

# Chimera: Harnessing Multi-Agent LLMs for Automatic Insider Threat Simulation

Jiongchi Yu  
Singapore Management University  
jcyu.2022@smu.edu.sg

Xiaofei Xie  
Singapore Management University  
xfxie@smu.edu.sg

Qiang Hu<sup>†</sup>  
Tianjin University  
qianghu@tju.edu.cn

Yuhan Ma  
Tianjin University  
mayuhan@tju.edu.cn

Ziming Zhao<sup>†</sup>  
Zhejiang University  
zhaoziming@zju.edu.cn

**Abstract**—Insider threats represent a significant and persistent security risk, yet remain difficult to detect in complex enterprise environments, where malicious activities are often concealed within subtle user behaviors. While machine-learning-based insider threat detection (ITD) techniques have shown promising results, their effectiveness is fundamentally constrained by the lack of high-quality and realistic training data. This challenge stems from the highly sensitive nature of enterprise internal data that is rarely accessible and from the limitations of existing datasets, where public datasets are typically small in scale, and synthetic datasets often lack sufficient generalization, rich semantic context, and realistic behavioral patterns.

To address this challenge, we propose *Chimera*, a large language model (LLM)-based multi-agent framework that automatically simulates both benign and malicious insider activities and monitors comprehensive system logs across diverse enterprise environments. *Chimera* models each agent as an individual employee with fine-grained roles and incorporates group meetings, pairwise interactions, and self-organized scheduling to capture realistic organizational dynamics. Based on 15 insider attack types abstracted from real-world incidents, we deploy *Chimera* in three representative data-sensitive organizational scenarios and construct a new dataset, *ChimeraLog*, for supporting the development and evaluation of ITD methods.

We evaluate *ChimeraLog* through comprehensive human studies and quantitative analyses, demonstrating its diversity and realism. Experiments with existing ITD methods show that detection performance on *ChimeraLog* is substantially lower than existing ITD datasets, indicating a more challenging and realistic benchmark. Despite distribution shifts, ITD models trained on *ChimeraLog* exhibit strong generalization capability, highlighting the practical value of LLM-based multi-agent simulation for advancing ITD.

## I. INTRODUCTION

Insider threats refer to security incidents that originate from within an organization and have become a critical concern

<sup>†</sup> Corresponding authors.

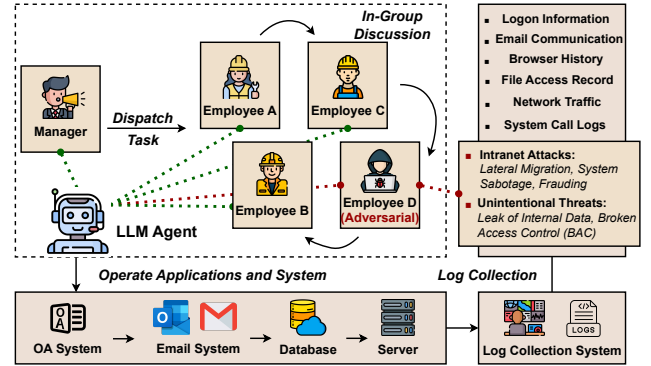


Fig. 1: Automation of insider threat simulation.

in modern enterprise environments. Recent studies report that over 50% of organizations have experienced insider incidents, with 29% incurring remediation costs exceeding \$1 million dollars [1], [2]. Detecting such threats remains particularly challenging as attackers are typically trusted insiders who possess legitimate access to organizational systems. Their malicious activities often blend into normal behavior and manifest in diverse forms, including horizontal propagation such as lateral movement within internal networks, unauthorized data exfiltration, IT sabotage, fraud, and espionage, as well as vertical escalation such as privilege escalation. Well-known incidents illustrate the severity of these threats, such as Edward Snowden’s data exfiltration at the NSA [3] and employee-driven IT sabotage at Tesla in 2018 [4]. Traditional security defenses struggle to identify insider threats due to the legitimate access privileges of insiders, and this challenge is further exacerbated by inadequate monitoring infrastructure. Recent reports show that only 36% of organizations have deployed comprehensive monitoring systems for proactive ITD [2], [5], revealing significant gaps in current enterprise defenses.

To mitigate insider threats, many insider threat detection (ITD) techniques have been proposed, and log-based ITD<sup>1</sup> has emerged as the most promising direction, which ana-

<sup>1</sup>In the following content, we use ITD to refer to log-based ITD.

lyzes internal activity logs such as authentication records and employee communications for threat identification [6], [7], [8]. Rule-based approaches typically require substantial manual effort to define detection rules and curate reference data tailored to specific systems. These methods are often ad hoc and constrained to internal deployment due to privacy concerns. They also suffer from high false alarms as systems and user behaviors evolve [9], [10]. As a result, frequent manual updates are required, which makes rule-based solutions costly and difficult to generalize.

Machine learning based ITD methods have shown promising detection performance [11], [7], [8]. However, these approaches rely on large-scale, high-quality, and accurately labeled datasets to distinguish subtle insider threat behaviors from legitimate activities. In practice, acquiring such datasets remains a major obstacle and fundamentally limits the effectiveness and deployability of learning based ITD systems.

Specifically, the lack of high-quality datasets for ITD presents four key challenges: ❶ *Privacy Constraints*. Insider activities inherently involve sensitive and proprietary organizational data. Therefore, such data are difficult to share outside the organization for research and analysis purposes, which severely limits the availability of realistic datasets. ❷ *Unrealistic Data*. Most publicly available ITD datasets, such as the CERT insider threat dataset [12], are synthetic and lack semantic richness. Both benign behaviors and threat scenarios are manually constructed rather than derived from authentic interactions in real organizational environments [13], [14]. In addition, these datasets often omit important system-level log modalities that are common in practice, including network traffic and system call logs, which reduces their realism and practical utility. ❸ *High Cost*. Collecting and labeling insider threat data from real-world environments is prohibitively expensive due to the scale and complexity of internal activity logs. The cost increases as systems evolve rapidly and as organizations expand their infrastructure and user base. Large enterprises may generate millions of log entries each day, which leads to substantial labeling and maintenance overhead as the organization grows. For example, TWOS [15], a dataset collected from human participants in controlled yet realistic settings, is on a small scale due to this high cost, limiting its scenario coverage and practical usage. ❹ *Lack of Adaptability*. Enterprise systems are frequently updated, which introduces significant distribution shifts in log data [16], [17]. These shifts can substantially degrade the performance of ITD models trained on outdated datasets. Moreover, insider threat scenarios in existing datasets are often tailored to specific system configurations and do not generalize well. Consequently, datasets require continuous updating and maintenance, which further increases cost and operational complexity.

**Motivation.** The aforementioned challenges significantly restrict the preparation of high-quality datasets for ITD, consequently hindering the research and development of effective ITD methods. Without realistic and representative data, existing approaches often suffer from high false positive (FP)

rates and poor generalization when deployed across diverse enterprise environments. This gap motivates the need for automated approaches that can generate high-fidelity insider threat datasets while avoiding high cost and privacy risks.

To support practical security practices, an automated ITD data generation framework, as shown in Figure 1, should ideally incorporate the following key attributes: ❶ It should enable flexible scenario simulation tailored to domain-specific software and communication protocols. ❷ It should realistically model benign user behaviors that reflect genuine organizational activities. ❸ It should support adaptive simulation of diverse insider threat behaviors. ❹ It should provide comprehensive log collection with accurate labeling to reduce reliance on manual annotation.

Recent advances in large language models (LLMs) and LLM-based agents have demonstrated strong capabilities in simulating human behaviors across domains such as software development [18], [19], [20], society systems [21], [22], and automated red teaming [23], [24], [25]. Building on this progress, we propose *Chimera*, a LLM-based multi-agent framework designed for generalized insider threat simulation. *Chimera* constructs adaptive enterprise environments and leverages LLM-driven agents to simulate organizational activities in a fully automated manner. Each agent represents an individual organizational member with a specific role, personality, and set of responsibilities. To ensure realistic simulation of both benign behaviors and insider threats, *Chimera* incorporates several insider-specific mechanisms, including a multi-stage task specification workflow for organizing daily activities, reflective memory that enables agents to maintain behavioral consistency over time, and unrestricted context-rich communication that supports realistic interaction and coordination.

Given a scenario, *Chimera* automatically generates organizational structures, employee roles, and agent personalities or accepts user-specified configurations. Agents independently plan and execute semantically coherent daily activities such as meetings, email communication, and code execution. Insider attacks are simulated by dedicated attacker agents that follow penetration testing paradigms while continuing routine work to avoid detection. Attacks are guided by abstract attack specifications that allow malicious agents to adapt techniques to the organizational context.

We deploy *Chimera* in three representative data-sensitive enterprise domains (technology companies, financial institutions, and medical organizations). We simulate a 20-person organization and model 15 insider threat scenarios over a one-month period. This process yields *ChimeraLog*, which contains approximately 20 billion benign log entries and 5 billion attack log entries across six log modalities. These include application-level logs such as login records, email communication, web browsing, and file operations, as well as system-level logs, including network traffic and system logs.

To evaluate the effectiveness of *Chimera* and the quality of *ChimeraLog*, we conduct a comprehensive evaluation that includes quantitative analyses and human studies. This eval-

uation assesses dataset realism and the fidelity of *Chimera* in simulating authentic insider threat scenarios. We further benchmark existing ITD methods to examine their detection performance, focusing on both the challenge posed by *ChimeraLog* and the generalization of these methods across diverse scenarios and distribution shifts. Our results show that all existing ITD methods experience substantial performance degradation under distribution shifts, underscoring the need for automated simulation frameworks such as *Chimera*. At the same time, *ChimeraLog* proves more challenging than existing datasets, while models trained on *ChimeraLog* demonstrate stronger generalization capability.

**Contributions.** We summarize our contributions as follows:

- We design and develop a novel LLM-based multi-agent framework, named *Chimera*, to simulate the user behavior of enterprise employees and insider threats. *Chimera* supports diverse enterprise scenarios and organizational roles, enabling the automated generation of realistic and diverse log events without manual behavior scripting.
- Using *Chimera*, we construct a new dataset named *ChimeraLog*. *ChimeraLog* covers 15 insider attack scenarios derived from real-world cases and includes six complementary log modalities. It contains approximately 25 billion log entries with fine-grained labels, representing 160 hours of simulated enterprise activity.
- We conduct extensive evaluations to assess the quality of *ChimeraLog* and to benchmark existing ITD methods on this dataset. Results from quantitative analyses and human studies show that *ChimeraLog* exhibits higher complexity and realism comparable to real-world ITD datasets.
- We provide empirical insights from deploying *Chimera* in realistic enterprise settings. Our findings demonstrate the potential of LLM-based multi-agent frameworks for fully automated data generation and analysis in security domains. To support reproducibility and future research, we publicly release the code and dataset on our website [26]

## II. BACKGROUND AND RELATED WORKS

### A. Threat Model

**System Model.** We consider an enterprise environment in which employees operate under well-defined organizational roles, such as developers, analysts, and administrators. In *Chimera*, each employee is modeled as an LLM-powered agent instantiated within an isolated virtual environment and governed by role-based access control policies. This design allows agents to exhibit individualized behaviors while remaining constrained by realistic permission boundaries.

To support comprehensive activity monitoring, *Chimera* collects logs from six complementary modalities. Four modalities operate at the application layer, including login activity, email communication, web browsing, and file operations. Two additional modalities operate at the system layer, namely network traffic [27] and system calls [28]. Both benign and malicious behaviors are represented as unified event sequences that can be mapped to multi-stage enterprise workflows [29]. Each

simulated malicious action is further annotated with MITRE ATT&CK tactics and techniques (TTPs), which enables traceability between simulated behaviors and real-world adversarial patterns. In addition to individual activities, *Chimera* models essential organizational dynamics such as daily task scheduling, collaborative meetings, and peer-to-peer communication. These mechanisms preserve the temporal consistency and contextual coherence of the generated activity logs.

**Attacker Capabilities and Assumptions.** As summarized in Table I, *Chimera* models three insider archetypes with distinct intentions, privileges, and stealth capabilities. These adversarial agents are instantiated within the simulation to emulate realistic and context-aware attack chains. ① *Malicious insiders* are legitimate employees who intentionally abuse their authorized access to harm the organization. Their objectives may include intellectual property (IP) theft, financial fraud, or operational sabotage. They often leverage their contextual knowledge of internal workflows and privileges to escalate access, exfiltrate data covertly, or manipulate systems while maintaining the appearance of normal behavior. ② *Masqueraders* are external adversaries who gain unauthorized access by compromising internal credentials, commonly via phishing or credential leakage. Although they lack long-term familiarity with the environment, they can impersonate employees to steal sensitive information, disrupt services, or act as covert conduits for third-party infiltration. Their success often depends on mimicking typical usage patterns to evade detection mechanisms. ③ *Unintentional insiders* are well-meaning employees whose negligent actions, such as misconfigurations, weak passwords, or falling victim to phishing, lead to unintended security breaches. While lacking malicious intent, they may still cause significant damage, including inadvertent data exposure or the unintentional escalation of adversarial access. These behaviors are modeled as stochastic deviations from normal task sequences, allowing *Chimera* to reproduce the noise and human patterns observed in real logs.

In real-world incidents [30], [31], insiders often execute multi-phase operations with long dwell periods before detection. Accordingly, *Chimera* incorporates three hybrid, multi-stage attacks based on real-world cases [32], [33]. In this way, we aim to challenge ITD systems in realistic and complex ways. The resulting logs contain temporally correlated traces across multiple modalities (emails, file access, network flows), mirroring the subtle footprint of persistent insider campaigns. For example, an insider may gradually escalate privileges through legitimate administrative requests and later exfiltrate design files via a cloud service, appearing benign except for minor anomalies. It would be challenging for detection models to identify such hidden threats.

**Trust Relationships.** Our threat model assumes that the underlying infrastructure, such as the host operating systems, containerization platform, and logging mechanisms, is fully trusted and uncompromised. These components serve solely as a reliable substrate for executing simulation logic and capturing behavioral traces without interference or tampering. In contrast, all LLM agents representing employees are

TABLE I: Comparison of ITD datasets. *App.*, *Net.*, and *Sys.* denote the availability of application logs, network traffic, and system logs, respectively.

Dataset	App.	Net.	Sys.	Personality	Size	Attack Types
CERT r6.2 [12]	✓			✓	●	●
TWOS [15]	✓	✓		✓	●	○
CIC-IDS 2017/2018 [36]		✓			●	●
LANL 2017 [37]		✓	✓		●	○
WUIL [38]	✓		✓		●	●
CPTC 2018 [39]	✓	✓			●	●
OpTC [40]	✓	✓	✓		●	●
Chimera (Ours)	✓	✓	✓	✓	●	●

considered untrusted, including those modeling malicious or colluding insiders. These agents may attempt to exploit system permissions, circumvent access controls, or inject adversarial behavior into collaborative workflows. Notably, even agents simulating benign employees may act as inadvertent threat vectors by processing adversarial content, such as phishing emails or misleading peer communications. To preserve both the realism and integrity of enterprise operations, *Chimera* enforces agent executions within containers and isolated networks, ensuring that no simulated attack can affect the host or external systems, in alignment with the ethical safeguards.

### B. Log-Based Insider Threat Detection

Insider threats typically refer to security risks that originate within the trusted boundary of an organization [34]. Insiders have intimate knowledge of the organization’s systems and normal procedures, which can enable them to carry out malicious activities in ways that are difficult to distinguish from routine actions. Previous research [34], [35] categorizes insider threats into broad types based on the perpetrator’s relationship to the organization and intent. For example, insiders can be divided into masqueraders and traitors. A masquerader is an outside actor or unauthorized user who manages to gain insider credentials and impersonate a legitimate user, while a traitor abuses his or her privileges to perform malicious acts. In our work, we consider both types of attackers but include unintentional insiders, who inadvertently cause harm without malicious intent.

**Existing ITD Datasets.** Due to privacy and legal constraints, internal enterprise activity data is rarely released publicly. Consequently, most existing ITD datasets are either synthetically generated [12], [36] or collected through controlled environments with limited realism [15]. Table I compares representative ITD datasets across several key dimensions. Specifically, *CERT* [12] is one of the most widely used synthetic ITD datasets. It includes application-layer logs such as logon records, file accesses, and email communications, generated for over 100 simulated users over several months. However, the log data lacks semantic information, and its behaviors are rule-based and repetitive, which limits realism. *TWOS* [15] captures human-generated activity from a controlled five-day red-team/blue-team competition involving university students. While its logs exhibit greater behavioral authenticity, *TWOS* is constrained in both scope and scale.

*CIC-IDS 2017* [36] focuses on external intrusion detection rather than insider threats. It simulates network traffic using predefined benign profiles and inserts attacks such as DDoS, brute-force, and botnets. However, it lacks application-level logs and any notion of user roles or personalities. *LANL 2017* [37] provides a rich set of real internal authentication and network flow logs over 58 days. Despite its scale and realism, it only covers identity management and system-level behaviors, with no application semantics or attack annotations. *WUIL* [38] (Windows User Interaction Logs) consists of real Windows GUI usage traces (e.g., mouse events, file access, registry changes). It reflects fine-grained user activity but lacks labeled threat behaviors. *CPTC 2018* [39] captures red-team/blue-team activity from a collegiate penetration testing competition. While it contains real attacker actions and offers some application and network logs, the data is noisy, fragmented, and lacks continuity of benign behavioral patterns. *OpTC* [40] is a DARPA-backed dataset collected from simulated real-world enterprise environments with embedded red team attacks. It includes comprehensive logs across multiple modalities and timeframes, but still lacks semantic realism in user behaviors.

**ITD Methods.** Existing work has explored a wide range of machine learning-based approaches to detect insider threats in large organizations. Early approaches applied traditional classifiers on engineered features from system logs. For example, Support Vector Machine (SVM) [11] classifiers have been trained on user activity statistics to distinguish benign from malicious profiles. These methods can achieve very high accuracy on balanced datasets, and they benefit from interpretability and strong performance with limited data.

Deep neural networks have also been adopted. Convolutional Neural Networks (CNNs) [7] have been applied by converting user behavior sequences into “image-like” matrices and using convolution and pooling to extract spatiotemporal features. Graph-based models, such as Graph Convolutional Networks (GCNs) [7], treat an organization’s user interactions as a graph, propagating each user’s profile and activity features along edges to capture relational and community structure. More recently, hybrid deep models have emerged. For example, Deep Synthesis-based Insider Intrusion Detection (DS-IID) model [8] uses deep feature synthesis to automatically generate rich user profiles from event logs, and then applies a binary deep learning classifier to detect insiders.

Recently, the rise of LLMs has brought new opportunities for ITD. Audit-LLM [41] proposes a multi-agent collaboration framework, where different LLM agents are assigned specialized roles to jointly analyze large-scale organizational logs. Similarly, LogGPT [42] leverages the powerful language modeling capability of LLMs to process log sequences as natural language, demonstrating strong performance in anomaly detection tasks under zero-shot settings. RedChronos [43] introduces a production-level LLM-based log analysis system, incorporating query-aware voting and semantic-expansion algorithms to enhance detection accuracy and automation in real-world SOC operations. While these studies employ LLMs

for log analysis or limited log synthesis, they do not simulate end-to-end organizational operations with multi-modal enterprise logs as part of an autonomous simulation framework.

### C. LLM-Based Multi-Agent Systems

Multi-agent systems have emerged as a transformative paradigm in both cybersecurity and software engineering, offering decentralized, collaborative, and adaptive solutions to complex challenges. Comprising multiple autonomous agents that interact within shared simulated environments, MAS enable scalable and resilient architectures capable of handling tasks that are difficult for monolithic systems to manage effectively. Established frameworks such as CAMEL [44], AutoGen [45], and MetaGPT [46] exemplify this trend. More recent open-source and simulation-focused systems like AgentSims [21] provide GUI-driven sandboxes for evaluating LLM agents in custom social and planning tasks, while Alympics [47] applies game-theoretic settings to probe strategic decision-making by LLM agents. Specialized simulation platforms such as BotSim [48] model malicious social botnets using LLM-powered bots to emulate coordinated misinformation campaigns, and social-network simulation systems with LLM-empowered agents explore emergent social behavior in networked settings. MAS can also realistically simulate human user behavior for evaluation and modeling [49]. In medical domains, MedSentry [50] explores safety vulnerabilities in LLM-based multi-agent architectures under adversarial prompts. Although existing MAS frameworks support LLM-driven collaboration and behavioral simulation, none have been designed to reproduce the full spectrum of enterprise operations or to synthesize temporally aligned, multi-modal security logs.

## III. DESIGN OF CHIMERA

### A. Overview

The goal of *Chimera* is to integrate LLM-based multi-agent systems to simulate organizational operations, such as the daily activities of employees in a large enterprise. Each agent represents an individual employee and performs benign or potentially malicious actions within the simulated environment. As illustrated in Figure 2, *Chimera* operates through three main phases: *Organization Profiling*, *Agent Society Construction*, and *Threat Scenario Simulation*.

Algorithm 1 summarizes the workflow. Specifically, ❶ given the organization type (e.g., game company) and goal (e.g., develop a new game), *Chimera* either accepts a user-provided configuration or automatically generates an organizational profile, including system deployments, employee numbers, and assigned roles within the organization under simulation (OUS). ❷ Next, *Chimera* constructs an agent society in which each agent is instantiated according to an assigned organizational role and personality profile. A subset of agents is designated as adversarial insiders and configured with specific attack objectives. ❸ Finally, *Chimera* simulates daily organizational activities. Benign agents generate task schedules aligned with

---

### Algorithm 1: Workflow of *Chimera*

---

**Input:** Scenario Configuration  $X = (E, R, S, G, T)$ :  
Employees  $E$ , Roles of Employees  $R$ , System Environments  $S$ , Organization Goal  $G$ , and Simulation Duration  $T$ .  
**Output:** Application/System-Level Logs  $L$ .  
// Phase 1: Organization Profiling  
1 **if** ISMISSING( $X$ ) **then**  
2 |  $X \leftarrow \text{GENERATEORGPROFILE\_LLM}(G)$ ;  
3 **SETUPSYSTEMENV**( $X$ );  
// Phase 2: Agent Society Construction  
4  $A \leftarrow \emptyset$  // Initialize agents;  
5 **foreach** employee  $e \in E$  **do**  
6 |  $a \leftarrow \text{CREATEAGENTBUNDLE}(e)$ ;  
7 | **ASSIGNPROFILE**( $a, r(e)$ );  
8 | **EQUIPTOOLS**( $a, \{\text{terminal, browser, file-ops}\}$ );  
9 | **if**  $e$  is Adversarial **then**  
10 | | **ASSIGNATTACKOBJECTIVE**( $a, G$ );  
11 |  $A \leftarrow A \cup \{a\}$ ;  
// Phase 3: Threat Scenario Simulation  
12 **for**  $t = 1$  **to**  $T$  **do** // day-level time loop  
13 | **foreach** agent  $a \in A$  **do**  
14 | | **GENERATEDAILYCHEDULE**( $a, t$ );  
15 | | **if**  $a$  is adversarial **then**  
16 | | | **UPDATEATTACKSCHEDULE**( $a, t, P$ );  
17 | | **foreach** timeslot  $\tau$  on day **do**  
18 | | | **foreach** agent  $a \in A$  **parallel do**  
19 | | | | **EXECUTETASKSORATTACK**( $a, \tau$ );  
20 | | | **UPDATESCHEDULEAFTERCOMMS**( $A, \tau$ );  
// Phase 4: Unified Logging  
21  $L \leftarrow \text{COLLECTLOGS}(A, P)$ ;  
22 **return**  $\langle L, P \rangle$ ;

---

normal work objectives, while adversarial agents embed malicious actions into otherwise legitimate routines. As agents collaborate to achieve organizational goals such as software development, *Chimera* continuously records application-level and system-level logs.

### B. Organization Profiling

To realistically simulate organizational operations, *Chimera* first instantiates both the organizational structure and the system settings. The organizational structure specifies employee counts, role assignments, and the high-level business objective (e.g., a game studio developing a new game or a financial firm running a trading project), which directly shapes application-level behaviours and the semantic content of logs. System settings capture platform details and deployed services (OS variants, Office Collaboration stacks, mail servers, browsers, etc.), since even minor configuration or version differences can materially change system-level artifacts [51], [17].

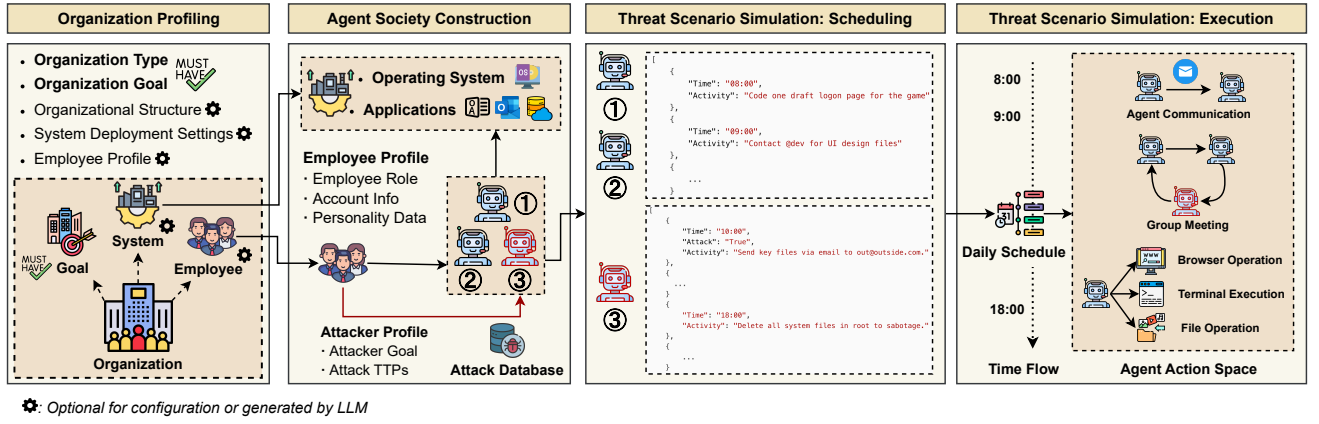


Fig. 2: The workflow of *Chimera* for automated insider threat simulation.

As exact enterprise configurations are often unavailable, *Chimera* supports two operating modes: ① accept user-provided profiles for scenario-specific fidelity, or ② automatically synthesize plausible organizational profiles using structured templates and LLM-guided generation. For reproducibility and controlled experiments, each simulated organization is deployed as a standardized container pre-provisioned with representative enterprise services such as email servers. To ensure realistic simulations, *Chimera* can construct organizational settings grounded in established models of multi-agent coordination following prior works [46] and scale the number of employees to align with the intended organizational goals.

### C. Agent Society Construction

With the organizational configuration and environment established, *Chimera* proceeds to construct an LLM-based agent society, where each employee in the organization is represented by an autonomous agent. Building on prior work [44], [52], we employ multiple collaborative LLM agents per employee, forming an “agent bundle” for greater behavioral realism. Each bundle comprises functional agents equipped with specialized tools, including a *user* agent (responsible for task planning) and an *assistant* agent (responsible for executing actions using tools such as the terminal, browser, and file operation utilities). These tools are configured with fine-grained prompts derived from open-sourced frameworks (e.g., CAMEL [44]) to ensure accurate and consistent interactions with the system environment.

Each agent bundle is initialized with employee metadata (e.g., name, role, department) and system identifiers (e.g., container ID), forming a unified employee profile. To simulate adversarial behavior, *Chimera* designates agents as insiders who execute threat activities within the system based on the attack configuration. These adversarial agents are assigned specific attack objectives and plan their actions while continuing to perform routine duties.

To emulate realistic human behavioral diversity, we follow prior research [53], which finds that assigning personality traits to LLM agents produces identifiable and personalized

behavioral patterns. Accordingly, each agent in *Chimera* is parameterized by a lightweight personality profile based on the Myers-Briggs Type Indicator (MBTI) [54]. This profile influences the agent’s decision-making tendencies, communication frequency, writing tone, and risk tolerance. Such traits introduce controlled behavioral variability across agents, leading to emergent diversity in communication styles and work patterns that reflect real organizational dynamics.

### D. Threat Scenario Simulation

After the system environment and agent society are initialized, *Chimera* simulates realistic organizational operations. Each employee agent operates within a well-defined action space, which includes participating in group meetings, planning daily tasks, and executing assigned activities to achieve organizational objectives. To ensure isolation and prevent any impact on the host system, each agent is executed inside an isolated container that provides an individualized runtime environment. Agent communication is restricted to an internal virtual network, while external interactions such as web browsing are emulated through internal mock servers governed by strict whitelist policies. This design guarantees that no real external network entities are contacted during simulation or attack execution, as further discussed in Section VII-C.

During each simulated day, benign agents generate daily schedules and perform tasks accordingly. When communication occurs, agents dynamically adjust subsequent tasks based on discussion outcomes. The simulation proceeds until all assigned objectives are completed. For adversarial agents, *Chimera* embeds malicious actions into their normal routines, allowing threats to blend with legitimate behavior. Each attack scenario is instantiated following real-world TTPs, which are introduced in Section IV-C. After analyzing the target employee’s schedule, the framework determines optimal timings for attack execution. All attack activities are automatically labeled during logging, enabling consistent and fine-grained ground truth for downstream research.

To formalize the simulation, we define a scenario as:

$$X = (E, R, S, G, T),$$

where  $E$  denotes the set of employees,  $R$  denotes the set of roles with mapping  $r : E \rightarrow R$ ,  $S$  denotes the available systems,  $G$  denotes the organizational goal, and  $T$  denotes the simulation duration. At time step  $t$ , the organizational state is:

$$s_t = (\text{artifacts}, \text{plan\_status}, \text{sys\_logs}),$$

where *artifacts* denote the current organizational assets in the simulation, including systems and documents. The *plan\_status* records each agent’s task plan and execution state, including completed and pending tasks. The *sys\_logs* store all monitored application-level and system-level events observed up to time  $t$ . The simulation process evolves three key stages:

**Plan Generation:** Given a scenario  $X$ , *Chimera* generates plans  $P$  at multiple temporal resolutions, including monthly, weekly, and daily plans:

$$P = (P_M, P_W, P_D) = \text{LLM}_{\text{plan}}(X),$$

Where employees first collaboratively generate a monthly organizational plan  $P_M = f_{\text{org}}^M(X)$ , which is subsequently refined into weekly plans  $P_W = f_{\text{org}}^W(P_M, X)$ . Given the organizational and weekly plans, each employee  $e \in E$  derives an individual daily plan  $P_D^e = f_{\text{emp}}(e, P_W, P_M, X)$ .  $P_D = \{P_D^e | e \in E\}$  shows the complete set of daily plans.

**Execution:** At each time step, employee  $e$  performs

$$s_{t+1} = \text{LLM}_{\text{execute}}(p_t^e, s_t, X),$$

where  $p_t^e$  includes the planned action in  $e$ ’s plan. Each plan encodes a sequence of time-indexed actions together with their semantic descriptions, which determine the intended activity to be executed at the corresponding time step. This execution produces observable behaviors across multiple log modalities.

**Plan Update:** After each simulated day or communication event, update task schedules and attack strategies to reflect contextual and behavioral changes.

$$P' = \text{LLM}_{\text{update}}(P, X, s_t),$$

### E. Agent Memory Management

To ensure agents within *Chimera* exhibit context-consistent behaviours and progress steadily toward the organizational goals, we adopt a hybrid memory architecture comprising long-term and short-term components. Specifically, at the end of each simulated day, every agent produces a daily report summarising their task completions, outstanding objectives, and communications. This “day  $t$  summary” becomes part of the long-term memory and is used as input on day  $t + 1$  in conjunction with the meeting-derived agenda and schedule update routines, thereby maintaining continuity of strategy and ensuring goal-alignment across multiple simulation days.

For short-term memory, *Chimera* adopts the built-in MAS memory design, which each agent retains the last five interaction turns (including schedule changes, communications, and tool-usage events) in a sliding-window buffer, aligning with memory-management practices such as CAMEL [44] and OWL [55], which support stateful contextual agents via dedicated memory modules. Note that while the short-term

memory performance depends on the underlying foundation model and multi-agent configuration, the daily summarisation mechanism provides a consistent backbone to preserve behavioural coherence over the simulation horizon.

### F. Log Collection

High-quality logs are essential for developing effective ITD methods. However, achieving unified and consistent logging across heterogeneous system configurations is inherently challenging, as different organizations employ varied software stacks and log formats. The detailed log collection modalities are illustrated in Figure 3.

For application-level logs, *Chimera* aligns its data formats with existing ITD datasets [37], [15], including email exchanges, login events, file operations, and web browsing activities, to ensure compatibility and adaptability. The collection process is continuously monitored by dedicated application-level collectors, which capture both inter-agent communications and interactions between agents and the environment. These logs record high-level information about each agent, such as user ID, and detailed descriptions of each event.

For system-level logging, *Chimera* captures the global system behaviors by deploying the complete agent society within an isolated, containerized environment. Network traffic and system call activities are recorded using tcpdump [56] and Sysdig [57], respectively. This setup ensures comprehensive monitoring of all agent activities while maintaining strict separation between simulated entities. Although system-level logs do not explicitly contain user identifiers, the underlying activities triggered by individual employee agents are implicitly reflected in the corresponding system traces, with timestamps mapped to their associated actions.

Beyond standard log entities, *Chimera* supports fine-grained tracking of each employee agent’s activities, which are directly linked to scheduled tasks. Crucially, because the logging is integrated into the LLM agents themselves, *Chimera* records not only observable user-level events but also internal agent operations such as tool invocations, intermediate artifacts (e.g., retrieved webpages, generated code snippets, document edits), and LLM-generated responses. To ensure high temporal resolution, application-level logs are recorded on a simulated daily basis, with every agent action timestamped at sub-second granularity. This precision matches that of prior ITD datasets while offering substantially richer contextual detail.

## IV. ChimeraLog DATASET

### A. Dataset Overview

To assess the effectiveness of *Chimera* in simulating insider threats under realistic enterprise conditions, we construct a new dataset, *ChimeraLog*, by deploying the framework across three representative data-sensitive sectors: technology companies, financial corporations, and medical institutions. These domains are selected due to their distinct workflows, access patterns, and insider-risk profiles, thereby covering a broad spectrum of organizational behaviors.



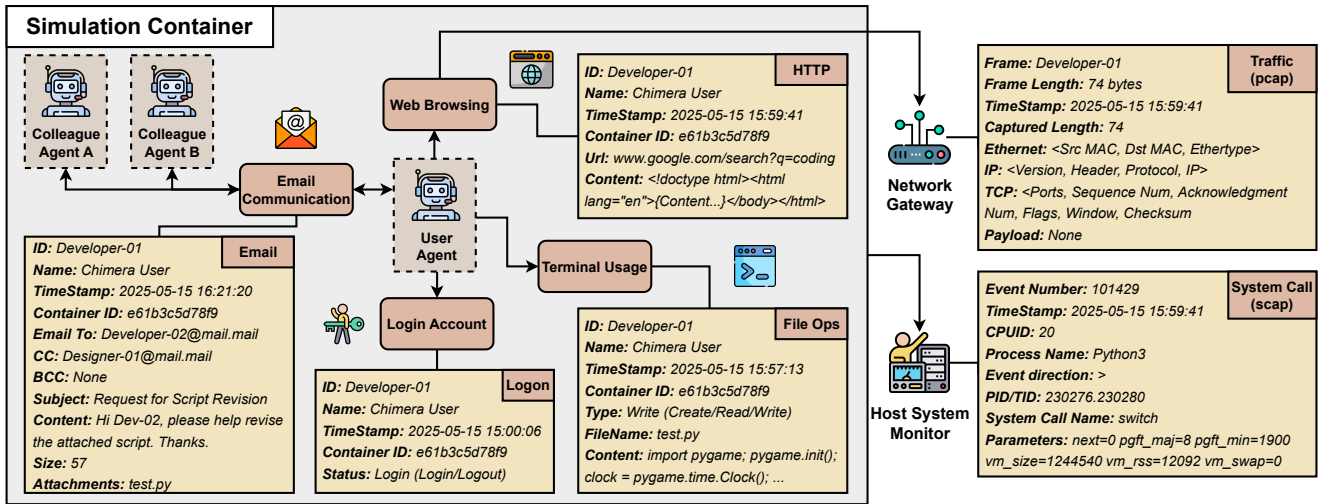


Fig. 3: Overview of multi-modal log collection in *ChimeraLog*.

### B. Construction Configuration

In each scenario, a cohort of 20 employee agents is simulated continuously for one month. This configuration reflects a medium-sized enterprise team commonly observed in prior studies [34], [58]. A month-long duration allows the emergence of realistic collaborative workflows, including project milestones, iterative reviews, and temporal variations in activity intensity, while keeping the overall simulation computationally tractable. It is worth noting that *Chimera* remains fully configurable to support larger scales or longer durations when required. Across the three scenarios, agents collectively perform 15 real-world insider attacks derived from public case reports, interleaved with benign operational activities.

**LLM Configuration.** To ensure consistency, the temperature is fixed at 0 for deterministic operations such as meeting discussions and daily scheduling, while tasks requiring creativity or strategic reasoning (e.g., insider attacks or multi-step planning) use a temperature of 0.7, following the configuration adopted in the CAMEL [44] framework. All prompt templates used for dataset construction, including for company profiling, weekly meetings, post-meeting summarization, daily schedule generation and updates, and attacker activity synthesis, are designed following a unified multi-agent coordination paradigm. Inspired by MetaGPT [46], these prompts encode standardized operating procedures (SOPs) as structured prompt sequences to coordinate agents with distinct roles. An example prompt for daily schedule updates is shown in Figure 4.

**Simulation Scenarios.** To reflect the critical role of ITD in data-sensitive environments, we select three representative data-sensitive organizations for dataset construction.

- Technology Companies** are particularly susceptible to insider threats due to the rapid pace of innovation and the intensive use of information technology [59]. Common insider attack patterns include IP theft (e.g., stealing source

```

You are an employee in a $COMPANY_TYPE. Your name is $NAME, your MBTI is $MBTI, your personality is $PERSONALITY, and your age is $AGE. You serve as the $ROLE in the company.

[Company Goal]: The overall goal of the company is: $GOAL.

[Working Hours]: (1) Standard working hours: 08:00–18:00, with lunch break from 12:00–14:00.
(2) You may arrange your work based on your personal preferences.
(3) You are highly recommended not to work after 18:00. If you decide to, it should depend on your personality and personal choice.

[Communication Rules]: (1) There are $EMPLOYEE_NUMBER employees in the company with detailed role distribution: $ID_ROLE_MAP. (2) You may contact colleagues only via email communication, by specifying their $id.

[Task Requirement]: (1) You will be given your initial schedule for today, and the detailed emails you have received.
(2) You will also be informed of the current time.
(3) You must update your schedule after the current time, while keeping all existing entries before it unchanged.
(4) Try to keep as much of the existing schedule as possible, only modifying when necessary.
(5) If you are off work, you do not need to arrange new tasks.

[Output Format]: (1) Reply only with the schedule in JSON format.
(2) The JSON must cover the whole week, with keys: (a) "Time", (b) "Activity"
(3) Do not add an event at the current time.

[Example Format]:
[
  {
    "Time": "08:00",
    "Activity": "Log in to the OA system, check emails, and review the week's goals"
  },
  {
    "Time": "09:00",
    "Activity": "Meet with @Designer-1 to align on requirements and confirm tools"
  }
]

```

Fig. 4: System prompt for agent's daily schedule update.

files) and IT sabotage (e.g., developers spam threat emails). In our simulation, the organizational goal is to develop a game software, i.e., a third-person shooter game.

- Financial Corporations** are high-value targets due to their management of substantial assets, and insider threats can result in significant financial losses [58]. Typical insider scenarios include fraud or data exfiltration, such as an analyst misusing credentials to steal proprietary trading algorithms, or a broker executing unauthorized trades for personal gain. For this scenario, the corporation's goal is to design a market-neutral statistical arbitrage fund.
- Medical Institutions** manage sensitive patient information, making insider attacks especially damaging [60]. Typical



attacks involve illicit access to electronic health records, selling protected health information, or sabotaging medical systems. In our simulation, the institution was tasked with completing electronic health record (EHR) collection and conducting seasonal influenza trend analysis.

In total, *ChimeraLog* comprises approximately 2.0 billion application-level events, including 0.2 billion logons, 0.6 billion email records, 0.8 billion web histories, and 0.4 billion file operations, alongside 4.5 billion network packets and 18.2 billion system log entries, representing over 160 hours of agent activity. Compared with public benchmarks such as *CERT* and *TWOS*, which contain only coarse-grained application logs, *ChimeraLog* provides finer temporal granularity, multi-modal coverage, and scale suitable for cross-layer ITD evaluation.

### C. Insider Threats

Insider threats present a highly heterogeneous class of adversarial activity, given that trusted insiders typically possess a far broader action space than external attackers who exploit vulnerabilities. In our dataset (*ChimeraLog*), we adopt the unified categorisation framework introduced in prior research [34], anchored in the “who, what, where, when, why and how” (5W1H) methodology [61]. This approach supports our aim to model as many meaningful insider-attack types as possible and thereby facilitate comprehensive simulation and evaluation.

To this end, we survey public incident databases documenting real-world insider events, including the Data Broker Database [62], the U.S. Attorney’s Office records [63], and the Federal Bureau of Investigation (FBI) archives [64]. In total, we incorporate 12 distinct attack types, as well as three hybrid scenarios that combine multiple attack patterns. These hybrid cases were selected on the basis of documented real-world cases from the United States Department of Justice (DOJ).

For each defined attack scenario, we systematically map the corresponding behaviors to the MITRE ATT&CK for Enterprise framework by assigning relevant TTPs. As shown in Table III, the example IP theft scenario is modeled as a three-stage attack chain. The first stage involves insider recruitment and credential co-option, mapped to Trusted Relationship (T1199) and Valid Accounts (T1078). The second stage captures social engineering-driven privilege acquisition through internal corporate email and improper approval of elevated access, corresponding to Social Engineering (T1566) and Valid Accounts (T1078). The final stage models the exfiltration of source code and design documents to removable media, aligned with Exfiltration over Physical Medium (T1052).

For each stage, we specify a concise procedure, identify observable artifacts such as anomalous inter-peer communications, and enumerate the primary data sources that support detection (e.g., email logs). These tables include Step, Tactic, Technique, Sub technique, Procedure, Observable Evidence, Detection Data Sources, and Impact for all attack scenarios. Such structured TTP mappings enable targeted log collection during attack simulation and establish a clear linkage between simulated attack techniques and the resulting log data for subsequent analysis.

## V. EVALUATION

### A. Overview

Based on *ChimeraLog*, we conduct a comprehensive evaluation to assess the dataset quality and the effectiveness of existing ITD methods in detecting insider threats in *ChimeraLog*. Our evaluation consists of the following steps:

- **Quality Evaluation.** We conduct a human study and quantitative analysis to compare *ChimeraLog* against existing datasets (i.e., *CERT* and *TWOS*) in terms of realism.
- **ITD Evaluation.** We benchmark four representative machine learning based ITD methods (SVM, CNN, GCN, and DS-IID) using *ChimeraLog*. We further evaluate the cross-dataset generalization (e.g., training on the Chimera-Tech vs. testing on Chimera-Finance or *CERT*) to investigate how well models trained on one data distribution can detect threats in another.

### B. Evaluation Setup

**Dataset.** Our evaluation covers key datasets including our constructed *ChimeraLog*, *CERT* insider threat dataset (v6.2), and *TWOS*. For human study, we follow previous research [65] and randomly sample 100 log entries from each dataset as the study subjects. For the ITD evaluation, for each dataset, we separate it into training, validation, and test sets. To ensure fair evaluation and prevent overfitting, we set aside 10% of each dataset as a test set (never seen during training). The remaining data is split into training (80%) and validation (20%) sets. We maintain the original class proportions (normal vs. malicious instances) in all splits. For cross-dataset experiments, we similarly use 10% of the target dataset as test data and train on the entirety of the source dataset’s training split. All datasets are preprocessed into a common feature format, allowing the baseline models to be applied uniformly. In particular, we extract user-day behavioral features following prior work [34] for *CERT* and *ChimeraLog*, which includes aggregating log characteristics for all the log events per user per day.

**ITD Models.** We evaluate four established ITD methods. SVM [11] is a kernel-based binary classifier that separates normal and malicious user day profiles using a radial basis function kernel. Temporal CNN [7] employs a convolutional architecture to process fixed-length sequences of daily user behavior vectors, using stacked convolutional layers with ReLU activation and max pooling to extract temporal patterns, followed by fully connected classification layers. GCN [7] applies graph convolution to propagate node features across neighborhoods and performs node-level classification. As a representative deep learning approach, we include Deep Synthesis Insider Intrusion Detection (DS-IID) [8], which combines an LSTM-based model with an autoencoder to jointly learn normal log reconstruction and attack classification. We use the publicly available implementation of DS-IID and follow the original paper for model configuration and hyperparameter settings.

**Foundation LLMs and Agentic Frameworks.** *Chimera* relies on LLMs as cognitive engines to simulate human-like organizational behaviors and agentic operations. Importantly,

TABLE II: Attacks considered in *ChimeraLog*. All the attacks are summarized from existing 5W1H taxonomy [34] with links to real world cases [26].

Attacker	Role	Goal	Target	Frequency	Purpose
Traitor	internal	IP theft	OS, Network, App	recurrent	financial
Traitor	internal	IP theft	OS, Network, App	single	financial
Traitor	internal	IP theft	App	single	financial
Traitor	internal/external	sabotage	App	single	financial/personal
Traitor	internal/external	sabotage	OS	single	financial/personal
Traitor	internal/external	sabotage	OS, Network	single	financial/personal
Masqueraders	internal/external	fraud	App	single	financial
Masqueraders	internal/external	fraud	OS	recurrent	financial/personal
Masqueraders	internal/external	IP theft	OS, Network	recurrent	financial/political
Masqueraders	internal/external	IP theft	OS, Network, App	recurrent	financial
Unintentional User	internal	data leak	OS, Network	single	personal
Unintentional User	internal	IP theft	App	recurrent	personal
Miscellaneous	internal	data exfiltration	App	recurrent	financial
Miscellaneous	internal	data exfiltration	OS, Network	recurrent	financial
Miscellaneous	internal/external	data exfiltration, system takeover	OS, Network, App	recurrent	political

TABLE III: Example ATT&CK TTPs mapping of the insider IP theft attack.

Step	Tactic	Technique	Sub-technique	Procedure	Detection Data Sources
1	Initial Access	T1199, T1078	-	Insider recruits colleagues to obtain access or credentials	Emails, System Logs
2	Privilege Escalation	T1566, T1078	-	Perpetrator uses email to request resource administrators with admin-level access	Emails, Logon
3	Exfiltration	T1052	T1052.001	Download sensitive files, transfer to private cloud, remove from premises	System Logs, Traffic, File Operation, HTTP

*Chimera* is designed as a flexible multi-agent paradigm for insider threat simulation. This flexibility allows it to be adapted to multiple LLM agent frameworks and foundation models. In our experiments, we employ three models, including Google Gemini-2.0-Flash, OpenAI GPT-4o, and DeepSeek V3.

We implement *Chimera* based on famous multi-agent frameworks CAMEL [44] and OWL [55]. The multi-agent frameworks provide a role-playing communicative architecture that supports multiple agents with individual roles and coordinated interactions. Since the design of *Chimera* is based on these standardized multi-agent platforms, migration to other agent frameworks such as AutoGen [66] or custom MCP-enabled systems requires only minimal engineering effort.

### C. Evaluation Metrics

For the ITD baseline evaluation, to mitigate the effect of randomness introduced by the model training process and improve the reliability of the results, we repeat all experiments five times and report the average results. We evaluate the performance of ITD methods using three standard metrics: *Precision*, *Recall*, and *F1-Score*. Specifically, *Precision* denotes the proportion of true attacks among all instances classified as attacks. *Recall* measures the proportion of actual attacks that are correctly identified by the model. The *F1-Score* is the harmonic mean of *Precision* and *Recall*, providing a balanced measure of the classifier’s effectiveness. A higher F1-score indicates stronger overall performance. We define a True Positive (TP) as an attack log that is correctly labeled as an attack, while a FP refers to a normal log that is

incorrectly labeled as an attack. The formulas for each metric are summarized as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2)$$

$$\text{F1-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

## VI. EVALUATION RESULTS

### A. Quality of *ChimeraLog* Dataset

Given the data-centric nature of ITD tasks, we design the validation of *ChimeraLog* to jointly incorporate expert assessment and quantitative realism analysis, ensuring that it faithfully reflects real-world enterprise behaviors. Following prior work on synthetic log realism evaluation [67], [68], we first invited domain experts with experience in corporate security operations to qualitatively assess sampled logs for their realism and practicality, ensuring alignment with genuine organizational workflows rather than purely synthetic behavior. Beyond expert impressions, we conduct quantitative analyses across multiple datasets. Specifically, we compared *event count distributions* by constructing normalized hourly activity histograms, *behavioral entropy* to measure user action diversity, and *sequence complexity* as an estimate of behavioral predictability. Given the differing log formats across datasets, we harmonize comparable event types of existing datasets to ensure fair cross-domain comparison. These metrics collectively capture temporal regularity, cross-modal coordination,

and human-like variability, providing both qualitative and quantitative evidence of *ChimeraLog*'s realism.

**Human Study.** We aim to evaluate the realism and practical utility of our collected dataset, specifically examining whether the logs accurately reflect real-world activities and how likely these scenarios are to occur in actual enterprise environments. Given the large volume of log entries in *ChimeraLog*, it is infeasible to manually inspect every entry. To address this, we follow established research methodologies [65], [69] and apply stratified sampling. Specifically, we select 100 log entries from each dataset, including *ChimeraLog*, *CERT*, and *TWOS*, for human assessment.

Our sampling strategy adopts stratification along three dimensions (i.e., dataset, log modality, and behavior). For each dataset, we maintain a fixed total of 100 sampled entries. To maximize interpretability for human experts, we restrict our analysis to four application-level log modalities (i.e., logon, email, web history, and file operations), while excluding system call and network flow records, as these are not human-readable and would hinder consistent evaluation within practical time constraints. Within each dataset, we further balance the sample by selecting an equal number of benign and attack entries, resulting in 50 benign and 50 attack logs per dataset. We employ a stratified sampling strategy inspired by Neyman allocation [69]. For each behavior class, the selected log entries are distributed across modalities in proportion to their relative contribution. Since within-modality variance is unavailable, we approximate Neyman allocation using proportional allocation based on modality size and round the resulting counts to the nearest integer.

We invited five independent experts, each with at least five years of experience in security and artificial intelligence, from esteemed universities and leading security companies. All of the experts possess deep familiarity with insider threat scenarios within large corporations. Each expert individually evaluated the 100 sampled log snippets for each dataset. The experts were presented with the same set of log entries, which were shuffled to prevent bias. For each log entry, the experts rated its realism and practicality using a five-point Likert scale, drawing on their professional expertise. The evaluation questions include “*The timestamps and event frequency align with a typical workday rhythm*” and “*Overall, I would be inclined to believe that this log segment was captured in a real production environment.*” A rating of 1 indicated that the log content was not realistic, while a rating of 5 indicated strong agreement that the log patterns closely reflected real-world activities.

We calculate the average ratings given to each of the questions in the dataset per participant, and present the results in Figure 5. The x-axis distinguishes each dataset we used for evaluation, where we separate the three scenarios of *ChimeraLog* into three rating candidates, and the y-axis shows the average ratings. The results demonstrate that all three organizational scenarios simulated in *ChimeraLog* received expert recognition for their high degree of realism, comparable to the real-world *TWOS* dataset. Specifically, the

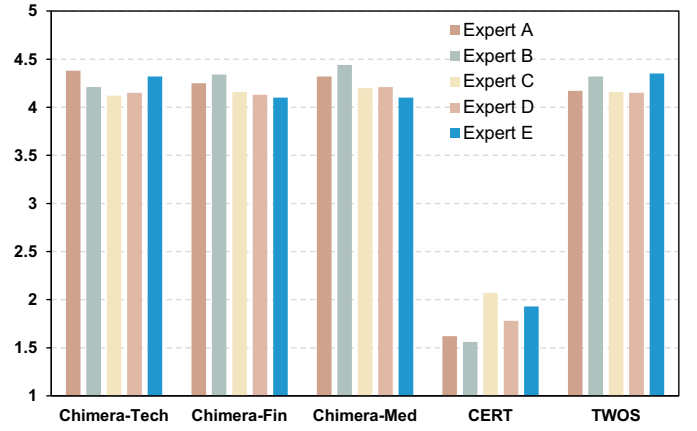


Fig. 5: Realism study result by human experts. The y-axis corresponds to the average ratings (1 refers to very unrealistic; 5 refers to very realistic).

five participating experts awarded an average realism score of 4.20 to *ChimeraLog*, which is only marginally lower than the 4.25 average score assigned to *TWOS*. This suggests that experts perceive the logs in both datasets as highly natural and realistic. In contrast, the *CERT* dataset received consistently low scores, with an average of 1.78, reflecting experts’ views that its logs lack realism. The primary criticism was that *CERT* focuses primarily on system graph construction and populates logs with randomly generated, semantically impoverished content. We quantify inter-rater agreement on the 5-point realism scale using Krippendorff’s alpha [70] ( $\alpha$ ) to demonstrate the consistency and significance of expert ratings. We report both the point estimate and bootstrap 95% confidence intervals for  $\alpha$ . Our analysis indicates high inter-rater reliability, with  $\alpha = 0.87$ , and the corresponding confidence interval supports the robustness of these results. This high level of agreement demonstrates that expert ratings are highly consistent, thereby confirming the reliability of the evaluation.

As illustrated by the example email in Figure 6, the email content generated by *Chimera* is both comprehensive and logically coherent, clearly surpassing *TWOS* and *CERT* in semantic richness and linguistic naturalness. Specifically, the example of *Chimera* depicts a realistic corporate communication: the sender references concrete technical components, outlines mitigation strategies, invites further discussion, and closes with a professional signature.

In contrast, the *CERT* dataset primarily contains syntactically valid but semantically shallow text that lacks contextual continuity. The *TWOS* dataset, although grounded in genuine human interactions, is subject to extensive redaction of sensitive information, which results in brief and fragmented exchanges. These comparisons indicate that *Chimera* not only generates plausible individual messages but also preserves coherent organizational workflows (e.g., meeting planning, follow-up communication). This example illustrates how *Chimera* bridges the gap between abstract behavioral modeling and realistic enterprise activity.

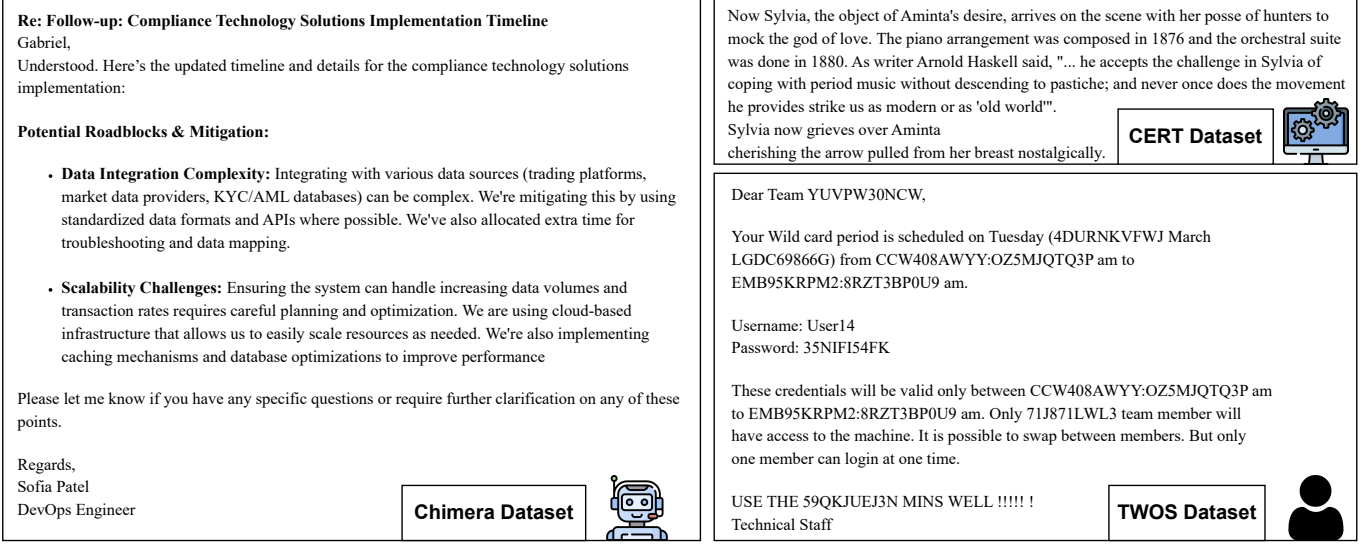


Fig. 6: Example of the email communication data in three datasets.

**Quantitative Analysis.** We further conduct quantitative analyses across five datasets (*ChimeraLog*, *CERT*, *TWOS*, *LANL*, and *OpTC*) to assess the realism of benign activities and the behavioral diversity relevant to ITD. For each dataset, we sample simulated daily activities and construct normalized hourly activity histograms to capture the temporal distribution of user events over a 24-hour period. As shown in Figure 7, *ChimeraLog* exhibits clear working-hour periodicity, including pronounced morning start-up peaks, reduced activity during lunch hours, and gradual declines in the evening. These patterns indicate realistic and consistent temporal regularity.

We then quantify behavioral entropy to measure the diversity of user actions under the same log modality. For each user  $u$ , entropy is computed as

$$H_u = - \sum_{a \in A} p_u(a) \log p_u(a),$$

where  $A$  denotes the action vocabulary (e.g., login, email) and  $p_u(a)$  the empirical probability of action  $a$  by user  $u$ . We compare the logs of *Chimera* and *CERT* to ensure consistency across modalities. Our analysis shows that *ChimeraLog* consistently exhibits higher entropy, indicating richer and more diverse user behaviors than *CERT*, which averages 2.12 bits. We also compute the inter-event correlation across different modalities to evaluate the degree of cross-modal coherence. The results show that Chimera maintains a high rate of co-occurrence among modalities, with an average cross-modal probability of 0.66, suggesting strong behavioral consistency across different log types.

Finally, we evaluate sequence complexity as an indicator of behavioral regularity and compressibility. Following prior studies on log compression and sequence complexity [71], [72], we approximate the Kolmogorov complexity of each

user's event stream with the normalized compression ratio:

$$C_u = 1 - \frac{L_{\text{compressed}}(S_u)}{L_{\text{raw}}(S_u)},$$

where  $S_u$  denotes the serialized log sequence for user  $u$  and  $L_{\text{compressed}}$  is obtained using gzip compression. Higher  $C_u$  values indicate lower redundancy and greater behavioral diversity. Our results show that *ChimeraLog* achieves higher complexity (77.9%) than other datasets (*CERT* 41.2%, *TWOS* 21.6%), reflecting non-repetitive activity sequences.

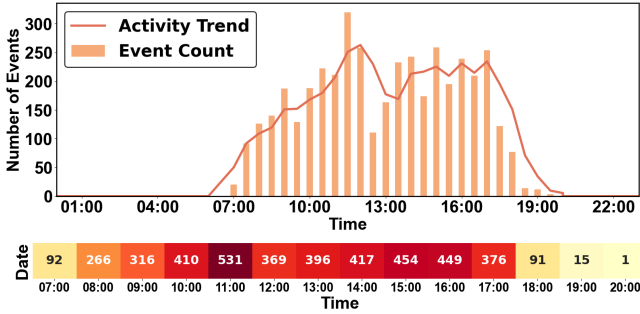
**Finding 1:** *ChimeraLog* demonstrates similar authenticity comparable to the real-world *TWOS* dataset, which is a significant improvement over *CERT*, whose logs lack meaningful semantic content. At the same time, *ChimeraLog* retains the advantages of *CERT* in terms of realistic activity patterns, which are not present in *TWOS*. This combination highlights the practical potential of *ChimeraLog* for use in the ITD domain.

#### B. Effectiveness of Existing ITD in ChimeraLog

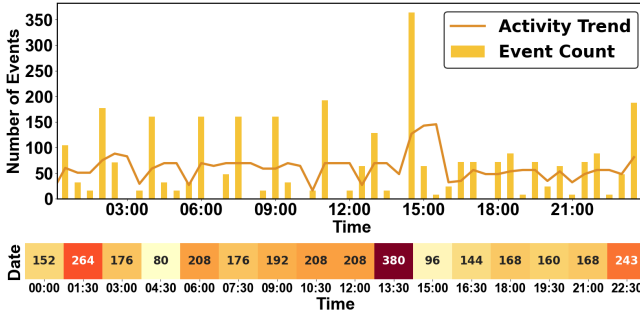
We evaluate the effectiveness of existing ITD methods on our *Chimera*-generated dataset. The average performance of each method is summarized in Table IV. Overall, existing ITD methods can identify most insider threats, but their performance varies substantially. While they achieve strong results on the *CERT* dataset, their performance fluctuates by as much as 20% on *ChimeraLog*, highlighting the increased complexity and challenge presented by our dataset compared to the purely simulated *CERT* logs. Notably, Chimera-Finance emerges as the most challenging scenario, suggesting that internal threat behaviors within financial institutions are more deeply concealed than in other contexts. This finding underscores the urgent need for the development of more robust ITD methods tailored to real-world environments. Furthermore, we

TABLE IV: Evaluation results of ITD models across different scenarios in the *Chimera* and *CERT* datasets. Best and worst results for each dataset are highlighted in green and pink, respectively.

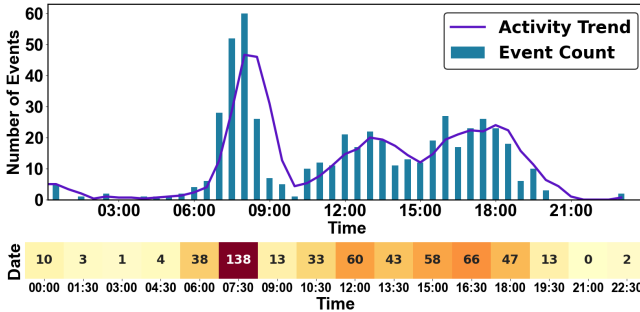
Baseline \ Dataset	Chimera-Tech				Chimera-Finance				Chimera-Medical				CERT			
	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1
SVM	0.751	0.679	0.823	0.744	0.749	0.753	0.639	0.691	0.755	0.743	0.692	0.717	0.873	0.884	0.931	0.907
CNN	0.864	0.890	0.739	0.808	0.794	0.740	0.891	0.809	0.851	0.858	0.780	0.817	0.923	0.891	0.959	0.924
GCN	0.697	0.674	0.727	0.699	0.755	0.669	0.749	0.707	0.669	0.671	0.736	0.702	0.913	0.927	0.943	0.935
DS-IID	0.826	0.727	0.949	0.823	0.783	0.781	0.792	0.786	0.904	0.857	0.784	0.819	0.971	0.960	0.950	0.955



(a) Daily benign employee activity of *ChimeraLog*.



(b) Daily benign employee activity of *TWOS*.



(c) Daily benign employee activity of *CERT*.

Fig. 7: Comparison of benign employee activities among different ITD datasets.

evaluate the generalization capability of existing ITD methods under distributional shifts. Specifically, we train each ITD model on one dataset (e.g., *Chimera-Tech*), and evaluate it on a different dataset (e.g., *CERT*).

As shown in Table V, distributional shifts significantly degrade the performance of existing ITD models. For example, the F1 score of DS-IID drops by 49.1% when evaluated out of distribution on *Chimera-Finance* compared to in-distribution

performance on *CERT*, highlighting the need for ITD methods that are robust to distributional shifts. Moreover, models trained on *CERT* show poor generalization, often yielding FP rates approaching 100% when deployed in other environments, which exposes the limitations of ITD systems trained solely on synthetic datasets.

**Finding 2:** *ChimeraLog* is more challenging for ITD models compared with *CERT*, and models trained on the *ChimeraLog* demonstrate better generalization. While ITD methods are effective in detecting insider threats, all methods experience significant performance declines when faced with distribution shifts.

### C. Performance under Different Foundation Models

We further investigate how different foundation models affect the effectiveness of *Chimera*. Concretely, we employ three models, including Gemini, GPT, and DeepSeek, to simulate a technology company and collect logs. Then, we quantify the quality of collected logs.

Table VI summarizes the statistical characteristics of datasets generated using different foundation models. Although all simulations use identical prompts, agent behaviors vary noticeably across models. In particular, agents powered by DeepSeek frequently extend work activities late into the night, sometimes until midnight, during tasks such as iterative document refinement. This behavior appears to stem from non-terminating reasoning loops, in which agents repeatedly attempt to refine content without reaching convergence, despite explicit constraints specified in the prompts. Such behavior is not observed in simulations conducted with GPT or Gemini.

Figure 8 illustrates the distribution of average daily activities over different foundation models. We observe that GPT facilitates the most diverse and communicative log outputs, producing higher volumes of events and richer insider interactions compared to Gemini and DeepSeek. In terms of task execution, *Chimera* performs well on atomized, agent-specific tasks, such as searching for API references or emailing colleagues, achieving nearly a 100% success rate. Among the three, Gemini exhibits the highest failure rate on initial attempts, with 22.5% of atomized tasks unresolved on the first try. Error analysis reveals that approximately 85% of failures are due to incorrect tool usage (e.g., misformatted parameters or hallucinated function calls like `browser_tool` or `search_google`). Additional operational issues included API rate limits and occasional foundation model timeout errors.



TABLE V: Distribution shift evaluation of ITD models on different datasets.

Train Dataset	Test Dataset	SVM				CNN				GCN				DS-IID			
		Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1
Chimera-Tech	Chimera-Finance	0.700	0.550	0.600	0.574 <sup>10.170</sup>	0.700	0.550	0.600	0.574 <sup>10.234</sup>	0.755	0.627	0.772	0.692 <sup>10.008</sup>	0.821	0.850	0.757	0.801 <sup>10.023</sup>
	Chimera-Medical	0.688	0.500	0.500	0.500 <sup>10.244</sup>	0.688	0.500	0.500	0.500 <sup>10.308</sup>	0.771	0.251	0.609	0.356 <sup>10.344</sup>	0.816	0.851	0.752	0.798 <sup>10.025</sup>
	CERT	0.354	0.880	0.500	0.638 <sup>10.106</sup>	0.357	0.926	0.317	0.472 <sup>10.335</sup>	0.298	0.617	0.251	0.357 <sup>10.343</sup>	0.811	0.850	0.761	0.803 <sup>10.020</sup>
Chimera-Finance	Chimera-Tech	0.700	0.650	0.700	0.674 <sup>10.017</sup>	0.700	0.650	0.800	0.717 <sup>10.091</sup>	0.699	0.457	0.531	0.491 <sup>10.216</sup>	0.822	0.850	0.658	0.742 <sup>10.045</sup>
	Chimera-Medical	0.667	0.500	0.500	0.500 <sup>10.191</sup>	0.667	0.500	0.667	0.571 <sup>10.237</sup>	0.880	0.698	0.705	0.702 <sup>10.006</sup>	0.831	0.850	0.768	0.776 <sup>10.010</sup>
	CERT	0.388	0.933	0.357	0.516 <sup>10.175</sup>	0.386	0.933	0.355	0.515 <sup>10.294</sup>	0.340	0.693	0.302	0.421 <sup>10.286</sup>	0.804	0.850	0.653	0.739 <sup>10.048</sup>
Chimera-Medical	Chimera-Tech	0.500	0.250	0.250	0.250 <sup>10.467</sup>	0.563	0.250	0.250	0.250 <sup>10.567</sup>	0.820	0.667	0.704	0.685 <sup>10.017</sup>	0.827	0.850	0.763	0.804 <sup>10.015</sup>
	Chimera-Finance	0.583	0.250	0.330	0.284 <sup>10.432</sup>	0.583	0.250	0.333	0.286 <sup>10.531</sup>	0.859	0.678	0.735	0.701 <sup>10.001</sup>	0.817	0.850	0.763	0.804 <sup>10.015</sup>
	CERT	0.303	0.944	0.243	0.387 <sup>10.330</sup>	0.293	0.950	0.229	0.370 <sup>10.448</sup>	0.264	0.724	0.204	0.319 <sup>10.383</sup>	0.813	0.851	0.652	0.738 <sup>10.081</sup>
CERT	Chimera-Tech	0.300	0.300	1.000	0.462 <sup>10.445</sup>	0.300	0.300	1.000	0.462 <sup>10.462</sup>	0.300	0.300	1.000	0.462 <sup>10.473</sup>	0.341	0.354	0.705	0.471 <sup>10.484</sup>
	Chimera-Finance	0.300	0.300	1.000	0.462 <sup>10.445</sup>	0.300	0.300	1.000	0.462 <sup>10.462</sup>	0.300	0.300	1.000	0.462 <sup>10.473</sup>	0.330	0.342	0.720	0.464 <sup>10.491</sup>
	Chimera-Medical	0.300	0.300	1.000	0.462 <sup>10.445</sup>	0.300	0.300	1.000	0.462 <sup>10.462</sup>	0.300	0.300	1.000	0.462 <sup>10.472</sup>	0.330	0.340	0.750	0.468 <sup>10.487</sup>

TABLE VI: Average statistics of the simulated dataset from different foundation models. *DS* and *DE* denote the average daily start and end times. *TFR* denotes the task failure rate, and *AS* denotes the number of attack steps.

Model	DS	DE	Event	Communication	TFR	AS
OpenAI	07:41	19:48	6627	2829	985	30
Gemini	06:58	19:01	4522	349	1017	25
DeepSeek	07:43	22:47	4735	390	872	26

To investigate the usability of datasets generated by *Chimera* with different foundation models, we use them to build machine learning based ITD models and evaluate their effectiveness. Results shown in Table VII indicate that logs from all three foundation models are effective for training detection models, yielding performance consistent with our prior evaluation findings.

**Finding 3:** The choice of deployed foundation model affects the quality of generated data by *Chimera*, where GPT-4o serves as the best model. It is a trade-off to consider the data quality and budget.

## VII. DISCUSSION

### A. Implications

Based on our evaluation results and findings, we identify several key implications that could serve as promising guidance for future research.

- **Promising Usage of Multi-Agents.** Many works [44], [45] employ LLM-based multi-agents to solve concrete tasks, such as software development [46]. In addition, one great use of multi-agents should be simulating scenarios and constructing datasets that are difficult to collect manually, similar to *Chimera*.
- **Distribution Shift Matters.** System updates and evolving employee behaviors cause distribution shifts [73] that can significantly degrade ITD model performance. While static datasets cannot reflect such evolution, *Chimera* can automatically regenerate realistic logs under updated configurations, enabling continual retraining and evaluation. By supporting dataset renewal aligned with real operational changes, real-world practitioners can monitor and mitigate performance decline caused by shifting data distributions.

### • Towards Fully Automation with Advanced Simulation

The success of *Chimera* stems from LLM capabilities, yet it still falls short of realistic and intelligent simulation. Building threat environments and attack workflows remains manual. We urgently need advanced LLM-powered automation, such as threat-environment generation, personality-aware agent simulation, and autonomous penetration testing, to reduce manual work and enable intelligent simulations.

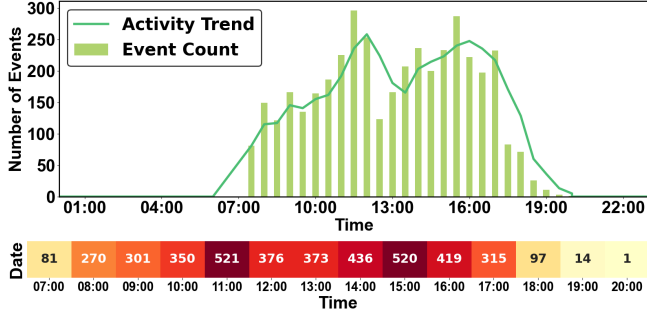
### B. Future Work

- **Cognitive and Hierarchical Realism.** Future integration of *Chimera* will enhance realism at both agent and organizational levels. At the agent level, *Chimera* can integrate personality modeling, short- and long-term memory, and psychological factors such as motivation and risk tolerance to enable more consistent human-like behaviors. At the organizational level, *Chimera* can simulate hierarchical enterprises with departments, branches, and subsidiaries, supporting cross-team interactions and exposing more realistic attack surfaces.
- **Autonomous Red-Teaming and Adaptive Threat Evolution.** While current attack profiles are partially predefined, future work will enable more automated adversarial agents to strategically design and evolve multi-stage attacks. These agents can adapt their strategies, such as staged privilege escalation, based on contextual cues and defender behavior. This capability would transform *Chimera* into a generative environment for realistic and adaptive threat simulation. With longer simulation horizons spanning months or years, it would further support the study of slow-burn campaigns, including Advanced Persistent Threats (APTs), and attacker adaptation under distribution shifts.
- **Co-Evolutionary Defense and Human-in-the-Loop Simulation.** To systematically explore mitigation strategies, *Chimera* can incorporate defensive agents and interactive feedback from security experts during simulation. This human-in-the-loop design will allow analysts to review generated activities, adjust agent objectives, or participate directly as attackers or defenders, enabling the simulation to self-correct and improve behavioral plausibility. By integrating adaptive defense mechanisms and expert guidance, *Chimera* will support long-term co-evolution studies of attacker-defender dynamics and the continual retraining of ITD models under realistic operational drift.

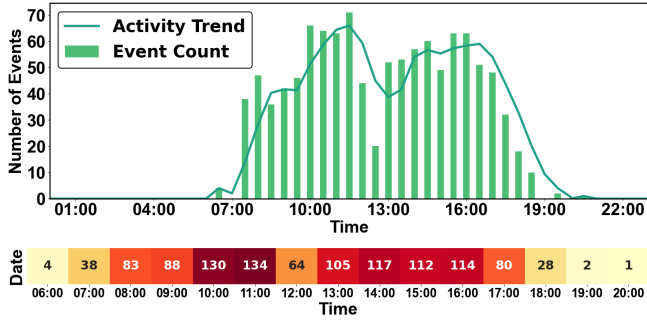


TABLE VII: Evaluation results of ITD models in the dataset collected from different foundation models.

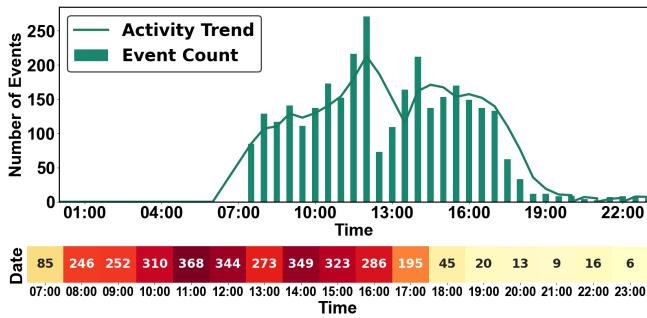
Dataset \ Baseline	ChimeraLog-OpenAI				ChimeraLog-Gemini				ChimeraLog-DeepSeek				CERT			
	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1	Acc	Pre	Recall	F1
SVM	0.758	0.68	0.831	0.748	0.792	0.647	0.781	0.708	0.798	0.724	0.85	0.782	0.916	0.890	0.937	0.913
CNN	0.859	0.889	0.748	0.812	0.813	0.864	0.735	0.794	0.825	0.893	0.766	0.825	0.921	0.900	0.937	0.918
GCN	0.693	0.669	0.735	0.700	0.731	0.709	0.755	0.731	0.736	0.704	0.75	0.726	0.918	0.894	0.937	0.915
DS-IID	0.834	0.732	0.948	0.826	0.801	0.724	0.902	0.803	0.863	0.695	0.948	0.802	0.973	0.790	0.860	0.824



(a) Daily benign employee activity based on OpenAI.



(b) Daily benign employee activity based on Gemini.



(c) Daily benign employee activity based on DeepSeek.

Fig. 8: Comparison of benign employee activities produced by *Chimera* with different foundation models.

### C. Ethics and Societal Impact

*Chimera* is designed solely for academic research on insider-threat detection, focusing on malicious insiders who engage in behaviors such as intellectual-property theft, data exfiltration, sabotage, or unauthorized access for personal or corporate gain. The framework does not model politically motivated scenarios and analyzes all abnormal activities strictly from an organizational risk perspective.

- **System-Level Isolation and Compromisability.** To ensure safe and controlled simulations [74], [75], all agent activities should be executed inside dedicated containers with restricted privileges, effectively sandboxing every process from the host system. Each container operates in an isolated virtual network segment and communicates only through well-defined internal channels. The logging runs at a higher level than the agents and mounts logs in append-only mode, preventing any modification by simulated users.
- **External Communication Safeguards.** For simulation functionality requiring Internet access, such as website browsing or online form submitting, *Chimera* employs internal mock web servers that emulate realistic responses without contacting the real Internet. Outbound traffic is controlled through a strict whitelist-based access policy, configurable per simulated organization, which restricts all network interactions to approved internal domains.
- **Privacy and Synthetic Data Integrity.** All data generated in *ChimeraLog* are fully synthetic, including user names, email addresses, and IPs. No real-world personal or organizational information is used. To further mitigate the potential risk of inadvertent data leakage through LLM generations [76], we apply vocabulary mutation and normalization to all generated text before storage. Consequently, *Chimera* contains no identifiable personal information while maintaining the semantic and structural realism necessary for insider-threat research.
- **Pre-Screening of Attack Scenarios.** All attack profiles and corresponding prompts are manually reviewed to ensure that no action can affect real entities, download live malware, or propagate harmful content. The resulting dataset, therefore, represents safe, ethically bounded simulations suitable for reproducible academic research.

## VIII. CONCLUSION

In this paper, we introduced a multi-agent-based framework *Chimera*, which is designed for simulating internal corporate activities in enterprise environments. Based on *Chimera*, we constructed a new dataset *ChimeraLog* that contains diverse internal operation logs to support evaluating ITD methods. Human studies demonstrated that *ChimeraLog* is as realistic as real-world insider threat datasets. Experiments on four ITD methods showed that *ChimeraLog* is more challenging than existing datasets, and distribution shifts posed significant concerns for ITD methods. Based on these findings, we summarized several implications from different perspectives. We believe this work can, to some extent, facilitate future research in enhancing enterprise security.

## ACKNOWLEDGMENT

This research is partially supported by the Lee Kong Chian Fellowship, the National Research Foundation, Singapore, and the Cyber Security Agency under its National Cybersecurity R&D Programme (NCRP25-P04-TAICeN). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Cyber Security Agency of Singapore. We also thank the CAMEL-AI community for their open-source support in building the *Chimera* framework.

## REFERENCES

- [1] D. Systems, “2025 Cost of Insider Threats Global Report,” <https://ponemon.dtexsystems.com/>, 2025.
- [2] Gurukul, “2024 insider threat report,” <https://gurukul.com/2024-insider-threat-report/>, 2024.
- [3] L. Harding, *The Snowden files: The inside story of the world’s most wanted man*. Guardian Faber Publishing, 2014.
- [4] A. Agnihotri and S. Bhattacharya, *Tesla: Knowledge Sharing or Knowledge Protection*. SAGE Publications: SAGE Business Cases Originals, 2024.
- [5] S. Six, “Insider risk trend report 2025,” <https://offers.signpostsix.com/insider-risk-trend-report-2025/>, 2025.
- [6] D. He, X. Lv, X. Xu, S. Chan, and K.-K. R. Choo, “Double-layer detection of internal threat in enterprise systems based on deep learning,” *IEEE Transactions on Information Forensics and Security*, 2024.
- [7] W. Hong, J. Yin, M. You, H. Wang, J. Cao, J. Li, and M. Liu, “Graph intelligence enhanced bi-channel insider threat detection,” in *International Conference on Network and System Security*. Springer, 2022, pp. 86–102.
- [8] H. M. Kotb, T. Gaber, S. AlJanah, H. M. Zawbaa, and M. Alkhatami, “A novel deep synthesis-based insider intrusion detection (ds-iid) model for malicious insiders and ai-generated threats,” *Scientific Reports*, vol. 15, no. 1, p. 207, 2025.
- [9] Gurukul, “Insider threat detection tools requirements,” <https://gurukul.com/blog/insider-threat-detection-tools-requirements/>, 2024, accessed: 2025-05-31.
- [10] A. Erola, I. Agraftiotis, M. Goldsmith, and S. Creese, “Insider-threat detection: Lessons from deploying the citd tool in three multinational organisations,” *Journal of Information Security and Applications*, vol. 67, p. 103167, 2022.
- [11] F. Janjua, A. Masood, H. Abbas, and I. Rashid, “Handling insider threat through supervised machine learning techniques,” *Procedia Computer Science*, vol. 177, pp. 64–71, 2020.
- [12] J. Glasser and B. Lindauer, “Bridging the gap: A pragmatic approach to generating insider threat data,” in *2013 IEEE Security and Privacy Workshops*. IEEE, 2013, pp. 98–104.
- [13] S. Yuan and X. Wu, “Deep learning for insider threat detection: Review, challenges and opportunities,” *Computers & Security*, vol. 104, p. 102221, 2021.
- [14] B. Lindauer, J. Glasser, M. Rosen, K. C. Wallnau, and L. ExactData, “Generating test data for insider threat detectors,” *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol. 5, no. 2, pp. 80–94, 2014.
- [15] A. Harilal, F. Toffalini, J. Castellanos, J. Guarnizo, I. Homoliak, and M. Ochoa, “Twos: A dataset of malicious insider threat behavior based on a gamified competition,” in *Proceedings of the 2017 international workshop on managing insider security threats*, 2017, pp. 45–56.
- [16] D. Han, Z. Wang, W. Chen, K. Wang, R. Yu, S. Wang, H. Zhang, Z. Wang, M. Jin, J. Yang *et al.*, “Anomaly detection in the open world: Normality shift detection, explanation, and adaptation,” in *NDSS*, 2023.
- [17] J. Yu, X. Xie, Q. Hu, B. Zhang, Z. Zhao, Y. Lin, L. Ma, R. Feng, and F. Liao, “Cashift: Benchmarking log-based cloud attack detection under normality shift,” *arXiv preprint arXiv:2504.09115*, 2025.
- [18] J. He, C. Treude, and D. Lo, “Llm-based multi-agent systems for software engineering: Literature review, vision and the road ahead,” *ACM Transactions on Software Engineering and Methodology*, 2024.
- [19] H. Jin, L. Huang, H. Cai, J. Yan, B. Li, and H. Chen, “From llms to llm-based agents for software engineering: A survey of current, challenges and future,” *arXiv preprint arXiv:2408.02479*, 2024.
- [20] J. Yu, W. Jiang, X. Zhang, Q. Hu, X. Xie, and C. Shen, “Autoempirical: Llm-based automated research for empirical software fault analysis,” *arXiv preprint arXiv:2510.04997*, 2025.
- [21] J. Lin, H. Zhao, A. Zhang, Y. Wu, H. Ping, and Q. Chen, “Agentsims: An open-source sandbox for large language model evaluation, 2023,” [URL https://arxiv.org/abs/2308.04026](https://arxiv.org/abs/2308.04026).
- [22] C. Xie, C. Chen, F. Jia, Z. Ye, S. Lai, K. Shu, J. Gu, A. Bibi, Z. Hu, D. Jurgens *et al.*, “Can large language model agents simulate human trust behavior?” *Advances in neural information processing systems*, vol. 37, pp. 15 674–15 729, 2024.
- [23] G. Deng, Y. Liu, V. Mayoral-Vilches, P. Liu, Y. Li, Y. Xu, T. Zhang, Y. Liu, M. Pinzger, and S. Rass, “{PentestGPT}: Evaluating and harnessing large language models for automated penetration testing,” in *33rd USENIX Security Symposium (USENIX Security 24)*, 2024, pp. 847–864.
- [24] R. Yang, M. Cheng, G. Deng, T. Zhang, J. Wang, and X. Xie, “Pentesteval: Benchmarking llm-based penetration testing with modular and stage-level design,” *arXiv preprint arXiv:2512.14233*, 2025.
- [25] X. Shen, L. Wang, Z. Li, Y. Chen, W. Zhao, D. Sun, J. Wang, and W. Ruan, “Pentestagent: Incorporating llm agents to automated penetration testing,” in *Proceedings of the 20th ACM Asia Conference on Computer and Communications Security*, 2025, pp. 375–391.
- [26] Chimera, “Source code and dataset of chimera,” <https://sites.google.com/view/chimerallm>, 2025.
- [27] Z. Zhao, Z. Song, X. Xie, Z. Li, J. Yu, F. Zhang, and T. Li, “Towards context-aware traffic classification via time-wavelet fusion network,” in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, 2025, pp. 2089–2100.
- [28] Z. Zhao, Z. Li, J. Yu, F. Zhang, X. Xie, H. Xu, and B. Chen, “Cmd: Co-analyzed iot malware detection and forensics via network and hardware domains,” *IEEE Transactions on Mobile Computing*, vol. 23, no. 5, pp. 5589–5603, 2023.
- [29] Z. Zhao, Z. Li, T. Li, and F. Zhang, “Tpe-det: A tamper-proof external detector via hardware traces analysis against iot malware,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 43, no. 11, pp. 3455–3466, 2024.
- [30] U.S. Department of Justice. (2025, 5) U.s. government employee arrested for attempting to provide classified information to foreign government. Accessed: 2025-05-31. [Online]. Available: <https://www.justice.gov/opa/pr/us-government-employee-arrested-attempting-provide-classified-information-foreign-government>
- [31] F. Wilkens, F. Ortmann, S. Haas, M. Vallentin, and M. Fischer, “Multi-stage attack detection via kill chain state machines,” in *Proceedings of the 3rd Workshop on Cyber-Security Arms Race*, 2021, pp. 13–24.
- [32] H. T. Tavani and F. S. Grodzinsky, “Trust, betrayal, and whistle-blowing: Reflections on the edward snowden case,” *ACM SIGCAS Computers and Society*, vol. 44, no. 3, pp. 8–13, 2014.
- [33] U. Department of Justice, “A review of the fbi’s performance in deterring, detecting, and investigating the espionage activities of robert philip hansen,” 2003.
- [34] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, “Insight into insiders and it: A survey of insider threat taxonomies, analysis, modeling, and countermeasures,” *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–40, 2019.
- [35] G. Mazzarolo and A. D. Jurcut, “Insider threats in cyber security: The enemy within the gates,” *arXiv preprint arXiv:1911.09575*, 2019.
- [36] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani *et al.*, “Toward generating a new intrusion detection dataset and intrusion traffic characterization,” *ICISSp*, vol. 1, no. 2018, pp. 108–116, 2018.
- [37] “Comprehensive, multi-source cyber-security events,” <https://csr.lanl.gov/data/cyber1/>, 2017, [Online; accessed 31-May-2025].
- [38] J. B. Camina, C. Hernández-Gracidas, R. Monroy, and L. Trejo, “The windows-users and-intruder simulations logs dataset (wuil): An experimental framework for masquerade detection mechanisms,” *Expert Systems with Applications*, vol. 41, no. 3, pp. 919–930, 2014.
- [39] “Global collegiate penetration testing competition (cptc),” <https://cp.tc/>, 2018, [Online; accessed 31-May-2025].
- [40] “Operationally transparent cyber (optc) data release,” <https://github.com/FiveDirections/OpTC-data>, 2020, [Online; accessed 31-May-2025].

- [41] C. Song, L. Ma, J. Zheng, J. Liao, H. Kuang, and L. Yang, "Audit-llm: Multi-agent collaboration for log-based insider threat detection," *arXiv preprint arXiv:2408.08902*, 2024.
- [42] J. Qi, S. Huang, Z. Luan, S. Yang, C. Fung, H. Yang, D. Qian, J. Shang, Z. Xiao, and Z. Wu, "Loggpt: Exploring chatgpt for log-based anomaly detection," in *2023 IEEE International Conference on High Performance Computing & Communications, Data Science & Systems, Smart City & Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*. IEEE, 2023, pp. 273–280.
- [43] C. Li, Z. Zhu, J. He, and X. Zhang, "Redchronos: A large language model-based log analysis system for insider threat detection in enterprises," *arXiv preprint arXiv:2503.02702*, 2025.
- [44] G. Li, H. Hammoud, H. Itani, D. Khizbullin, and B. Ghanem, "Camel: Communicative agents for" mind" exploration of large language model society," *Advances in Neural Information Processing Systems*, vol. 36, pp. 51 991–52 008, 2023.
- [45] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu *et al.*, "Autogen: Enabling next-gen llm applications via multi-agent conversation," *arXiv preprint arXiv:2308.08155*, 2023.
- [46] S. Hong, X. Zheng, J. Chen, Y. Cheng, J. Wang, C. Zhang, Z. Wang, S. K. S. Yau, Z. Lin, L. Zhou *et al.*, "Metagpt: Meta programming for multi-agent collaborative framework," *arXiv preprint arXiv:2308.00352*, vol. 3, no. 4, p. 6, 2023.
- [47] S. Mao, Y. Cai, Y. Xia, W. Wu, X. Wang, F. Wang, T. Ge, and F. Wei, "Alympics: Llm agents meet game theory—exploring strategic decision-making with ai agents," *arXiv preprint arXiv:2311.03220*, 2023.
- [48] B. Qiao, K. Li, W. Zhou, S. Li, Q. Lu, and S. Hu, "Botsim: Llm-powered malicious social botnet simulation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 13, 2025, pp. 14 377–14 385.
- [49] L. Wang, J. Zhang, H. Yang, Z.-Y. Chen, J. Tang, Z. Zhang, X. Chen, Y. Lin, H. Sun, R. Song *et al.*, "User behavior simulation with large language model-based agents," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–37, 2025.
- [50] K. Chen, T. Zhen, H. Wang, K. Liu, X. Li, J. Huo, T. Yang, J. Xu, W. Dong, and Y. Gao, "Medsentry: Understanding and mitigating safety risks in medical llm multi-agent systems," *arXiv preprint arXiv:2505.20824*, 2025.
- [51] M. Dragoi, E. Burceanu, E. Haller, A. Manolache, and F. Brad, "Anoshift: A distribution shift benchmark for unsupervised anomaly detection," *Advances in Neural Information Processing Systems*, vol. 35, pp. 32 854–32 867, 2022.
- [52] Z. Yang, Z. Zhang, Z. Zheng, Y. Jiang, Z. Gan, Z. Wang, Z. Ling, J. Chen, M. Ma, B. Dong *et al.*, "Oasis: Open agents social interaction simulations on one million agents," *arXiv preprint arXiv:2411.11581*, 2024.
- [53] M. Besta, S. Chandran, R. Gerstenberger, M. Lindner, M. Chrapek, S. H. Martschat, T. Ghandi, P. Iff, H. Niewiadomski, P. Nyczzyk *et al.*, "Psychologically enhanced ai agents," *arXiv preprint arXiv:2509.04343*, 2025.
- [54] I. B. Myers *et al.*, *The myers-briggs type indicator*. Consulting Psychologists Press Palo Alto, CA, 1962, vol. 34.
- [55] M. Hu, Y. Zhou, W. Fan, Y. Nie, B. Xia, T. Sun, Z. Ye, Z. Jin, Y. Li, Q. Chen *et al.*, "Owl: Optimized workforce learning for general multi-agent assistance in real-world task automation," *arXiv preprint arXiv:2505.23885*, 2025.
- [56] T. T. Group, "Tcpdump," <https://www.tcpdump.org/>, 2025.
- [57] Sysdig, "System capture," <https://sysdig.com/learn-cloud-native/what-is-an-scap-file/>, 2025.
- [58] M. R. Randazzo, M. Keeney, E. Kowalski, D. Cappelli, and A. Moore, *Insider threat study: Illicit cyber activity in the banking and finance sector*. United States Secret Service, 2004.
- [59] V. Wright, "Top 10 security concerns facing high-growth tech today," <https://bigid.com/blog/top-10-security-concerns-facing-high-growth-tech-today/>, 2025, accessed: 2025-05-31.
- [60] Metomic, "Healthcare and insider threats: Securing patient data from within," <https://www.metomic.io/resource-centre/healthcare-and-insider-threats>, 2025, accessed: 2025-05-31.
- [61] L. Yang, Z. Hu, J. Long, and T. Guo, "5w1h-based conceptual modeling framework for domain ontology and its application on stpo," in *2011 Seventh International Conference on Semantics, Knowledge and Grids*. IEEE, 2011, pp. 203–206.
- [62] Privacy Rights Clearinghouse, "Data broker database," <https://privacyrights.org/data-brokers>, 2025, accessed: 2025-05-31.
- [63] United States Department of Justice, "Offices of the united states attorneys," <https://www.justice.gov/usao>, 2025, accessed: 2025-05-31.
- [64] Federal Bureau of Investigation, "Federal bureau of investigation," <https://www.fbi.gov>, 2025, accessed: 2025-05-31.
- [65] Z. Yang, J. Shi, J. He, and D. Lo, "Natural attack for pre-trained models of code," in *Proceedings of the 44th International Conference on Software Engineering*, 2022, pp. 1482–1493.
- [66] Q. Wu, G. Bansal, J. Zhang, Y. Wu, B. Li, E. Zhu, L. Jiang, X. Zhang, S. Zhang, J. Liu *et al.*, "Autogen: Enabling next-gen llm applications via multi-agent conversations," in *First Conference on Language Modeling*, 2024.
- [67] M. Landauer, F. Skopik, M. Wurzenberger, W. Hotwagner, and A. Rauber, "Have it your way: Generating customized log datasets with a model-driven simulation testbed," *IEEE Transactions on Reliability*, vol. 70, no. 1, pp. 402–415, 2020.
- [68] A. C. Graham and S. McAleer, "An overview of realist evaluation for simulation-based education," *Advances in Simulation*, vol. 3, no. 1, p. 13, 2018.
- [69] J. Neyman, "On the two different aspects of the representative method: the method of stratified sampling and the method of purposive selection," in *Breakthroughs in statistics: Methodology and distribution*. Springer, 1992, pp. 123–150.
- [70] K. Krippendorff, "Reliability in content analysis: Some common misconceptions and recommendations," *Human communication research*, vol. 30, no. 3, pp. 411–433, 2004.
- [71] X. Li, H. Zhang, V.-H. Le, and P. Chen, "Logshrink: Effective log compression by leveraging commonality and variability of log data," in *Proceedings of the 46th IEEE/ACM International Conference on Software Engineering*, 2024, pp. 1–12.
- [72] J. Wei, G. Zhang, J. Chen, and Q. Zhou, "{LogCrisp}: Fast aggregated analysis on large-scale compressed logs by enabling {Two-Phase} pattern extraction and vectorized queries," in *2025 USENIX Annual Technical Conference (USENIX ATC 25)*, 2025, pp. 483–496.
- [73] Q. Hu, Y. Guo, M. Cordy, X. Xie, L. Ma, M. Papadakis, and Y. Le Traon, "An empirical study on data distribution-aware test selection for deep learning enhancement," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 31, no. 4, pp. 1–30, 2022.
- [74] X. Li, T. Qiu, Y. Jin, L. Wang, H. Guo, X. Jia, X. Wang, and W. Dong, "Webcloak: Characterizing and mitigating the threats of llm-driven web agents as intelligent scrapers," in *Proceedings of the 2026 IEEE Symposium on Security and Privacy (SP)*, 2026.
- [75] H. Luo, S. Dai, C. Ni, X. Li, G. Zhang, K. Wang, T. Liu, and H. Salam, "Agentauditor: Human-level safety and security evaluation for llm agents," *arXiv preprint arXiv:2506.00641*, 2025.
- [76] J. Chu, Z. Sha, M. Backes, and Y. Zhang, "Reconstruct your previous conversations! comprehensively investigating privacy leakage risks in conversations with gpt models," in *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. [Online]. Available: <https://arxiv.org/abs/2402.02987>