


Present at *IEEE International Conference on Communications 2025*

A Cascade Approach for APT Campaign Attribution in System Event Logs: Technique Hunting and Subgraph Matching



Yi-Ting Huang*, Ying-Ren Guo†, Guo-Wei Wong‡, Meng Chang Chen†

*National Taiwan University of Science and Technology

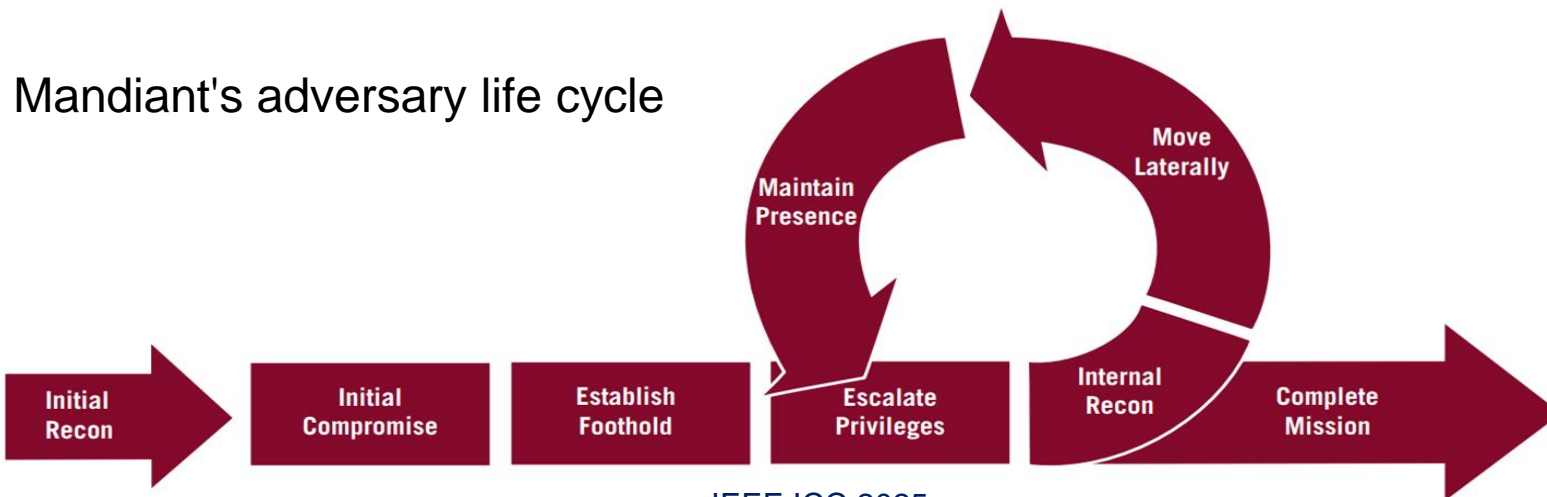
†Academia Sinica

‡National Taiwan University

TAIWAN

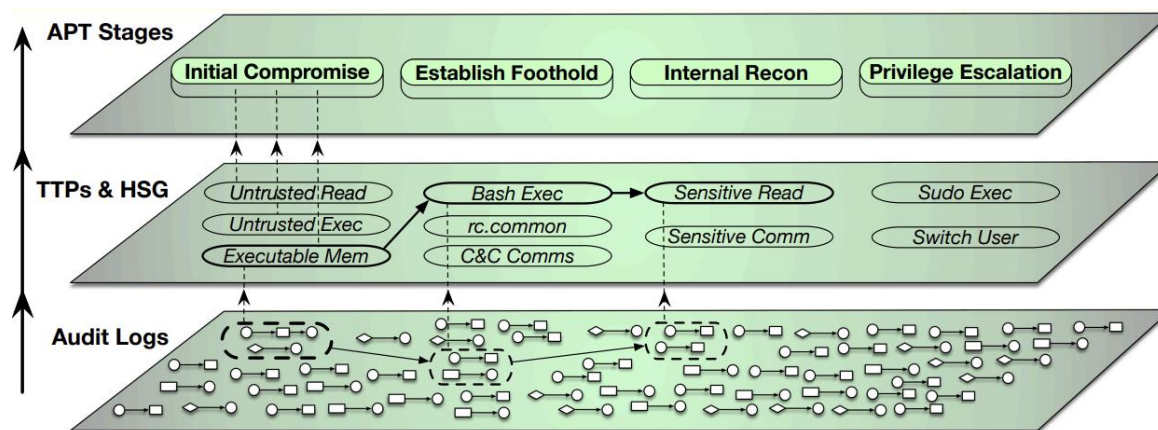
Background

- **Advanced Persistent Threats (APTs)** has posed significant challenges to the cybersecurity community.
 - BlackEnergy
 - SolarWinds Compromise
- Differ from traditional malware or botnet attacks, APT campaigns are **multistage operations**, that is often begin with *gaining a foothold* in a target environment, followed by prolonged periods of undetected activity, *data exfiltration*, and *system compromise*.

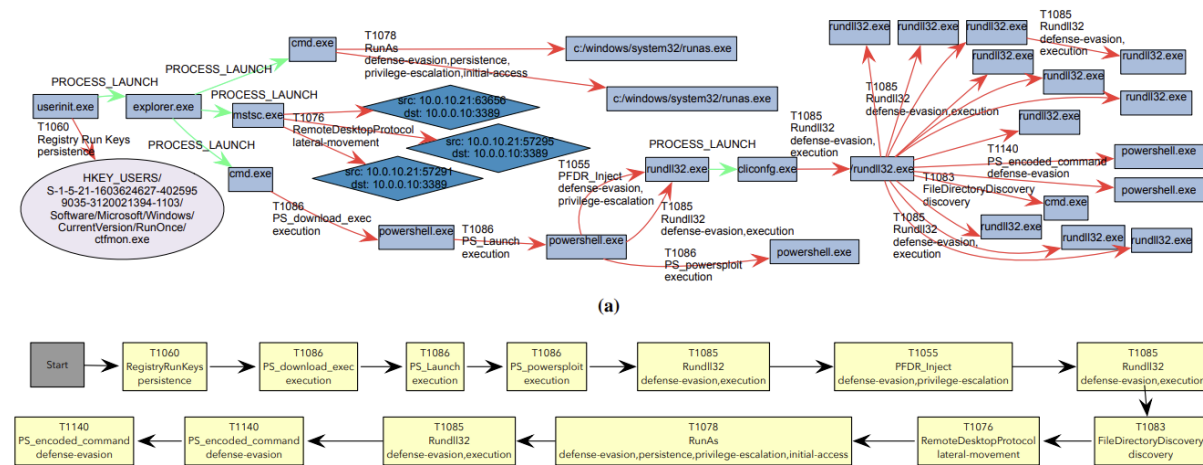


Motivation 1

- **Holmes** [6] and MORSE [7] have shown that combining coarse-grained analysis (which classifies events as benign or malicious) with **fine-grained analysis (which maps events to Tactics, Techniques, and Procedures, TTPs)** can significantly enhance threat detection capabilities.
- **RapSheet** [8] and KRYSTAL [9] focus on detecting known attack descriptors to construct contextual attack scenarios, further improving understanding of intrusion activity.



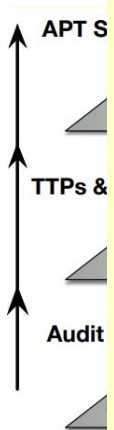
Holmes [S&P19]



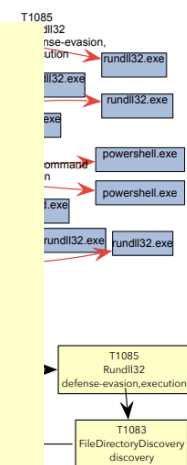
RapSheet [S&P20]

Motivation 1

- Holmes and MORSE have shown that combining coarse-grained analysis (which classifies events as benign or malicious) with **fine-grained analysis** (**which maps events to Tactics, Techniques, and Procedures, TTPs**) can significantly enhance threat detection capabilities.
- RepSheet and KRYSTAL focus on detecting known attack descriptors to construct contextual attack scenarios, further improving understanding of intrusion activity.



However, these methods typically require **manual input to define mapping rules** for recognizing attack patterns, which **limits their scalability and automation potential**.



Motivation 2

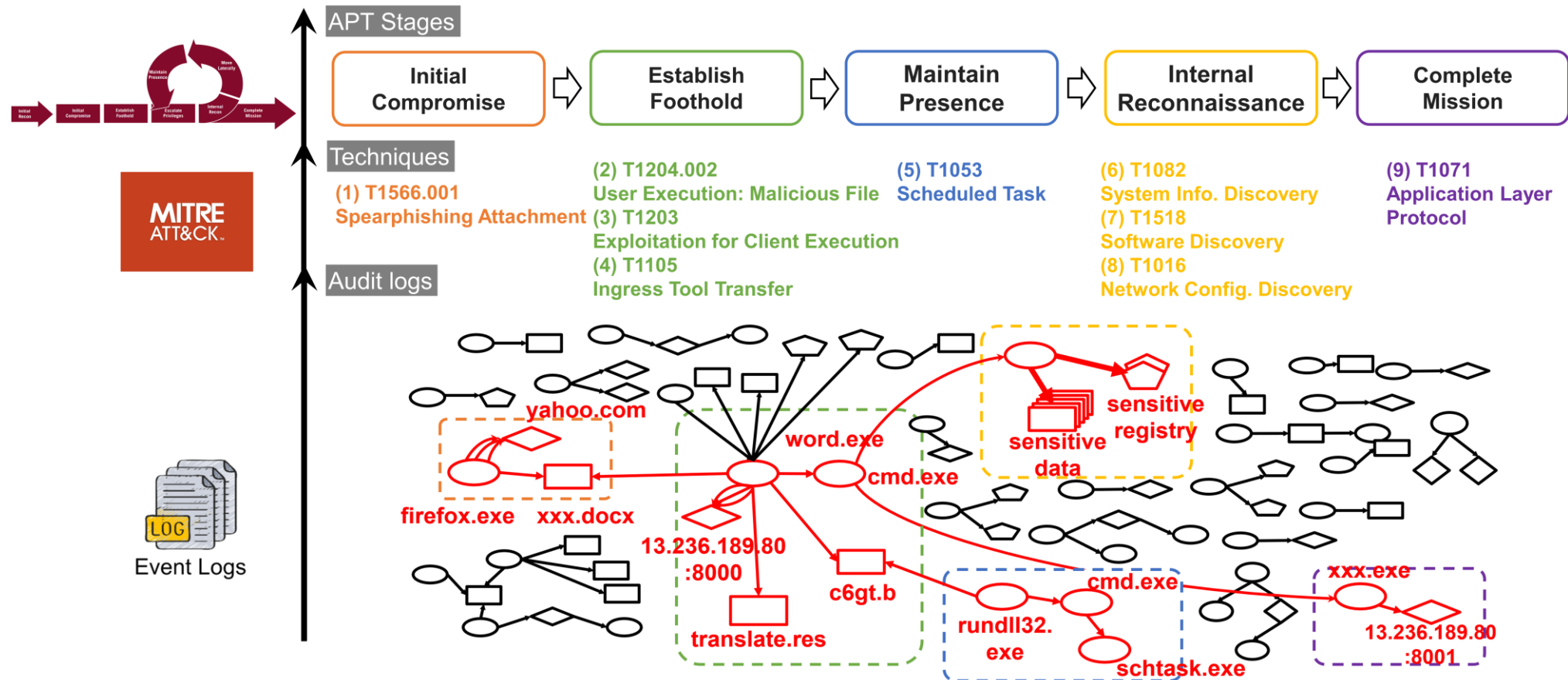
- **Forensic analysis** of security incidents, whether to attribute attacks to specific threat actors or align them with known campaigns based on observable artifacts, remains a labor-intensive process.
- Few studies have explored cyber threat attribution based on
 - observable attack stages [10]
 - attacker profiling [11]
 - artifact analysis [12]
- Recognizing intrusion activities as part of known APT campaigns is equally important for improving system defenses and accelerating incident response.

Research purpose

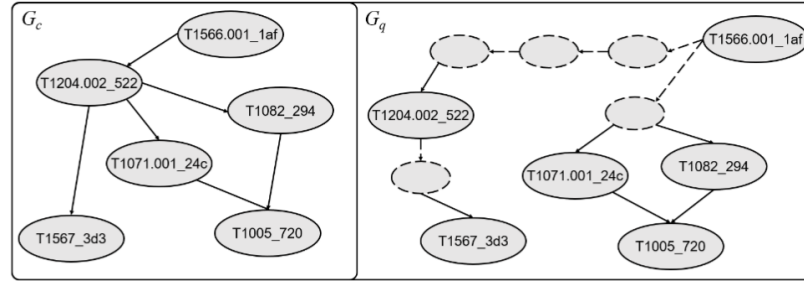
- We propose a machine learning-based **Straight Forward Method (SFM)** for **audit log analysis** and **APT campaign detection**.
- Specifically, the tasks of this study are:
 1. **Malicious behavior identification:** design a neural network detection model to discover **malicious behaviors (MITRE ATT&CK TTPs)**
 2. **APT campaign attribution:** identify **the most likely APT campaign** by matching the discovered behaviors with known APT campaigns.



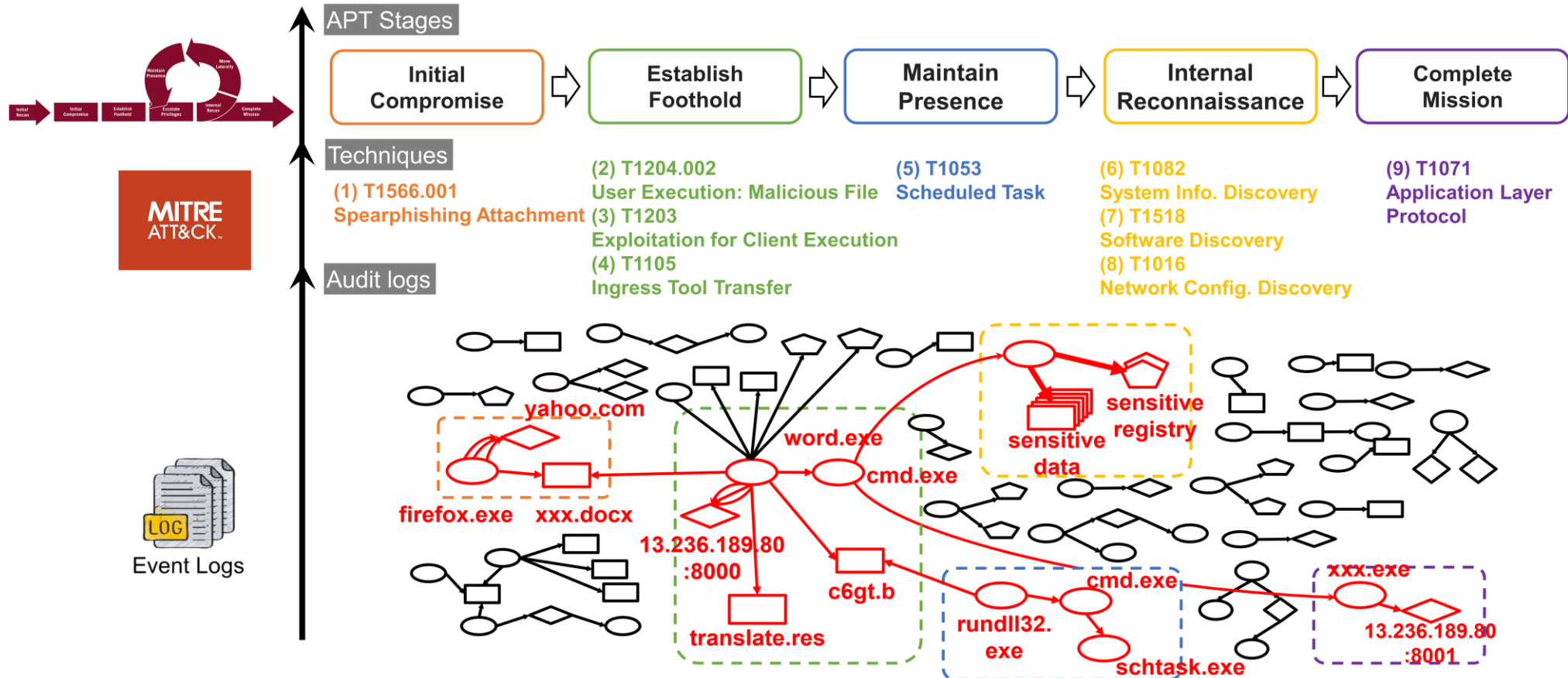
Our intuition



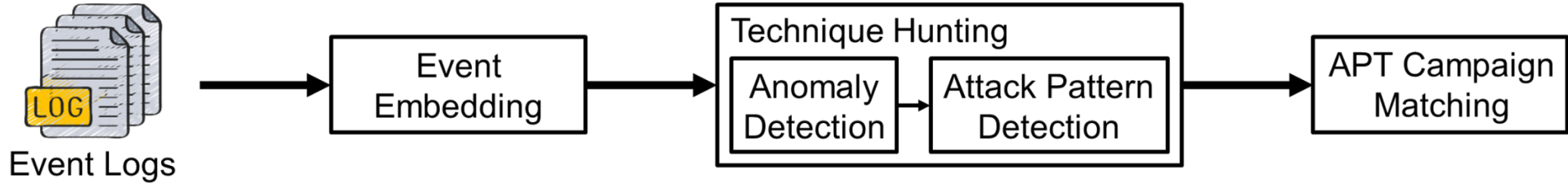
Our intuition



Campaign Graph Query Graph



Straight Forward Method (SFM)



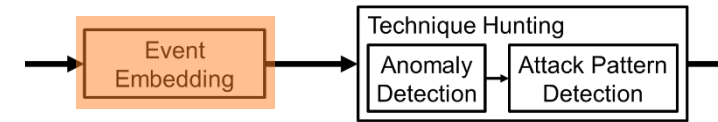
- **Event Embedding:** Converts textual logs into numerical vectors.
- **Technique hunting:**
 - **Anomaly Detection:** Handles event imbalance to highlight suspicious behavior.
 - **Attack Pattern Detection:** Uses sequence modeling to detect specific TTPs.
- **APT campaign matching:** Matches to potential actors via graph-based similarity.

Event logs

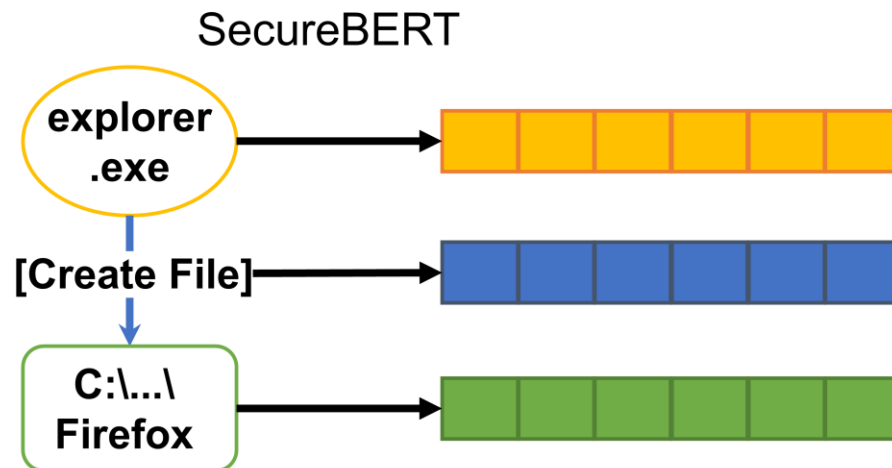
Time of Day	Process Name	PID	Operation	Path
09:00:42.519...	groupagent.exe	5216	Process Create	C:\Users\ezk\AppData\Local\Microsoft\Windows\groupagent.exe
09:00:42.519...	groupagent.exe	10264	Process Start	
09:00:42.519...	groupagent.exe	10264	Thread Create	
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager\AppCertDlls
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager\AppCertDlls
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\System\CurrentControlSet\Control\SafeBoot\Option
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\Software\WOW6432Node\Policies\Microsoft\Windows\Safer\Coc
09:00:42.519...	groupagent.exe	5216	RegOpenKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\Safer\CodeIdentifiers
09:00:42.519...	groupagent.exe	5216	RegSetInfoK...	HKLM\SOFTWARE\Policies\Microsoft\Windows\safer\codeidentifiers

- Event logs are collected from the Process Monitor (ProcMon), which records detailed system activities such as **process creation** and **registry access**.
- These logs provide critical information for analyzing system behavior.

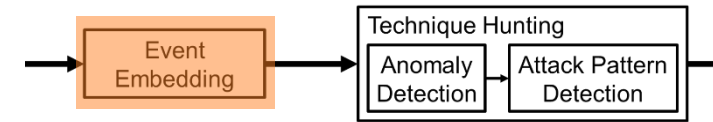
Event Embedding



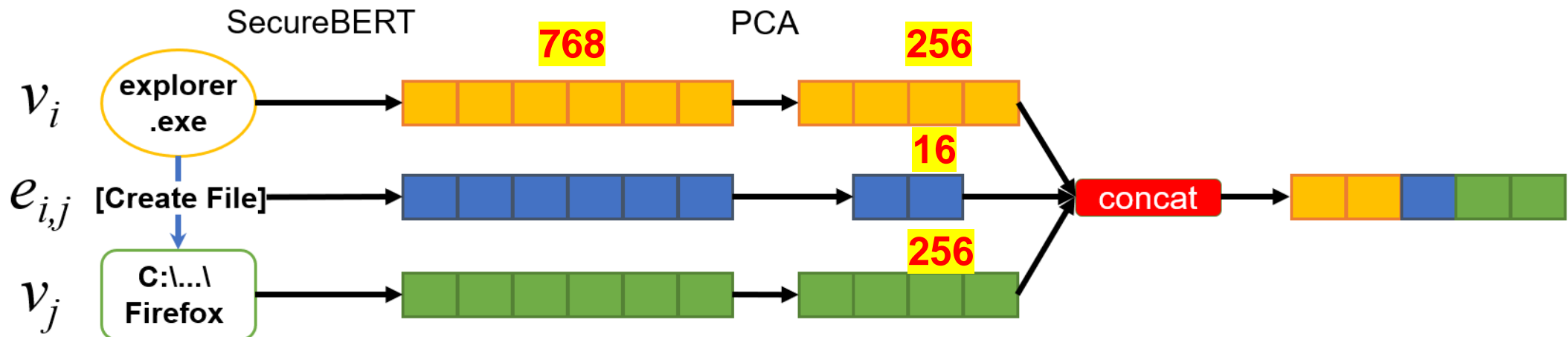
- To process system events, we use an embedding function, **SecureBERT** [14], to convert a single system event into numerical vectors.
- SecureBERT is a domain-specific language model which is trained on a large amount of cybersecurity textual data.
- Event embedding preserves meaningful semantics and contextual relations.



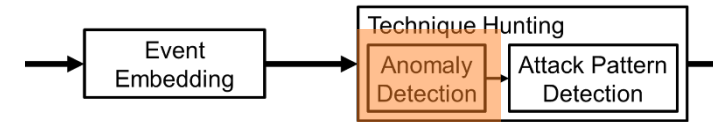
Event Embedding



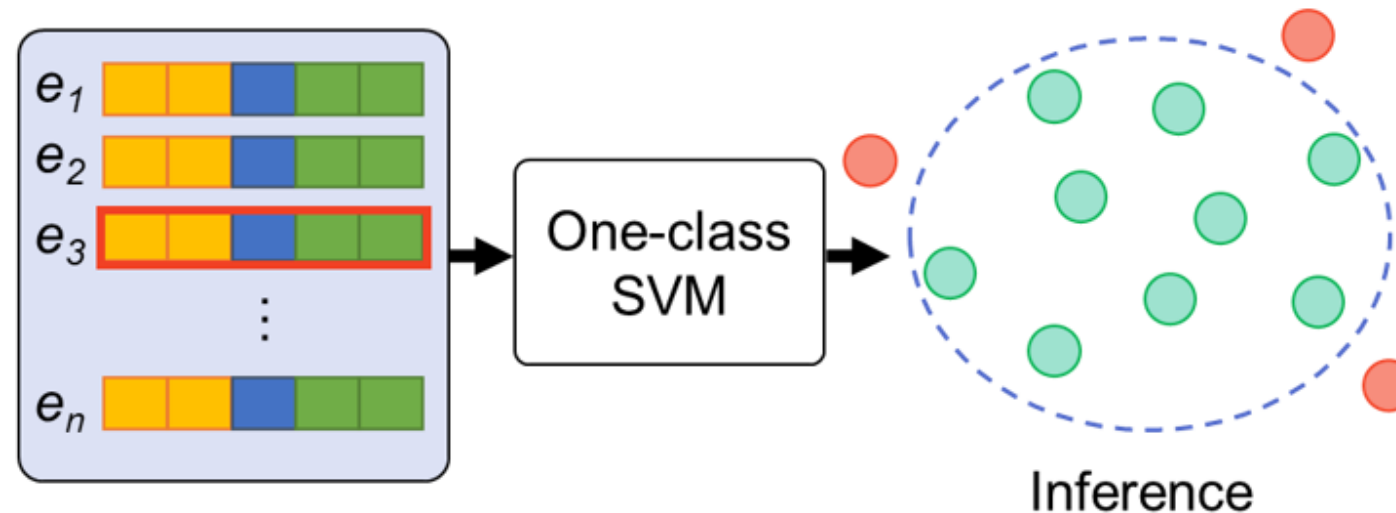
- Since the SecureBERT embeddings are high-dimensional (768), we further apply **principal component analysis (PCA)** to reduce dimensionality.
- The resulting embeddings serve as features of individual events for subsequent tasks, i.e. **anomaly detection** and **attack pattern detection**.



Anomaly Detection

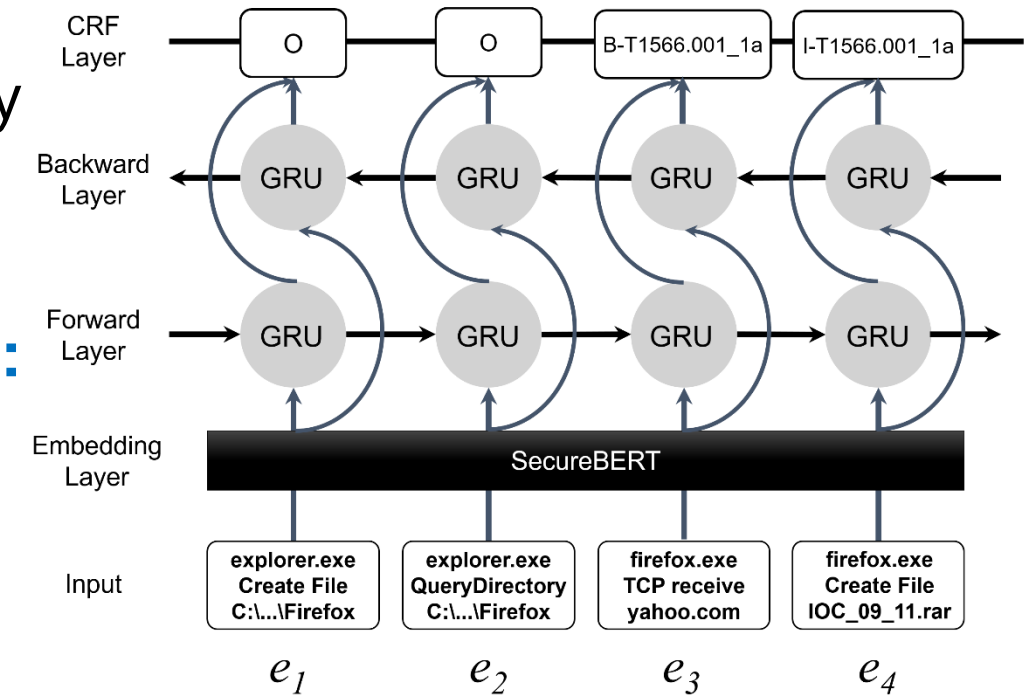
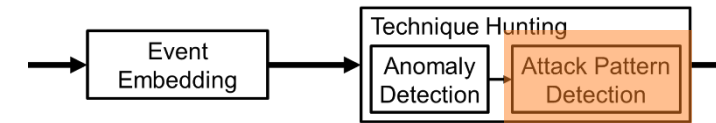


- In real-world scenarios, there is often a significant imbalance between attack and benign events.
 - E.g. in the DARPA TC3 dataset, compared to over 14 million benign events is collected in one day, attack events number only around 5,300 (2600:1).
- To mitigate this, we use a **one-class support vector machine (SVM)** to preserve likely malicious processes.

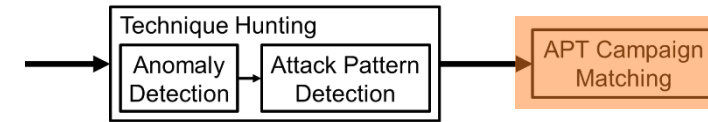


Attack Pattern Detection

- Since a **Technique** may involve in more than **one events**, **BiGRU-CRF** is employed to identify TTPs within the malicious events.
- **Bidirectional Gated Recurrent Units (BiGRU):** to process the sequence in both directions.
- **Conditional Random Field (CRF):** to jointly decode labels across sequences by capturing dependencies among neighboring labels.



APT Campaign Matching

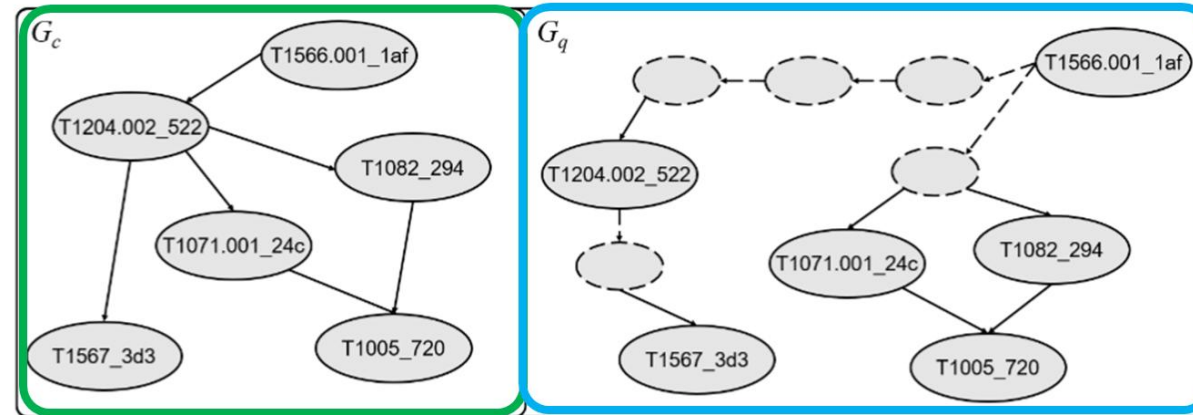


- Determining the most likely APT campaign is formulated as graph-matching problem.

G_c : campaign graphs from CTI reports

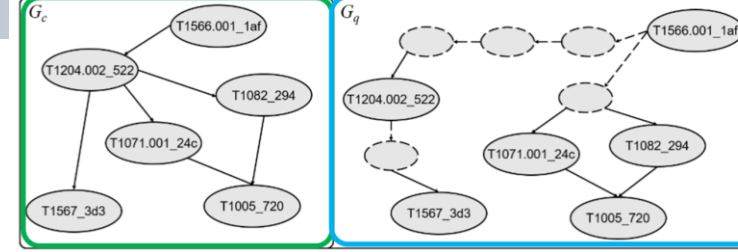
G_q : discovered TTPs graph from event logs

- Node: TTPs
- Edge: a temporal relationship between two TTPs involving the same system entities.



- Subgraph isomorphism problem** is NP-complete.
- We observe that nodes within G_q often do not align consistently with nodes in the known campaign G_c due to high FP and FN rates.

APT Campaign Matching



- Graph Edit Distance (GED):

$$GED(G_q, G_c) = \min_{o_1, \dots, o_m \in \gamma(G_q, G_c)} \sum_{i=1}^m cost(o_i)$$

costs associated with operation

- deletion
- substitution
- insertion

- A measure of similarity between two graphs based on the minimum cost needed to transform one graph into another.
 - **Insertion** (e.g., adding a new technique),
 - **Deletion** (e.g., removing an unmatched technique), and
 - **Substitution** (e.g., replacing one technique with another).
- The **lower the total cost** of these operations, the **more similar** the graphs are.
- The threat actor whose campaign graph has the **smallest GED** to the query graph is considered the **most likely match**.

Evaluation Settings

- **Dataset:**
 - Five synthetic campaigns from SAGA [30][31]
 - 21 Technique labeling

APT Campaign	Attack Stage	Techniques	Event	MalEvent
Higaisa [25]	{1,2,6,4,4,6,6}	PA, MFE, RK, SID, SNCD, MTOS, ST	607,416	0.005%
APT28 [26]	{1,2,2,4,4,7}	PA, WP, MFE, SID, DLS, EWS	1,203,013	1.175%
CobaltGroup [27]	{1,2,4}	PA, RAS, NSD	961,920	0.118%
Gamaredon [28]	{1,2,2,6,6,4,4,6,7}	PA, WP, MFE, MR, RK, WMI, SID, ST, DF	442,729	0.013%
Patchwork [29]	{1,2,3,4,4,4,6,5}	PA, PS, BUAC, DLS, UD, SD, RK, RDP	155,296	9.095%

PA = phishing Attachment, MFE = Malicious File Execution, RK = Registry Run Keys, SID = System Information Discovery, SNCD = System Network Configuration Discovery, MTOS = Masquerade Task or Service, ST = Scheduled Task, WP = Web Protocols, DLS = Data from Local System, EWS = Exfiltration Over Web Service, RAS = Remote Access Software, NSD = Network Service Discovery, MR = Modify Registry, WMI = Windows Management Instrumentation, DF = Defacement, PS = PowerShell, BUAC = Bypass User Account Control, UD = System Owner/User Discovery, SD = Security Software Discovery, RDP = Remote Desktop Protocol, PEI = Portable Executable Injection, SM = Shortcut Modification, DMT = Disable or Modify Tools, HW = Hidden Window. The subsequent number of a technique represents a distinct ability used to implement that technique [30].



- **Baseline: Sigma**
 - open and widely used signature format as fine-grained attack patterns

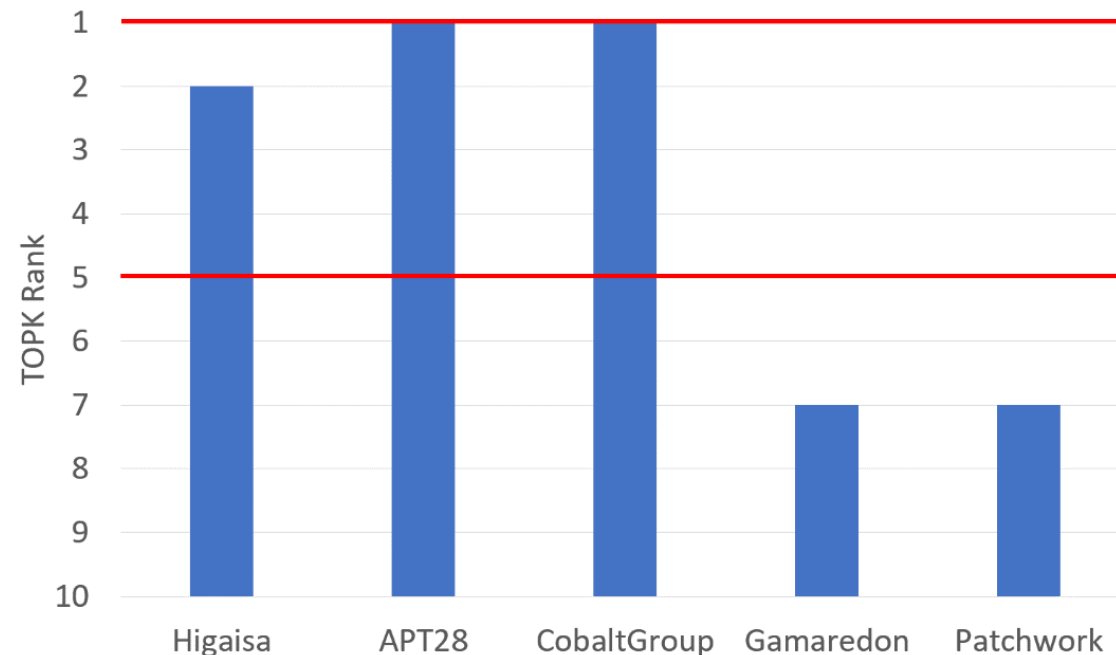
Evaluation on TTPs detection

	Sigma			SFM		
APT Campaign	P	R	F1	P	R	F1
Higaisa	33.37%	36.11%	33.40%	90.32%	90.48%	87.00%
APT28	0.00%	0.00%	0.00%	56.30%	62.45%	57.02%
CobaltGroup	0.28%	29.75%	0.54%	54.82%	72.31%	58.44%
Gamaredon	25.02%	17.08%	16.71%	73.51%	77.75%	73.21%
Patchwork	8.13%	21.96%	9.14%	68.60%	68.87%	67.55%
Avg.	13.36%	20.98%	11.96%	68.71%	74.37%	68.64%

- Our methodology exhibits substantial performance compared to Sigma.
 - Sigma rules, while designed by experts, only cover portions of attack behaviors, leaving numerous malicious activities undetected

Evaluation on APT campaign attribution performance

- **Top-1 ranking:** 40% correctly matched.
- **Top-5 ranking:** 60% correctly matched.
- **Implication:**
 - GED tolerates minor detection errors
 - It narrow down the pool of likely threat actors, even in real-world scenarios.



Conclusion

- This study presents a **machine learning-based SFM** for identifying potential APT threat actors.
- Results show SFM
 - detects over 60% of techniques successfully from system event logs
 - attributes APT campaigns to the correct threat group within the top 5 ranks in 60% of cases.
- These highlight SFM as a promising approach for APT detection and attribution, helping to narrow down likely threat actors in real-world scenarios.