CE 815 – Secure Software Systems

Causal Analysis (Poirot)

Mehdi Kharrazi
Department of Computer Engineering
Sharif University of Technology

Acknowledgments: Some of the slides are fully or partially obtained from other sources. A reference is noted on the bottom of each slide, when the content is fully obtained from another source. Otherwise a full list of references is provided on the last slide. Thanks to Zahra Fazli for the help on the slides.

Cybersecurity Stats in 2022

- An estimated 2,200 cyberattacks per day.
- 255 million phishing attacks occurring in a six-month span, with over 853,987 domain names reported for attempted phishing.
- 2.8 billion malware attacks launched in the first half of 2022 alone.
- 60% more malicious DDoS attacks occurring in the first six months of 2022 than the entirety of 2021.
- 1.51 billion IoT breaches were reported in the first six months of 2022.
- More than 500,000 users were negatively impacted by malicious mining software.
- 92% of malware was successfully delivered via email.
- 71% of organizations worldwide became victims of ransomware at least once.

Biggest Data Breaches in 2022

- Twitter was accused of concealing data breaches that impacted millions of users' data.
- More than 1.2 million credit card numbers were leaked on the hacking forum BidenCash.
- 11 million people were impacted by the Optus personal and medical cyberattack.
- Threat actors attempted to sell the data of 500 million WhatsApp users on the dark web.
- Both Uber and Rockstar had their internal servers compromised.
- A student loan breach released 2.5 million social insurance numbers.

Cybercrime Cost

https://www.independent.co.uk/advisor/vpn/cybercrime-statistics

Advanced Persistent Threats Attacks

New APT Group Red Stinger Targets Military and Critical Infrastructure in Eastern Europe

May 11, 2023 · Ravie Lakshmanan · Advanced Persistent Threat

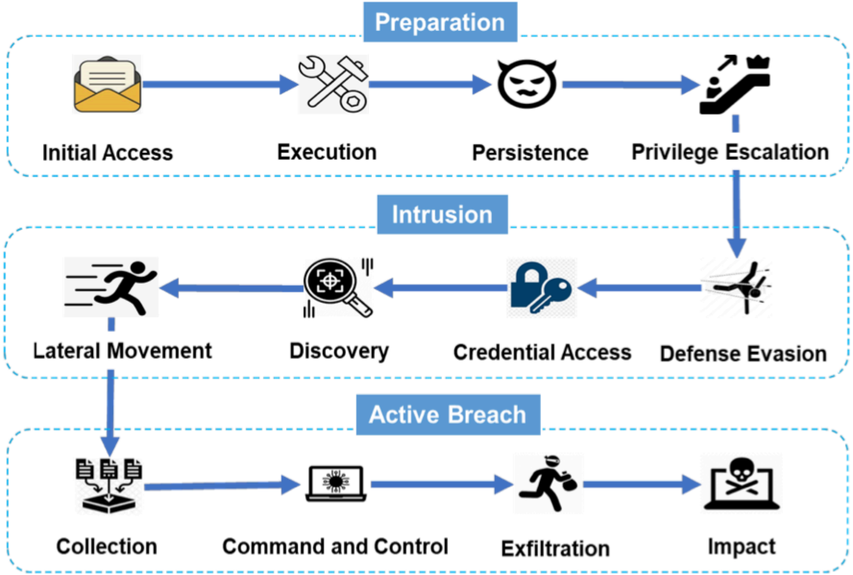
NEWS 3 NOV 2022

Threat Actor "OPERA1ER" Steals Millions from Banks and Telcos

Home > News > Security > Dark Pink hackers continue to target govt and military organizations

Dark Pink hackers continue to target govt and military organizations

By Bill Toulas · May 31, 2023



A Big Problem Affecting Many Nations and Industries

Long Duration and Stealthy

Advanced Persistent Threat (APT) and its challenges

- Targeted cyber attacks on organizations getting more sophisticated and stealthy.
  - Goal: to steal data, disrupt operations or destroy infrastructure.
- APTs combine many different attack vectors each appearing in some log sources.
- Firewall, IDS/IPS, Netflow, DNS logs, Identity and access management tools.
- Might occur over a long duration.
- Correlating heterogeneous alarms using heuristics like timestamp is not so effective Lacking the full picture (root cause, affected entities, etc.).

Evidence to investigate the attack

- System Audit log : ETW , Auditd , Sysmon , Sysdig

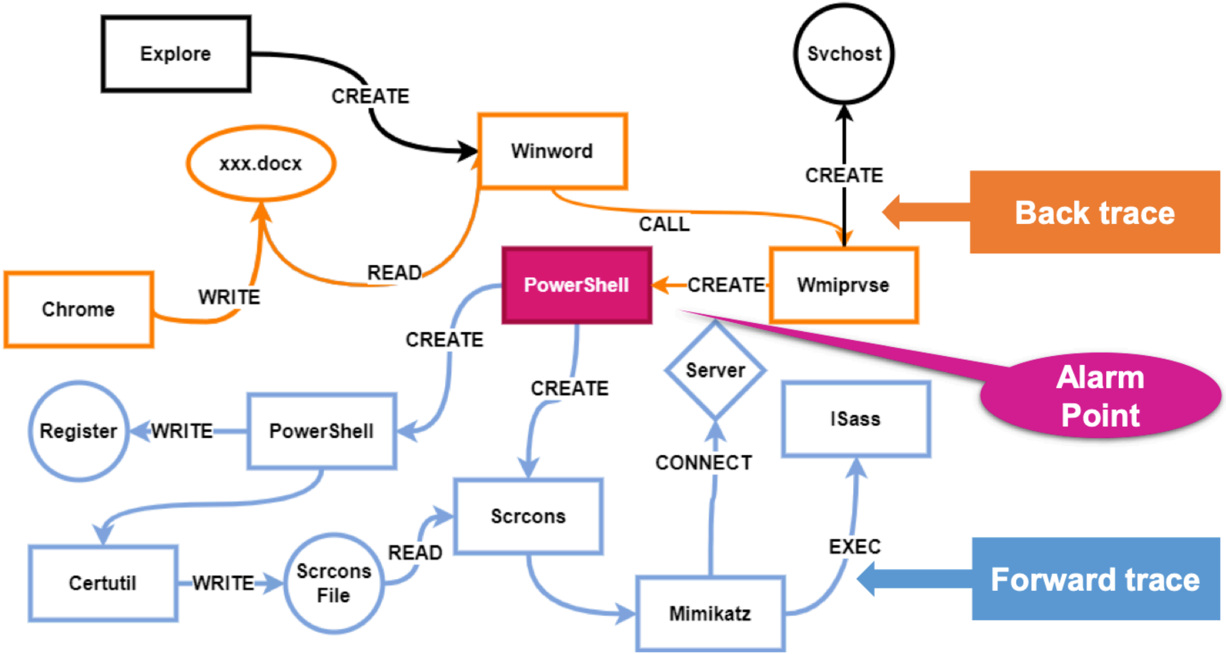
{"MSec": "166.6093", "PID": "1004", "PName": "msvsmon", "TID": "15336", "EventName": "FileIO/Read", "FileName": "C:\\Users\\Administrator\\Desktop\\ConsoleApp1\\ConsoleApp1\\bin\\x64\\Release\\System.Runtime.CompilerServices.Unsafe.dll", "Offset": "0", "IrpPtr": "0xFFFFE38FF04D2358", "FileObject": "0xFFFFE38FF047B900", "FileKey": "0xFFFF9407EF3EF700", "IoSize": "23,600", "IoFlags": "395,520"}
{"MSec": "597.7318", "PID": "4880", "PName": "pgAdmin4", "TID": "13740", "EventName": "Image/Load", "ImageBase": "0x00007FF9130F0000", "ImageSize": "0x0012A000", "ImageChecksum": "1,265,582", "TimeDateStamp": "-1,130,476,303", "DefaultBase": "0x00007FF9130F0000", "FileName": "C:\\Windows\\System32\\ole32.dll"}
{"MSec": "953.0958", "PID": "6624", "PName": "SearchApp", "TID": "-1", "EventName": "TcpIp/Send", "size": "80", "daddr": "202.89.233.101", "saddr": "192.168.0.74", "dport": "443", "sport": "58,197", "startime": "3,607,236", "endtime": "3,607,237", "seqnum": "0", "connid": "0x00000000"}

An example of windows ETW

Provenance Graph

- Use Provenance Graph to enable alert correlation for attack campaign detection.
- Vertices:
  - system entities (socket, process, file, memory, etc.), and agents (user, groups, etc.)
- Edges: system calls (causal dependencies or information flow)
- Leverage the full historical context of a system.
- Reason about interrelationships between different events and objects.

Detect APT Attacks with Provenance Graph



With data provenance, we can capture **full historical context** and all **casual relationships** among system subjects (e.g., process) and objects (e.g., files).

Poirot: Aligning Attack Behavior with Kernel Audit Records for Cyber Threat Hunting, S. M. Milajerdi, B. Eshete, R. Gjomemo, V. N. Venkatakrishnan, CCS, 2019.

Threat Hunting

- IOC: Indicators of Compromise (IOCs) related to an Advanced Persistent Threat (APT) detected in an organization.
- Post-detection, a prevalent query among security analysts is the potential targeting of their enterprise by the APT.
- The endeavor to ascertain if the enterprise was targeted, termed as Threat Hunting.
- Requires extensive and complex searches plus analysis on enterprise's host and network logs.
- Identifying entities from IOC descriptions in logs and evaluating the likelihood of the APT's successful infiltration.

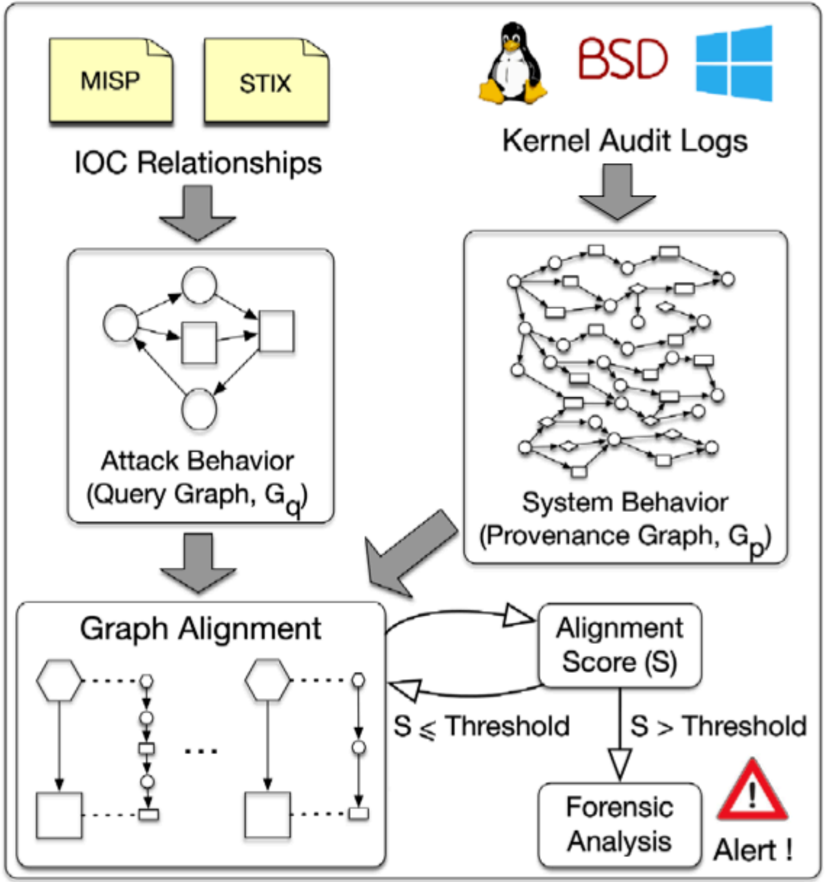
Enterprise Threat Hunting Challenges

- Design approaches to link related IOCs over long attack durations, enabling search among millions of log events.
- Ensure sound identification of attack campaigns despite mutated artifacts, and uncover the entire threat scenario.
  - Attacker might have mutated the artifacts like file hashes and IP addresses to evade detection.
- Facilitate timely understanding and reaction to threats by minimizing false positives and enabling prompt cyber-response operations.

Threat Hunting Limitations

- Information often shared via Cyber Threat Intelligence (CTI) reports in various formats like natural language, structured, and semi-structured forms.
  - OpenIOC, STIX, and MISP standards to facilitate IOC exchange and adversarial TTPs (techniques, tactics, and procedures) characterization.
- Current threat hunting largely operates on fragmented views like signatures, file/process names, and IP addresses.

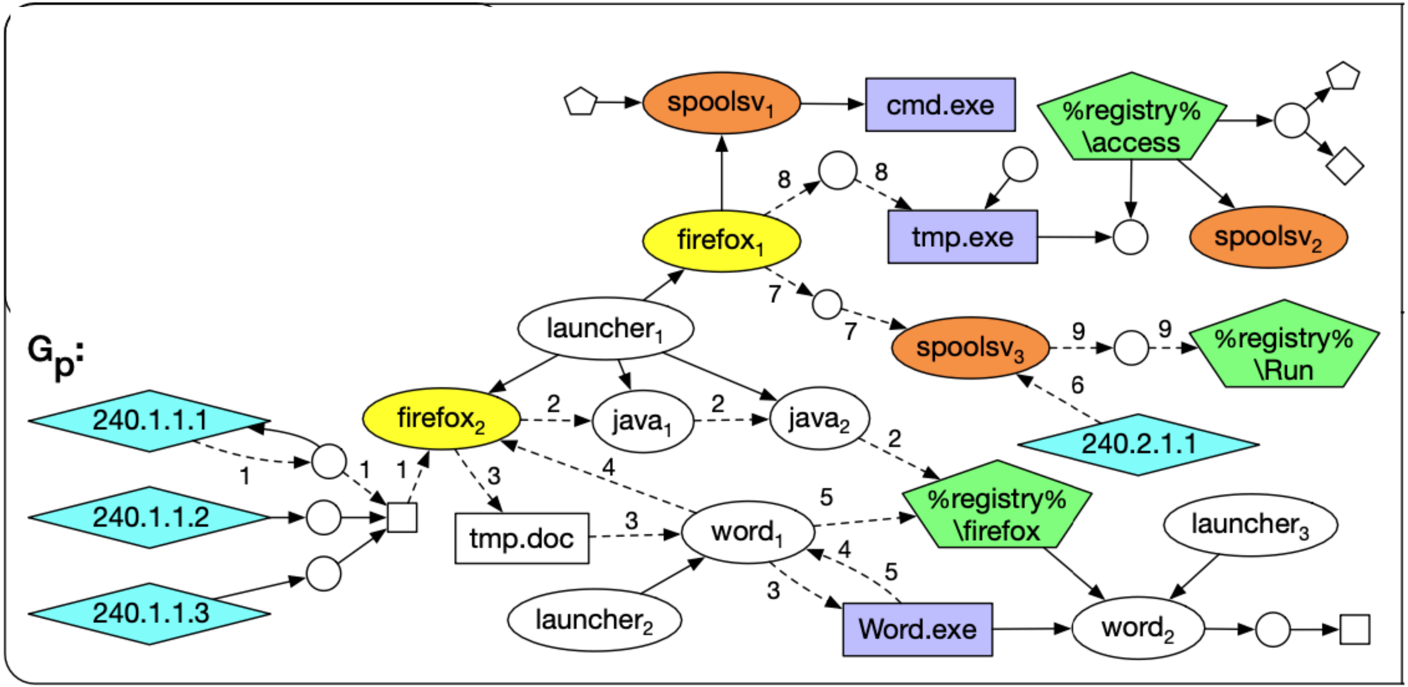
Poirot


MISP
STIX
IOC Relationships
BSD
Kernel Audit Logs
Attack Behavior
(Query Graph, Gq)
System Behavior
(Provenance Graph, Gp)
Graph Alignment
...
Alignment
Score (S)
S ≤ Threshold
S > Threshold
Forensic
Analysis
Alert !

Provenance Graph Construction (Gp)

- Determine APT actions in the system by modeling kernel audit logs.
- labeled, typed, and directed graph representation of kernel audit logs for efficient causality and information flow tracking.
- Nodes Representation: System entities involved in kernel audit logs like files and processes.
- Edges Representation: Information flow and causality among nodes, considering direction.
- Supports kernel audit logs from Windows, Linux, and FreeBSD, constructing an in-memory provenance graph with efficient searching features like fast hashing and reverse indexing for process/file name to unique node ID mapping.

Provenance Graph Construction (Gp)

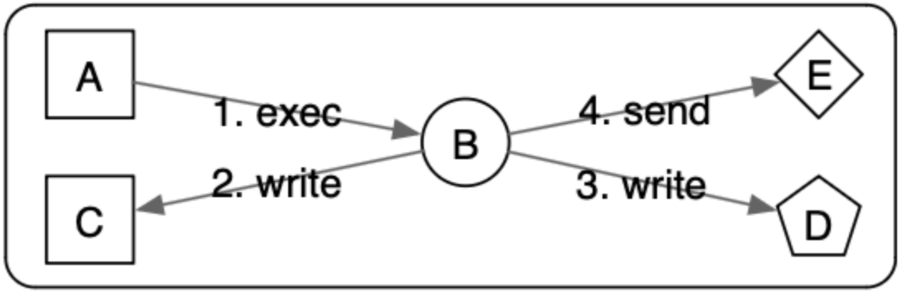


Query Graph Construction (Gq)

- IOCs and relationships among them are extracted from CTI reports related to known attacks, obtained from various sources like security blogs, threat intelligence reports, and forums.
- Automated tools help in initial feature extraction to generate query graphs, with manual refinement by security experts to reduce noise and enhance quality.
- The behavior from CTI reports is modeled as a labeled, typed, and directed graph, with entities transformed into nodes and relationships into directed edges.

Example: Report on DeputyDog malware

Upon execution, 8aba4b5184072f2a50cbc5ecfe326701 writes "28542CC0.dll" to this location: "C:\Documents and Settings\All Users\Application Data\28542CC0.dll". In order to maintain persistence, the original malware adds this registry key: "%HKCU%\Software\Microsoft\Windows\CurrentVersion\Run\28542CC0". The malware then connects to a host in South Korea (180.150.228.102).



Graph Alignment

- Aligning query graph G_q representing attack, with provenance graph G_p representing system activity.
- Matching single edges in G_q to paths in G_p , critical for algorithm design to handle noise added by attackers.
- Existing graph matching problems are NP-complete, with practical limitations in threat hunting context.
  - Hence, finds possible candidate alignments, expands search from high likelihood seed nodes, employing a novel metric called influence score to prioritize flows.
  - Upon alignment, a score representing similarity is calculated; if above a threshold, an alert is raised for analysts, otherwise, the process iterates with the next seed node candidate.

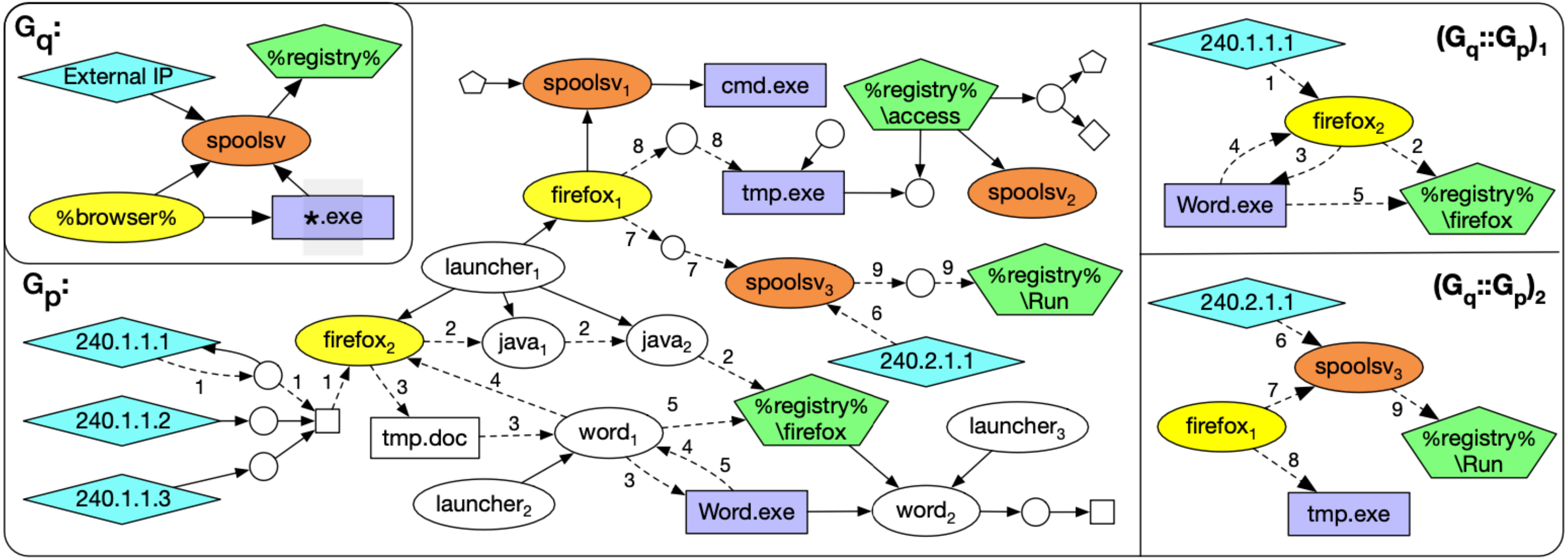


Fig. 3: Simplified Provenance Graph (G_p), Query Graph (G_q), and two sample graph alignments ($G_q :: G_p$). Node types are shown with different shapes, and possible alignments for each node is shown with the same color. The numbers on the edges are merely to illustrate possible paths/flows and do not have additional meaning.

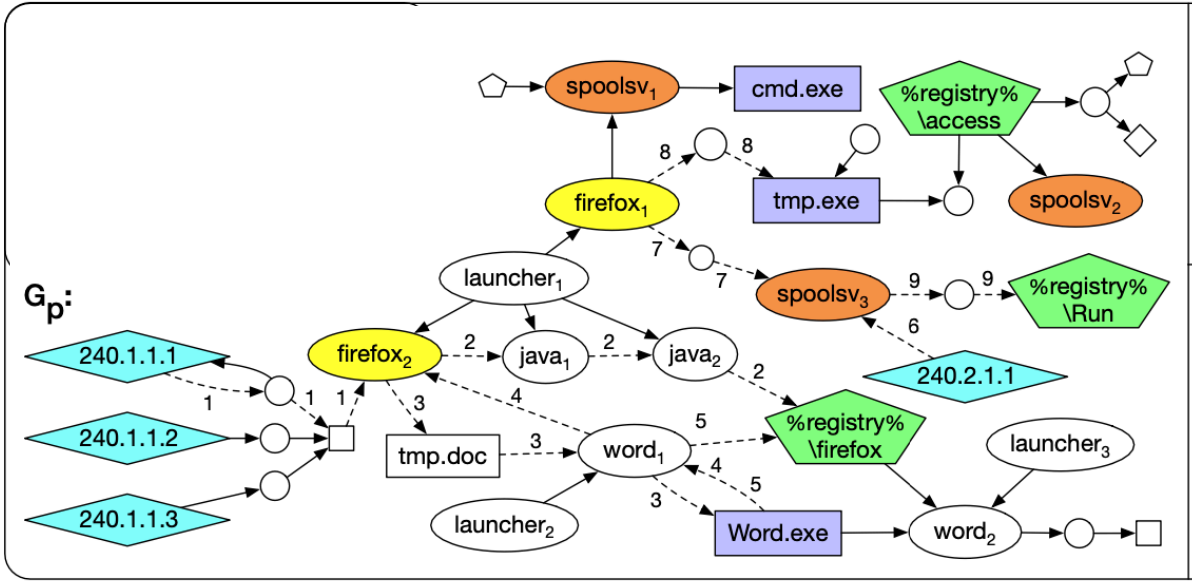
Algorithm Details

- Two Types of Alignments: Node alignment (between two nodes in different graphs) and graph alignment (a set of node alignments).
- Node Alignment Example: A node representing a commonly used browser in G_q and a node representing a Firefox process in G_p.
- Many-to-Many Relationship from $V(G_q)$ to $V(G_p)$, indicating multiple possible alignments.
- Find the best possible graph alignment among candidate graph alignments.
- Determine the best candidate alignment based on the number of aligned nodes and correspondence of flows to edges in G_q .

Algorithm Details

- Path scoring function to quantify the "goodness" of a graph alignment.
- Likelihood of an attacker producing a flow between nodes.
- Two flows from node firefox2 to %registry%\firefox in graph G_p, with different likelihoods based on attacker control.
- Not dependent on flow length but on the number of processes and distinct ancestors in the process tree.
- Robust against evasion attempts, as activities adding noise have the same common ancestors unless attacker incurs higher compromise costs.

Provenance Graph Construction (Gp)



Algorithm Details

- C_{min}: Minimum number of distinct compromises needed to create a flow from node i to node j.
  - Common Ancestor: C_{min} value of 1 if all processes in a flow share a common ancestor.
  - Multiple Ancestors: Higher C_{min} values indicate more compromises and a harder flow for attackers.
- Assumption that attackers are unlikely to compromise many processes due to resource constraints.
  - C_{thr} Limit: A threshold limiting C_{min} values to identify likely attacker-initiated flows.
- Influence Score: Inverse of C_{min}, higher values indicate easier control by an attacker.
- Maximum and Minimum Scores: Scores range from 1 (easy control) to 0 (no flow exceeding C_{thr}).

Algorithm Details

- $S(G_q :: G_p)$ calculates alignment score based on influence scores.
  - Sum of influence scores normalized by maximal possible value.
  - Higher $S(G_q :: G_p)$ value indicates more node alignments and similar flows under potential attacker control.
- Score Range: Value between 0 and 1, with 1 indicating high likelihood of attacker control.
- Alarm Threshold: Predefined threshold τ to trigger an alarm.
- Threshold Calculation: τ determined based on maximum number of distinct entry points an attacker is likely to exploit.
- Alarm Condition: Alarm raised if $S(G_q :: G_p) \geq \tau$.

Algorithm Details

- Maximize alignment score by finding $G_q :: G_p$ in a large provenance graph G_p
  - Size of G_p reaching millions of nodes and edges.
- Step 1 (Find Candidate Node Alignments):
  - Search G_p nodes for candidate alignments for each G_q node.
  - Candidate alignment based on node name, type, annotations.
  - Initial step focuses on nodes in isolation without path/flow information.

Algorithm Details

- Step 2 (Selecting Seed Nodes):
  - Identify starting points based on likely attack activities having fewer alignments.
  - Sort nodes by increasing order of candidate alignments and select seed nodes with fewest alignments first.
- Step 3 (Expanding the Search):
  - From selected seed node, iterate over all aligned nodes in G_p initiating graph traversals to find other aligned nodes.
  - Stop search expansion along a path once influence score reaches 0 to reduce search complexity.
  - Multiple forward/backward tracking cycles may be needed based on G_q shape.
  - Repeat traversals from nodes adjacent to unvisited but previously visited nodes until all G_q nodes are covered.

Algorithm Details

- Step 4 (Graph Alignment Selection):
  - Produce final result or iterate search from Step 2 if no result is found.
  - Identify a subset of candidate nodes in G_p for each node in G_q .
  - Determine total possible graph alignments based on candidate alignments per node.
  - Maximize alignment score by starting from a seed node, select node in G_p maximizing alignment score contribution, and fix this node alignment. Follow edges in G_q to fix alignment of additional nodes, selecting those maximizing score contribution.
- Selection Function
  - Approximates each alignment's contribution to final alignment score, aiming for highest contribution.
  - Evaluation reveals attack graph usually found within the first few iterations.

Evaluation

- Experiment 1: Utilized DARPA Transparent Computing (TC) program scenarios, simulating adversarial engagements in an enterprise network setting.
- Experiment 2: Tested Poirot on real-world incidents replicated from publicly available threat reports in a controlled environment.
- Experiment 3: Assessed Poirot's false signal robustness in an attack-free dataset.
- C_{thr} set to 3 across experiments, influencing false positives/negatives rate.
- Manual analysis of matched attack subgraphs to validate correct pinpointing of actual attacks in query graphs.

Evaluation on DARPA TC Dataset

- Experiment Setup: Utilized a dataset from DARPA TC program's red-team vs. blue-team adversarial engagement, with various servers and benign activities simulated.
- Attack Scenarios Evaluated: Ten scenarios across BSD, Windows, and Linux systems.
- BSD Attacks: Executed on a back-doored Nginx server on FreeBSD 11.0 (64-bit).
- Windows Attacks: Win-1 involved a phishing email with malicious Excel macro; Win-2 exploited a vulnerable Firefox browser on Windows 7 Pro (64-bit).
- Linux Attacks: Conducted on Ubuntu 12.04 (64-bit) and 14.04 (64-bit); Linux1&3 had in-memory browser exploits, while Linux2&4 involved malicious browser extensions.

Evaluation on DARPA TC Dataset (Con't)

Scenario	**subjects** $\in$ $\lvert V(G_q)\rvert$	**objects** $\in$ $\lvert V(G_q)\rvert$	$\lvert E(G_q)\rvert$	$\lvert F(G_q)\rvert$
BSD-1	4	9	19	81
BSD-2	1	7	10	32
BSD-3	3	18	34	159
BSD-4	2	8	13	43
Win-1	13	8	26	149
Win-2	1	13	19	94
Linux-1	2	9	19	62
Linux-2	5	12	24	112
Linux-3	2	8	22	48
Linux-4	4	11	22	96

Evaluation on DARPA TC Dataset (Con't)

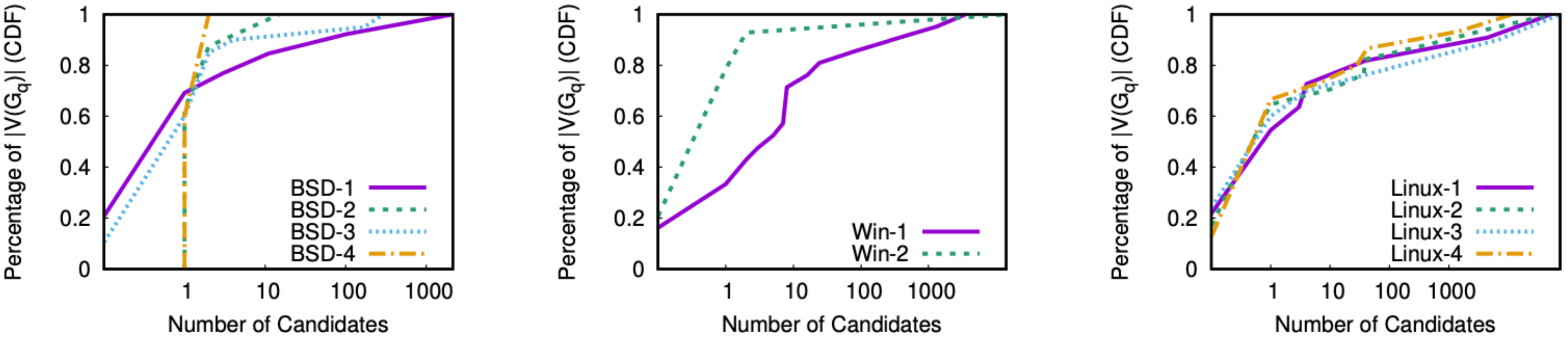


Fig. 4: Cumulative Distribution Function (CDF) of number of candidates in $|G_p|$ for each node of $|G_q|$. From left to right: BSD, Windows, and Linux Scenarios.

Evaluation on DARPA TC Dataset (Con't)

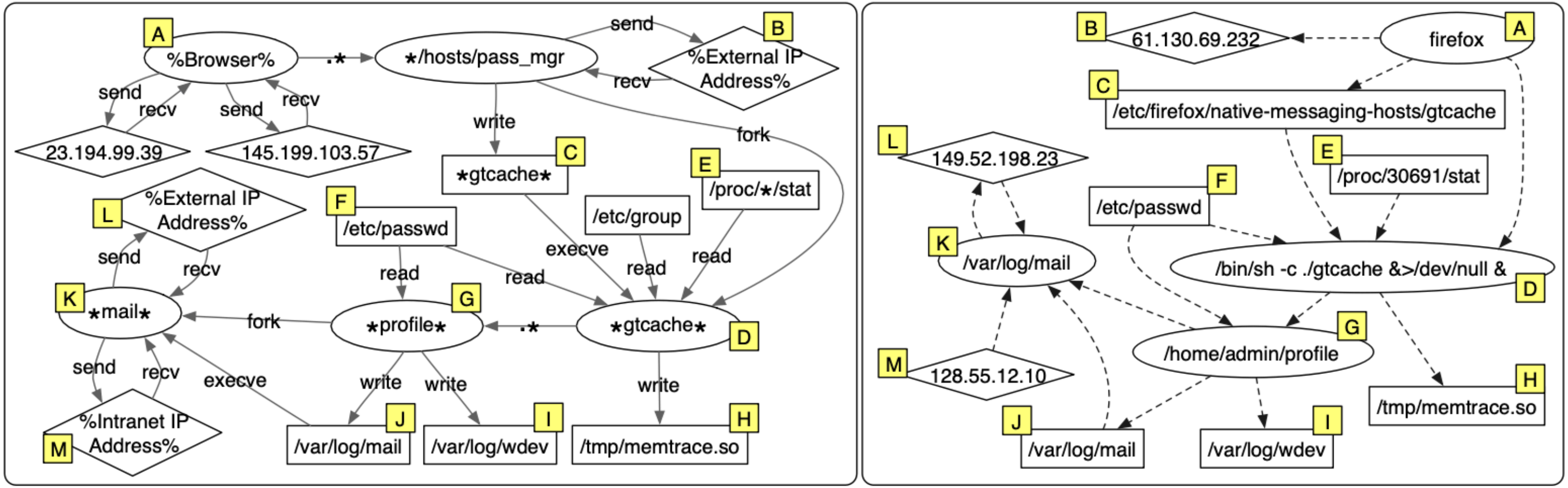


Fig. 5: Query Graph of Scenario: Linux-2 (on the left) and its Detected Alignment (on the right).

Evaluation: real-world incidents

Malware Name	Report Source	Year	Reported Samples	Analyzed Malware MD5	Sample Relation	Isolated IOCs	Detection Results			
							RedLine	Loki	Splunk	POIROT
njRAT	Fidelis [58]	2013	30	2013385034e5c8dfbbe47958fd821ca0	different	153	F+H	F+H	P	B (score=0.86)
DeputyDog	FireEye [50]	2013	8	8aba4b5184072f2a50cbc5ecfe326701	subset	21	F×2+H+R	F×2+H	P+R	B (score=0.71)
Uroburos	Gdata [5]	2014	4	51e7e58a1e654b6e586fe36e10c67a73	subset	26	F+H	F+H	R	B (score=0.76)
Carbanak	Kaspersky [22]	2015	109	1e47e12d11580e935878b0ed78d2294f	different	230	-	PE	S	B (score=0.68)
DustySky	Clearsky [65]	2016	79	0756357497c2cd7f41ed6a6d4403b395	different	250	-	-	-	B (score=1.00)
OceanLotus	Eset [6]	2018	9	d592b06f9d112c8650091166c19ea05a	subset	117	F+R	F+PE	P+R	B (score=0.65)
HawkEye	Fortinet [7]	2019	3	666a200148559e4a83fabb7a1bf655ac	different	3	-	PE	-	B (score=0.62)

Table 4: Malware reports. In the Detection Results, B=Behavior, PE=PE-Sieve, F=File Name, H=Hash, P=Process Name, R=Registry, S=Windows Security Event.

Evaluation: Benign Dataset

Scenario	Size on Disk (Uncompressed)	Consumption time	Occupied Memory	Log Duration	sub ∈ $\|V(G_p)\|$	obj ∈ $\|V(G_p)\|$	$\|E(G_p)\|$	Search Time (s)
BSD-1	3022 MB	0h-34m-59s	867 MB	03d-18h-01m	110.66 K	1.48 M	7.53 M	3.28
BSD-2	4808 MB	0h-58m-05s	1240 MB	05d-01h-15m	213.10 K	2.25 M	12.66 M	0.04
BSD-3&4	1828 MB	0h-21m-31s	638 MB	02d-00h-59m	84.39 K	897.63 K	4.65 M	26.09 (BSD-3), 1.47 (BSD-4)
Win-1&2	54.57 GB	4h-58m-30s	3790 MB	08d-13h-35m	1.04 M	2.38 M	70.82 M	125.26 (Win-1), 46.02 (Win-2)
Linux-1&2	9436 MB	1h-26m-37s	4444 MB	03d-04h-20m	324.68 K	30.33 M	51.98 M	1279.32 (Linux-1), 1170.86 (Linux-2)
Linux-3	131.1 GB	2h-30m-37s	21.2 GB	10d-15h-52m	374.71 K	5.32 M	69.89 M	385.16
Linux-4	4952 MB	0h-04m-00s	1095 MB	00d-07h-13m	35.81 K	859.03 K	13.06 M	20.72

Table 8: Statistics of logs, Consumption and Search Times.

Conclusion

- Cyber threat hunting cast as graph pattern matching.
- Efficient alignment algorithm for embedding threat behavior graph in kernel audit records provenance graph.
- Tested on real-world cyber attacks, ten red-team attack scenarios across three OS platforms.
- All attacks detected confidently, no false signals, and completed within minutes.

Acknowledgments

- [packetlabs] 239 Cybersecurity Statistics (2023) [https://www.packetlabs.net/posts/239-cybersecurity-statistics-2023/]
- [Prographer] PROGRAPHER: An Anomaly Detection System based on Provenance Graph Embedding, F. Yang, J. Xu, C. Xiong, Z. Li, K. Zhang, Usenix Sexurity 2023.
- [Holmes] HOLMES: Real-Time APT Detection through Correlation of Suspicious Information Flows, S. Momeni Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, V. N. Venkatakrishnan, IEEE Symposium on Security and Privacy 2019.
- [Poirot] Poirot: Aligning Attack Behavior with Kernel Audit Records for Cyber Threat Hunting, S. M. Milajerdi, R. Gjomemo, B. Eshete, V.N. Venkatakrishnan, CCS, 2019.