AI Spark Program Technical Tackling List

1. Research on Key Technologies for Artificial Intelligence Lifecycle Governance
2. Research on Key Algorithms for Intelligent Multi-Source Knowledge Graph Construction in the Security Domain Based on Large Language Models
3. Development of High-Performance Operators for Deterministic Inference Results
4. Research and Practice on Key Technologies for AI-Driven Automated Network Attacks and Defenses
5. Innovation and Security Governance Research on Full-Chain Embodied Intelligence
6. Research on Key Technologies for Secure Training and Inference of Large Models Supported by Confidential Computing
7. Research on Zero-Intrusion Full-Chain Traceability and Cross-Modal Digital Watermarking Technologies in Heterogeneous Environments
8. Research on Key Technologies for Trusted Data Synthesis and Privacy Security

---

1. Research on Key Technologies for Artificial Intelligence Lifecycle Governance

I. Research Direction and Overall Objective

Focus on the prominent governance challenges faced by artificial intelligence systems in the whole process of R&D, training, deployment, application and continuous evolution, and carry out systematic and implementable technological innovation research. Through original breakthroughs in methodologies, algorithms, system frameworks and engineering solutions, build verifiable, transferable and sustainable AI governance capabilities, and provide a technical foundation for the development of safe, trustworthy and controllable artificial intelligence.

II. Project Background and Necessity

With the rapid development of generative AI and agent technologies, the scale of AI models continues to expand and application scenarios are constantly extended, leading to AI risks showing systematic, cross-stage and cumulative characteristics. Hallucinatory output, corpus contamination, identity forgery, uncontrollable behaviors of agents, and the lack of a unified and effective security evaluation system have become key bottlenecks restricting the high-quality development of AI. It is urgent to break through a number of cutting-edge governance technologies, form a general framework,

reusable tools and verification mechanisms, and promote source innovation and global competitiveness in the field of AI governance.

III. Key Research Tasks
(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)
1. Hallucination Suppression and Robust Generation Technology
- Methods for hallucination recognition, root cause analysis and real-time suppression of large models
- Robust generation mechanism based on knowledge enhancement and verifiable reasoning
- Multi-modal hallucination detection and semantic consistency verification technology

2. Corpus Security and Trustworthy Data Construction
- Technology for training data source audit and risk identification
- Mechanisms for data contamination defense, copyright identification and sensitive information elimination
- Trustworthy and high-quality corpus construction and evaluation system for model training

3. Identity Authentication and Behavior Constraints of Models and Agents
- Anti-forgery model identity fingerprint, model watermarking and tracing technology
- Monitoring of agents' autonomous behaviors, unauthorized access detection and boundary constraint mechanisms
- Methods for security coordination and conflict control of large-scale agent groups

4. Security Evaluation and Governance Framework for AI Systems
- Multi-dimensional security evaluation index system for large models and agents
- Automated, scenario-based and adversarial security evaluation tool chain
- Design of interpretable, verifiable and traceable governance technologies and system architectures

5. Integrated Governance Solution for the Whole Life Cycle
- System-level governance framework covering data, models, inference and applications
- In-depth integration of governance mechanisms with large model training/inference frameworks
- Engineerable and deployable governance tool suite

IV. Outcome Requirements

1. Form no less than one set of key technologies, methods or system frameworks with originality and engineering value
2. Submit verifiable prototypes, prototype systems or tool chains
3. Release technical documents, security evaluation reports or standard proposals
4. Support pilot verification in key industries or typical application scenarios

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

2. Research on Key Algorithms for Intelligent Multi-Source Knowledge Graph Construction in the Security Domain Based on Large Language Models

I. Research Direction and Overall Objective

Study the core construction algorithms of knowledge graph in the security field based on large language models, break through key technical bottlenecks such as multi-source heterogeneous knowledge extraction, cross-source entity alignment and knowledge conflict resolution, form a complete algorithm system and prototype system, and provide theoretical and technical support for the automated and intelligent construction of security knowledge graphs.

II. Project Background and Necessity

- Explosive growth of knowledge in the cyber security field (the number of CVE vulnerabilities increases by more than 30% annually), and manual management has four major pain points: knowledge is scattered in CVE databases, threat intelligence and other sources without a unified system; updates lag behind and fail to keep up with the evolution of threats; multi-source knowledge conflicts are difficult to handle; and the correlation analysis between vulnerabilities, attacks and defenses is weak.
- Outstanding technical bottlenecks: insufficient generalization of multi-source heterogeneous knowledge extraction, low accuracy of cross-source entity alignment, lack of intelligent resolution mechanism for knowledge conflicts, and imperfect modeling of security domain ontologies (e.g., attack chains).
- Domestic and international status quo: Foreign frameworks such as MITRE ATT&CK rely on manual maintenance, and academic research is limited to single tasks; domestic research mainly focuses on manually maintained intelligence databases, lacking an end-to-end graph construction system.
- Technology trend: Large language models (LLMs) have brought breakthroughs (F1 ≥ 90% for related tasks), but it is necessary to solve three major problems: domain prompt strategy adaptation, cross-source entity alignment, and interpretability of knowledge fusion.

III. Key Research Tasks
(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)
1. LLM-driven Security Knowledge Extraction: Study prompt engineering methods, Few-shot learning strategies and domain adaptation technologies for the security field, achieving an entity recognition accuracy ≥ 90%, recall ≥ 85%, and a relation extraction F1 score ≥ 85%.
2. Cross-source Knowledge Alignment Algorithm: Study entity alignment methods based on semantic embedding and LLM reasoning, break through challenges such as inconsistent naming and descriptive differences, achieving a cross-source entity alignment accuracy ≥ 88%.
3. Knowledge Fusion and Conflict Resolution Mechanism: Study conflict detection, credibility evaluation and intelligent resolution strategies for multi-source knowledge, achieving a conflict detection accuracy ≥ 85% and supporting efficient fusion of incremental knowledge.
4. Security Domain Ontology Modeling Method: Design a security domain ontology model covering core elements such as vulnerabilities, attack technologies, defense measures and threat actors, forming an ontology specification with ≥ 10 types of node entities and ≥ 20 types of relations.
5. Dynamic Knowledge Update Strategy: Study efficient processing algorithms for incremental data, knowledge timeliness evaluation methods and graph consistency maintenance mechanisms, supporting a knowledge update delay ≤ 24 hours.

The core algorithms researched in this project can support the following security application scenarios:
1. Threat intelligence analysis: Quickly locate the exploitation methods, affected assets and defense solutions of vulnerabilities through graph correlation analysis.
2. Attack tracing: Assist security analysts in attack path restoration and attribution analysis based on the attack chain knowledge graph.
3. Security knowledge Q&A: Support natural language queries, such as "What is the difficulty of exploiting CVE-2023-1234? What defense measures are available?"
4. Vulnerability priority assessment: Intelligently evaluate the priority of vulnerability repair by combining information such as vulnerability exploitation chains and asset exposure surfaces in the graph.

IV. Outcome Requirements
1. Form of Outcomes
- Algorithm prototype system: A Python-based core algorithm package for knowledge graph construction, including modules for knowledge extraction, entity alignment, knowledge fusion, dynamic update, etc., with clear code structure, complete documentation and strong reusability.

- Technical reports: Detailed algorithm design documents, experimental evaluation reports and research summary reports.
- Experimental dataset: Build annotated datasets for tasks such as security domain knowledge extraction, entity alignment and knowledge fusion to support algorithm evaluation and follow-up research.

2. Deliverable List: Algorithm code and documentation, research outcome documents, datasets and resources.

3. Acceptance Criteria
- Algorithm performance indicators: Knowledge extraction accuracy, entity alignment accuracy, knowledge fusion accuracy, dynamic update efficiency.
- Function completeness indicators: Support knowledge extraction from ≥ 5 types of security data sources (CVE, NVD, security papers, technical blogs, threat intelligence); support ≥ 10 types of entity and ≥ 20 types of relations; provide a complete algorithm API and call documentation; highly reusable code with modular design and flexible configuration support.
- Research outcome indicators: Submit ≥ 2 academic papers (at least 1 to CCF B-class or above conferences/journals); build an experimental dataset with ≥ 1000 annotated data entries; submit complete technical documents (algorithm design, experimental reports, API documents).
- Code quality indicators: Clear code structure in line with Python coding standards (PEP 8); unit tests for key modules with a test coverage ≥ 70%; complete README, installation documents and usage examples; open-source code or complete source code delivery (including comments).

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

3. Development of High-Performance Operators for Deterministic Inference Results

I. Research Direction and Overall Objective

Develop a set of deterministic high-performance operator libraries and execution mechanisms supporting mainstream inference frameworks to achieve:

1. Complete consistency of inference results (bitwise equality of floating-point numbers) under different concurrency levels and request sequences with the same input tensor and execution configuration;
2. Performance not lower than 95% of the corresponding non-deterministic implementation (PyTorch) on the premise of ensuring determinism;
3. Select one target platform from GPU, DCU, NPU or other domestic graphics card architectures for development and verification to ensure the

complete runnability and reproducibility of operators in the domestic independent computing power environment.

II. Project Background and Necessity

With the wide application of large models in cloud inference, online services and edge deployment, the uncertainty of inference results has become a key pain point for trustworthy AI.

In current mainstream inference frameworks such as vLLM and PyTorch, even with the same input tensor and execution configuration, minor differences may exist in the floating-point numbers of output results under different concurrency, scheduling sequences and batch merging strategies. This computational non-determinism mainly stems from the following reasons:

- Thread competition and accumulation order differences in parallel reduction;
- Changes in concurrent request scheduling and execution order;
- Non-deterministic selection of algorithm paths in underlying libraries (cuBLAS, cuDNN);
- Differences in kernel floating-point rounding and fusion optimization.

This phenomenon leads to inconsistent model results in multiple runs, making it difficult for the system to reproduce inference results, conduct accurate debugging and trustworthy verification, and seriously affects the reliability and regulatory traceability of the model. Especially in high-concurrency online service scenarios, result fluctuations not only reduce credibility but also may trigger business risks. Internationally, the open-source community has developed some deterministic operators, but there are still problems such as a more than 30% reduction in inference throughput in multi-threaded concurrent inference scenarios and the lack of a systematic solution. Therefore, it is urgent to break through the technical bottleneck of "determinism guarantee for high-performance concurrent inference", build independent and controllable deterministic operators and execution specifications, and support the construction of a trustworthy large model inference ecosystem.

III. Key Research Tasks

(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)

1. Research on Deterministic Computing Principles
    - Analysis of floating-point error propagation and rounding consistency;
    - Reduction order control;
    - Research on concurrent request scheduling consistency strategies.

2. Implementation of High-performance Deterministic Operators

- Develop deterministic kernels for core operators such as matrix multiplication, LayerNorm, Softmax and Attention;
- Optimize performance using technologies such as CUDA Graph, Warp-Level Reduction and Memory Barrier;
- Select one target platform from GPU, DCU, NPU or other domestic graphics card architectures for development and verification to ensure the complete runnability and reproducibility of operators in the domestic independent computing power environment.

3. Concurrent Deterministic Execution Mechanism
- Implement execution plans with fixed scheduling sequences and dynamic batching consistency strategies;
- Eliminate thread randomness to achieve "consistent results with different concurrency levels".

4. Deterministic Verification and Performance Testing Platform
- Develop an automated verification framework for bitwise comparison under different concurrency levels and request sequences;
- Generate deterministic reports and performance benchmarking results.

5. Typical Application Verification and Integration
- Verify determinism on models such as Qwen and GLM;
- Output application performance reports and standardized development interface documents.

IV. Outcome Requirements
1. Form of Outcomes
- Independent and controllable deterministic high-performance operator library (source code and compilation package) that can run stably on the selected computing power platform (GPU/DCU/NPU/domestic graphics card);
- Concurrent deterministic verification and performance testing platform;
- Technical standards and development documents;
- Large model application verification report.

2. Deliverables and Acceptance Criteria
- Deterministic operator library: ≥ 10 high-frequency operators; bitwise consistent results under different concurrency levels and request sequences with the same input and configuration.
- Verification platform: Support multi-concurrency testing and automated comparison; output deterministic detection reports.
- Technical documents: Complete API, execution specifications and environmental requirement descriptions.
- Application report: ≥ 2 large model inference examples demonstrating determinism and performance indicators.

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

4. Research and Practice on Key Technologies for AI-Driven Automated Network Attacks and Defenses

I. Research Direction and Overall Objective

The core objective is to build an intelligent agent that can understand and execute tactical and technical processes similar to MITRE ATT&CK, realize an automated closed loop from vulnerability discovery and exploitation to link attacks, and solve the pain points of current security tools such as "high false alarm rate, difficult exploitation and broken links".

II. Project Background and Necessity

With the increasing complexity and intelligence of cyber attack methods, traditional rule-based and manual penetration testing/vulnerability discovery are difficult to cope with massive assets and unknown threats. This project aims to collect AI-based automated cyber offense and defense technologies, build an intelligent agent that can understand and execute tactical and technical processes similar to MITRE ATT&CK, realize an automated closed loop from vulnerability discovery and exploitation to link attacks, and solve the pain points of current security tools such as "high false alarm rate, difficult exploitation and broken links".

III. Key Research Tasks

(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)

1. AI-driven Smart Fuzzing and Vulnerability Discovery

1.1 Corresponding ATT&CK phases: Reconnaissance, Initial Access

1.2 Core pain points: Traditional Fuzzing has strong blindness, low code coverage and is ineffective against logical vulnerabilities.

1.3 Technical requirements:

- Use AI models (e.g., LLM or code large models) to understand the target source code or binary files and generate high-quality seed mutation strategies.
- Realize semantics-aware Fuzzing for specific protocols or application interfaces, rather than simple random byte flipping.
- Automatically identify crashes and conduct preliminary exploitability assessment (Triage).

2. Automated Vulnerability Scanning and Exploit Generation (Auto-Exploit Generation)

2.1 Corresponding ATT&CK phases: Execution, Persistence, Privilege Escalation

2.2 Core pain points: High false alarm rate of scanners; POC (Proof of Concept) is often difficult to convert into EXP (Exploit); difficult to adapt to different environments.

2.3 Technical requirements:

- Build an AI-based vulnerability verification engine to automatically write or modify Payloads to adapt to the target environment (e.g., bypass WAF, adapt to different OS versions).
- Use reinforcement learning (RL) or planning algorithms to automatically attempt privilege escalation paths in a simulated environment.
- Practical requirements: It must be proven that the tool can complete the fully automated process of "vulnerability discovery -> EXP generation -> Shell acquisition".

3. Vulnerability Chaining Based on Offense and Defense Knowledge Graph

3.1 Corresponding ATT&CK phases: Lateral Movement, Collection, Impact

3.2 Core pain points: Single vulnerabilities have limited harm; lack the ability to combine multiple low-risk vulnerabilities into high-risk attack paths.

3.3 Technical requirements:

- Build an offense and defense knowledge graph to endow the AI agent with "tactical planning" capabilities.
- Realize multi-hop attack decision-making: e.g., enter the intranet through Web vulnerabilities -> scan weak passwords in the intranet -> move laterally to the database.
- Scenario requirements: Automatically plan and execute a complete attack chain containing at least 3 attack steps in a given target environment.

IV. Outcome Requirements

1. Core Assessment Indicators

- ATT&CK coverage: The tool or algorithm must cover at least 3 tactical phases and 10 technical points in the MITRE ATT&CK matrix.
- Automation level: The number of manual interventions in the attack/detection process must be < 2 (0 intervention is ideal).
- Accuracy: The false alarm rate of vulnerability verification must be < 5%; the missed alarm rate of vulnerability verification must be < 10% (based on standard target environment testing).
- Efficiency indicator: The automated decision-making and detection time for assets in a standard Class C network segment must be reduced by more than 50% compared with traditional manual penetration testing.
- Interpretability: The system must output a complete attack/defense path map, clearly marking the ATT&CK technical ID (e.g., T1190) used in each step.

2. Deliverable Requirements

- Runnable prototype system/tool: Provide a Docker image or executable program for retesting in an isolated environment.
- Technical white paper: Elaborate on the AI model architecture (e.g., the LLM and reinforcement learning algorithms used), the source of training datasets, and the mapping logic for the corresponding offense and defense phases.
- Test report: Actual test reports based on no less than 3 typical target environments (e.g., DVWA, Metasploitable3 or self-built real network simulation environments).

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

5. Innovation and Security Governance Research on Full-Chain Embodied Intelligence

I. Research Direction and Overall Objective

Focus on the key links of embodied intelligence systems in the full link such as perception, decision-making, action execution, system upgrade and multi-agent collaboration, and carry out systematic technological innovation research. Through methodology, algorithm, tool chain and framework design, realize efficient perception, robust decision-making, accurate execution and controllable collaboration of embodied intelligence in complex real environments.

In the full-link governance, security and compliance are important links, including control instruction verification, behavior compliance monitoring, OTA upgrade security, boundary risk prevention and control, etc. The project aims to form a verifiable and engineerable solution, realize the safe, trustworthy, deployable and efficient operation of embodied intelligence systems, and provide technical support for the landing of intelligent agents in application scenarios such as industry, services and scientific research.

II. Project Background and Necessity

Embodied intelligence systems independently perceive, reason and execute tasks in the real environment, bringing improvements in efficiency and functions, but also introducing multi-dimensional technical challenges:

- Perception and understanding: Multi-modal information fusion, dynamic scene understanding, object and relationship prediction.
- Action planning and execution: High-precision control, robust actions in uncertain environments, multi-task collaboration.

- System upgrade and boundary control: OTA upgrade vulnerabilities, software and firmware security, agent behavior boundary constraints.
- Security and behavior monitoring: Abnormal detection of control instructions, behavior compliance verification, traceability and interpretability.

At present, embodied intelligence systems still lack a full-link technical system and security governance solution. It is urgent to break through the key technologies of perception, decision-making, execution and security management, form implementable, verifiable and deployable solutions, and promote the safe and reliable application and industrialization of embodied intelligence in complex scenarios.

III. Key Research Tasks
(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)
1. Scene Perception and Understanding
    - Multi-modal perception fusion and environment modeling technology
    - Semantic understanding and task reasoning in dynamic environments
    - Object recognition, relationship understanding and prediction methods in complex interactive scenarios

2. Action Planning and Execution Optimization
    - High-precision motion control and path planning algorithms
    - Robust execution and adaptive strategies in uncertain environments
    - Multi-task collaboration and conflict scheduling methods

3. Behavioral Security Detection and Monitoring
    - Abnormal detection of control instructions and behavior compliance verification
    - Rule-based and data-driven security monitoring framework
    - Traceable, interpretable and verifiable mechanisms for agent behaviors

4. OTA Upgrade and System Boundary Security
    - Security detection and vulnerability protection during the upgrade process
    - Security management strategies for agent software and firmware
    - Security boundary constraints in the collaboration of multi-agent systems

5. Full-chain Security Governance of Embodied Intelligence
    - End-to-end security governance framework from perception and decision-making to execution
    - In-depth integration of security strategies and control mechanisms with system architecture

- Engineerable verification and security evaluation tool chain in simulated environments
- Provide verification solutions deployable in simulated and real scenarios

IV. Outcome Requirements
1. Full-link technological innovation: Propose original methods, algorithms or frameworks for embodied intelligence systems in perception, decision-making, action execution and multi-agent collaboration.
2. Security governance capabilities: Realize verifiable solutions for control instruction verification, behavior compliance monitoring, OTA upgrade security and boundary risk prevention and control.
3. Engineering and deployment: Build a prototype system or tool chain verifiable in simulated or real scenarios, supporting phased deployment and iterative optimization.
4. Documentation and standardization: Submit technical documents, evaluation reports or reusable tool templates and method guidelines.

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

6. Research on Key Technologies for Secure Training and Inference of Large Models Supported by Confidential Computing
I. Research Direction and Overall Objective
Through breakthroughs in confidential computing technology, realize practical, cost-controllable technical capabilities that can provide security protection for data and models in the pre-training, fine-tuning and inference phases of large models.

II. Project Background and Necessity
The performance of large language models (LLMs) is highly dependent on the scale and diversity of training data. However, the most valuable data — such as medical images, financial transaction records and personal identity information — is strictly restricted in their respective "data silos" due to compliance requirements, commercial confidentiality or privacy concerns, making it impossible to achieve cross-domain circulation and fusion analysis.

The cost of training and inference for large models is often tens of millions of US dollars, and the training data, inference data and model weights are the core intellectual property of developers. This process involves multiple participants such as data providers, model providers and algorithm providers, each with different security considerations. When conducting training or providing inference services on the cloud, it is necessary to ensure that the

models and data provided by all parties will not be stolen by potential malicious attackers including cloud service providers and malicious privileged users.

III. Key Research Tasks

(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)

1. Security Architecture for Large Model Training and Inference Supported by Confidential Computing

- Design of security evaluation model and security system architecture in the confidential computing environment
- Data security isolation and data compliance management mechanism for multi-tenant services in the confidential computing environment
- Full life cycle threat modeling and security strategies for large model training and inference in the confidential computing environment

2. Fusion Technology of Confidential Computing and Large Model Frameworks

- In-depth fusion scheme of mainstream model underlying scheduling frameworks in the confidential computing environment
- Orchestration and abstraction of large model training/inference pipelines in the confidential computing environment
- Integrated development and debugging tool chain for confidential computing + large models for developers

3. Performance Bottleneck Optimization of Confidential Computing Technology

- Analysis and optimization of large model computing and memory overhead in the confidential computing environment
- Optimization of large model I/O and network bottlenecks in the confidential computing environment
- Hardware acceleration and system-level collaborative optimization

4. Interconnection Technology of Confidential Computing in Large Model Scenarios

- Trustworthy, secure interconnection and collaborative scheduling between heterogeneous confidential computing platforms

IV. Outcome Requirements

1. Core Assessment Indicators

To ensure the applicability of the technology, all directions are required to deliver runnable prototype systems, technical reports and reproducible test materials; the indicators cover security system and evaluation, performance

comparison and optimization, interconnection and portability, as well as engineering quality and demonstration landing.

- Security architecture and functional completeness: Build a security system and evaluation model for confidential computing scenarios, form an implementable solution covering "data — code — runtime — access control — audit and traceability", and provide prototype implementation and verification results.

- Performance and overhead control: Propose and implement optimization solutions for the key performance bottlenecks of mainstream confidential computing technology stacks (e.g., memory encryption overhead, I/O, network, proof process, confidential containers/virtualization), provide reproducible comparative verification, and prove that the solution has an engineering available performance level.

- Interconnection and portability: Form docking capabilities across platforms, environments and implementations, support standardized integration with existing business systems/data infrastructure, reduce migration costs, and ensure that the solution can be deployed and run on different confidential computing hardware and software stacks.

- Others: To encourage the participation of different technical routes and innovative solutions, in addition to the above directions, the applicants are allowed and encouraged to propose differentiated technical paths and extended capabilities on the premise of not reducing security and usability, and deliver tangible outcomes.

2. Deliverable Requirements

2.1 Runnable Prototype System

- Provide a deployable prototype (including installation package/mirror/deployment script) that can complete deployment and demonstration in the specified hardware/cloud environment.

- Output core module list, interface description, operation manual and troubleshooting manual.

2.2 Reviewable Technical Report

- System architecture and threat model, security design, key mechanism description, implementation details, performance evaluation methods and results, boundaries and limitations, applicable scenarios and risk prompts.

2.3 Reproducible Test and Acceptance Materials

- Performance benchmarks and security test cases, test datasets/generation scripts, test environment configuration descriptions.

- Form an acceptance demonstration script: one-click deployment → function demonstration → security verification → performance comparison → interface interconnection.

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

7. Research on Zero-Intrusion Full-Chain Traceability and Cross-Modal Digital Watermarking Technologies in Heterogeneous Environments

I. Research Direction and Overall Objective

Aim to solve the link breakage problem of data flow between "applications — databases — files" for enterprises in a "black box" environment where application source code is unavailable (no code audit permission). The core objective is to build a non-intrusive traceability system, use the logical reasoning and causal inference capabilities of large language models (LLMs) to realize dynamic lineage reconstruction based on traffic and logs; at the same time, combine cross-modal watermarking technology to open up the tracking path from structured data (DB) to unstructured data (documents/images), and realize the visualization, management and traceability of the whole life cycle.

II. Project Background and Necessity

1. Practical pain point (zero-intrusion demand): In the security construction of modern enterprises, security teams often find it difficult to intervene in the development process or obtain source code permissions, leading to the failure of traditional static application security testing (SAST). In a microservice architecture, database audit logs and API gateway logs are fragmented, making it impossible to restore the real data flow on the business side.
2. Cross-modal regulatory gap: When data is queried and exported from the database as Excel, PDF or screenshots, the original security marks are lost, resulting in the inability to trace the source after leakage.
3. Technology combination point: Traditional rule-based log correlation has a high false alarm rate. Using the powerful semantic understanding capabilities of LLMs, it is possible to accurately infer the causal relationship between "API requests" and "SQL execution", and deeply understand the business intentions behind data flow.

III. Key Research Tasks

(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)

1. Zero-intrusion Lineage Reconstruction Based on Full Traffic Parsing and Causal Correlation of Multi-source Logs (Core Focus)

Collect database communication protocol traffic and application layer HTTP traffic, and use LLMs to extract high-dimensional features from API logs and SQL audit logs. Based on time windows, parameter similarity and access patterns, build a causal inference model to automatically deduce the mapping

relationship between "API interfaces — database tables/fields", and supplement application-level lineage without touching the code.

2. Automatic Discovery and Anomaly Detection of Data Flow Topology in Microservice Links
For legacy systems with missing Trace ID or complex microservice chains, study agentless tracking technology based on traffic fingerprints. Combine graph neural networks (GNN) to restore the data flow topology between microservices and identify non-business logic data aggregation or detour access.

3. Cross-modal Implicit Fingerprint Embedding from Structured Data to Unstructured Documents
Focus on data export scenarios (ETL/BI reports), study algorithms for generating attack-resistant implicit fingerprints at the moment when structured data is converted into Excel/CSV/PDF. Ensure that the fingerprints can resist format conversion and truncation attacks, and can reversely parse the identity and time of the exporter.

4. Ultra-large-scale Dynamic Lineage Storage and Second-level Query Based on Graph Database
Aim at the massive instantaneous lineage relationships generated by full traffic analysis, study an efficient time-series graph storage architecture. Solve the problems of real-time writing and historical traceback, and support second-level retrieval of the "flow path in the past 30 days" of sensitive fields.

5. Data Flow Intention Recognition and Risk Semantic Analysis
Use LLMs to conduct joint semantic analysis of multi-source logs, focusing not only on identifying "what happened" but also reasoning "why". Distinguish between "normal batch export" and "malicious crawling", "compliant operation and maintenance" and "privilege abuse", and attach dynamic risk semantic labels to the traceability graph.

IV. Outcome Requirements
1. Academic/Theoretical Outcomes
- Publish no less than 2 papers in CCF-A/B class conferences/journals or SCI journals (focusing on log causal inference, intention recognition and cross-modal watermarking).
- Apply for or be authorized no less than 3 national invention patents.
- Form a White Paper on Enterprise Zero-intrusion Data Lineage Construction and Risk Traceability Technology.

2. Industrial/Implementation Outcomes

- Prototype system: Provide a full-traffic data lineage analysis platform demonstrating the complete link from API clicks, DB access to file export.
- Core components/SDK: Log correlation engine with an API-SQL correlation accuracy > 85% in a Trace ID-free environment; intention recognition module with a recall rate > 90% for intention recognition of typical high-risk flow behaviors; cross-modal watermarking tool supporting watermark embedding and extraction for Excel/PDF/images.
- Verification report: Complete deployment verification in at least 1 actual enterprise environment (100 million-level SQL logs per day) to prove the effectiveness of the solution.

(The specific form of outcomes can be appropriately adjusted according to the research direction)

---

8. Research on Key Technologies for Trusted Data Synthesis and Privacy Security

I. Research Direction and Overall Objective

Through technological breakthroughs in key directions such as data generation, quality evaluation and privacy protection of synthetic data, form implementable, cost-controllable, regulable and traceable synthetic data production and governance capabilities.

II. Project Background and Necessity

Large models (LLMs, VLMs, etc.) are increasingly dependent on high-quality data, but real data in key fields such as medical care, finance and government affairs is difficult to share directly between institutions due to privacy protection, compliance policies and commercial confidentiality, forming a prominent contradiction of "data visible but unavailable". Synthetic data is regarded as an important means to alleviate data scarcity, improve data quality and meet compliance constraints, but the existing technologies still have obvious shortcomings in the following aspects: first, the fidelity of synthetic data in statistical distribution, structural characteristics and business semantics is insufficient, which is easy to lead to the expansion of model deviation or performance degradation; second, the lack of a systematic privacy risk quantification and verifiable mechanism makes it difficult to prove that synthetic data will not leak the original individual information; third, a standardized quality evaluation index system and engineering tool chain have not been formed in real business scenarios, making it difficult to support cross-industry and cross-scenario reuse. How to greatly improve the authenticity, effectiveness and usability of synthetic data under the premise of privacy and security, and deeply integrate it with the training and evaluation process of large models is an urgent key problem to be solved.

III. Key Research Tasks
(Applicant teams may select one or more entry points around the following directions, or propose new and high-value challenge directions)
1. Generation Mechanism of High-fidelity Synthetic Data
- Study generation methods that can effectively capture the high-order feature relationships, cross-field structural dependencies and time patterns of real data, realizing high-fidelity synthesis of different data types (tabular, time-series, text, images).
- Study controllable generation mechanisms, and realize accurate expression of key statistical laws and business structures through strategies such as feature constraints, sparse event modeling and distribution alignment.
- For easy-to-missing areas such as long-tail samples, rare events and abnormal patterns, study enhanced generation strategies to improve the coverage and diversity of synthetic data and alleviate the problems of mode collapse and distribution degradation.

2. Multi-dimensional Synthetic Data Quality Evaluation System
- Build an index system for evaluating the quality of synthetic data, covering multiple dimensions such as statistical consistency, structural dependency retention, sample diversity, sparse feature retention capability and abnormal pattern restoration degree.
- Establish structural consistency detection methods suitable for different data types to effectively identify generation defects such as field relationship damage, time disorder and key feature loss.
- Establish synthetic data evaluation benchmarks and standardized processes for typical industry scenarios (e.g., medical care, finance, cyber security, government services), and form a reusable evaluation sample library and script tools.

3. Privacy Risk Modeling, Offense and Defense Verification and Safe Usage Boundary of Synthetic Data
- Establish a privacy leakage risk model for synthetic data, and systematically analyze the sensitivity of different generation mechanisms to attacks such as membership inference, attribute inference and distribution inversion.
- Study the quantitative relationship between privacy protection and information retention, and clarify the leakage probability and recoverability of sensitive features under different privacy budgets or generation strategies.
- Integrate technologies such as differential privacy, federated learning and confidential computing to explore privacy risk mitigation mechanisms for synthetic data, including feature sensitivity suppression, synthetic space limitation, perturbation injection, training structure transformation and other

methods, so as to improve the safety and reliability of synthetic data applications.

IV. Outcome Requirements

1. Core Assessment Indicators

- Synthetic data quality and usability: In typical downstream tasks, the gap between the key indicators of large models trained/fine-tuned based on synthetic data and those trained based on real data does not exceed the agreed threshold (e.g., precision, recall, AUC), and the impact of synthetic data on model collapse does not exceed the agreed threshold.
- Privacy and compliance: In typical privacy attack scenarios such as membership inference and attribute inference, the attack success rate is significantly lower than the real data baseline, meeting the agreed privacy budget/risk upper limit in differential privacy or relevant compliance requirements.
- Completeness and systematicness: The evaluation objects selected by the synthetic data quality evaluation system have real scenario significance; the construction of the quality evaluation system is systematic and can fully evaluate all data conditions in the agreed scenarios; the theoretical basis and derivation of the quality evaluation system can be provided.
- Transferability and scalability: The synthetic data generation and evaluation capabilities can be transferred to at least 2 industry or regional scenarios, and have good scalability and stability under different data scales.

2. Deliverable Requirements

- Runnable synthetic data generation and evaluation prototype system/tool: Provide a prototype platform or tool kit for easy retesting, supporting the synthesis and evaluation of at least 1 type of structured data and 1 type of unstructured data (e.g., text or images); open-source code or reproducible scripts are encouraged.
- Typical industry demonstration datasets and experimental reports: Form synthetic data samples covering several typical business scenarios and experimental reports comparing them with real data, including quality evaluation indicators, privacy risk assessment results and typical application cases.
- Technical white paper/report: Elaborate on the problems solved, technical routes and system architecture design, and provide the synthetic data generation methods, evaluation system, privacy protection mechanisms, engineering implementation routes and practical experience summary.
- Test report: Systematically demonstrate the comprehensive effect of synthetic data based on the actual test results of different business scenarios.

(The specific form of outcomes can be appropriately adjusted according to the research direction)