Industry White Book

AI-Driven Investigation Framework Based on Behaviour Chain Semantics

Let Ai Investigate, Let Human Decide

June 2025

Author: Qimin Zhao

Version: V1.0

Date: June 2025

Patent Disclosure and Confidentiality Notice

Contents

I.Executive Summary

The proposed model, named the **Semantic Chain Security Model (SCSM)**, integrates:

1.Natural language understanding (LLM/NLP)

2.Structured behavior chain modeling

3.AI-based semantic reasoning

4.Micro-model anomaly scoring

5.Human-machine collaboration

6.Knowledge feedback and evolution mechanisms

This model transforms isolated logs into structured behavior chains, allowing AI to interpret and investigate attack paths in ways previously only possible for experienced human analysts.

At the core of the model is the belief that AI cannot investigate what it cannot understand—and without structured behavior sequences, AI lacks the semantic substrate required for reasoning.

The Key Innovations of this model Include:

1.Natural Language Interfaces

Analysts use English, Chinese, or German to initiate investigations—no more SPL or KQL.

2.Democratized Threat Investigation

Anyone who can describe a suspicious behavior can launch an AI-powered inquiry.

3.Behavior Chain Construction

Logs are chronologically and semantically transformed into coherent behavioral narratives for each actor (IP, host, user).

4.Semantic Reasoning Engine

AI doesn't just match patterns—it infers intent, detects anomalies in context, and identifies pivotal transitions.

5.Pivot Point Detection (BPP)

Novel concept of pivot strength identifies critical semantic transitions (e.g., from reconnaissance to exploitation), even when actions appear "normal."

6.Expert Feedback + Knowledge Write-Back

Human-validated behavior chains are written back into a dynamic knowledge base, enabling memory, comparison, and cumulative learning.

7.Micro-Level Anomaly Models

Pluggable, role-specific models analyze single logs for character usage, API paths, time-of-day anomalies, and more.

8.Three-Field Mapping for Accurate Attribution

Introduces external_ip → srcip → dstip mapping to resolve actor ambiguity in edge device logs (e.g., WAF).

9.Investigative Layer Standardization Proposal

Recommends the formal inclusion of an “ Investigative Layer ” into security standards (SIEM/SOC/XDR), bridging the gap between alert generation and incident response.

II.Problem Statement and Problem Definition

A.Structural Gaps in the Current Industry Framework

1.Existing Security Operations Investigation Architecture:

Mainstream security platforms (such as SIEM, XDR, EDR) generally follow the operational workflow:

Log Collection → Anomaly **Detection** (via Rules/Models) → **Alert** Generation → **Manual Response**

2.Core Issues:

An alert is **not an investigation** result—it is merely a detection outcome.

A response is not an analytical judgment—it is often a default action.

The system lacks the ability to connect behavior fragments into a coherent "attack narrative"—analysts are forced to manually piece everything together.

B.The Missing Investigative Layer: A Semantic Blind Spot in the Industry

Table1：Global Standardization Landscape

Security Domain	Standardization Status	Representative Standards
Prevention	Mature / Established	NIST CSF, ISO/IEC 27001
Detection	Mature / Established	MITRE ATT&CK, NIST SP 800-94
Investigation	Absent — No globally unified architectural standard	(Gap — no established global framework)
Response	Partially Standardized	NIST SP 800-61 (Computer Security Incident Handling Guide)

1.No Behavior Chain Structure → Logs Are Fragmented

Each log entry is treated as an isolated "point"—there is no structural connection to form a "line" or "chain".

There is no structured perspective to answer:

Who performed the behavior? When did it happen? What actions were taken?

2.No Structural Language → No Standardized Modeling

Without an Entity Behavior Database (EBD), it is impossible to model behavior around a specific actor.

Behavior coordinates—such as which step, what stage, or is this a critical point—are entirely absent.

3.No Attack Path Mapping → No Stage-Based Intent Reasoning

Frameworks like MITRE ATT&CK define tactical stages of attacks, but the industry lacks mechanisms to map raw behavior sequences to these stages.

Can we determine if an attack has escalated? Or if it's preparing for outbound communication?

Without structure, AI cannot make such judgments—it can only passively assign scores.

C.Semantic Disconnection: Alert ≠ Explanation, Detection ≠ Reasoning

Current Reality:

Detection systems are good at identifying anomalous behaviors,

but true investigation requires understanding the behavioral path, intent at each stage, and the actor behind the actions.

The absence of an investigative layer leads to semantic misinterpretation:

Table2：Traditional Detection Conclusion VS SCSM Semantic Interpretation

Example Behavior Chain	Traditional Detection Conclusion	SCSM Semantic Interpretation
Failed login ×50 → Successful login	Successful login is treated as "normal"	Successful login is a pivot point, indicating a shift in attack stage
3 failed logins → Brute-force attack detected	Normal fluctuation in login outcome is flagged as "suspicious"	Normal behavior by the same actor is misclassified as an attack

SCSM is not about detecting anomalies — it's about reconstructing the attack story.

Table3：Seven Consequences of a Missing Investigative Layer

Missing Structural Element	Immediate Symptom	Systemic Impact
No Behavior Chain	Alerts are fragmented	No context; AI cannot reconstruct a coherent "story"
No Role-Based Path Mapping	Actor confusion across logs	Cannot determine if actions belong to the same entity
No Stage Coordinates	Logs lack positional reference	AI cannot judge whether behavior belongs to an attack phase
No Pivot Point Recognition	Phase transitions are vague	Attack escalations blend with normal activity
Missing Structural Element	**Immediate Symptom**	**Systemic Impact**
No Semantic Query Interface	No semantic search	Analysts must rely on handwritten

	capability	KQL/SPL queries
No Expert Feedback Mechanism	Human insight is lost	Judgments cannot be structured and written back for reuse
No Knowledge Base Growth Mechanism	AI has no "memory"	Cannot reuse or evolve historical investigative structures

Let’s take another look at three major unsolved challenges in today’s cyber security landscape &Industry Exploration Trends :

Table4：Three major unsolved challenges

No.	Threat Type / Problem	Current Limitation	Root Cause Analysis	Reasoning
1	Zero days False Positives / Missed Alerts / Alert Overload	Single-rule decisions lack context, easily bypassed or become ineffective	No behavior chain structure; rules operate without semantic context	SCSM does not rely on known signatures or predefined indicators. Instead, it identifies abnormal behavior chain structures and intent transition signals by reasoning over behavior chain paths and attack phases—even when the individual actions may appear normal, as is often the case in 0-day attacks.
2	APT (Advanced Persistent Threats)	Log fragmentation, no context, no chronological action reconstruction	No unified behavior chain per entity; semantic gaps remain	SCSM constructs role-centric temporal chains through the **Entity Behavior Database (EBD),** ensuring that the attack path remains continuous even across devices and systems.
3	DDoS (Distributed Denial of Service)	Knowledge can't accumulate; no reusable templates for identifying patterns across actors	No knowledge feedback or generalization mechanism; behaviors remain isolated	SCSM does not determine DDoS attacks based on access frequency alone; instead, it evaluates behavior chain similarity and overlapping multi-source activity paths to identify coordinated role-based attack behaviors.

While some vendors claim to offer "investigation" capabilities, Actually， their systems typically only aggregate or correlate data after alerts have been triggered. In the SCSM (Semantic Chain Security Model) framework, **true investigation requires:**

1.Reconstruction of behavioral sequences — not isolated or single-point queries

2.Role continuity across actions — not raw IP matching

3.Path-based semantics and event coordinates — not flat or surface-level correlations

4.Reasoning over tactical phase transitions — not rigid rule-based triggers

5.Feedback-integrated memory and structured knowledge base

Therefore, these so-called “investigation” features in commercial platforms do not constitute semantic chain investigation. They remain alert-centric, descriptive, and superficial, lacking the structural layer that should exist between detection and response.

Table5: Global Vendor Gap vs SCSM

Vendor	Investigation Capability Claim	Actual Layer Coverage	Semantic Behavior Chain Modeling	Pivot Strength / Phase Reasoning	Expert Feedback + Knowledge Memory	Investigation Layer Standardization	Comments
Google Chronicle	Natural language search + threat hunting	Alert Correlation Layer	No	No	No	No	Primarily enhances Alert Searchability
Microsoft Sentinel	KQL + Workbooks + Hunting Queries	Alert Correlation Layer	No	No	No	No	Focused on Alert Enrichment & Hunting
Splunk Security Suite	SPL-based correlation & dashboards	Alert/Detection Layer	No	No	No	No	Powerful detection but lacks Behavior Chain semantics
Crowdstrike Falcon	EDR + Threat Graph	EDR-level Process Tracking	Partial (Process Chains)	No	No	No	Strong EDR focus, lacks cross-system Behavior Chain and semantic investigation
Palo Alto Cortex XDR	Analytics + Playbooks	XDR Alert Handling	Partial (limited path correlation)	No	No	No	Emphasizes playbook-driven response, no semantic layer
SCSM	Semantic Chain Modeling + AI Reasoning	Dedicated Investigation Layer	Full Behavior Chain Modeling	Yes,Pivot Strength + Phase Reasoning	Yes,Expert Feedback + Knowledge Memory	Yes,Proposes Standardization	Fills global architecture gap between Detection and Response

In Summary:

Whether it's the gradual failure of traditional rule-based systems against novel attack scenarios, or the current industry's fragmented attempts to integrate AI, both trends point to a fundamental root cause: the absence of an investigative layer and structured behavior chains.

III.Solution Theory&Solution Framework

Table6：Industry Paradigm Evolution for cyber security

Evolution Stage	Paradigm Innovation	Representative Technologies/Products	Industry Architectural Layer	SCSM Contribution	Impact
First Stage (1990s)	Perimeter Security (Firewall)	Cisco PIX, Checkpoint FW	Perimeter Layer	—	Defined boundary security
SecondStage (2000s)	Real-time Intrusion Detection (IDS/IPS)	Snort, Suricata, Bro/Zeek	Detection Layer	—	Introduced real-time detection
Third Stage (2005–2015)	Centralized Log Analysis (SIEM)	Splunk, ArcSight	Detection → Alert Layer	—	Enabled cross-device correlation
Fourth Stage (2015–2020)	Tactical Phase Modeling (MITRE ATT&CK)	MITRE ATT&CK Framework	Detection → Attack Understanding	—	—
Fifth Stage (2017–Present)	Automated Response (SOAR)	Cortex XSOAR, DFLabs	Response Layer	—	Enabled response automation
Sixth Stage (2025.6)	**Investigation Layer**	**SCSM**	New Investigation Layer	First structured investigation layer definition	Bridges Detection → response gap
Seventh Stage (2025.6)	**Knowledge Loop / AI-Evolving Layer**	**SCSM**	New Knowledge Memory Layer	First expert feedback + AI learning loop	Establishes AI-driven investigative paradigm

Over the past three decades, the cybersecurity field has undergone multiple paradigm shifts — each introducing a new architectural layer: perimeter defense, real-time detection, centralized analysis, tactical phase modeling, and automated response.

Yet **two critical architectural gaps remain**: the absence of a native Investigation Layer and a structured, AI-driven Knowledge Memory Layer.The Semantic Chain Security Model (SCSM) addresses **both gaps simultaneously** — defining the world's **first structured Investigation Layer**, and **introducing an AI-evolving Knowledge Loop** that enables dynamic learning and memory within security operations.This marks not one, but two paradigm breakthroughs in cybersecurity architecture — moving beyond traditional detection and response toward **AI-driven investigation** and **knowledge-based adaptive security.**

A.Solution Theory

Original Semantic Constructs Introduced for AI-Driven Security Investigation Architecture.As the original author, I hereby introduce the following core semantic constructs into the AI-driven security incident investigation system. These foundational concepts are designed to support behavior chain modeling, path inference, human-AI consensus, and knowledge base evolution through write-back mechanisms.

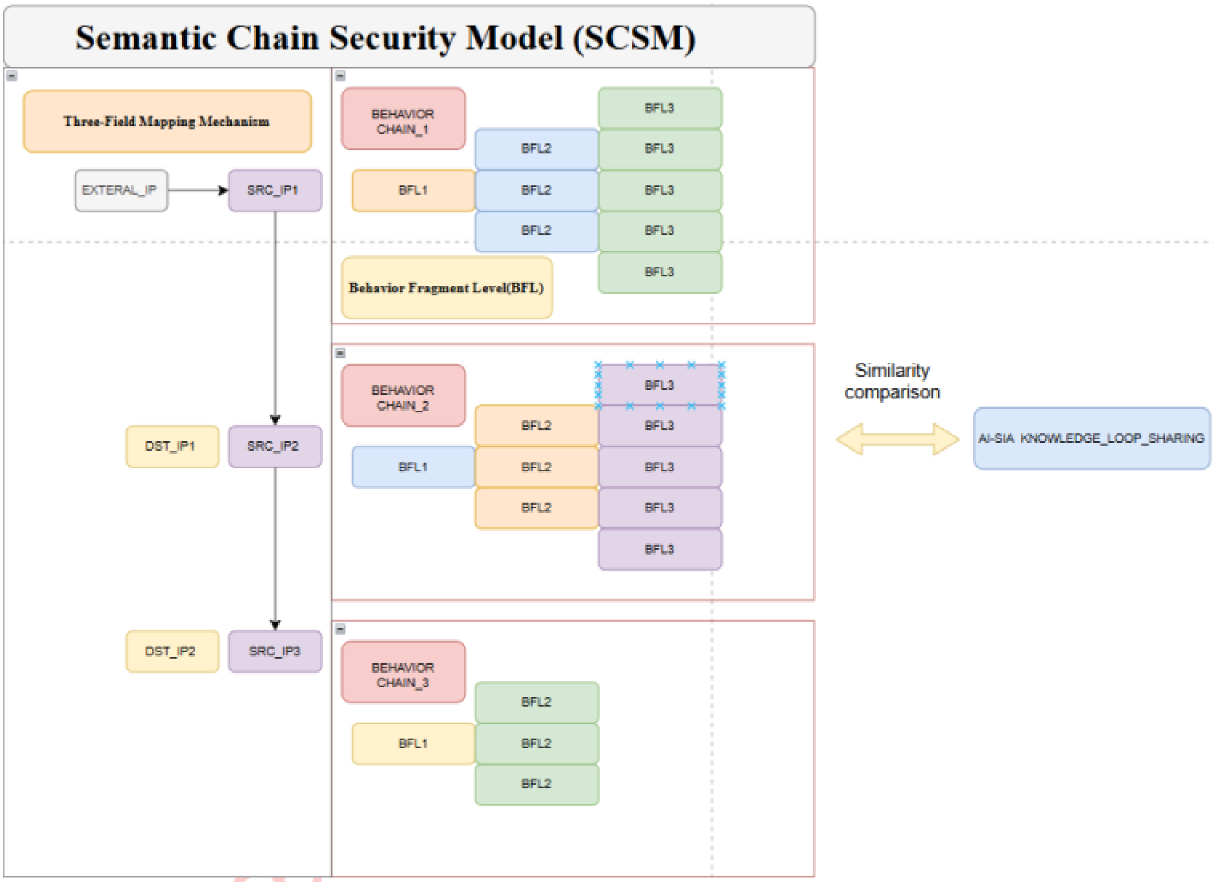

Figure1: Semantic Chain Security Model

1.Semantic Chain Security Model (SCSM)

The Semantic Chain Security Model (SCSM) is an AI-driven security operations framework centered on behavioral semantics. It establishes a full-cycle investigation process that spans from log ingestion, behavior reconstruction, and semantic inference to expert validation and knowledge write-back.

SCSM breaks away from the traditional SOC paradigm built on rule-based detection and manual log querying. By introducing a structural semantic layer, it empowers AI to understand behavior—enabling proactive risk assessment, attack path recognition, and continuous self-evolution.

2.Character/Entity Behavior Database (EBD)

The Entity Behavior Database (EBD) is a structured, role-centric data repository that chronologically records and organizes all observable actions performed by a specific entity—such as a user, IP address, host, or device.

It provides the foundational structure for behavior reconstruction and actor-centric semantic reasoning in security investigations.

3.Knowledge Write-Back Mechanism

The Knowledge Write-Back Mechanism refers to the process by which human security experts review, validate, and optionally revise the AI-generated behavior chains and risk evaluations.

This mechanism is the cornerstone of human-AI collaboration. It ensures that expert judgments are not lost but instead structurally recorded and fed back into the system's evolving knowledge base, thereby improving semantic inference over time.

4.Three-Field Mapping Mechanism

The Three-Field Mapping Mechanism addresses semantic ambiguities in traditional binary log field mappings (e.g., srcip, dstip), particularly in edge device logs (e.g., WAF, proxy).

It introduces a third semantic field, external_ip, to explicitly represent the origin of the access request. This results in a clarified behavioral structure:

External_ip → srcip → dstip

enabling accurate actor attribution and path-level modeling in a three-stage behavior chain format.

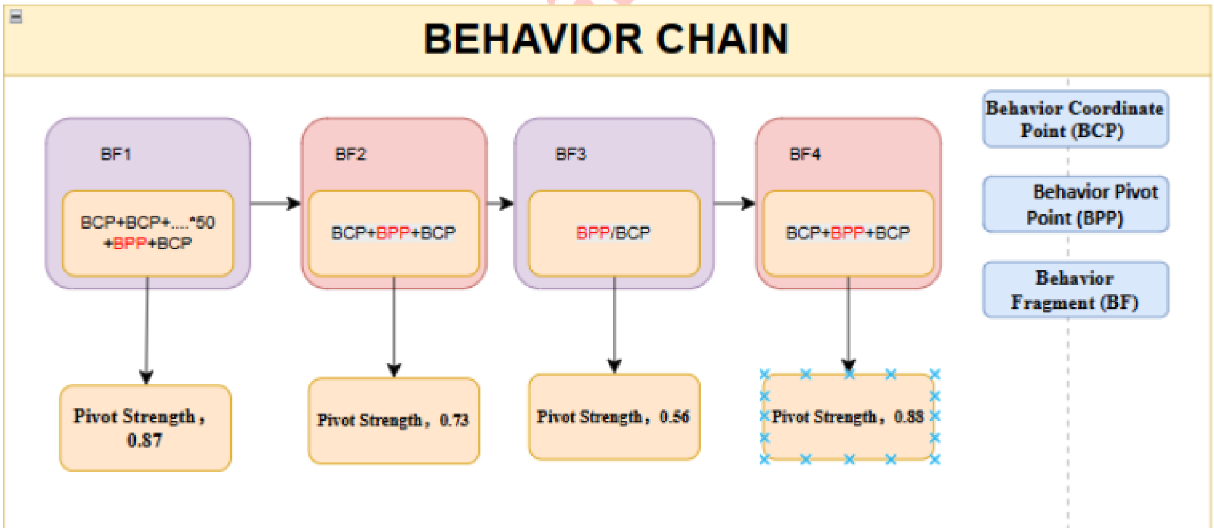

Figure2：Behavior Chain

5.Behavior Chain / User Behavior Chain (BC / UBC)

A Behavior Chain is a time-ordered, semantically consistent sequence of behavioral events linked to a single entity (e.g., user, IP, host, or device) within a defined time window. It integrates both normal and abnormal behaviors and provides the structural foundation for AI to perform reasoning, identify anomalous stages, and infer malicious intent. It is the precondition for building an attack chain.

6.Behavior Chain Path (BP)

A Behavior Chain Path refers to a reconstructed sequence of operations performed by a specific role (e.g., IP, host, account, or process) over time. Each path represents the continuous behavioral trajectory of one entity and forms the semantic backbone for behavior modeling, attack path reconstruction, and AI-driven inference

7.Behavior Coordinate Point (BCP)

A Behavior Coordinate Point is the smallest semantic unit within a behavior chain, marking the position of a behavior event in the structural path. Each BCP contains: the behavior itself, its position in the sequence (e.g., step number), the tactical stage it belongs to, its contextual semantics, scoring results from micro-models, and a flag indicating whether it is a critical node (e.g., a pivot point).BCPs are essential for path reasoning, similarity comparison, and knowledge visualization.

8.Behavior Pivot Point (BPP)

A Behavior Pivot Point acts as a semantic transition hub within the behavior chain, bridging different tactical stages (e.g., from initial access to lateral movement). It represents a behavioral turning point and decision junction in the AI reasoning process.

Typical characteristics include:

①Preceded by multiple anomalous or failed attempts;Followed by a successful or impactful action;

②Often aligned with changes in tactics or objectives;

③BPPs are the entry nodes where behavior shifts from exploration to execution, and their Pivot Strength determines how confidently the AI recognizes the phase transition.

④The detection is not based on the intrinsic abnormality of a single behavior, but on its semantic role within the behavior chain, particularly whether it serves as a pivot point indicating a phase shift in the attack sequence.It functions as a semantic transition node within the attack chain—acting as both the logical bridge between ATT&CK stages and the semantic trigger point for AI to assess whether a subject has entered a malicious trajectory.

9.Behavior Fragment (BF)&Behavior Fragment Level(BFL)

Behavior Fragment (BF) A Behavior Fragment (BF) represents the entire process of a single behavior type (e.g., login, file upload, command execution) from start to end.Each BF is composed of multiple state judgments that describe how the behavior unfolds.A BF can represent a normal process (e.g., multiple login failures followed by a successful login within acceptable limits) or an abnormal process exhibiting attack traits (e.g., dozens of failed logins followed by a successful login within a short timeframe).By combining these states, a BF provides a complete semantic view of the behavior.

Behavior Fragment Level (BFL) A Behavior Fragment Level (BFL) is a lifecycle classification that organizes BFs into three stages: Early, Mid, Late.This mapping reflects the tactical role of each BF in the attack chain and aligns with MITRE ATT&CK tactics.

Behavior Fragment (BF) Structural Components:

Behavior States: Different outcomes or phases of the same behavior (e.g., success, failure).

Behavior Coordinate Point (BCP): An individual state event in the process (e.g., one failed login).

Behavior Pivot Point (BPP): A special type of BCP representing a decisive transition to another phase (e.g., multiple failures followed by a final success).

Conditions: Contextual rules for evaluating the fragment (e.g., time window, frequency thresholds, file type).

Behavior Fragment (BF) Key Features:

Captures the **entire lifecycle of a behavior**, not just isolated events.

Can be benign or malicious, depending on the combination of states.

Serves as the fundamental analytic unit for mapping behaviors to the attack lifecycle.

Examples of Behavior Fragments

Example 1: Login Behavior (BF) — Abnormal Brute-Force Behavior

States included:

Multiple failed login attempts (BCPs)

One successful login (BPP)

Condition: ⩾ 50 failed attempts within 24 hours

Interpretation: A complete login behavior fragment → successful brute-force attempt.

Example 2: Login Behavior (BF) — Normal Login Behavior

States included:

A few failed login attempts (BCPs)

One successful login (BPP)

Condition: < 10 failures within 1 hour

Interpretation: A complete login behavior fragment → normal login activity.

Behavior Fragment Level (BFL)

Level 1 (Early Stage / Initial Access & Discovery) Representative BF: **Login (Brute-Force)**

MITRE ATT&CK Mapping: Initial Access, Credential Access

Level 2 (Mid Stage / Execution, Persistence, Lateral Movement) Representative BF: **File Upload**

MITRE ATT&CK Mapping: Execution, Persistence, Lateral Movement

Level 3 (Late Stage / Command & Control, Exfiltration) Representative BF: **Command Execution**

MITRE ATT&CK Mapping: Command & Control, Exfiltration, Impact

Table7：Semantic Phase Mapping Between Behavior Fragment Levels (BFL) and MITRE ATT&CK Framework

BFL Semantic Phase	Definition	Corresponding MITRE ATT&CK Tactic(s)
Level 1 – Access Phase	Also known as Recon Phase. This level describes the process from external probing to successful access. For example, brute-force attempts followed by successful login.	*Initial Access*, *Credential Access*, *Discovery*, *Reconnaissance*
Level 2 – Execution Phase	Represents the stage where the attacker leverages obtained access to execute tools, scripts, or malicious payloads within the environment.	*Execution*, *Privilege Escalation*, *Defense Evasion*, *Persistence (early)*
Level 3 – Divergence Phase	After tool execution, multiple paths may emerge including lateral movement, credential theft, communication with C2 servers, or data exfiltration.	*Lateral Movement*, *Collection*, *Command & Control*, *Exfiltration*, *Impact*, *Persistence (sustained)*

The following three structural elements form the foundational prerequisites for enabling AI-driven knowledge base feedback:

1.A behavior chain must exist

Providing the structural backbone that organizes actions chronologically and semantically around a specific actor (user, IP, host, etc.).

2.Each behavior must have a semantic coordinate

Marking its position and meaning within the overall context (e.g., login attempt, file upload, lateral movement), enabling role-aware interpretation.

3.AI must identify pivot points

Detecting critical transitions that signify a shift in the attack phase, such as a successful login following multiple failures.

Only when these conditions are met can a human analyst validate the event severity based on the reconstructed path and perform accurate knowledge write-back.This is not just a technical process—it is the semantic essence of the system. These structural anchors are what allow the AI to truly “understand the attack story” and evolve from detection to reasoning.

(Note:Pivot Point Identification: From Binary Judgment to Weighted Confidence)

10.Pivot Strength

Pivot Strength refers to the semantic confidence score assigned to a specific behavior node when it is identified as a pivot point within an attack behavior chain. It reflects the logical support for judging whether the action represents a tactical phase transition—such as a shift from reconnaissance to execution—and serves as a key signal indicator in AI inference pathways.

In semantic investigation, whether a behavior constitutes a pivot point should not be treated as a binary decision (yes/no), but rather as a probabilistic weight known as Pivot Strength.

Two Key Decay Factors Impact **Pivot Strength**:

1.Time Decay The longer the time interval between consecutive failure and eventual success, the lower the confidence in it being a malicious transition.If the time gap exceeds a threshold T (e.g., 15 minutes), the event may indicate a legitimate login rather than a post-brute-force compromise.

2.Pattern DivergenceIf the "successful behavior" differs significantly in method or context from previous failed attempts (e.g., different IP, device, or user agent), the system assumes actor inconsistency, reducing confidence in it being a true transition.

pivot_confidence = f(base_pattern_match_score, temporal_proximity, actor_consistency)where:

- base_pattern_match_score = Evaluate whether the sequential actions reflect a consistent behavioral intent

- temporal_proximity = exp(-⊿t / T)   // The longer the time gap, the lower the weight

- actor_consistency = Whether the action is attributable to the same entity, based on IP address, session ID, or device fingerprint

pivot_point = True if pivot_confidence > threshold

B.Solution Framework

Table8：Five-Layer Foundation

Level	Name	Function Description
1	**Field Normalization Layer**	Standardizes heterogeneous log formats from multiple sources; constructs key fields (e.g., srcip, dstip, eventid, source_from, external_ip)
2	**Micro-Scoring Layer**	Applies multiple AI models to each log entry to generate threat score fields (model1 to model4) for downstream semantic computation
3	**Behavior Chain Modeling Layer**	Reconstructs behavior chains for the same entity based on temporal, semantic, and role consistency, forming inference material
4	**AI Semantic Reasoning Layer**	Performs path inference, attack chain recognition, and pivot point detection on behavior chains; supports natural language interface via LLM
5	**Expert Feedback & Knowledge Loop**	Integrates human feedback into a knowledge base with behavior chains, pivot points, labels, and attack names, enhancing future reasoning capabilities

IV.Methodology

A.Field Normalization Layer

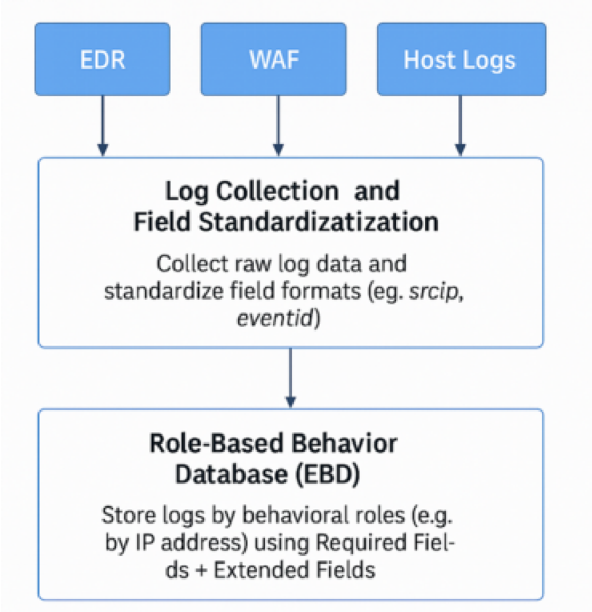

Figure3：Log Standardization and Collection Module + Role-Based Behavior Database (EBD) Construction

This figure illustrates the collaborative workflow of an AI-driven behavior chain modeling and intelligent investigation system, composed of five core modules. These modules form a logical feedback loop, highlighting the structural and intelligent design of the system.

1.Log Collection and Field Standardization Module

This module is responsible for collecting raw log data from a variety of security devices, such as EDRs, WAFs, and host-based logging systems. It unifies field formats—such as srcip, eventid, and others—to standardize heterogeneous data. This standardization process lays the groundwork for downstream semantic modeling and AI analysis.

2.Role-Based Behavior Database (EBD) Construction

Once standardized, logs are written into a structured database based on behavioral roles (e.g., by IP address). Each unique role entity (such as an IP) is mapped to its own table or partition. The database is built following a "Required Fields + Extended Fields" principle:

Required Fields ensure semantic consistency and traceability (e.g., srcip, dstip, eventid, external_ip, source_from, Date, Time);Extended Fields allow compatibility with additional device-specific metadata without compromising structural uniformity. This approach preserves the unique attributes of different log sources while enabling unified behavior tracking and flow analysis across systems.

Table9：Essential Fields Definition for Role-Based Behavior Database

Field Name	Description
external_ip	Public IP address of an external visitor. When the log source is **a boundary device** (e.g., WAF, firewall), the original srcip is mapped here to support external threat attribution.
srcip	Initiating IP address of the behavior. Represents the behavioral subject in internal systems and is the starting point of a behavior chain. Special case: When the log source is a boundary device, the original dstip (external actor) is mapped here.
dstip	Target IP address of the behavior. Represents the destination, either the endpoint or an intermediate node in the behavior chain.
Model_X	Micro-model feedback field. Used to store scores or results from AI micro-models. Naming convention: Model_<purpose>_<number>, e.g., **Model_malicious_detect_1**.
source_from	Log source identifier. Specifies the device that generated the log (e.g., WAF, EDR, host), which is used for interpreting field semantics. Naming convention: <deviceType>_<deviceName>_<additionalInfo>, e.g., **WAF_modsecurity_01**.
Time	Time of the event in **24-hour** format.
Date	Date of the event in **YYYY-MM-DD** format.
Detail	Compatibility field for extended device-specific data. Used as Detail.<key> or in composite formats.
Event ID	Event classification for the local host. Describes the action performed, the outcome, and a unique identifier for the log entry.
Log_id	Globally unique identifier for each log entry. Serves as the primary key in the role-based behavior database to ensure entity traceability, indexing efficiency, and integrity of behavior chain reconstruction.

B.Micro-Scoring Layer

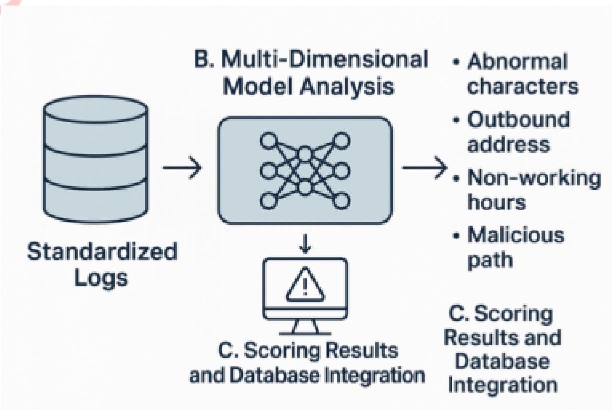

Figure4：Micro-Model Threat Scoring Module

This diagram illustrates the end-to-end logic of the Micro-Model Threat Scoring Module.

1.Standardized Input

The process begins with standardized logs on the left. These logs have been cleansed and normalized across heterogeneous sources, ensuring consistent fields such as srcip,source_from, and time&date. They serve as the input data for model evaluation.

2.Multi-Dimensional Model Analysis

In the central module, multiple machine learning models or external API services process the logs across several behavioral dimensions. These models assess characteristics such as unusual characters in commands, outbound connection attempts, non-work-hour activity, or access to suspicious paths. Based on this contextual understanding, the models generate threat scores that reflect the severity or abnormality of each log event.

3.Scoring Results and Database Integration

Finally, the scored outputs are written into the database as structured threat labels. These scores and annotations support downstream components like behavior chain construction and AI semantic reasoning. The module is built to be pluggable, scalable, and interpretable—serving as the foundational unit for single-point anomaly detection in the broader intelligent investigation system.

C.Behavior Chain Modeling Layer

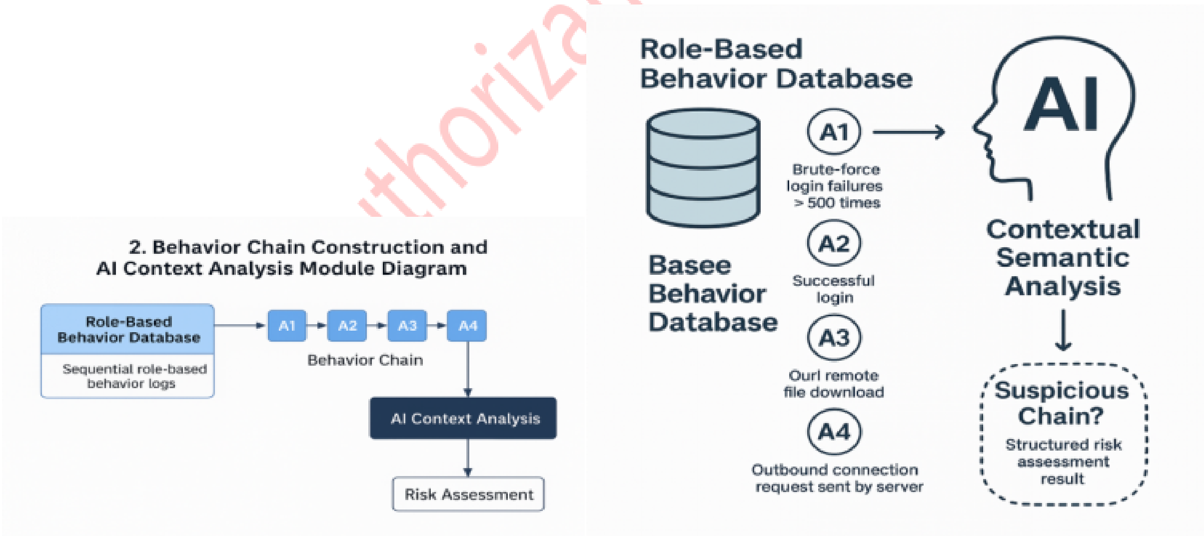

Figure5：Behavior Chain Construction and AI Context Analysis Module

This diagram is designed to visually illustrate the core workflow of the "Behavior Chain Construction and AI Context Analysis" module. The system begins by extracting a sequence of logs associated with a specific behavioral role (e.g., an IP address) from the Role-Based Behavior Database within a defined time window.

These logs — such as multiple failed login attempts (e.g., brute-force attack), a subsequent successful login, remote file downloads, and outbound connection attempts — are arranged in chronological order to form a behavior chain (e.g., A1 → A2 → A3 → A4).

The constructed chain is then passed to the AI Context Analysis Module, where each event is semantically interpreted. The AI not only identifies the meaning of each individual action but also performs logical reasoning across the full chain to assess whether it constitutes a potential threat— such as a classic mining trojan delivery path.

Finally, the system outputs structured conclusions based on this analysis, which can trigger alerts or support human investigation. The diagram emphasizes the intelligent investigation loop: **Raw logs → Behavior sequence → AI semantic analysis → Risk judgment.**

D.AI Semantic Reasoning Layer

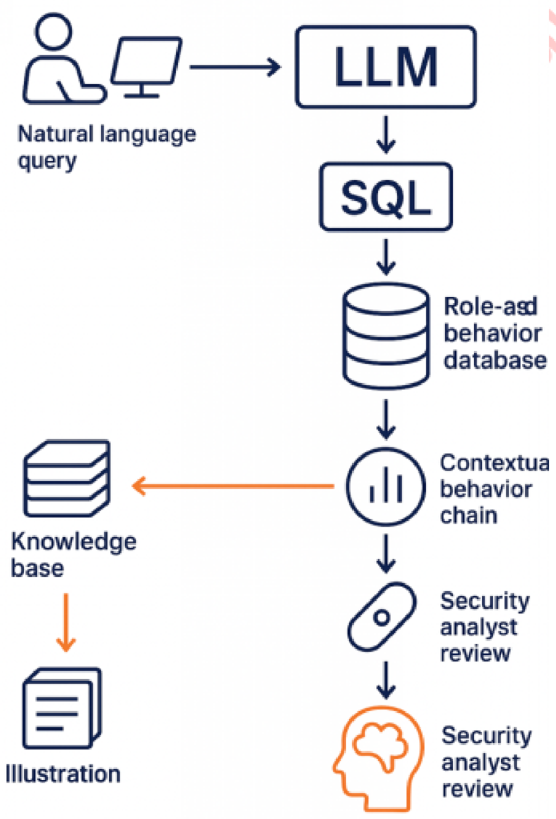

Figure6：Natural Language Input and Feedback Output Module

This diagram illustrates the workflow of the Natural Language Input and Feedback Output Module, emphasizing the role of AI as a bridge in human-machine interaction.

First, a security analyst poses a question related to a security event using natural language, such as:

"Did a specific IP perform any suspicious login activity During 2:00- 3:00 pm 03/06/2025?".

The system uses a Large Language Model (LLM) to understand the semantic meaning of the question and converts it into a structured SQL query.

The SQL query retrieves relevant raw behavior data for the specified subject from the Role-Based Behavior Database.

Next, the AI performs contextual analysis on the query results, reconstructing a complete behavior chain based on temporal sequence and semantic correlation. The LLM then conducts semantic interpretation and classification of the chain to preliminarily identify possible attack patterns or anomalous behaviors.

The result is then handled in two ways:

On one hand, it is reviewed and confirmed by the security analyst; on the other, if the behavior chain is identified as a novel or representative attack pattern, it is written into the knowledge base along with analytical tags, enabling knowledge retention and reuse

This entire workflow reflects an efficient human-AI collaboration, moving from natural language to structured analysis, and from AI generation to expert confirmation.

E.Expert Feedback & Knowledge Loop

Figure7:Expert Confirmation and Knowledge Base Feedback Module

This diagram is designed around two distinct scenarios, together illustrating the bidirectional feedback loop of the Expert Confirmation and Knowledge Base Writing Module.

Scenario One (upper section) shows how, after the AI constructs a behavior chain (e.g., A1 – A4), the results are handed over to a security expert for review. The expert reviews the chain in context, using AI-generated scores and semantic information to either confirm or modify the sequence. Once confirmed, the typical attack behavior chain is written into the knowledge base as reusable structured knowledge.

Scenario Two (lower section) demonstrates the reverse flow. When the system detects a new log sequence, the AI retrieves similar historical behavior chains from the knowledge base and compares them with the new subject's behavior. If the similarity reaches 75%, the system automatically notifies the analyst and provides a recommended response.

This process reflects the ability of the system to store expert-reviewed knowledge structurally and to intelligently match future events based on learned patterns. It forms an efficient collaborative model of Expert + AI + Knowledge Base

V.Use Cases

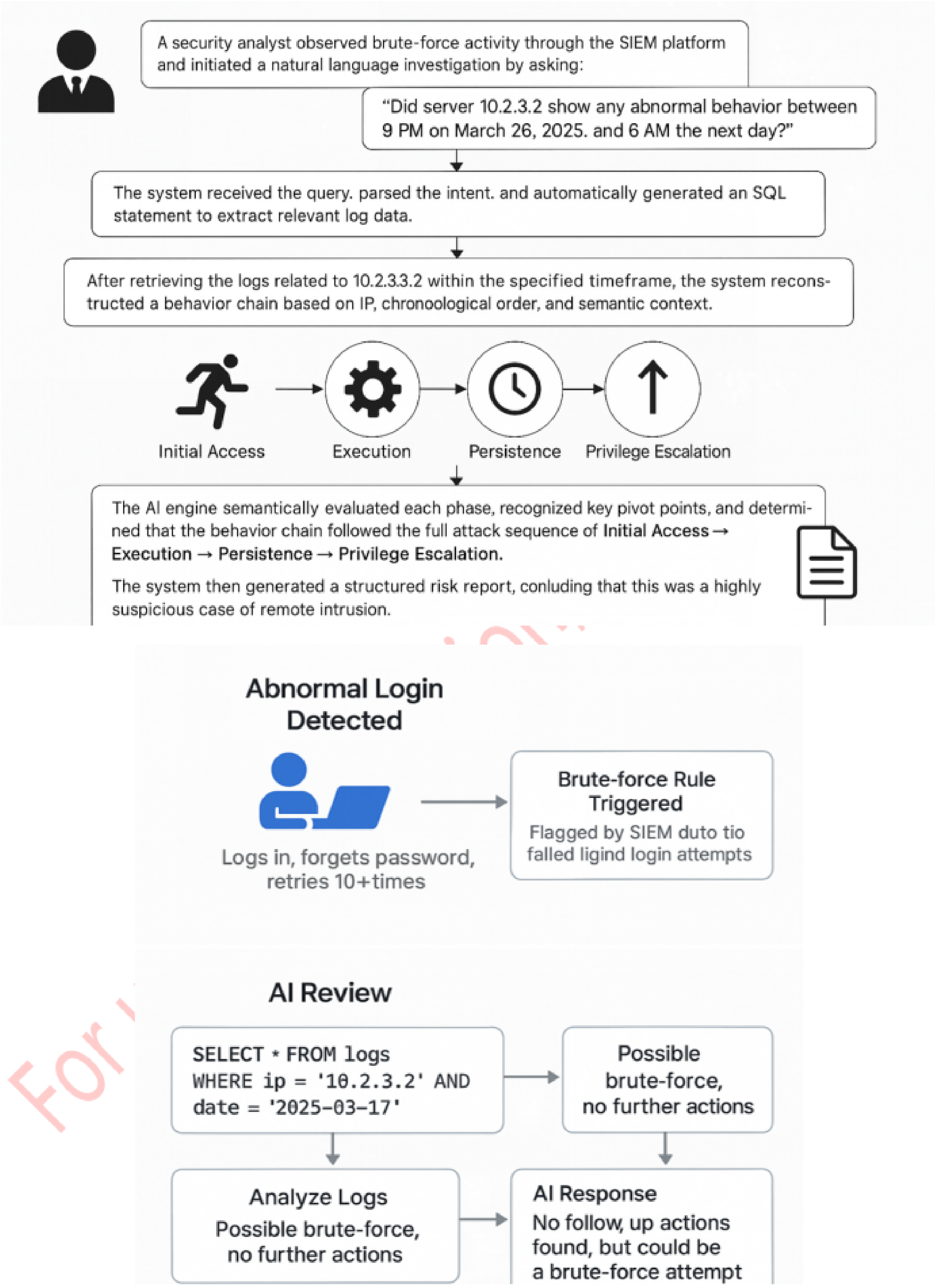

Figure8:AI-Driven Behavior Chain Analysis Flowchart(1.1)

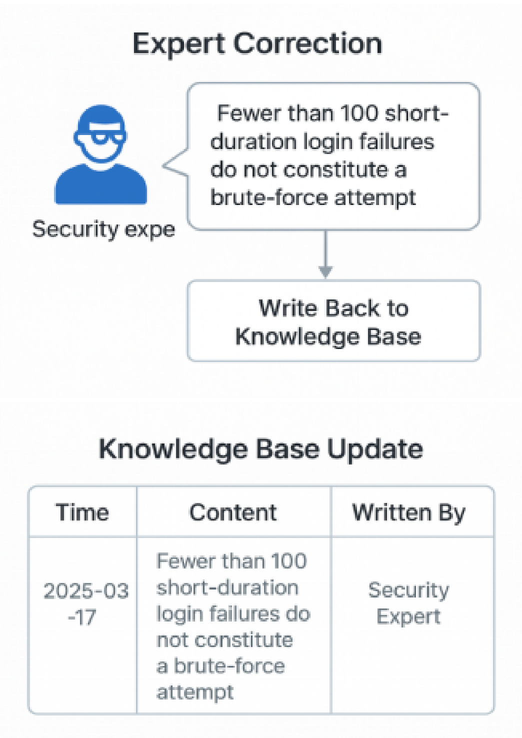

Figure9:Expert Feedback and Knowledge Base Update Diagram(1.2)

VI.Industry Impact

A.Impact

To address the lack of structured investigative semantics between “alert” and “response” in current SOC/SIEM architectures, it is recommended that the “Investigative Layer” be formally introduced as a distinct architectural component in industry standards. It mainly includes:

1.Inputs: Sequences of behavioral logs and alert events

2.Outputs: Structured behavior chains, attack path graphs, and AI-generated reasoning suggestions

3.Core Functions: Behavior modeling, pivot point identification, and expert knowledge feedback integration

4.Necessity: Enables the response layer to execute precise, explainable, and automated decisions

B.Systemic Value

1.Language replaces experience: Transitions from manual, experience-driven investigation to AI-driven semantic reasoning and interpretation.

2.Structure captures expertise: Uses behavior chains to structurally store human expertise and AI feedback

3.AI understands behavior: Enables AI to understand not just “logs” but actual “behaviors”

4.Adaptive to attack-defense evolution: Supports continuous evolution, knowledge accumulation, and reasoning over behavioral variants

5.Transforms traditional SIEM investigation: Fully replaces the manual “log piecing” paradigm—AI investigates, humans validate

Table10:Structured Solutions to Core Industry Challenges

Industry Challenge	Root Cause	Structured Solution
0day Attacks	0days fundamentally bypass fixed detection rules; in traditional defenses, rule bypass = full defense failure	Structural-semantic language does not rely on static rules. It identifies anomalies through behavior chain paths and AI semantic reasoning, making it resilient to rule bypasses
Pivot Point Detection	Traditional systems misclassify post-exploit “normal” behaviors as benign, failing to detect lateral movement stages	Based on the ATT&CK tactical framework, any "normal behavior" following a labeled attack stage is flagged as a pivot point. The system includes decay mechanisms and path-based scoring
Industry	**Root Cause**	**Structured Solution**

Challenge		
APT Attacks	Logs are fragmented and lack contextual continuity, making it impossible to reconstruct the full attack chain	Builds a Role-Based Behavior Database (EBD), storing logs by entity and sorting them chronologically. AI semantic analysis is applied to reconstruct the complete attack path
DDoS Attacks	Distributed, multi-source, concurrent access depletes resources. Traditional frequency-based thresholds are easily bypassed	Trains micro-models targeting specific DDoS features to detect patterns in path, frequency, and distribution. AI synthesizes these into behavior chains for organizational defense. Detection strategy includes: 1. Micro-models for fine-grained analysis, macro-logic for attack inference 2. Behavior chain knowledge base with expert feedback integration

Table11:SCSM vs Traditional SOC: Structural Innovation Comparison

Innovation Point	**Traditional SOC**	**SCSM Advantages**
Log Structure	Non-standard fields, inconsistent formats	Field semantic normalization with triple-field mapping (e.g., external_ip)
Detection Method	Rule-based matching, signature-dependent	AI-based micro-scoring, multi-dimensional models, self-evolving detection
Behavior Understanding	Evaluates each log independently	Constructs semantic chains via temporal and role-based behavior linking
Attack Recognition	Relies on static attack rules	Dynamically identifies attack chains and pivot points; supports variant inference
Expert Knowledge Retention	High loss of individual insights	Writes back into structured knowledge chains, enhancing system memory
LLM Integration	Absent	Supports full flow: natural language → SQL/semantic reasoning → structured output
Structural Layer Completion	Detection → Alert → Response (Investigation is manual)	Completes the gap: Detection → Alert → Investigation → Response

Table12:SCSM vs. Three Historical Protocol Standards

Protocol	**Initial Phase**	**Structural Release**	**Adopted by Standards Bodies**	**Global Ecosystem Expansion**
HTTP	Defined at CERN as URL structure	Adopted by W3C → Unified browser language	Referenced by all web protocols	Became the universal language of web communication
TLS	Netscape’s proprietary SSL	Extended by IETF as TLS standard	Adopted by all secure browser platforms	Became the de facto encrypted communication standard for the Internet
Protocol	**Initial Phase**	**Structural Release**	**Adopted by Standards Bodies**	**Global Ecosystem Expansion**

MITRE ATT&CK	MITRE research project	Released as an open tactical structure model	Referenced by NIST, widely adopted by security companies	Became the factual standard for threat detection and behavior modeling
SCSM	Independently invented + patent-defined	Released via industry white paper + five supporting documents		

VII.Collaboration

In Summary:

1.Compatible with all data source platforms via standardized field mapping, regardless of vendor or system architecture.

2.Supports multilingual natural language input (e.g., Chinese, English, German).

3.Seamlessly integrates into existing SOC workflows with AI-powered investigation layers—no need to replace current detection engines.

4.Supports various log formats including Syslog, CEF, JSON, and Windows Event Logs.

5.Enables behavior chain construction based on diverse role types such as IP addresses, users, endpoints, and assets.

6.Micro-models are pluggable and support custom training and on-premise deployment.

7.Behavior chain database can be implemented using mainstream relational databases.

8.AI models are deployable locally and do not rely on external APIs.

9.Enables cross-platform migration and standardized knowledge feedback mechanisms.

10.Facilitates organization-wide knowledge accumulation and reuse through structured, shareable insights.

VIII.Technical Risk Boundary Statement

The SCSM model and the AI-SIA Alliance framework are designed as an industry-generic, explainable, and structured AI framework for enhancing investigation capabilities within security operations centers (SOCs). The framework operates exclusively at the investigation and standardization layer, and is not tailored to any specific national critical infrastructure domain or restricted sectoral application.The architecture does not involve or control critical national infrastructure, nor does it participate in automated system control or response operations. The solution is inherently human-in-the-loop, designed to augment human understanding of behavior chains and investigation context, rather than performing automated decision-making or controlling security responses.The deployment model is flexible and fully adaptable to operational requirements. The solution supports both online and offline modes, and can be deployed on-premise using self-hosted, open-source components (such as TensorFlow, LLMs, ChatGLM, MySQL, Filebeat, Logstash, etc.), without dependency on external cloud services or third-party unvetted APIs. Any future integration with commercial components will be subject to applicable national AI governance and compliance requirements.

The SCSM model does not inherently process national security sensitive data. The framework operates on structured behavior chain reconstruction based on input data sources determined by the deploying entity. Any inclusion of sensitive data remains under the governance and compliance of the deploying entity, with no inherent requirement or pre-configuration for processing such data within the core SCSM architecture.Usage of Large Language Models (LLMs), if incorporated, is strictly confined to auxiliary functions such as generating human-readable reports, facilitating investigation context summarization, and supporting knowledge augmentation. LLM components do not participate in core investigation decision-making logic or automated control pathways within the solution.