METHOD AND SYSTEM FOR SEMANTIC BC (BEHAVIOR CHAIN) RECONSTRUCTION AND AI-DRIVEN SECURITY INVESTIGATION

TECHNICAL FIELD

The invention relates to the cross-field of network security and artificial intelligence, and particularly to a method and system for semantic BC (behavior chain) reconstruction and AI-driven security investigation.

BACKGROUND ART

Existing security operation platforms primarily rely on SIEM systems for log aggregation and event analysis. Their core mode typically involves matching single logs based on fixed rules to generate alerts. During actual investigation processes, analysts need to manually write query statements to locate key information from multi-source logs and manually piece together BCs to infer threat paths. This method suffers from problems such as high syntax barriers, fragmented logs, information omission, and delayed response. Simultaneously, various log formats are heterogeneous, fields are inconsistent, security devices operate independently, and there is a lack of contextual correlation capability, resulting in an inability to fully reconstruct attack processes. Moreover, system alerts heavily rely on static rules, lacking detection capabilities for unknown attacks and highly covert behaviors, often failing to identify and block advanced threats like APT and ZERO DAY in advance.

Furthermore, current SOC investigations rely on experienced personnel to complete behavior reconstruction and risk judgment. Once personnel change, knowledge and processes lack accumulation, investigation quality fluctuates easily, and reusable knowledge structures are lacking. Traditional platforms also do not support interactive methods using natural language questioning. Analysts still need to master specific syntax or rule languages to query the system, creating a high barrier to data access and reasoning capabilities.

Therefore, it can be concluded that existing security operation platforms generally have the following structural defects:

1. lack of capability for unified subject behavior modeling across devices and dimensions;

2. inability to automatically construct BCs and provide contextual explanations for role behaviors;

3. lack of intelligent Q&A interaction mechanisms and semantic output capabilities;

4. inability to accumulate historical attack knowledge into a dynamically reusable knowledge base;

5. inability to break free from rule dependence, resulting in weak response capabilities to advanced threats.

SUMMARY OF THE INVENTION

The object of the invention is to overcome the above problems, proposing a method for semantic BC (behavior chain) reconstruction and AI-driven security investigation, defined as the missing investigation layer in current security operation architectures, achieving for the first time a structured investigation process and knowledge evolution loop across logs, devices, and roles. To achieve the above object, the invention adopts the following technical solutions:

a method for semantic BC reconstruction and AI-driven security investigation, including the following steps:

step S11: collecting log data from multiple security devices or systems, performing field normalization for the purpose of unified behavior subject modeling;

step S12: constructing a BC structure associated with an entity, and the BC structure is a time series, graph structure, tag chain structure, or equivalent path structure;

step S13: receiving a user-input natural language security investigation question;

step S14: mapping the natural language question to a query target related to the BC structure using an artificial intelligence engine;

step S15: performing semantic reasoning, path search, or context analysis operations on the query target based on the BC structure;

step S16: outputting a structured or unstructured query result, including path diagrams, node lists, behavior segments, risk prompts, expert feedback information, or suggestive results.

Further, in step S11, the log data undergoes field normalization, entity unification, platform alignment, and identifier aggregation during the collection process, achieving

cross-platform and cross-system log BC integration.

Further, in step S12, the BC structure is established based on log chronological order, operational association, tag linkage, causal dependency, or knowledge graph paths, supporting graphical display and real-time or persistent storage of behavior nodes; the BC structure embeds enhancement information including node risk scores, structural path weights, behavior intent tags, anomaly confidence levels; pivot point identifiers: the pivot point identifiers are used to mark key nodes indicating changes in attack phases within the BC; behavior coordinates: the behavior coordinates record the phase position, role attribution, and temporal sequence of each behavior within the chain path; phase pivot intensity scores: the phase pivot intensity scores are used to measure the credibility of pivots between nodes, comprehensively considering temporal continuity, role consistency, and behavior pattern deviation; the enhancement information is used to optimize the artificial intelligence model's path ranking, anomaly identification, and phase pivot reasoning capabilities.

Further, in step S12, the BC structure is either a pre-constructed structure or a dynamically aggregated collection of event segments formed during a query process based on user-specified entity identifiers, time ranges, or behavior characteristic conditions; the collection of event segments has relevance or logical sequence at the semantic level; the BC structure serves as the foundation for triggering a structured event investigation process, including BC reconstruction for specific IPs or users, comparison of similar attack paths, behavior attribution, and collaborative analysis tasks; BC node composition includes communication actions, login behaviors, file operations, configuration changes, process invocations, URL accesses, API requests, or other identifiable security behavior events.

Further, in step S14, the artificial intelligence engine comprises a large language model, NLP, a graph neural network, a rule-learning hybrid engine or an equivalent semantic reasoning system; the artificial intelligence engine converts the natural language question into a structured query template, graph structure traversal request, or SQL statement, and combines Prompt engineering to achieve efficient mapping between natural language and structured queries, supporting the semantic reasoning process.

Further, the user continuously asks questions in natural language, and the artificial intelligence maintains a semantic session based on the BC context state, supporting context

zooming, chain structure focus switching, reverse question generation, and interactive behavior path evolution, and supporting cross-turn context memory and expert feedback-driven dialogue evolution capabilities.

Further, in step S16, the output result includes attack path diagrams, structural node lists, behavior trajectories within a time window, matching segments, risk explanations, reasoning failure prompts, or semi-structured results requiring manual confirmation.

A system for semantic BC reconstruction and AI-driven security investigation, comprising: a field normalization module, a BC modeling module, an AI semantic reasoning module, a micro-model scoring, and an expert feedback and knowledge write-back mechanism; the field normalization module, BC modeling module, AI semantic reasoning module, micro-model scoring, and expert feedback and knowledge write-back mechanism form a logical feedback loop, supporting the BC construction, attack path reconstruction, and artificial intelligence semantic analysis process;

the field normalization module performs field normalization, entity normalization, platform alignment, and key field mapping on collected multi-source log data;

the BC modeling module constructs a BC structure based on chronological order, semantic consistency, and role consistency, serving as the input foundation for AI reasoning and path reconstruction;

the AI semantic reasoning module performs path reasoning, phase pivot identification, and risk assessment on the BC, supporting natural language interface invocation;

the micro-model scoring performs multi-dimensional model scoring on standardized log fields, generating threat scoring fields to support BC modeling and AI reasoning;

the expert feedback and knowledge write-back mechanism feeds back BCs and phase tags confirmed by human experts to a knowledge base, forming a sustainably evolving AI knowledge memory layer.

Further, in the micro-model scoring, the log data processing and scoring analysis method performs external analysis processing on structured log fields and uniformly writes them to a database, including the following steps:

step S21: log preprocessing: upon receiving raw log data, utilizing local script programs, applications, or equivalent modules to perform field extraction, data cleaning, and

normalization operations;

step S22: interface invocation: sending the preprocessed log fields in a structured format via an API interface to one or more external analysis services;

step S23: service processing: the external analysis services includes micro-model services, scoring engines, feature extractors, tag classifiers, or rule-based evaluators, used to perform processing operations such as behavior evaluation, threat scoring, field statistics, and context parsing;

step S24: result reception and encapsulation: uniformly receiving structured results returned by the analysis services and performing encapsulation processing;

step S25: database writing: writing the processed result data to predefined fields in the database for use in BC modeling, attack phase identification, visual presentation, or knowledge base updates.

Further, in the field normalization module, processing security logs based on a ternary field mapping includes the following steps:

step S31: collecting log data from security devices and extracting raw fields srcip and dstip;

step S32: constructing a ternary structure external_ip→srcip→dstip, wherein when the log source is a border device directly accessible from the public network, mapping the original srcip to external_ip, mapping the original dstip to a logical srcip, and retaining the original srcip field content without using it for BC modeling.

Further, the expert feedback and knowledge write-back mechanism comprises a structured BC knowledge base; the structured BC knowledge base writes verified attack BCs and their associated information into the knowledge base, and performs automatic semantic judgment and BC similarity retrieval based on the knowledge base;

in the expert feedback and knowledge write-back mechanism, the expert determines a security event through an AI-driven investigation process and feeds back knowledge, manually triggers a comparison process to compare the current BC under investigation with historical BCs stored in the knowledge base for similarity, assisting in judging as a similar attack path or phase pivot risk; the comparison process operates independently of the AI automatic reasoning process;

the knowledge base comprises a BC reproduction library, a field semantic parsing library, a risk tag and intent library, a graph relationship database, a path semantic coordinate library, and other optimization information knowledge bases; the path semantic coordinate library stores the structural position, phase tag, behavior coordinates, pivot point identifier, and pivot intensity score of each node in the BC;

the structured BC knowledge base utilizes structural vectors, semantic tags, or graph structure overlap for similarity matching, performs high-confidence attack path comparison and phase pivot reasoning based on historical BCs, and uses the matching results as input to the artificial intelligence model or as auxiliary reference for security analysts;

the structured BC knowledge base and the artificial intelligence model form a closed-loop interaction mechanism; during the training, update, and inference stages of the artificial intelligence model, the artificial intelligence model dynamically depends on structured BC data, path semantic coordinates, pivot point identifiers, and phase pivot intensity scoring information within the knowledge base.

The advantages of the invention are as follows.

The invention collects multi-source log data, performs field normalization and entity unification, constructs BC structures based on time series, graph structures, etc., embeds enhancement information such as node risk scores and behavior coordinates, achieving cross-platform, cross-system log BC integration and complete attack path reconstruction, solving problems of log fragmentation and information omission, and providing a unified data foundation for security investigation.

The invention utilizes AI engines such as large language models to convert natural language questions into structured queries, supports semantic sessions with continuous questioning and interactive behavior path evolution, achieving intelligent Q&A interaction, lowering the syntax barrier for security investigation, enabling analysts to efficiently obtain results like attack paths and risk prompts without mastering specific syntax.

The invention, through the expert feedback and knowledge write-back mechanism, writes verified BCs and tags into a structured knowledge base, combined with the micro-model scoring performing multi-dimensional scoring of logs, achieving dynamic accumulation and reuse of historical attack knowledge, improving detection capabilities for

unknown attacks and advanced threats, and solving the problems of traditional platforms relying on static rules and delayed responses.

BRIEF DESCRIPTION OF ACCOMPANY DRAWINGS

Drawings forming a part of the invention are used to provide further understanding of the invention, making other features, objectives, and advantages of the invention more apparent. Schematic embodiment drawings of the invention and their descriptions are used to explain the invention and do not constitute an improper limitation to the invention.

In the drawings:

FIG. 1 is a schematic diagram of the closed loop of the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 2 is a module interaction diagram of the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 3 is a schematic diagram of the principles of BC construction and AI context analysis in the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 4 is a schematic diagram of the principles of expert confirmation and knowledge base write-back in the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 5 is a schematic diagram of the principles of the micro-model scoring in the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 6 is a schematic diagram of natural language input and output feedback in the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

FIG. 7 is a comparison diagram of mapping methods in the method for semantic BC reconstruction and AI-driven security investigation in Embodiment 1 according to the invention.

SPECIFIC EMBODIMENT OF THE INVENTION

In order to make the objectives, technical solutions, and advantages of the embodiments

of the invention clearer, the technical solutions in the embodiments of the invention will be described clearly and completely hereinafter with reference to the drawings in the embodiments of the invention. Obviously, the described embodiments are part of the embodiments of the invention, rather than all of the embodiments. The components of the embodiments of the invention generally described and illustrated in the drawings herein may be arranged and designed in various different configurations.

The invention will be introduced in detail hereinafter through specific embodiments to enable better understanding, but the following embodiments do not limit the protection scope of the invention.

Embodiment 1

As shown in FIGS. 1-2, a method for semantic BC reconstruction and AI-driven security investigation, including the following steps:

step S11: collecting log data from multiple security devices or systems, performing field normalization for the purpose of unified behavior subject modeling;

step S12: constructing a BC structure associated with an entity, and the BC structure is a time series, graph structure, tag chain structure, or equivalent path structure;

step S13: receiving a user-input natural language security investigation question;

step S14: mapping the natural language question to a query target related to the BC structure using an artificial intelligence engine;

step S15: performing semantic reasoning, path search, or context analysis operations on the query target based on the BC structure;

step S16: outputting a structured or unstructured query result, including path diagrams, node lists, behavior segments, risk prompts, expert feedback information, or suggestive results.

Further, in step S11, the log data undergoes field normalization, entity unification, platform alignment, and identifier aggregation during the collection process, achieving cross-platform and cross-system log BC integration.

Further, as shown in FIG. 3, in step S12, the BC structure is established based on log chronological order, operational association, tag linkage, causal dependency, or knowledge graph paths, supporting graphical display and real-time or persistent storage of behavior

nodes; the BC structure embeds enhancement information including node risk scores, structural path weights, behavior intent tags, anomaly confidence levels; pivot point identifiers: the pivot point identifiers are used to mark key nodes indicating changes in attack phases within the BC; behavior coordinates: the behavior coordinates record the phase position, role attribution, and temporal sequence of each behavior within the chain path; phase pivot intensity scores: the phase pivot intensity scores are used to measure the credibility of pivots between nodes, comprehensively considering temporal continuity, role consistency, and behavior pattern deviation; the enhancement information is used to optimize the artificial intelligence model's path ranking, anomaly identification, and phase pivot reasoning capabilities.

Further, in step S12, the BC structure is either a pre-constructed structure or a dynamically aggregated collection of event segments formed during a query process based on user-specified entity identifiers, time ranges, or behavior characteristic conditions; the collection of event segments has relevance or logical sequence at the semantic level; the BC structure serves as the foundation for triggering a structured event investigation process, including BC reconstruction for specific IPs or users, comparison of similar attack paths, behavior attribution, and collaborative analysis tasks; BC node composition includes communication actions, login behaviors, file operations, configuration changes, process invocations, URL accesses, API requests, or other identifiable security behavior events.

Further, in step S14, the artificial intelligence engine comprises a large language model, NLP, a graph neural network, a rule-learning hybrid engine or an equivalent semantic reasoning system; the artificial intelligence engine converts the natural language question into a structured query template, graph structure traversal request, or SQL statement, and combines Prompt engineering to achieve efficient mapping between natural language and structured queries, supporting the semantic reasoning process.

Further, as shown in FIG. 6, the user continuously asks questions in natural language, and the artificial intelligence maintains a semantic session based on the BC context state, supporting context zooming, chain structure focus switching, reverse question generation, and interactive behavior path evolution, and supporting cross-turn context memory and expert feedback-driven dialogue evolution capabilities.

Further, in step S16, the output result includes attack path diagrams, structural node lists, behavior trajectories within a time window, matching segments, risk explanations, reasoning failure prompts, or semi-structured results requiring manual confirmation.

Further, a system for semantic BC reconstruction and AI-driven security investigation, comprising: a field normalization module, a BC modeling module, an AI semantic reasoning module, a micro-model scoring, and an expert feedback and knowledge write-back mechanism; the field normalization module, BC modeling module, AI semantic reasoning module, micro-model scoring, and expert feedback and knowledge write-back mechanism form a logical feedback loop, supporting the BC construction, attack path reconstruction, and artificial intelligence semantic analysis process;

the field normalization module performs field normalization, entity normalization, platform alignment, and key field mapping on collected multi-source log data;

the BC modeling module constructs a BC structure based on chronological order, semantic consistency, and role consistency, serving as the input foundation for AI reasoning and path reconstruction;

the AI semantic reasoning module performs path reasoning, phase pivot identification, and risk assessment on the BC, supporting natural language interface invocation.

This module, based on a locally deployed large language model, enables users to initiate behavioral investigations directly through natural language, eliminating the need to write SQL statements. The system generates structured SQL through prompt engineering. The specific process is as follows: a standard prompt template is encapsulated; when the user inputs a natural language question, the system invokes a locally deployed LLM (such as DeepSeek) to perform semantic parsing of the question; combined with technical frameworks like LangChain, it automatically translates the natural language question into a corresponding SQL query statement to query relevant behavioral data in the EBD (entity behavior database). Experts conduct interactive questioning and result verification with the AI through an integrated chat platform. The query is sent to the database via a post request for execution, and the results are returned to the user. Examples of platforms include AI chat platforms like OpenChat, self-built web interfaces, IM tool-embedded dialog boxes, or AI dialog modules built into the security operations platform. For instance, if the user asks, "what actions did IP

1.1.1.1 perform between 14:00 and 18:00 on March 26, 2025, and were there any abnormal behaviors?", the system will translate this into a valid SQL query to retrieve structured logs from the relevant data table in that IP's EBD. The AI will then perform comprehensive reasoning based on multidimensional data, including the behavior sequence from the BC modeling layer, attack patterns from the historical knowledge base, micro-scoring results, and path coordinate information. It returns the results in a structured or graphical format to the chat platform interface for further expert analysis and confirmation. Wherein the micro-scoring result can be the Model_X field, the coordinate information includes 0s and BPP (behavior pivot points). A BCP (behavior coordinate point) is the smallest semantic unit in the BC, marking the position and semantic information of a behavior event within the chain. A BPP is a key semantic node in the BC where a phase pivot occurs, typically located at the "tactical turning point" of an attack path.

Experts can achieve contextually progressive investigation during the dialogue by appending natural language questions. The AI maintains the BC context state, supporting chain structure focus switching, phase pivot reasoning, and similarity comparison, which ultimately assists experts in completing the full investigation loop and writing confirmed results back to the structured BC knowledge base, continuously optimizing the AI's reasoning capabilities and knowledge accumulation. This module realizes an intelligent Q&A-style investigation experience where "human language is the query", significantly lowering the barrier of security analysis.

Based on the behavioral logs for each role in the "EBD", the AI automatically reconstructs BCs in chronological order. When the user initiates a query via natural language, the AI retrieves all logs for the specified role within the specified time from the EBD and constructs the BC by sorting them chronologically. For example, using 300 logs as a sample return, the system can identify potential attack paths such as "Login Failure→Login Success →Upload→Command Execution→External Connection". This chain-building method does not rely on fixed rules or device characteristics but constructs logical behavior sequences through contextual understanding and semantic composition. The system automatically integrates multiple log sources of host, traffic, security, etc., possesses the capability of building chains across time periods and from multi-source heterogeneous data, and can

effectively identify complex tactics such as asynchronous behaviors and delayed attacks. After the AI judges and analyzes the constructed BC, it returns the final analysis results for expert judgment or further human-machine interaction. It is recommended to attach a score value, log source, and field explanation to each node in the BC to enhance the traceability of the analysis.

the micro-model scoring performs multi-dimensional model scoring on standardized log fields, generating threat scoring fields to support BC modeling and AI reasoning;

the expert feedback and knowledge write-back mechanism feeds back BCs and phase tags confirmed by human experts to a knowledge base, forming a sustainably evolving AI knowledge memory layer.

Further, as shown in FIG. 5, in the micro-model scoring, the log data processing and scoring analysis method performs external analysis processing on structured log fields and uniformly writes them to a database, including the following steps:

step S21: log preprocessing: upon receiving raw log data, utilizing local script programs, applications, or equivalent modules to perform field extraction, data cleaning, and normalization operations;

step S22: interface invocation: sending the preprocessed log fields in a structured format via an API interface to one or more external analysis services;

step S23: service processing: the external analysis services includes micro-model services, scoring engines, feature extractors, tag classifiers, or rule-based evaluators, used to perform processing operations such as behavior evaluation, threat scoring, field statistics, and context parsing;

step S24: result reception and encapsulation: uniformly receiving structured results returned by the analysis services and performing encapsulation processing;

step S25: database writing: writing the processed result data to predefined fields in the database for use in BC modeling, attack phase identification, visual presentation, or knowledge base updates.

The system uses Python to call Flask/FastAPI interfaces connecting to pre-trained TensorFlow models to perform micro-threat scoring on each log. This score value is written back to a designated field in the database, providing a quantitative basis for subsequent

analysis. First, the system uses Python and TensorFlow to train and generate custom anomaly scoring models based on requirements. These models cover multiple dimensions, including models using autoken and random forests trained on normal data to assess whether a log is normal or anomalous, malicious detection models trained on datasets with malicious behavior features for single-statement detection, time-dimension models based on working hours and non-working hours, special character count models, and access anomaly path models trained by statistically analyzing all API service paths to identify abnormal paths. The system packages these machine learning-generated models of H5 and Keras into microservices using Flask/FastAPI. Python scripts periodically access these microservices to score each piece of data pushed into the database and write the results back to designated fields in the SQL database. Furthermore, the system supports model fusion judgment and explainable output, such as future integration with SHAP/LIME. This module is the core support for data-driven analysis.

As shown in FIG. 7, in the field normalization module, processing security logs based on a ternary field mapping includes the following steps:

step S31: collecting log data from security devices and extracting raw fields srcip and dstip;

step S32: constructing a ternary structure external_ip→srcip→dstip, wherein when the log source is a border device directly accessible from the public network, mapping the original srcip to external_ip, mapping the original dstip to a logical srcip, and retaining the original srcip field content without using it for BC modeling.

This module is responsible for collecting logs from various security devices, performing normalization processing, and constructing a EBD. Log normalization is performed first: relevant fields are extracted from logs of different sources and mapped to standard key fields including, but not limited to: Date, Time, external_ip, srcip, dstip, source_from, Event_ID, Detail, and Model_X. A unique identifier (Log_id) is then generated for the log in the database. The field set can be adjusted according to specific implementation scenarios to ensure structural and semantic consistency between fields, supporting subsequent BC modeling and AI semantic analysis processes.

The external_ip field is used to determine whether to apply the ternary mapping

mechanism based on the scenario of the srcip field in the original border device log; if judged to be an external address, it records the corresponding external access IP address; the srcip field represents the internal behavior subject; the dstip field records the target address of the behavior; the source_from field annotates the source device information of the log; the Detail field encapsulates the content of the remaining original fields, stored in JSON format; the Model_X field stores micro-model scoring results, and these results are obtained by calling external analysis services via API and are written back to the database after retrieval. The external analysis services include, but are not limited to, micro-model services, scoring engines, feature extractors, tag classifiers, or rule-based evaluators, used to perform processing operations such as behavior evaluation, threat scoring, field statistics, and context parsing.

Subsequently, a relational database is initialized. Database tables are created separately with SRCIP as the role subject, constructing the EBD.

In this embodiment, filebeat, logstash, and python are used to write logs into their respective database tables within each role's database based on the log subject. Logs are sorted by time. By mapping to standard key fields (including but not limited to Date, Time, external_ip, srcip, dstip, source_from, Event_ID, Detail, and Model_X) and generating a unique identifier (Log_id) for each log, logs from different sources acquire a unified structural semantics for subsequent processing.; the added source_from field aids in later modeling and traceability. The role database uses srcip as the primary key; each role corresponds to one table, storing related logs sorted in chronological order. This establishes a security data model centered on the "behavior subject" rather than the "event". This structure inherently supports attack traceability, BC reconstruction, and behavior pattern learning.

The system first collects raw logs from various security devices, standardizes field formats to achieve heterogeneous data normalization, and clarifies the subject-object attribution relationship of behaviors by introducing the ternary field mapping, which lays a unified data foundation for subsequent BC modeling and AI semantic analysis. The standardized logs then enter the micro-model threat scoring layer. By integrating multi-dimensional threat detection models, each log undergoes fine-grained scoring, generating scoring fields like Model_X to assist subsequent BC modeling and risk

discrimination. The scored log data is aggregated by role entity and enters the BC modeling layer. The system reconstructs BCs based on chronological order, behavioral semantic consistency, and role continuity, forming structured behavior sequences, e.g., "A1→A2→A3→A4".

The modeled BC is further processed by the AI semantic reasoning layer. The system supports experts asking questions to the AI via natural language input, e.g., "did IP [X] exhibit any abnormal login behavior during time period [Y]?". The AI converts the natural language into structured result queries (e.g., using langchain and LLM input; the result is output to the LLM for analysis). Experts interact with the AI system using natural language. The AI understands and analyzes whether the sequence of actions for the specified behavior subject within the specific time period contains suspicious features. During the analysis, the AI engine comprehensively utilizes multidimensional data resources such as the historical BC reproduction library, micro-model scoring library, and key field parsing library for semantic reasoning and judgment. The micro-model scoring library includes fine-grained threat scores for individual logs or behavior nodes, e.g., the model4 field in MySQL, where 0% represents normal and 100% represents malicious, serving as a reference for anomalous behavior.

The key field parsing library contains fields like Host_eventid, such as the eventid field in Windows security logs, used to assist in determining host behavior intent and phase attribution. Combining the BC, historical knowledge, and contextual information, the system performs path reasoning, pivot point identification, and anomaly assessment, generating analysis results and reasoning justifications.

After the AI engine completes the preliminary analysis of the BC, the system submits the analysis results to experts for review, forming a "human-machine collaboration" judgment loop. Based on the AI reasoning results and their own analysis, experts confirm whether an attack behavior or anomalous BC exists. If confirmed, the key BP (behavior chain path) exhibiting attack characteristics and its corresponding features are written into the AI knowledge base, including modules like the BC reproduction library, risk tag & intent library, and path semantic coordinate library. This forms an accumulation of empirical data for subsequent analysis and model optimization. For example, a key BP could be: "A1 brute force→A2 login success→A3 upload behavior→A4 command execution→A5 external connection

behavior". As this knowledge base continuously evolves, it can be further upgraded to a graph database structure to support efficient attack similarity comparison and knowledge reasoning, enhancing the system's overall intelligent investigation and decision support capabilities.

Further, as shown in FIG. 4, the expert feedback and knowledge write-back mechanism comprises a structured BC knowledge base; the structured BC knowledge base writes verified attack BCs and their associated information into the knowledge base, and performs automatic semantic judgment and BC similarity retrieval based on the knowledge base.

In the expert feedback and knowledge write-back mechanism, the expert determines a security event through an AI-driven investigation process and feeds back knowledge, manually triggers a comparison process to compare the current BC under investigation with historical BCs stored in the knowledge base for similarity, assisting in judging as a similar attack path or phase pivot risk; the comparison process operates independently of the AI automatic reasoning process.

the knowledge base comprises a BC reproduction library, a field semantic parsing library, a risk tag and intent library, a graph relationship database, a path semantic coordinate library, and other optimization information knowledge bases; the path semantic coordinate library stores the structural position, phase tag, behavior coordinates, pivot point identifier, and pivot intensity score of each node in the BC.

The structured BC knowledge base utilizes structural vectors, semantic tags, or graph structure overlap for similarity matching, performs high-confidence attack path comparison and phase pivot reasoning based on historical BCs, and uses the matching results as input to the artificial intelligence model or as auxiliary reference for security analysts.

The structured BC knowledge base and the artificial intelligence model form a closed-loop interaction mechanism; during the training, update, and inference stages of the artificial intelligence model, the artificial intelligence model dynamically depends on structured BC data, path semantic coordinates, pivot point identifiers, and phase pivot intensity scoring information within the knowledge base.

The above has provided a detailed description of specific embodiments of the invention, serving as examples. The invention is not limited to the specific embodiments described hereinabove. For those skilled in the art, any equivalent modifications and substitutions made

to the invention also fall within the scope of the invention. Therefore, equivalent transformations and modifications made without departing from the spirit and scope of the invention should be encompassed within the protection scope of the invention.