* packets while the remaining packets are considered to be *background*.

The CALDERA report contains timestamps indicating every start and end time of the current technique [11]. When iterating the attack list, LADEMU$_N$ checks if the packet's timestamp falls within this range, and if so, relabel the packet with relevant technique ID from the CALDERA report, and updates the attack list with this refined entry. Packets that do not fall into any of the time ranges of the CALDERA report will keep the *C&C* label.

B. Labelling host behaviour

The LADEMU module for host-based labelling is written in C# and called LADEMU$_H$[23]. It uses both the JSON-converted Sysmon logs generated by Winlogbeat and the CALDERA report.

LADEMU$_H$ first iterates the CALDERA report to find the PIDs of the events related to CALDERA attacks. It reads each attack step (conducted experiment/emulation) as a separate

[19] https://github.com/olafhartong/sysmon-modular

[20] https://www.elastic.co/beats/winlogbeat

[21] A more detailed explanation of the experimental setup of LADEMU$_N$ can be found in [19].

[22] https://www.wireshark.org/docs/man-pages/editcap.html

[23] A more detailed explanation of the experimental setup of LADEMU$_H$ can be found in [25].

Dette er en postprint-versjon/This is a postprint version.
DOI til publisert versjon/DOI to published version: 10.1109/BigData55660.2022.10020549

object, and extracts the agent PID (the first child process for each operation), the fields containing values for the ATT&CK tactic and technique related to the specific ATT&CK ability, and timestamps for each step. These values are placed in a temporary list called *maliciousPID*.

For each PID in the list, LADEMU$_H$ then iterates through the JSON-file again, searching for any relations with this PID. Relations between processes include: interactions, child/parent/target relations, or ProcessAccess to name a few. Any new process IDs found are linked to the initial malicious process and inherits the ATT&CK technique, ATT&CK technique tactic and the timestamps, and are added to the *maliciousPID* list. This process repeats itself until there are no new malicious process IDs observed. By utilising this technique, LADEMU$_H$ is able to generate a process tree with all processes observed in relation to the malicious operation.

LADEMU$_H$ then uses this list to find any matching PIDs in the events within the JSON-file. When iterating the log files, LADEMU$_H$ looks for any process or parent process ID (PPID) corresponding to the PIDs in the malicious PID list. If matching PIDs are observed, the event is considered malicious, and LADEMU$_H$ applies the label according to the ATT&CK technique from the CALDERA report. However, a PID leveraged by a malicious operation earlier is not necessary conducting malicious activity at a later time. In order to avoid mislabelling these events, a function which checks if the malicious event in the JSON-file is within the time frame of the malicious operation was implemented, down to milliseconds. Each operation in the CALDERA log has a delegated and finished time stamp. This was leveraged in order to check if the detected malicious event is within these two timestamps. The argument for this time-based approach is that each operation occurs within the delegated and finished time stamp.[24]

If an event is considered malicious, but is outside of the expected operation time, the label "*background* +" technique id and tactic is applied in addition to the *"isMalicious: False"* verdict. The *"background"* label indicates uncertainty of the label, and that the event may need a manual review to determine its true nature. If the event is within the operation time, the label "*isMalicious: True*" and the "*verdict:*" + technique id and tactic labels are applied. This is then repeated for all attack steps of the CALDERA report, and each event containing a malicious PID is labelled accordingly. Once LADEMU$_H$ has processed the entire CALDERA report and labelled all malicious events, the remaining events are labelled as *benign*.

We consider there to be no benign actions performed by the CALDERA agent, since the agent is the initial access point of the attack. Therefore, any interactions in the environment performed by the agent are regarded as malicious in our case. An example of this could be the agent performing process injection by executing code in the address space of a separate process.

In addition to labelling techniques from ATT&CK, we further adjusted LADEMU$_H$ to label *C&C* traffic correctly. *Event ID 3* in Sysmon is Network Connection. We programmed our labelling tool to check if a malicious event has *Event ID 3* and occurs within an expected time frame. If so, the event is labelled as *C&C* traffic.

V. Experiments

APT29 is a Russian Foreign Intelligence Service (SVR)-affiliated threat group that has been active since at least 2008 [10]. The group has received allegations of substantial breaches targeting U.S. governments and organisations. APT29 reportedly compromised the Democratic National Committee in 2016 [38], attacked Covid-19 vaccine development labs in 2020 [31], and the group was publicly announced by US and British authorities to be the attackers behind the SolarWinds supply chain campaign [32], [41]. The group is known for their stealth and use of sophisticated techniques [5] [16]. Their characteristic sophistication has made the group an ideal object for emulation [14].

Mitre released the APT29 EMU plan[25] in early 2021 and is available as a plugin for CALDERA. The emulation plan was developed from publicly available sources, describing the motivations, objectives and attributed tactics, techniques and procedures mapped to Mitre ATT&CK. It consists of techniques chained into a logical order observed across previous APT29 campaigns. The attack scenarios include abilities related to: discovery, C&C, credential dumping, executing and defence evasion. The plan contains two distinct scenarios with 20 defined stages from 79 attack steps[26]. The first scenario has a "smash-and-grab" approach, starting with noisy techniques before proceeding to a quick espionage mission for collecting data and exfiltration. The second scenario has a more "low and slow" approach and involves compromising the initial target, establishing persistence, obtaining credentials, and then enumerating and compromising the entire domain more stealthily and slowly. The EMU plan is a plugin for CALDERA, which made for easy integration with the framework and ATT&CK in this work.

Using DetectionLab, we created an enterprise environment consisting of 5 VMs to experiment with the APT29 EMU plan. They were connected to distinct networks in order to separate the various activities carried out during the experiment. With the plan's final step shutting down the victim VM and bringing the experiment to an end, the operation lasted approximately one hour and produced $19,567$ network packets, while Sysmon generated $3,928$ events. Note that LADEMU$_N$ had to label $17,000$ network packets at a time, due to restrictions with Editcap. This resulted in multiple smaller and labelled logfiles (PCAP) which was merged using Mergecap [27].

For each attack step execution, the resulting CALDERA report showed it to be successful, failed or skipped. Most

[24] Recall that the finished time stamp is applied once the agent reports back the outcome of the operation.

[25] https://github.com/center-for-threat-informed-defense/adversary_emulation_library

[26] An attack step is defined as the execution of an ability

[27] https://www.wireshark.org/docs/man-pages/mergecap.html

Dette er en postprint-versjon/This is a postprint version.
DOI til publisert versjon/DOI to published version: 10.1109/BigData55660.2022.10020549

of these these attack steps executed successfully, however a large share had been skipped. A small quantity also failed. The CALDERA documentation notes that an attack step may be skipped if conditional data is missing from CALDERA's knowledge database e.g., failed to update or receive expected output from previous steps, or when insufficient facts were learned during previous executions. We have also observed cases where attack steps reported as successful by CALDERA actually failed. For instance, certain scripts for persistence establishment was found, but others were missing. We have not investigated this any further, but we have indications of the report containing conflicting verdicts.

A. Results

1) Network labelling: LADEMU labelled $19,161$ out of $19,567$ network packets, giving a total labelling percentage of 97.92%. Table I details the label distribution for the packets. Attack data accounts for 0.02% of the whole dataset, which is to be expected in environments with high activity. Among the unlabelled packets, we observe that they were related to ARP queries, ICMP traffic and other background activity that LADEMU failed to detect as background. Despite this, we consider the labelling coverage to be satisfactory with sufficient representation from different attack categories present in the emulation plan.

Further inspection shows that IP addresses of our labelled packets match our expectations. For instance, the benign packets had destination IPs that were external and did not belong to our enterprise environment. Moreover, the background packets did belong to processes tied to updates, clock synchronisation, DHCP and multicast DNS. We have indications of the benign, and background traffic being appropriately labelled as long as the attack do not propagate to other machines and network protocols tied to numerous network services remain static. (e.g. NTP, DHCP)

Most attack packets were labelled with the correct ATT&CK technique, however labelling conflicts can occur. For instance, we are aware of situations where CALDERA's C&C channel is periodically transmitting beacons during attack steps. These are labelled with the current ATT&CK technique, rather than its' original technique. Moreover, we have not ruled out that certain attack steps may be executed in sequence before the CALDERA server is given feedback, and may thus provide overlapping time ranges affecting labelling. We return to this in the section VI below.

2) Host labelling: For the hosts, $3,928$ security events were generated, where 971 were malicious (24.7%), 35 were background (1.2%) and $2,922$ were benign (74.1%). We note that the benign activity is limited, due to the emulation environment not being configured with Microsoft Office license keys and GHOSTS not performing as intended. GHOSTS was only able to successfully simulate a user browsing the web by automating web requests towards different domains. According to the GHOSTS documentation [43], it should be able to automate terminal commands and office document management. However, neither of these did function properly in our testbed and GHOSTS only executed the web browsing function. This caused the rate of malicious activity in the host logs to be much higher than on the network, where all the benign activity worked as intended.

Label category	**No. of packets**	**Percentage**
Attack	1765	9.02%
Benign	16 704	85.36%
Background	692	3.53%
Empty label	406	2.07%
Total no. labelled packets	**19 161**	**97.92%**
Total no. packets	**19 567**	**100%**

TABLE I
NUMBER OF PACKETS AND PERCENTAGE OF EACH CATEGORY.

Src IP	**Dst IP**	**Timestamp**	**Label**
192.168.56.104	192.168.56.107	29-09-2022 20:10:46	T1518
192.168.56.107	192.168.56.104	29-09-2022 20:10:47	T1518
192.168.56.104	192.168.56.107	29-09-2022 20:12:27	T1071.001
192.168.56.102	192.168.56.107	29-09-2022 20:12:55	Benign
192.168.56.104	192.168.56.107	29-09-2022 20:13:26	T1033

TABLE II
EXAMPLE NETWORK PACKETS WITH DIFFERENT LABELS.

A smaller 30 minute timeframe of the data looks credible, which might indicate that the rest of the results are credible. A considerable majority were labelled with their associated ATT&CK technique. The remaining events were labelled benign, and were related to the host sending DNS queries to enterprise services, which occurred even when the host was idle ahead of our timeframe. We also investigated our background labelled data, for which we identified that some events could not be traced back to a malicious PID or had blank PPID entries. Without this relation, our approach is unable to label correctly. Moreover, we discovered that some data were mislabelled due to a malicious process interfering with a benign process, and the process performing both benign and malicious activity after completion of the operation. The malicious activity were limited to execution of scripts shortly after completion. We outline how these limitations can be addressed in the future in section VI.

VI. CONCLUSION & FUTURE WORK

Building on CALDERA, we have developed and implemented a proof-of-concept modular approach for generated labelled APT datasets, capturing both host and network logs, and providing high-quality labels at the Mitre ATT&CK technique level. By labelling at this level, the dataset can be used to develop APT detection and hunting capabilities, including

Verdict	**Techn. Count**
Privilege-escalation - Process Injection - T1055	202
Execution - Command and Scripting Interpreter: PowerShell - T1059.001	67
Discovery - Process Discovery - T1057	63
Discovery - System Information Discovery - T1082	60
Credential-access - Credential Dumping - T1003	53
Discovery - Permission Groups Discovery - T1069	44
Remote System Discovery - T1018	41

TABLE III
TOP 7 LABELS APPLIED TO HOST DATASET.

for kill-chains and multi-step attacks, as the labels can differentiate the different stages of an attack. The developed tool, called LADEMU, has been applied on the existing APT29 emulation plan as means of evaluation.

The advantage of our approach is the partial decoupling of emulation and labelling – meaning we can directly benefit new emulation plans and techniques from CALDERA and, at the same time, exploit timing and process information provided by CALDERA to improve the label quality compared with more ad-hoc methods. We have not conducted direct experiments to evaluate if such additional information from emulation indeed provides higher quality labels; for this we rely on others [8], [28]. We do however observe that host logs, which had more information from CALDERA compared with network logs, provided better separation between the attack steps compared with network logs, which go some way of justifying this claim, at least anecdotally.

As LADEMU is a proof-of-concept, we have identified several limitations – some minor, some more substantial, some with respect to LADEMU itself and some related to integration with other tools. LADEMU was created with CALDERA in mind, meaning it needs to be adapted before used with other tools. It creates detailed labels by leveraging an emulation plan. This means that it is not directly transferable to areas without such plans (or something equivalent). All use still require manually labelling techniques at some point in the process. As discussed in V, CALDERA sometimes failed to execute certain steps in the plan, and the information provided by the CALDERA report was rather minimal or nonexistent. There were also cases where the report was incomplete, such as missing timestamps, and cases where the report indicated successful execution but nothing was captured by the log. As CALDERA is developed to test defences (and not generate datasets), such bugs in CALDERA may be difficult to find in its normal operation, and pointing to secondary use of our approach: as a debugging tool for emulators. For benign traffic, GHOST had minimal effect when generating benign host logs. It was predominantly generating web traffic, and had very limited impact on the host logs, which needs to be adressed in the future. It would also be interesting to explore if GHOSTS, or other frameworks for emulating benign behaviour, can provide an emulation report comparable to the CALDERA report, both to improve label quality and to generate more fine grained labels (e.g. the application executed).

Malicious processes observed outside the operation time frame were given *background* labels for host-based logs. In order to reduce the number of such labels, the labelling function can be further developed by calculating the difference between the time stamps. For example, malicious PIDs observed ten seconds after the operation time frame is more likely to be malicious than if observed ten minutes later. A more advanced and accurate approach would be to analyse the event ID of the last process seen in relation to the PID outside the operation time frame. For example, if the last operation was a Sysmon *Event ID 1* (Process Created), the following operation conducted by this newly created process can with high accuracy be considered malicious if observed outside the operation time frame. This would require an evaluation score to be applied to the various Sysmon events ahead of time.

For the network logs, a first step for improvement would be to use fine grained time steps to separate between attack steps. Longer-term we need to investigate which additional information from the simulation and experimental setup can improve the quality of the labels. Initially, we were planning to use the containerised approach used in the DetGen tool [8] to separate malicious and benign traffic. However, integrating this with CALDERA would require resources not available in the time frame available for this project and was therefore put on hold for now. This is however something to explore in the future. We have previously shown promising results using DetGen with a single malware (Mitre ATT&CK C2 tactic) [4] to capture label network traffic, and we are confident that this approach will be able to separate malicious (C&C) and benign traffic. However, it unlikely to provide any help in separation between different (malicious) techniques/steps in the emulation, as each step is likely to have temporal dependencies on previous steps, and each step cannot therefore not be executed in separate containers. Furthermore, it is unclear to which degree host-level labelling can benefit from such approach. Another approach is to explore the correlation of network logs with host logs describing process network connectivity.

LADEMU may have been too tailored to our experiments and could be further generalised and automated more when setting up and running on new environments. One type of attack we have not addressed yet is lateral movement where the CALDERA agent spreads to other hosts. Similarly, LADEMU could be adapted to be used within cyber exercises, where an interesting question is how required information now provided by the CALDERA report can be extracted in this context.

Finally, one of the motivation for a modular approach is to be able to reuse the approach (and LADEMU) across different emulation tools – given that they provide the necessary information from emulation. Such generality is not something we have addressed in this paper. A first step would be to include the newly developed *Micro Emulation Plan* extension. Another tool, which is also able to emulate APTs, is Mnemonic's *Adversary Emulation Planner*[28]. As CALDERA only emulates post-compromise steps, other tools may be able to also cover the initial steps of a kill-chain.

REFERENCES

[1] Adel Alshamrani, Sowmya Myneni, Ankur Chowdhary, and Dijiang Huang. A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. *IEEE Communications Surveys & Tutorials*, 21(2):1851–1877, 2019.

[2] Andy Applebaum, Doug Miller, Blake Strom, Chris Korban, and Ross Wolf. Intelligent, automated red team emulation. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, pages 363–373, 2016.

[3] Rody Arantes, Carl Weir, Henry Hannon, and Marisha Kulseng. Operationally transparent cyber (optc), 2021.

[28] https://github.com/mnemonic-no/aep

Dette er en postprint-versjon/This is a postprint version.
DOI til publisert versjon/DOI to published version: 10.1109/BigData55660.2022.10020549

[4] Markus Asprusten, Julie Gjerstad, Gudmund Grov, Espen Kjellstadli, Robert Flood, Henry Clausen, and David Aspinall. A containerised approach to labelled c&c traffic. In *Norsk IKT-konferanse for forskning og utdanning*, number 3, 2021.

[5] ATT&CK Evaluations. Apt29 enterprise evaluation 2019. https://attackevals.mitre-engenuity.org/enterprise/apt29, 2019. [Online; last updated June 14, 2021], [Last accessed: April 19, 2022].

[6] Jon Baker and Forrest Carver. Introducing the all-new adversary emulation plan library, September 2020. [Online; published September 10, 2020], [Accessed: April 5, 2022].

[7] Ping Chen, Lieven Desmet, and Christophe Huygens. A study on advanced persistent threats. In *IFIP International Conference on Communications and Multimedia Security*, pages 63–72. Springer, 2014.

[8] Henry Clausen, Robert Flood, and David Aspinall. Traffic generation using containerization for machine learning. *arXiv preprint arXiv:2011.06350*, 2020.

[9] Henry Clausen, Gudmund Grov, and David Aspinall. Cbam: A contextual model for network anomaly detection. *Computers*, 10(6):79, 2021.

[10] The Mitre Corporation. MITRE ATT&CK apt29. https://attack.mitre.org/groups/G0016/. Last accessed: May 9, 2022.

[11] The Mitre Corporation. Welcome to caldera's documentation! https://caldera.readthedocs.io/en/latest/index.html. Last accessed: May 10, 2022.

[12] Gideon Creech. *Developing a high-accuracy cross platform Host-Based Intrusion Detection System capable of reliably detecting zero-day attacks.* PhD thesis, University of New South Wales, Canberra, Australia, 2014.

[13] Robert K Cunningham, Richard P Lippmann, David J Fried, Simson L Garfinkel, Isaac Graf, Kris R Kendall, Seth E Webster, Dan Wyschogrod, and Marc A Zissman. Evaluating intrusion detection systems without attacking your friends: The 1998 darpa intrusion detection evaluation. Technical report, MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB, 1999.

[14] Frank Duff. Round 2 of ATT&CK Evaluations is Now Open. https://medium.com/mitre-attack/attack-evals-round-2-c3ea383ba55d, May 1, 2019. Accessed: April 7, 2022.

[15] Håkon Svee Eriksson, Gudmund Grov, Gil Christian Tinde, and Espen Hammer Kjellstadli. 'svak' merking av nettverkstrafikk ved hjelp av snorkel. Notat, FFI, 2021.

[16] F-Secure. The dukes: 7 years of russian cyberespionage. Technical report, F-Secure, 2015.

[17] Mohamed Amine Ferrag, Leandros Maglaras, Sotiris Moschoyiannis, and Helge Janicke. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, 50:102419, 2020.

[18] Sebastian Garcia, Martin Grill, Jan Stiborek, and Alejandro Zunino. An empirical comparison of botnet detection methods. *computers & security*, 45:100–123, 2014.

[19] Julie Lidahl Gjerstad. Generating labelled network datasets of APT with the MITRE CALDERA framework. Master's thesis, University of Oslo, 2022. http://urn.nb.no/URN:NBN:no-98161.

[20] Jorge Guerra, Carlos Catania, and Eduardo Veas. Datasets are not enough: Challenges in labeling network traffic. *arXiv preprint arXiv:2110.05977*, 2021.

[21] Tommy Gustafsson and Jonas Almroth. Cyber range automation overview with a case study of crate. In Mikael Asplund and Simin Nadjm-Tehrani, editors, *Secure IT Systems*, pages 192–209, Cham, 2021. Springer International Publishing.

[22] Waqas Haider, Gideon Creech, Yi Xie, and Jiankun Hu. Windows based data sets for evaluation of robustness of host based intrusion detection systems (ids) to zero-day and stealth attacks. *Future Internet*, 8(3):29, 2016.

[23] Hanan Hindy, Elike Hodo, Ethan Bayne, Amar Seeam, Robert Atkinson, and Xavier Bellekens. A taxonomy of malicious traffic for intrusion detection systems. In *2018 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA)*, pages 1–4. IEEE, 2018.

[24] Information and Irvine Computer Science, University of California. Kdd cup 1999 data. "http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html", October 1999. [Online; published October 28, 1999], [Accessed: July 05, 2022].

[25] Fikret Kadiric. APT attack emulation and data labeling. Master's thesis, University of Oslo, 2022. http://urn.nb.no/URN:NBN:no-98204.

[26] Nicolas Känzig, Roland Meier, Luca Gambazzi, Vincent Lenders, and Laurent Vanbever. Machine learning-based detection of c&c channels with a focus on the locked shields cyber defense exercise. In *2019 11th International Conference on Cyber Conflict (CyCon)*, volume 900, pages 1–19. IEEE, 2019.

[27] Marcus LaFerrera. Generating Known Unknowns through Known Knowns. Presented at FloCon 2022.

[28] Max Landauer, Florian Skopik, Markus Wurzenberger, Wolfgang Hotwagner, and Andreas Rauber. Have it your way: Generating customized log datasets with a model-driven simulation testbed. *IEEE Transactions on Reliability*, 70(1):402–415, 2020.

[29] Gabriel Maciá-Fernández, José Camacho, Roberto Magán-Carrión, Pedro García-Teodoro, and Roberto Therón. Ugr '16: A new dataset for the evaluation of cyclostationarity-based network idss. *Computers & Security*, 73:411–424, 2018.

[30] Sowmya Myneni, Ankur Chowdhary, Abdulhakim Sabur, Sailik Sengupta, Garima Agrawal, Dijiang Huang, and Myong Kang. Dapt 2020-constructing a benchmark dataset for advanced persistent threats. In *International Workshop on Deployable Machine Learning for Security Defense*, pages 138–163. Springer, 2020.

[31] The National Cyber Security Centre (NCSC). Advisory: Apt29 targets covid-19 vaccine development. Technical report, The National Cyber Security Centre (NCSC), 2020.

[32] The National Cyber Security Centre (NCSC). Uk and us call out russia for solarwinds compromise. https://www.ncsc.gov.uk/news/uk-and-us-call-out-russia-for-solarwinds-compromise, April 2021. [Online; published April 15, 2021], [Accessed: April 6, 2022].

[33] Alex Ratner, Stephen Bach, Paroma Varma, Chris Ré, and members of Hazy Research. Weak supervision: A new programming paradigm for machine learning. Accessed May 11, 2022.

[34] Markus Ring, Sarah Wunderlich, Deniz Scheuring, Dieter Landes, and Andreas Hotho. A survey of network-based intrusion detection data sets. *Computers & Security*, 86:147–167, 2019.

[35] Florian Skopik, Giuseppe Settanni, Roman Fiedler, and Ivo Friedberg. Semi-synthetic data set generation for security software evaluation. In *2014 Twelfth Annual International Conference on Privacy, Security and Trust*, pages 156–163. IEEE, 2014.

[36] Branka Stojanović, Katharina Hofer-Schmitz, and Ulrike Kleb. Apt datasets and attack modeling for automated detection methods: A review. *Computers & Security*, 92:101734, 2020.

[37] Blake Strom. Audit Policy Recommendations. https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/audit-policy-recommendations, July 2021. Accessed: August 09, 2022.

[38] Editorial team. Crowdstrike's work with the democratic national committee: Setting the record straight. https://www.crowdstrike.com/blog/bears-midst-intrusion-democratic-national-committee/, June 2020. [Online; published June 5, 2020], [Accessed: April 6, 2022].

[39] The Center for Threat-Informed Defense. Adversary emulation library — github. https://github.com/center-for-threat-informed-defense/adversary_emulation_library, 2021. [Online; Last accessed 8-May-2022].

[40] The MITRE Corporation. Emu plugin — github. https://github.com/mitre/emu, 2022. [Online; Last accessed 8-May-2022].

[41] Cybersecurity The National Security Agency (NSA), Infrastructure Security Agency (CISA), and Federal Bureau of Investigation (FBI). Russian svr targets u.s. and allied networks. "https://media.defense.gov/2021/Apr/15/2002621240/-1/-1/0/CSA_SVR_TARGETS_US_ALLIES_UOO13234021.PDF", April 2021. [Online; published April, 2021], [Accessed: April 6, 2022].

[42] Ciza Thomas, Vishwas Sharma, and N Balakrishnan. Usefulness of darpa dataset for intrusion detection system evaluation. In *Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security 2008*, volume 6973, pages 164–171. SPIE, 2008.

[43] Dustin D Updyke, Geoffrey B Dobson, Thomas G Podnar, Luke J Osterritter, Benjamin L Earl, and Adam D Cerini. Ghosts in the machine: A framework for cyber-warfare exercise npc simulation. Technical report, CARNEGIE-MELLON UNIV PITTSBURGH PA, 2018.

[44] Chika Yinka-Banjo and Ogban-Asuquo Ugot. A review of generative adversarial networks and its application in cybersecurity. *Artificial Intelligence Review*, 53(3):1721–1736, 2020.