# KAIROS: Practical Intrusion Detection and Investigation using Whole-system Provenance (supplementary material)

Zijun Cheng[1,2], Qiujian Lv[2], Jinyuan Liang[3], Yan Wang[2], Degang Sun[2], Thomas Pasquier[3], and Xueyuan Han[4]

[1]School of Cyber Security, University of Chinese Academy of Sciences
[2]Institute of Information Engineering, Chinese Academy of Sciences
[3]University of British Columbia
[4]Wake Forest University

## 1. Introduction

This document contain materials supplementary to our paper published in the $43^{rd}$ IEEE Symposium on Security and Privacy (S&P'24) [1]. In Section 2 of this document, we report attack descriptions based on the ground truth information provided by the Defense Advanced Research Projects Agency (DARPA) [2–4] and the associated attack summary graphs generated by KAIROS. In Section 3, we report representative false positive samples.

## 2. Attack Reconstruction

**E3-THEIA (Fig. 1).** The attacker leverages a `Firefox` vulnerability to establish a foothold on a victim machine, which enables the attacker to write a malicious payload called `clean` to `/home/admin/` on disk. The attacker then executes the payload with escalated privileges. This new attack process (with root privileges) communicates with the attacker's command-and-control (C&C) server at `161.116.88.72` to download and execute another malicious payload called `profile`, again with root privileges. `profile`, in turn, fetches a third payload called `xdev` from the C&C server and stores the payload in `/var/log/`. `profile` and `xdev` lurk in the victim host to prepare for subsequent attacks. A few days later, the attacker uses `profile` to inject malicious code in the `mail` process and executes `mail` with root privileges. `mail` then performs port scans of all known hosts on the victim's network.

**E3-CADETS (Fig. 2).** The attacker (`81.49.200.166`) connects to a vulnerable `Nginx` server and obtains a shell. Through the shell, the attacker successfully downloads a malicious payload to `/tmp/vUgefal` and executes the payload with root privileges. The elevated process `vUgefal` attempts to move laterally to `154.145.113.18` and `61.167.39.128`. However, only the attempt at infecting `61.167.39.128` is successful. `vUgefal` further plans to inject malicious payload to the `sshd` process. To do so, the attacker downloads the payload to `/var/log/devc`, but the attempted process injection fails.

**E3-ClearScope (Fig. 3).** The attacker exploits a vulnerability in `Firefox` to gain control of a victim host by browsing www.mit.gov.jo. The attacker then loads the payload `shared_files` into the victim's filesystem at `/data/data/org.mozilla.fennec_firefox_dev` and runs it with root privileges. The attacker's subsequent attempts to (1) load additional attack modules and (2) inject the attack payload into a process fail. Note that we were unable to find any `execute`- or `fork`-related behaviors in the data. It is unclear whether ClearScope records them in E3. Consequently, malicious activity that leverages such behaviors is not invisible to KAIROS (since it is not included in the data to begin with).

**E5-THEIA (Fig. 4).** The attacker exploits a vulnerability in `Firefox` by browsing www.nhra.com, connecting the victim host to C&C servers (`189.141.204.211` and `208.203.20.42`). The attacker then uses `/run/shm` to escalate privileges and gain access to a list of processes. The attacker finds the `sshd` process and injects malicious code into the process. The compromised `sshd` process writes to a file `/var/log/sshdlog` on the host's filesystem.

**E5-CADETS (Fig. 5).** The attacker sends a malformed `HTTP POST` from `128.55.12.167` to the victim host to trigger the `Nginx`'s backdoor. The `HTTP` request carrying a payload executes shell code, establishing a connection to a C&C server (`4.21.51.250`). The attacker obtains `username` and `hostname` information before unexpectedly losing control. The next day, the attacker sends the malformed `HTTP POST` again and continues the attack from the C&C server (`128.55.12.233`). The attacker exfiltrates the host's environment information such as `hostname`, `passwd`, and `username`, in the subsequent attack.

**E5-ClearScope (Fig. 6).** A user accidentally installs a malicious `appstarter` APK `de.belu.appstarter`, which loads an attack module called `busybox`. This module gives the attacker control from `77.138.117.150`. The attacker then installs the driver `msm_g711tlaw` into the victim host for privilege escalation. The attack exfiltrates `calllog.db`, `calendar.db`, and `mmssms.db` and takes a screenshot. Two days later, the attacker exploits `appstarter` again to try to connect to the C&C server

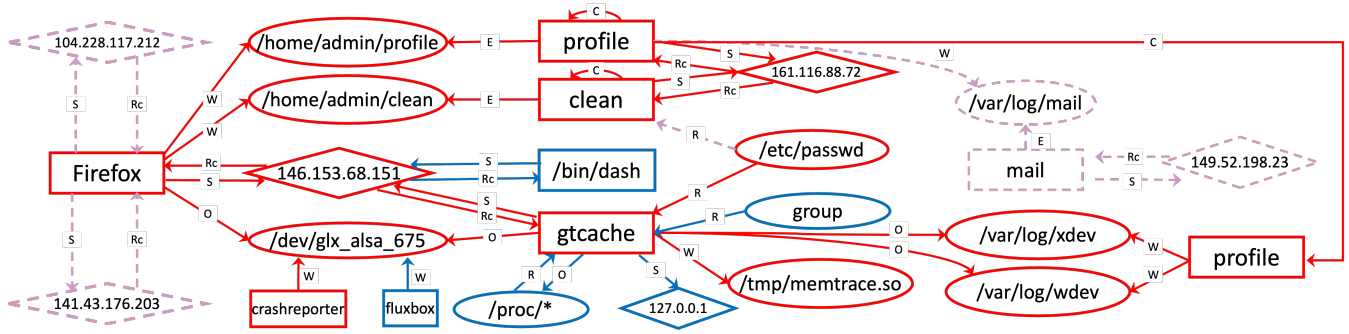


Figure 1. A provenance summary graph from DARPA E3-THEIA that describes attack activity in the motivating example, as automatically generated by KAIROS. Rectangles, ovals, and diamonds represent processes, files, and sockets, respectively. R=Read, W=Write, O=Open, S=Send, Rc=Receive, C=Clone, and E=Execute. We add colors and dashed elements for clarity to highlight the output that KAIROS generates. Solid nodes and edges are extracted by KAIROS from the original provenance graph to reconstruct the attack. Dashed pink nodes and edges are attack-related activities missed by KAIROS, according to the attack ground truth. Blue nodes and edges are activities not explicitly mentioned in the ground truth but included by KAIROS.

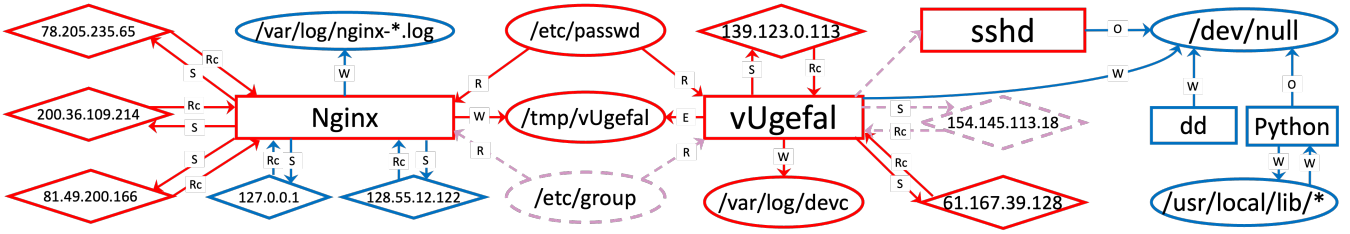


Figure 2. A summary graph that describes attack activity in DARPA's E3-CADETS dataset, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

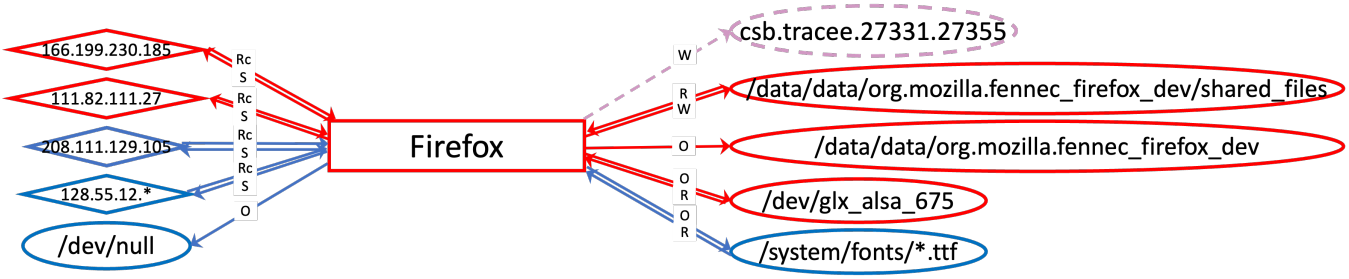


Figure 3. A summary graph that describes attack activity in DARPA's E3-ClearScope dataset, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

(`128.55.12.233`) but failed. The ground truth also describes some malicious activity of attack payloads called `lockwatch` and `mozilla`. Upon close inspection, we discover that the provenance data related to the malicious activity is corrupted. We remove the corrupted data and omit the malicious activity in Fig. 6.

**OpTC Day 1 (Fig. 7).** The attacker uses a C&C server (`132.197.158.98`) to connect to the victim host and executes a `powershell` script `runme.bat`. The attacker then injects the process `lsass` to collect the victim's credential and host information. The attacker also scans the network (e.g., using `ping` and `smb`) and uses `wmiprvse` to move laterally to a host at `142.20.57.147`. Eventually, the attacker moves to a host at `142.20.58.149` and runs more `powershell` scripts `*.ps1` to collect information.

**OpTC Day 2 (Fig. 8).** The attacker sends a phishing email with a payload (`payroll.docx`) to the victim. After executing the payload, the attacker uses the elevated `powershell` to performs network scan and `wmiprvse` to establish persistence. The attacker uploads `plink.exe` to forward port `3389` and uses RDP to connect to the host. Finally, the attacker collects and compresses host information into `*.zip` and runs `nx.exe` to exfiltrate the files.

**OpTC Day 3 (Fig. 9).** `Notepad++` downloads a malicious file `update.exe`, which runs an attack payload `cKfGW.exe`. The attacker controls `cKfGW.exe` through a malicious DNS server (`53.192.68.50`) to inject system processes and collect and modify host information.

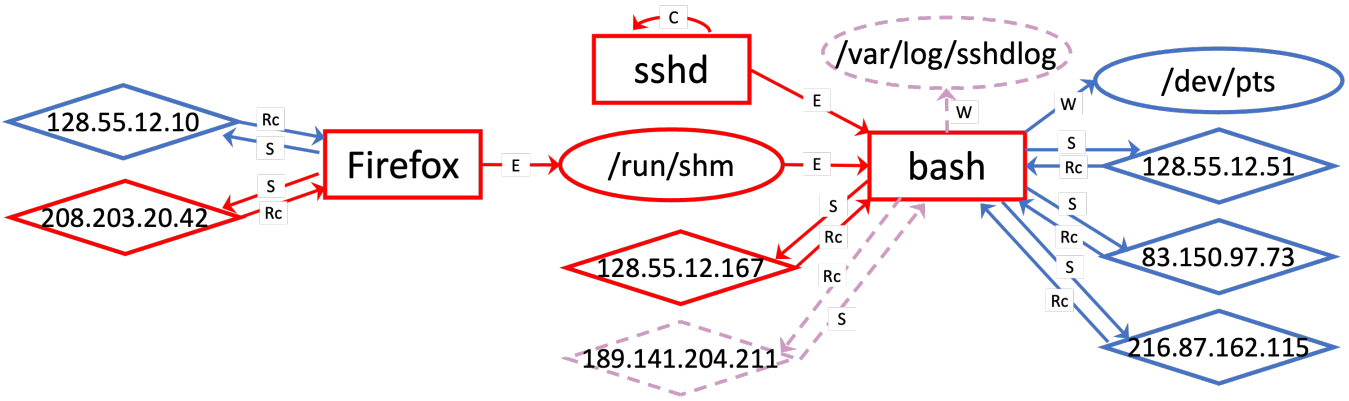


Figure 4. A summary graph that describes attack activity in DARPA's E5-THEIA dataset, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

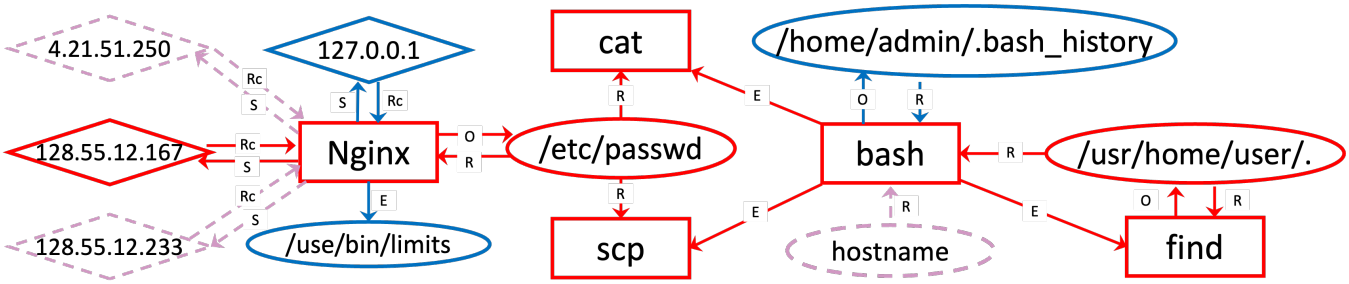


Figure 5. A summary graph that describes attack activity in DARPA's E5-CADETS dataset, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

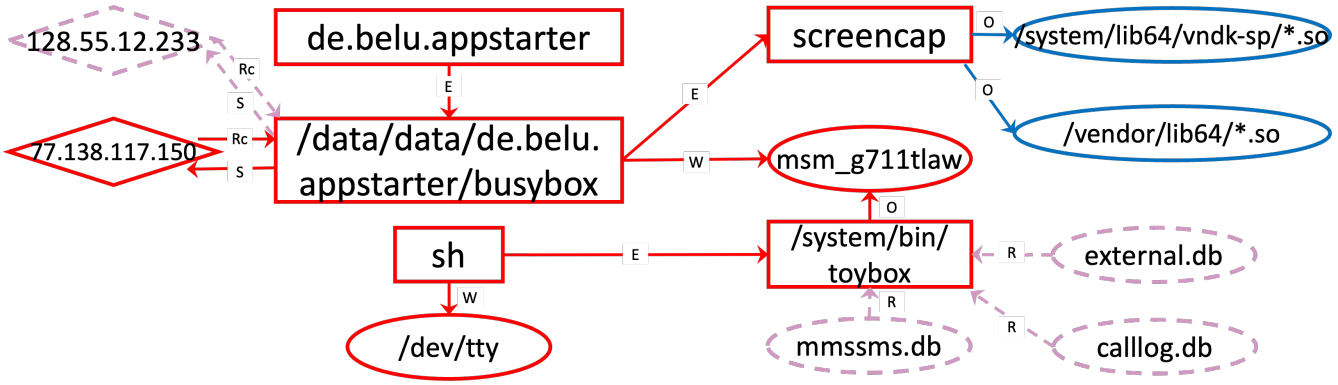


Figure 6. A summary graph that describes attack activity in DARPA's E5-ClearScope dataset, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

## 3. Benign Summary Graph Examples

**E3-CADETS (Fig. 10).** `wget` is a Linux utility used to download files from the Internet. It might connect to any external IP or URL. To determine whether `wget`'s behavior is related to attack activity, sysadmins might either check whether any connected IP is in a blocklist or confirm with the user the identities of the files they download. Any file not recognized by the user might be downloaded by the attacker through a C&C server.

**E3-ClearScope (Fig. 11).** `system_server` is an Android core process used to serve various Android system services, e.g., activity manager service (AMS). Sysadmins can manually identify whether a new legitimate system service triggered KAIROS' intrusion alert.

**E5-THEIA (Fig. 12).** `dbus-daemon` provides one-to-one communication between two applications. `upowerd` is a Linux middleware for power management, which provides its services through `dbus`. Sysadmins can check whether any application using `dbus-daemon` is suspicious.

**E5-CADETS (Fig. 13).** `scp` is used to securely copy files

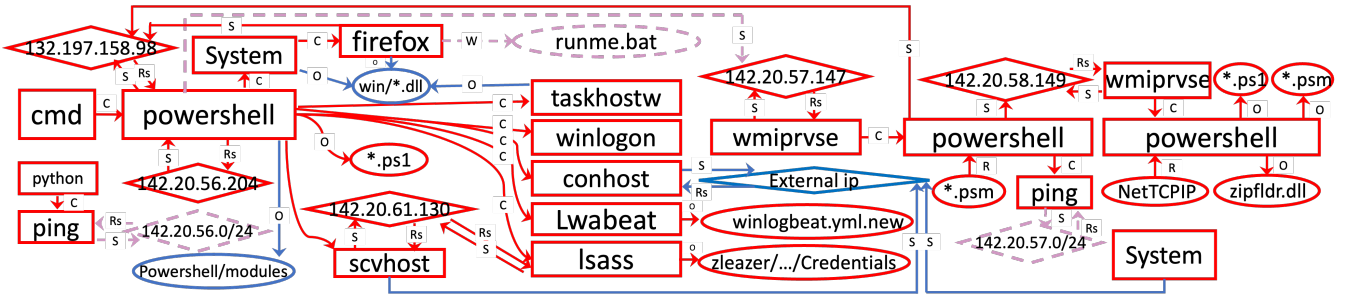


Figure 7. A summary graph that describes attack activity in DARPA's OpTC dataset in day 1, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

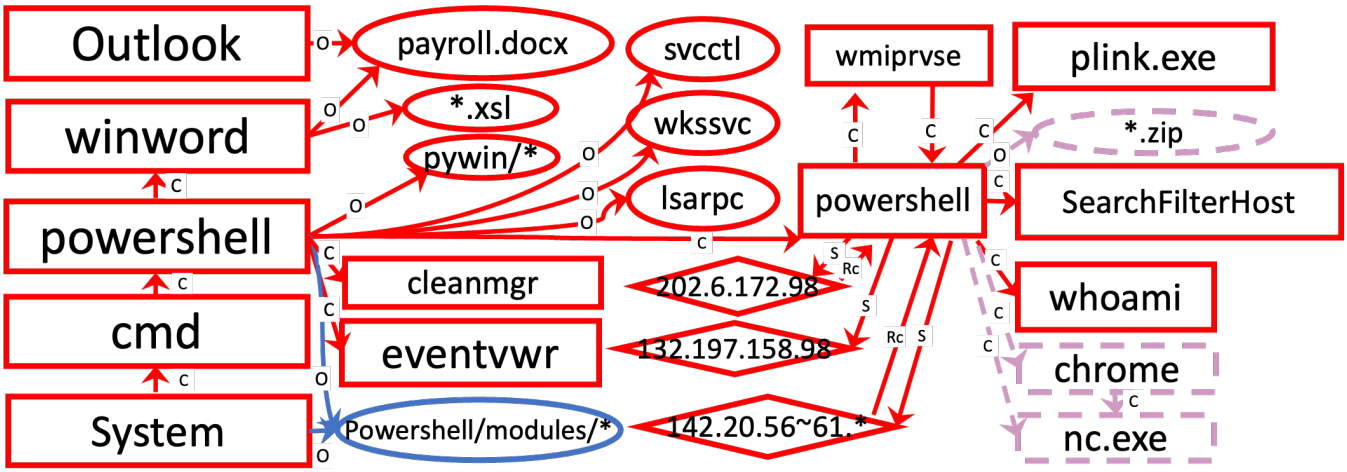


Figure 8. A summary graph that describes attack activity in DARPA's OpTC dataset in day 2, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

between Linux systems. Unlike `wget`, it serves point-to-point file transfers, so the user should know the identity of the other host. `du` is a Linux utility that collects a host's disk usage. Sysadmins could determine the legitimacy of `scp` and `du` by checking if the user issued the commands.

**E5-ClearScope (Fig. 14).** `defcontainer` is a system process associated with APK file installation. Sysadmins might confirm with the user the identities of the APK files they install. Sysadmins should further inspect the installed APK files to ensure that they are from legitimate vendors.

**OpTC (Fig. 15).** `Installagent` is Microsoft Windows Store's update agent, which uses the system services `System`, `backgroundTaskHost`, and `svchost`. Sysadmins need to investigate `Installagent` only when suspicious files (e.g., files not in the system path) appear in its activity.

## 4. Availability

KAIROS's code is available online at https://github.com/ProvenanceAnalytics/kairos. StreamSpot's [5] code and its datasets are available online at https://sbustreamspot.github.io/. Unicorn's [6] code is available online at https://github.com/crimson-unicorn. ThreaTrace's [7] code is available online at https://github.com/threaTrace-detector/threaTrace. DARPA E3 and E5 datasets are available online at https://github.com/darpa-i2o/Transparent-Computing. DARPA OpTC datasets are available online at https://github.com/FiveDirections/OpTC-data.

## Acknowledgments

We thank S&P 2023 and 2024 anonymous reviewers for their insightful comments. We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC). Nous remercions le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG) de son soutien. This work was partially supported by research funding from the National Research Council Canada (NRC). This material is based upon work supported by the U.S. National Science Foundation under Grant CNS-2245442. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## References

[1] Z. Cheng, Q. Lv, J. Liang, Y. Wang, D. Sun, T. Pasquier, and X. Han, "Kairos: Practical Intrusion Detection and

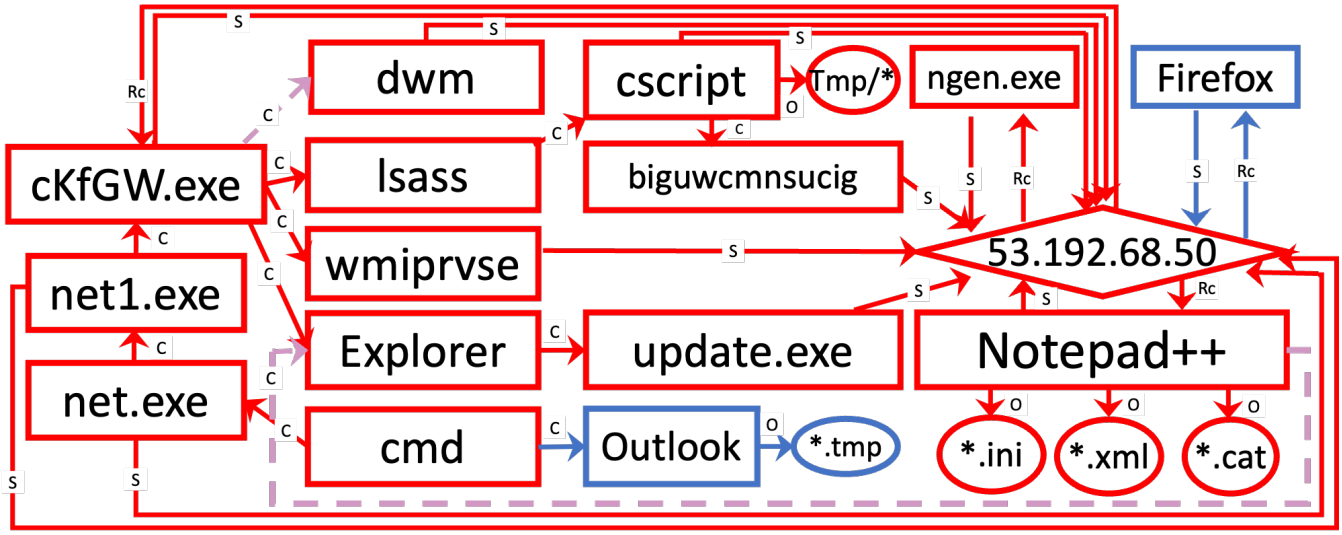


Figure 9. A summary graph that describes attack activity in DARPA's OpTC dataset in day 3, as automatically generated by KAIROS. Colors and dashed elements are added to ease comparison with the ground truth.

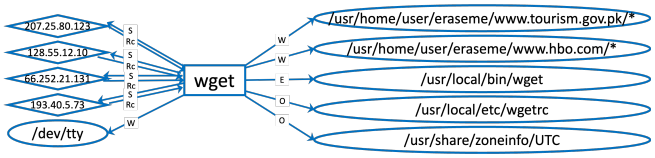


Figure 10. A benign summary graph in DARPA's E3-CADETS dataset.

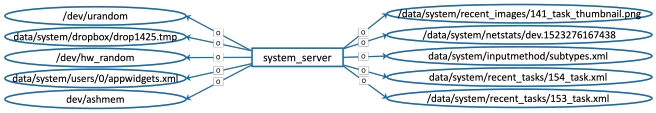


Figure 11. A benign summary graph in DARPA's E3-ClearScope.

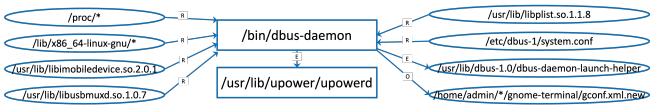


Figure 12. A benign summary graph in DARPA's E5-THEIA dataset.

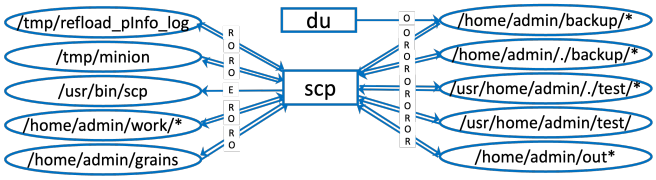


Figure 13. A benign summary graph in DARPA's E5-CADETS dataset.

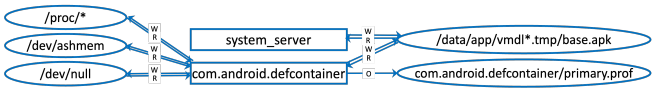


Figure 14. A benign summary graph in DARPA's E5-ClearScope.

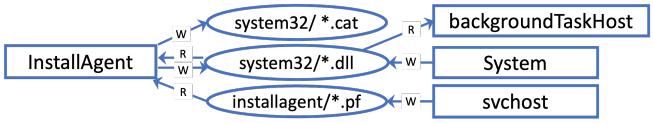


Figure 15. A benign summary graph in DARPA's OpTC.

Investigation using Whole-system Provenance," in *Symposium on Security and Privacy (S&P'24)*. IEEE, 2024.

[2] A. D. Keromytis, "Transparent Computing Engagement 3 Data Release," 2018, https://github.com/darpa-i2o/Transparent-Computing/blob/master/README-E3.md.

[3] J. Torrey, "Transparent Computing Engagement 5 Data Release," 2020, https://github.com/darpa-i2o/Transparent-Computing.

[4] M. van Opstal and W. Arbaugh, "Operationally Transparent Cyber (OpTC) Data Release," 2019, https://github.com/FiveDirections/OpTC-data.

[5] E. Manzoor, S. Momeni, V. Venkatakrishnan, and L. Akoglu, "Fast memory-efficient anomaly detection in streaming heterogeneous graphs," *International Conference on Knowledge Discovery and Data Mining (KDD'16)*, 2016.

[6] X. Han, T. Pasquier, A. Bates, J. Mickens, and M. I. Seltzer, "Unicorn: Runtime provenance-based detector for advanced persistent threats," in *Network and Distributed System Security Symposium (NDSS'20)*. The Internet Society, 2020.

[7] S. Wang, Z. Wang, T. Zhou, X. Yin, D. Han, H. Zhang, H. Sun, X. Shi, and J. Yang, "Threatrace: Detecting and tracing host-based threats in node level through provenance graph learning," *IEEE Transactions on Information Forensics and Security*, 2022.