



世界数字技术院（WDTA）

生成式人工智能应用安全测试标准

世界数字技术院标准

WDTA AI-STR-01

版本：2024-04

© WDTA 2024 - 保留所有权利。

世界数字技术标准 WDTA AI-STR-01 被指定为 WDTA 规范。本文件为世界数字技术院(WDTA)的财产，受国际版权法保护。未经 WDTA 事先书面许可，禁止使用本文件，包括复制、修改、分发或重新发布。WDTA 对于本文件中的任何错误或遗漏不承担责任。
更多 WDTA 标准和相关出版物，请访问 <https://wtdacademy.org/>

目录

前言.....	5
1 范围	9
2 目标受众	12
3 规范参考	13
4 术语和定义.....	14
5 人工智能应用安全和验证标准.....	16
5.1 基础模型选择测试标准.....	17
5.1.1 模型合规性和上下文测试.....	17
5.1.2 数据使用检查测试.....	20
5.1.3 基础模型推理 API 安全测试.....	26
5.2 嵌入和向量数据库.....	33
5.2.1 数据清洗和匿名化测试	33
5.2.2 向量数据库安全测试.....	34
5.3 使用 RAG（检索增强生成）的提示和知识检索	37
5.3.1 提示构造测试.....	38
5.3.2 外部 API 集成测试（函数调用、插件）	43
5.3.3 向量数据库检索测试.....	44
5.4 提示执行/推理.....	45
5.4.1 LLM 应用 API 测试	45
5.4.2 缓存和验证测试	49
5.5 代理行为	50
5.5.1 提示响应测试.....	50
5.5.2 记忆利用测试.....	51
5.5.3 知识应用测试.....	52
5.5.4 规划能力测试.....	52
5.5.5 行动执行测试.....	53
5.5.6 工具利用测试.....	53
5.5.7 过度代理测试.....	54
5.6 微调	55
5.6.1 数据隐私检查测试.....	55
5.6.2 用于微调的基础模型选择测试.....	56
5.6.3 用于微调的基础模型存储测试.....	56
5.6.4 训练数据污染测试.....	57
5.6.5 微调后的模型部署测试	57

5.7	响应处理	58
5.7.1	基本事实检查测试	58
5.7.2	相关性检查测试	59
5.7.3	不良内容检查测试	59
5.7.4	伦理检查测试	60
5.7.5	不安全输出处理测试	60
5.7.6	后门攻击测试	61
5.7.7	隐私和版权合规检查	61
5.7.8	妥善处理未知或不支持的查询	62
5.8	AI 应用运行时安全	63
5.8.1	数据保护测试	63
5.8.2	模型安全测试	63
5.8.3	基础设施安全测试	66
5.8.4	API 安全测试	66
5.8.5	合规和审计追踪测试	67
5.8.6	实时监控和异常检测测试	67
5.8.7	配置与态势管理测试	67
5.8.8	事件响应计划测试	68
5.8.9	用户访问管理测试	68
5.8.10	依赖和第三方组件安全测试	69
5.8.11	安全鲁棒性测试与验证	69
5.8.12	可用性测试	70
5.8.13	侦察防护测试	70
5.8.14	持久性缓解测试	70
5.8.15	权限提升防御测试	71
5.8.16	防御规避检测测试	71
5.8.17	发现抗性测试	72
5.8.18	数据采集防护测试	72
5.9	附加测试规范	72
5.9.1	供应链漏洞测试	72
5.9.2	安全的 AI 应用开发过程	76
5.9.3	AI 应用治理测试	77
5.9.4	安全模型共享和部署	81
5.9.5	决策透明度	83

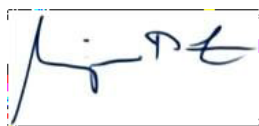
前言

世界数字技术院（WDTA）致力于成为全球数字技术创新的先驱，并作为非政府组织（NGO）与联合国框架保持一致。WDTA 秉承“速度、安全、共享”的 3S 原则，努力加速数字规范的制定，推动研究前沿，促进国际合作，并在技术进步中保持领先地位。通过合作努力，WDTA 致力于推动数字技术的发展，以造福社会。

AI STR（安全、可信、负责任）计划作为 WDTA 国际倡议的重要组成部分，旨在应对 AI 系统快速普及所带来的复杂挑战。鉴于全球 AI 技术的迅速扩展与整合，AI STR 站在全球技术进步的前沿。本标准文件为生成式人工智能应用的安全测试和验证提供了一个框架。该框架涵盖了 AI 应用生命周期中的关键领域，包括基础模型选择、在检索增强生成设计模式中的嵌入和向量数据库、提示执行/推理、代理行为、微调、响应处理以及 AI 应用运行时安全。

其主要目标是确保 AI 应用在其生命周期内能够安全地运行并符合预期设计。通过为 AI 应用堆栈的每一层提供一套专注于安全性和合规性的测试标准与指南，本文件旨在帮助开发者和组织提升其基于大型语言模型（LLM）构建的 AI 应用的安全性与可靠性，降低潜在安全风险，提升整体质量，并推动 AI 技术的负责任发展和部署。

AI STR 计划代表了我们在 AI 技术开发和部署方式上的范式转变。倡导 AI 系统的安全、信任与责任，为更具道德、安全和公平的数字未来奠定基础，使 AI 技术成为进步的推动者，而非不确定性和危害的来源。《生成式人工智能应用安全测试标准》是 AI STR 标准之一。



Founding Chairman of WDTA



Executive Chairman of WDTA

致谢

WDTA AI STR 工作组联席组长

Ken Huang (CSA GCR)

Josiah Burke (Anthropic)

英文版编写专家组

主要作者

Ken Huang (CSA GCR)

Heather Frase (Georgetown University)

Jerry Huang (Kleiner Perkins)

Leon Derczynski (Nvidia)

Krystal (A) Jackson (University of California, Berkeley)

Patricia Thaine (Private AI)

Govindaraj Palanisamy (Global Payments Inc)

Vishwas Manral (Precize.ai)

Qing Hu (Meta)

Ads Dawson (OWASP® Foundation)

Amit Elazari (OpenPolicy)

Apostol Vassilev (National Institute of Standards and Technology)

Bo Li (University of Chicago)

评审专家

Cari Miller (Center for Inclusive Change)

Daniel Altman (Google)

Dawn Song (University of California, Berkeley)

Gene Shi (Learning-Genie)

Jianling GUO (Baidu)

Jing HUANG (iFLYTEK)
John Sotiropoulos (Kainos)
Josiah Burke (Anthropic)
Lars Riddigkeit (Microsoft)
Guanchen LIN (Ant Group)
Melan XU (World Digital Technology Academy)
Nathan VanHoudnos (Carnegie Mellon University)
Nick Hamilton (OpenAI)
Rob van der Veer (Software Improvement Group)
Sandy Dunn (BreachQuest, acquired by Resilience)
Seyi Feyisetan (Amazon)
Yushi SHEN (NovNet Computing System Tech Co., Ltd.)
Song GUO (The Hong Kong University of Science and Technology)
Steve Wilson (Exabeam)
Swapnil Modal (Meta)
Tal Shapira (Reco AI)
Anyu WANG (OPPO)
Wicky WANG (ISACA)
Yongxia WANG (Tencent)

中文版翻译专家组

黄连金（云安全联盟大中华区）

党超辉（中国广电青岛 5G 高新视频应用安全重点实验室）

李 岩

何伊圣（山石网科）

崔 崑

卜宋博

卞超轶（北京启明星辰信息安全技术有限公司）

杨天识（北京启明星辰信息安全技术有限公司）

陶瑞岩（上海安几科技有限公司）

高健凯

罗智杰（云安全联盟大中华区）

1 范围

《生成式人工智能应用安全测试标准》提供了一套完整的框架，旨在评估或验证基于大型语言模型（LLM）构建的下游人工智能应用的安全性，该框架涵盖定义了跨 AI 应用程序堆栈的各个层面的测试和验证范围（见图 1）。将生成式（GenAI）模型融入集成到更广泛的支持 AI 的系统或下游应用程序中可能会带来一定的安全问题。因此，无论其基础人工智能（GenAI）模型在集成到下游应用之前是否经过了严格的测试，都需要在其部署之后进行安全测试和符合性验证。

虽然本文档是一个初始版本，其主要陈述重点是大型语言模型（LLM）。但需要注意的是，其范围也可扩展到生成式人工智能（GenAI）。在本文档的后续迭代版本中，也会整合多模态及更大范围的生成式人工智能（GenAI）模型。

人工智能安全测试和验证工作需要同步进行，以确保 AI 应用程序能够安全地运行并按照预期的方式工作。在可行的情况下，应该在整个开发生命周期中采用稳健的方法，使用诸如提示注入（prompt injection）、扫描和红队演练（red teaming exercises）等技术手段来主动识别潜在的问题。然而，仅依靠测试是有局限性的，尤其是在涉及第三方组件时，可能无法进行测试或者测试受到限制。在这种情况下，与专注于审计人工智能治理、流程和程序的专业外部专家或组织机构合作，对于验证第三方组件的安全性至关重要。对人工智能应用进行全面审计，确保其在所有生命周期部署环境中都符合安全标准，是不可或缺的步骤之一。

对于下游 AI 应用程序的全面审查可确保其遵循安全标准，即使在模型层面评估不充分的情况下也应如此。通过结合强有力的测试实践和对政策、流程和性能持续验证的集成保证方法，可以确保在系统持续自主学习的过程中提供负责任的 AI 成果。总之，它们共同提供了有关系统优势和劣势的信息，帮助判断最终用途的应用是否恰当提供了参考，并协助进行风险控制。

本标准规定了基于大语言模型（LLM）之上构建的下游应用程序的安全测试要求，但并未包括这些基本大语言模型（LLM）自身的安全性测试规范。在不久的将来、我们将会单独发布一份针对基本大语言模型（LLM）的安全测试规范文件。

本标准针对如下几个核心关键领域部分进行了阐述：

1. 基础模型选择：下游人工智能应用程序的候选模型应该在选择之前进行检查。该领域部分主要介绍了如何验证基本模型的合规性、数据使用的适当性和 API 安全性。在本标准文件的第 5.1 章节中提供了详细的指导方针，以确保所选择的模型符合法律法规、道德和运营要求，这是确保人工智能应用安全性的重要环节。所覆盖的选择范围既包含了开源模型，也包含了闭源模型选择。

2. 嵌入和向量数据库：这些是大多数下游人工智能应用的关键组件，用于存储和检索语言数据块。在本标准文件的第 5.2 章节中详细描述了测试数据完整性、质量以及匿名化处理过程，旨在保护用户隐私并确保符合相关法规。此外，本规范还提供了一套指导原则，用以测试向量数据库的机密性、完整性以及服务的可用性。

3. 基于检索增强生成（RAG）的快速知识检索：使用检索增强生成（RAG）可以显著提高生成式 AI 应用（如大型语言模型）的事实准确性和可靠性。RAG 通过在文本生成时动态地整合来自外部来源的实时领域特定知识来实现这一目标。在本标准文件的第 5.3 章节中提供了对“如何构建高效提示、创建和应用提示模板，以及集成外部 API 的相关指导。同时，它还包括了对向量数据库检索过程的测试，确保人工智能应用程序能够精准地获取并利用相关信息。

4. 提示执行/推理：在本标准文件的第 5.4 章节中该文档详细介绍了提示执行与推理层面中大语言模型（LLM）API 的测试过程，包括对缓存机制和验证过程的测试，以优化性能和准确性。此外，该层还涉及对提示的审查，以确保大语言模型（LLM）不被用于执行超出使用案例允许范围的未授权操作。

5. 代理行为：这些是高级大语言模型（LLM）应用能力。在本标准文件的第 5.5 章节中详细概述了对提示执行、内存利用、知识应用、规划和行动启动等方面的测试流程。这包括测试集成到 AI 应用程序中的工具，以安全地增强其能力。

6. 微调：生成式人工智能（GenAI）通常针对特定的下游 AI 应用程序进行微调。在本标准文件的第 5.6 章节中详细介绍了数据隐私测试、基本模型选择的重新评估和模型部署，以确保人工智能的持续改进和相关性。

7. 响应处理：在本标准文件的第 5.7 章节中将会专注于对人工智能的响应进行准确性验证、相关性评估、病毒性检测以及道德考量，以保持人工智能交互的可信赖性和安全性。

8. AI（人工智能）应用运行时安全性：运行时安全性涉及对 AI 应用程序的持续且实时监控。其主要涵盖了数据保护、模型安全性、基础设施安全性以及与审计追踪的合规性。这种全面的安全策略确保了人工智能应用在其整个生命周期中能够抵御各种潜在的威胁和漏洞，从而保障其安全性。

总体而言，《生成式人工智能应用安全测试标准》为 AI 应用栈的各层测试提供了详细而结构化的方法，确保对人工智能应用程序的每个环节都进行了深入的安全性和合规性审查评估。



图 1：AI 应用程序栈

2 目标受众

本文档的目标受众是参与确保生成式人工智能应用的安全性和完整性的专业人员和利益相关者。本文档特别适用于如下群体：

- **AI安全工程师和分析师：**这些专业人员承担着执行和维护本规范文件中所描述的安全措施的主要责任。他们主要评估AI应用程序的威胁，设计安全架构，并监控系统以预防、检测和响应安全事件。此外，这些工程师还需密切监控并解决人工智能应用可能存在的潜在偏见和安全威胁，确保系统的安全性和公正性。

- **AI开发人员、机器学习运维（MLOps）和AI人工智能工程师：**他们主要负责构建、维护和自动化人工智能应用的工作流程。通过遵循安全规范，他们能够深入理解并将安全最佳实践融入应用程序开发生命周期中。

- **合规官和监管专家：**主要负责确保人工智能应用符合持续更新的法律和监管要求，他们利用本规范作为指导参考，特别是在那些对数据保护和隐私法规要求严格的行业中，以确保合规性的努力得到有效实施。

- **数据保护官：**主要负责监督人工智能应用在数据处理上是否安全且遵循数据保护法律法规。本安全规范文件为他们提供了适当的数据管理及保护策略的指导方针。

- **IT和网络管理员：**主要负责维护AI应用的底层基础设施。这些专业人员将依据安全规范来加固网络、服务器和其他组件，防止恶意行为者在与人工智能相关的过程中利用漏洞。

- **风险管理专家：**主要负责评估和管理与人工智能应用相关的风险。通过本安全规范文件可帮助他们识别潜在的安全风险，并采取措施降低风险。

- **伦理审查委员会：**主要负责监督人工智能的应用符合伦理标准，并且能够抵御不当使用或有害的操控。他们依据安全规范来审核人工智能应用，以保障其伦理性和安全性。

- **AI项目中的项目经理和产品负责人：**主要负责确保人工智能项目能够安全且高效地交付。本安全规范文件为他们提供了指导，帮助他们确立与安全相关的项目目标和评估标准。

- **第三方或外部安全审计员和顾问：**这些专家主要负责对人工智能应用的安全状况进行独立审查。他们以本规范作为评估标准，检验应用是否遵循了安全最佳实践。

● **最终用户或业务利益相关者**：尽管他们不直接参与安全措施的执行，但他们对于人工智能应用的安全性有着切实利益。深入了解安全规范有助于他们判断人工智能应用的稳定性和可信性。

这些不同的群体在人工智能应用的安全性保障中扮演着至关重要的角色，涵盖了从开发、部署到运维的全过程。他们以《生成式人工智能应用安全测试标准》文档为指导性框架，确保应用的安全性得到全面保证。

3 规范参考

下面列出的参考资料对于应用和理解本文档至关重要。它们提供了对安全和负责任的开发和部署人工智能应用程序至关重要的基础理论、实践、法律框架和指导方针：

- [WDTA全球人工智能治理宣言 \(WDTA Declaration on Global AI Governance\)](#)
- [生成人工智能安全：理论与实践 \(Generative AI security: Theories and Practices\)](#)
- [CCM的AIS域在生成式人工智能中的应用 \(Applying the AIS Domain of the CCM to Generative AI\)](#)
- [欧盟人工智能法案 \(EU AI Act\)](#)
- [拜登关于安全、可靠和值得信赖的人工智能的行政命令 \(Biden Executive Order on Safe, Secure, and Trustworthy Artificial Intelligence\)](#)
- [NIST值得信赖和负责任的人工智能NIST AI 100-2e2023 \(NIST Trustworthy and Responsible AI NIST AI 100-2e2023\)](#)
- [NIST人工智能风险管理框架 \(AI RMF 1.0\) \(NIST Artificial Intelligence Risk Management Framework \(AI RMF 1.0\)\)](#)
- [中国生成型人工智能监管 \(Chinese Generative AI Regulation\)](#)
- [中国对人工智能监管的态度 Chinese Approach to AI Regulations](#)
- [机密计算联盟 Confidential Computing Consortium](#)
- [OWASP 大语言模型 \(LLM\) 应用程序前10名 OWASP Top 10 for LLM Applications](#)
- [CSA云控制矩阵 \(CCM v4\) CSA Cloud Controls Matrix \(CCM v4\)](#)
- [MITRE ATLAS™ \(人工智能系统的对抗威胁格局\) MITRE ATLAS™ \(Adversarial Threat Landscape for Artificial-Intelligence Systems\)](#)
- [NIST安全软件开发框架 \(SSDF\) NIST Secure Software Development Framework \(SSDF\)](#)

- OWASP API十大安全风险[OWASP Top 10 API Security Risks](#)
- 减轻检索增强生成（RAG）LLM应用中的安全风险[Mitigating Security Risks in Retrieval Augmented Generation（RAG）LLM Applications](#)
- OWASP人工智能交汇[OWASP AI Exchange](#)

4 术语和定义

本文档适用以下术语和定义：

基础模型选择：在考虑 AI 安全性的情况下选择适当的基础模型。此时要评估各种因素，诸如性能基准、训练数据质量、潜在偏差、安全程序、潜在有害输出、预期用例和法规合规性要求等因素。健全的模型来源、透明度、对数据/培训方法的审计、跨职能审查流程以及对行为准则的遵守是在负责任地部署大语言模型（LLM）时维护安全性、合规性和道德标准的重要考虑因素。

嵌入与向量数据库：向量数据库作为基础事实，帮助将知识从训练时间扩展到运行时间，并减少生成式人工智能模型中的错误。它们允许将大量真实世界的数据（图像、文本、分子结构等）存储为捕获语义概念和特征的矢量表示。然后，这些向量数据集作为推理过程中生成模型的参考，使其输出更接近现实，避免制造错误的细节。从生成模型输出的数据库中检索最接近的向量提供了一种自动检测和过滤错误内容的方法。这种数据集调节对于药物发现和内容创建等生成式人工智能的安全关键应用至关重要。优化的向量搜索和可扩展性使得 PG vector、Milvus、Weaviate 和 Pinecone 等数据库非常适合为现实世界中部署的生成式人工智能应用程序提供大规模的错误检测。

检索增强生成（RAG）：检索增强生成通过使用从矢量数据库中实时提取的相关知识来增强生成式 AI 应用程序，如大型语言模型的事实准确性。在推理过程中，检索模块首先使用生成器的内部状态向量来查询存储外部知识（文本、图像等）的向量数据库。然后将与生成上下文最相关的检索向量与内部状态交叉以生成下一个生成的输出。这个过程动态地使模型的生成更接近现实，纠正错误的假设并减少错误的内容。这种可扩展的检索基础设施在开放式推理期间为生成器提供持续的相关外部数据。这种检索增强生成方法抵消了生成器的知识限制和伪造信息的倾向，从而提高了开放域生成人工智能应用中的事实一致性、安全性和信任度。

提示执行/推理：该文档详细说明了在提示符中大语言模型（LLM）API 的测试过程执

行/推理层，包括缓存机制和验证过程的测试，以优化性能和准确性。这一层还包括用于检查提示和确保大语言模型（LLM）不被用于执行未经授权的操作的测试，这在用例中是不允许的。

代理行为：大语言模型（LLM）应用程序通过诸如内存利用、知识应用、计划和基于提示执行操作等行为来展示代理的能力。

微调（Fine Tuning）：针对特定任务或数据集调整基于模型的过程，以提高性能、相关性和数据隐私合规性（使用未来的技术，如微调以消除敏感数据）。

响应处理：这部分涉及对人工智能的响应、相关性、毒性和道德考虑进行事实检查，以保持人工智能交互的可信赖性和安全性。

AI 应用运行时安全：为保护 AI 应用运行时安全而实施的综合安全措施。它包括数据保护、模型安全性和基础设施安全性。

人工智能治理：对人工智能风险的框架、要求、监督和问责制。这些结构支持将风险映射到组织（如团队、身份、优先级和规则），同时提供证明、可跟踪性、度量和监督。

人工智能响应处理：涉及处理和评估人工智能响应的准确性、相关性、无毒性、隐私性、保密性和道德一致性。

API 安全检查：检查与模型接口的 API 的安全措施，如身份验证、授权、数据加密等，防止未经授权的访问和数据泄露。

基础语言模型：基础模型（有时称为基础模型）是一个大型的语言模型，它已经被其原始模型构建者使用诸如从人类反馈强化学习（RLHF）的技术进行了训练和微调，以获得其功能。这些基础模型（例如 OpenAI 的 GPT-4、Anthropic 的 Claude 3、Google 的 Gemini 1.5、Cohere Command、Amazon Titan 或 Meta 的开源 LLaMA2）是进一步特定任务定制的坚实基础。通常，开发人员调整基本模型输出，在下游应用程序中表现出广泛的语言能力和对专业用例的适应性。然后，工程师和公司将这些基本模型作为有效开发和部署定制的人工智能解决方案的起点，以满足他们的精确需求和应用。基本模型消除了从头开始训练完整模型的需要，提供了封闭和开源的人工智能应用启动平台。

缓存：用于存储 AI 模型推断输出的技术，以避免在推断期间重复计算。由于神经网络模型的运行计算成本很高，因此缓存它们的输出可以在实时请求期间加快响应时间。典型的解决方案包括为聊天机器人缓存问答对话，为计算机视觉模型缓存分类，或为大型语言模型生成文本。

缓存验证：在将 AI 应用程序的缓存输出返回给用户之前，检查它们的准确性、相关性

和安全性。这可能涉及信任检查、语义分析、敏感主题的输入阻塞或人工确认。验证与缓存一起工作，以确保可靠和安全的实时 AI，同时受益于更快的性能。在利用缓存时，应用深思熟虑的验证方法是必不可少的。

闭源模型：其权重、推理代码和训练数据清单不是公开可用的模型。

数据清理和匿名化：从数据中删除不准确和不一致的信息，并对个人或敏感信息进行匿名化，以维护隐私和合规性。

数据使用检查：确保用于培训和操作模型的数据是适当的、合乎道德的，并且符合数据保护法规。

外部 API 集成：将外部 API 集成到 LLM 应用程序中以增强功能，例如访问额外的数据源或执行专门的计算。

大型语言模型（LLM）：大型语言模型（LLM）是一个在庞大的文本语料库上训练的神经网络，通过预测下一个单词或令牌来生成智能文本，从而实现开放式文本生成应用程序，如会话 AI 聊天机器人。

模型合规性检查：评估所选模型是否符合法律法规和道德标准。这包括数据隐私法和偏见最小化等考虑因素。请记住，合规性将随着时间的推移而改变，不要认为它总是给定的。也不要将一个供应商的合规性推断为另一个供应商的合规性。很少对模型本身进行认证，而是对其托管的解决方案进行认证。

模型注册：用于存储、版本控制和机器学习/人工智能模型目录和相关元数据（例如，模型卡片）的数据库、存储库或系统。模型花园是注册表的管理版本，包含提供商的管理模型。它通常需要模型与使用的训练数据和推理数据点之间建立元数据脉络。

提示构建和模板：为大型语言模型（LLM）创建有效、安全的提示，并开发模板以标准化和简化提示的生成。

提示处理：大型语言模型（LLM）解释和处理提示以生成响应的过程。这个过程包括理解提示、访问相关知识，并根据提示和上下文知识生成输出。

5 人工智能应用安全和验证标准

要确保 AI 应用程序的安全性和完整性，需要对 AI 应用程序堆栈中的所有组件进行全面且细致的测试。全面的测试制度可验证下游 AI 应用的各个方面（从基础模型选择到程序部署运行）都能按预期安全运行，且不存在漏洞。细致的测试规范设定了明确的测试要

求、测试方法和预期结果，以实现评估的透明性。本节提供了 AI 应用架构各层的详细测试标准。

5.1 基础模型选择测试标准

基本模型选择是确保 AI 应用程序的安全性和合规性的关键环节。测试和验证过程需要针对开源模型和闭源模型进行不同的考量，因为闭源模型可能有较多现成的合规性文档，而开源模型则可能缺乏相应既定的合规性文档。不过，两者都需要进行全面的测试和验证。

重要的是，基础模型的测试和验证是一个连续的过程，特别是在上游基本模型更改的情况下。随着基本模型的迭代和更新，必须重新验证模型以确保其仍然满足所需的安全性和合规性。这种持续的验证过程有助于维护 AI 应用程序的完整性和可靠性，即使底层基础模型普通的修改或旨在提升性能的改进。

5.1.1 模型合规性和上下文测试

检查模型合规性涉及针对每种模型类型的不同方法，要考虑它们的独特特征和信息可用性。

测试要求：

确保无论是开源还是闭源的 AI 模型，符合法律法规、安全和道德标准。

测试方法：

(1) 对于闭源模型，根据相关法律、行业法规和道德准则，对供应商提供的合规性文档进行评估。为确保合规，还应审查和评估模型的训练数据质量（数据是否符合用途）、输出行为、操作参数和社区用户反馈。闭源模型的权限和访问控制可能会限制这种评估。

(2) 对于所有模型，诸如 Model Cards¹和 Data Statements²等工具为模型和数据文档提供了基线检查标准。但有些缺乏正式文档领域的合规性条目，可能需要咨询法律和行业专家。测试模型在特定任务上的准确性、相关性、一致性和性能是否满足预定的要求。使用预设场景和数据集对模型进行基准测试，衡量其性能和输出质量。

注意事项：

用于测试的资源，它们的效用可能会随着时间的推移而变化和降低。对于闭源和开源

模型，都有许多公开可用的即时安全性测试结果。由于它们使用不同的测试数据集，这些安全评估工作可能为相同的模型产生不同的结果。不过，参考多个结果可以指出大语言模型（LLM）更可能表现出哪些类型的有害行为。随着新的故障模型和攻击的发现，基准也会随着时间的推移而改变；需考虑并报告用于评估模型的基准的日期/版本。

此外，需识别并列已知漏洞，参考来源：**MITRE Atlas**（™）³、**AVID**⁴、**AI 风险数据库**⁵和**AI 事件数据库**⁶等。

预期结果：

无论模型是否闭源，基础模型完全满足所有法律法规、安全和道德标准。对于闭源模型，任何不合规的领域都被明确标出，对于开源模型，则严格地推断并记录在案。

检查上下文元数据，每个模型的训练和微调谱系都存在于模型卡中。模型卡提供了每个应用程序所使用的模型的详细信息。

测试要求：

确保人工智能模型（无论是开源还是闭源）具有详细说明模型来源、数据敏感度、训练数据集和模型监护人的模型卡。这些信息应可通过模型卡直接或间接地访问。

测试方法：

- (3) 对所有模型（包括托管模型）的可用模型卡进行详细的检查。
- (4) 确保模型片具有模型谱系和所有权的详细信息。模型卡应该能够提供训练和微调的（如果适用）数据集。
- (5) 对于闭源模型，如果已知，请从供应商处获取模型卡的详细信息。
- (6) 检查模型管理员、所有权和数据集随时间变化后，是否维护和管理模型卡的措施。

预期结果：

无论开源还是闭源，基本模型都具有模型应用程序的完整元数据。

1. Model Cards: <https://dl.acm.org/doi/10.1145/3287560.3287596>

2. Data Statements: <https://techpolicylab.uw.edu/data-statements/>

3. MITRE Atlas (™): <https://atlas.mitre.org/>

4. AVID: <https://avidml.org/>

5.AI风险数据: <https://airisk.io/>

6.AI 事件数据库 <https://incidentdatabase.ai/>

5.1.2 数据使用检查测试

1. 保护用户在与AI应用程序交互时的隐私

测试要求:

在用户输入的提示词中保护用户隐私

测试方法:

- (1) 对敏感和个人数据的用户提示实现数据匿名化或假名化技术。
- (2) 进行定期审计，确保个人身份信息被有效地隐藏。
- (3) 确保输入的提示词和输出的结果的存储时间不超过政策规定。这可能会限制存储的内容和存储时间期限。
- (4) 进行对抗性测试检查数据泄漏，使用诸如连续性测试、完形填空任务测试等方法来检查用户数据泄漏。

预期结果:

用户提示词的处理不会泄露个人身份信息，确保隐私并符合数据保护法律。如有需要，应有控制措施防止敏感信息被收集或发送到 API。

2. 在处理和使用时保持道德标准和法律合规。

测试要求:

数据使用道德且合法

测试方法:

- (1) 建立符合道德标准和法律要求的数据使用指南。
- (2) 定期执行合规性检查和审计，以监控对数据使用指南的遵守情况。
- (3) 持续确保模型文档合规性的评估要求。

预期结果：

来自用户的提示词、微调训练数据和向量数据库的数据在道德和法律上能合理使用，不存在滥用或处理不当的情况。

3. 确保数据的使用遵守国际和本地数据保护法规

测试要求：

数据的使用遵守法律法规。

测试方法：

(1) 检查是否存在获取用户同意、确保数据透明度和为用户提供对其数据控制权的程序。

(2) 数据采集最小化，仅收集必要的个人数据，避免收集过度。

(3) 通过技术手段尽量减少个人数据的使用和存储期限。

(4) 根据数据的敏感度采取差异化的隐私保护措施。

(5) 定期对员工进行数据保护法培训，并对培训合规性进行审计

(6) 检查是否存在响应信息访问请求和被遗忘请求的流程和程序。

预期结果：

数据的使用完全遵守GDPR或CCPA等数据保护法律，并能通过审计结果和用户反馈证明。

4. 利用数据谱系和元数据的完整性，确保数据可溯源

测试要求：

确保用于训练机器学习模型的数据集（无论开源或闭源）都拥有数据卡，并且可以通过数据卡直接或间接访问数据集的数据源、数据敏感度、合规制度和数据管理员等信

息。

测试方法：

- (1) 对数据集和数据卡进行审查。
- (2) 验证每个数据集（尤其是那些已用于训练或微调模型的数据集）是否拥有数据卡。
- (3) 确保数据卡包含数据集谱系和所有权的详细信息。
- (4) 检查是否存在维护和管理数据卡的措施。随着数据管理员、所有权和数据集随时间变化，数据卡中的收集和內容也会发生变化。

预期结果：

无论开源或闭源模型，基础模型都具有模型应用的完整元数据。

5. AI应用开发者与基础模型供应商之间的数据使用协议

测试要求：

AI应用开发者与基础模型供应商之间存在数据使用协议。

包括以下测试或验证：

- **子要求1：**明确协议双方及适用范围。

测试方法：

审查数据协议，核实所有参与方都被正确识别，并且协议的范围，包括具体的基础模型或数据集，都被明确定义。

预期结果：

协议准确地识别了所有参与方并概述了范围，所涉及的模型或数据集的歧义较小。

- **子要求2：明确使用权和限制。**

测试方法：

对协议进行详细分析，确保明确说明并理解使用权和限制，包括对修改和再分发的任何限制。

预期结果：

清楚理解和记录使用权，确保许可类型（独占或非独占）和任何限制都被明确定义并遵守。

- **子要求3：数据处理合规。**

测试方法：

审查处理用户提示词、微调训练数据和向量数据库内容的流程。检查是否遵守数据保护法律，以及数据匿名化、数据驻留和安全程序。

预期结果：

数据处理方法完全符合协议和法律标准，并实施了安全合规的数据管理实践。

- **子要求4：知识产权。**

测试方法：

(1) 核实协议是否明确概述了有关基础模型、微调模型、输入数据、微调数据和输出数据的知识产权。

(2) 检查实际操作中的合规性。

(3) 查看模型提供商赋予其用户的任何赔偿条款。通过审查提供商的赔偿条款来确定法律风险。

预期结果：

知识产权得到尊重，并为使用、修改和再分发模型输出提供了明确的指导方针。根据公司的需求，理解任何赔偿条款和条件。

- **子要求5：保密和非公开条款。**

测试方法：

评估保密和非公开条款的执行情况，特别是关于敏感数据的条款。

预期结果：

严格遵守保密义务，所有敏感信息都按照组织关于处理公司机密信息和重大非公开信息的政策受到保护。

- **子要求6：责任与担保条款。**

测试方法：

(1) 审查组织处理与人工智能系统相关的责任和保证的方法。

(2) 评估组织实施相关条款，应对潜在的模型故障或数据泄露，并遵守该领域的适用标准和法规。

预期结果：

组织表现出尽其所能妥善管理责任和履行保证的承诺，并制定了旨在处理可能出现的任何问题的政策和流程。努力遵守相关的行业标准、最佳实践和法律要求，这些要求涉及责任分配和故障或违规的补救措施，同时认识到可能无法始终完美执行。

- **子要求7：终止与续约条款。**

测试方法：

审查协议中关于终止和续约条件的明确性，包括通知期限和终止程序。

预期结果：

明确规定终止和续约的条款，任何一方都可以遵循一个简单的流程。

- **子要求8：争议解决条款。**

测试方法：

检查争议解决条款，评估在发生分歧或违约时参与概述流程的准备情况。

预期结果：

具备有效的争议解决机制，与协议条款保持一致。

- **子要求9：管辖法律。**

测试方法：

确保明确说明并理解管辖法律和管辖权，并检查与AI应用程序开发或使用地当地法律之间是否存在任何潜在冲突。

预期结果：

确保明确说明并理解管辖法律和管辖权，并检查与AI应用程序开发或使用地当地法律之间是否存在任何潜在冲突。

- **子要求10：签署。**

测试方法：

确认协议由双方授权代表签署。

预期结果：

具有法律约束力的协议，所有相关方均已签署，确保其可执行性。

5.1.3 基础模型推理 API 安全测试

本节概述了全面评估客户端应用程序如何与第三方模型推理API集成的具体测试规范。当应用程序与外部API交互时，这些测试至关重要，需要采用与传统API测试不同的方法。

本节侧重于从客户端应用程序的角度进行测试，这与API提供商进行的测试不同，如第5.4.1节所述。这种区分至关重要，因为我们正在处理将使用第三方推理API的客户端应用程序。为了确保安全的集成，需要一种针对此用例量身定制的独特测试方法。

为确保全面、结构化地测试与第三方模型推理API集成的客户端应用程序的安全性，我们列出了以下测试规范。

1. 身份认证和授权：

测试要求：

所有对API的请求都必须经过身份认证和授权，以确保客户端既有权限又有适当的权限访问请求的资源。

测试方法：

模拟各种身份验证场景，测试协议实现和密钥/令牌的管理。

预期结果：

客户端必须在请求标头中包含有效的身份验证令牌，通常为承载令牌（Bearer Token）。API应使用适当的状态代码进行响应：

- 如果请求成功，则返回 **200 OK** 状态代码。
- 如果令牌丢失、无效或已过期，则返回 **401** 未授权状态代码。
- 经过身份验证的客户端缺乏执行请求操作的权限，则返回 **403** 客户端错误状态代码。

提供清晰简洁的错误消息来指示特定的授权问题，例如缺乏访问权限或令牌过期。

2. 数据加密：

测试要求：

必须在所有状态下应用数据加密：传输中、存储时和使用中。

(1) 数据传输子要求：

对通过网络传输的数据使用强加密协议（如TLS 1.2或更高版本），以确保安全机密性和完整性。实现完美的前向保密性，即使过去的密钥被泄露，也能保护过去的加密通信免遭解密。在传输之前，敏感数据必须进行加密，确保在到达目的地时才能获得授权解密。

测试方法：

对基础模型推理API的端点进行全面的漏洞评估和渗透测试，以评估其抵御潜在攻击的能力。确保其加密配置和加密协议强大且不可渗透。持续监控往返端点的网络流量，验证是否执行了严格的加密标准并采用了安全协议（例如最新版TLS），并优先考虑保持完美的前向保密性。

预期结果：

所有传输的数据均使用符合当前加密标准的最新安全协议进行安全加密。加密密钥应在每个会话中动态更改，以防止将来密钥被泄露时过去的会话被解密。

(2) 数据存储子要求：

实施多层安全方法去除敏感数据的身份信息，包括令牌化、匿名化和假名化。在无法对敏感数据或个人信息进行去标识化的情况下，应采用AES-256或同等强度的加密标准来安全存储敏感数据。加密密钥应与加密数据分开存储以增强安全性。必须实施严格的访问控制和有效的密钥管理策略。

测试方法：

根据定义的测试程序验证合规性要求。

对于令牌化，评估令牌生成的安全性和随机性。评估令牌字典访问安全性。验证所有令牌化操作的日志记录和审计。评估令牌-数据的映射的加密和访问控制。测试令牌化数据使用时的数据泄露风险，尤其是与频率攻击相关的风险。

对于匿名化，验证其不可逆匿名化和不可重新识别数据。检查匿名化数据是否用于预期用途。对具有风格代表性的数据进行潜在重新识别的风险分析。审查所使用的匿名化技术及其有效性。

对于假名化，确保假名的唯一性、安全性以及与源数据的分离。分析考虑数据相关性的恢复身份信息的风险。验证访问控制、审计跟踪和授权访问。

预期结果：

数据应在各种情况下得到充分保护，而不会损害其对合法业务流程的效用。根据数据的敏感度，在理想情况下，个人被通过匿名数据重新识别的风险应在0.04%至0.1%之间或更低。只有被授权后才可以访问并将令牌或假名链接到原始数据，并需要完整的审计跟踪。

必须从实践上使从匿名化或假名化数据中重新识别个人变得不可能，从而将隐私风险降至最低。

(3) 使用中数据子要求：

对内存中正在处理的敏感数据实施加密。应用程序必须遵守安全编码规范来防止内存转储和侧信道攻击。应采用最小权限原则限制处理期间对敏感数据的访问。应考虑使用提供硬件信任根（由机密计算联盟定义）的机密计算硬件或服务以及机密 GPU。

测试方法：

- (1) 通过评估应用程序和系统，确认它们在内存中有效地加密敏感数据，从而验证内存加密的有效性。
- (2) 评估应用程序如何处理内存中的敏感数据，重点防止通过内存转储泄漏以及防止侧信道攻击。
- (3) 测试侧信道攻击的漏洞，检查数据在内存中的处理和存储方式，以识别潜在的泄漏点。
- (4) 审查用户和进程的访问权限，确保它们是最小且必要的，符合最小权限原则，以降低未经授权访问内存中敏感数据的风险。
- (5) 可信执行环境（TEE）提供安全的执行环境，在处理过程中将敏感数据隔离在受保护的 CPU 区域中。
- (6) 检查数据使用中机密计算是关键部分，允许从可信根（RoT）逐步建立对系统的实际信任。

预期结果：

确保敏感数据在处理时保持加密、假名化或匿名化状态，并确保其安全。只有必要用户和进程才能访问正在使用的敏感数据，并且此类访问必须受到严格控制和记录。机密计算和 TEE 验证结果应表明环境是安全的，并已通过完整性测试。

3. 输入审核和数据净化：

测试要求：

针对对从API接收的数据进行全面的验证和净化。

测试方法：

- (1) 评估应用程序处理和净化各种潜在攻击向量（包括超出常规用例参数的输入）的能力。
- (2) 执行模糊测试，应涵盖API接口的所有功能点，包括各种HTTP方法（例如GET、POST、PUT、DELETE等）。

(3) 执行渗透和安全漏洞测试，例如SQL注入、跨站点脚本（XSS）、命令注入、溢出错误等。

预期结果：

应用程序可有效过滤和净化输入，防止注入、无关输入和其他数据操纵攻击。所有被识别为恶意的提示词都会被记录并发送以供分析。

4. 错误处理和日志安全：

测试要求：

安全且不会泄露敏感信息的错误处理和日志记录。

测试方法：

触发错误条件并分析日志是否存在信息泄露。

如果可能，验证日志在存储之前是否会自动删除敏感信息。

预期结果：

错误得到安全处理，不会泄漏敏感数据，并且日志得到安全维护。

5. 速率限制和资源管理：

测试要求： 遵守API的速率限制和高效的资源管理。

测试方法： 在不同的负载条件下测试应用程序，评估是否符合速率限制和资源使用情况。

预期结果： 应用程序遵守速率限制并有效管理API资源。

6. API密钥和凭证的机密管理：

测试要求：

使用机密管理方法将API密钥和凭据存储在安全保管库中，从而安全地管理它们。API密钥和机密必须定期轮换（间隔不得超过180天），或在出现潜在泄露迹象时立即轮换。轮换过程必须支持安全密钥撤销和配置，以最大限度地减少攻击窗口期。

测试方法：

- (1) 实施安全保管库来存储和检索API密钥和凭证，确保它们不会暴露或泄露。
- (2) 观察并验证安全的密钥管理流程，包括但不限于密钥生成、密钥轮换、禁用旧密钥、密钥销毁和安全处理密钥材料。
- (3) 采访负责人员并审查培训材料，以确认相关团队了解并理解安全的API密钥轮换和机密管理流程。
- (4) 通过尝试使用旧/已撤销的API密钥和机密访问资源并验证访问是否被适当拒绝来执行示例测试。

预期结果： API密钥和凭证得到安全存储，降低未经授权访问和数据泄露的风险。

7. 依赖项和库安全：

测试要求：

用于API通信的库和依赖项是最新的且安全的。

测试方法：

执行漏洞扫描检查过是否存在过时和已弃用的组件。

预期结果：

所有组件都是最新的并且没有已知漏洞。

8. 遵守API安全策略：

测试要求：

完全遵守API供应商的安全政策。

测试方法：

审查并测试应用程序的数据处理和集成是否符合API的安全协议。

预期结果：

应用程序符合API的所有指定安全指南和协议。

9. 监控和事件响应：

测试要求：

有效监控异常API活动并制定健壮的事件响应计划。

测试方法：

确定用例的正常API行为基线（API输入和输出），包括相关性、典型请求率、响应大小和模式。此基线有助于检测异常，尤其是由审核或用例过滤API触发的异常。保留详细日志并使用自动化工具分析它们是否存在可疑活动。

预期结果：

及时检测异常并有效执行事件响应计划。

10. 隐私和数据保护合规性：

测试要求：

遵守数据保护法律和隐私设计原则。

测试方法：

审计应用程序中的数据处理实践和隐私措施。

预期结果：

应用程序符合相关数据保护法规，并有效保护用户隐私。

11. 定期安全审计：

测试要求：

持续进行安全评估，重点关注API交互。

测试方法：

定期进行安全审计，以识别和解决漏洞。

预期结果：持续识别并及时修复与API交互相关的安全问题。

5.2 嵌入和向量数据库

对于人工智能应用程序的嵌入和向量数据库组件，测试规范可以按如下方式构建：

5.2.1 数据清洗和匿名化测试

通过验证数据的清洁度和有效的匿名化，确保用于创建嵌入的数据的完整性和隐私合规性。

测试要求：

确保用于创建嵌入的数据，特别是面向公众的应用程序，根据用例被有效地清洗和匿名化。这可以通过诸如标记化等技术来实现。

测试方法：

(1) 实施测试以评估数据清洗过程的全面性，确保识别和纠正或删除不相关、冗余或错误的数据库。

(2) 另外，测试匿名化过程，以确认个人或敏感信息已根据GDPR等隐私标准得到有效遮蔽或删除。这可能涉及审查匿名化算法、技术及其在各种场景中的有效性。

预期结果：

嵌入过程中使用的数据库是干净、相关且无错误的。匿名化过程有效地保护了个人和敏感信息，使其无法识别。

5.2.2 向量数据库安全测试

通过实施和验证高级加密、基于角色的数据访问控制 (RBAC)、稳定的密钥管理、全面的身份与访问管理 (IAM) 策略以及其他关键安全措施来增强向量数据库的安全性。以下是测试规范：

1. 高级加密技术：

测试要求：

评估包括端到端加密在内的先进加密技术的使用情况，以确保静态和传输中的数据安全。考虑使用始终加密的数据库和机密计算来保护数据。

测试方法：

对正在使用的加密协议和加密标准进行全面评估，分析其在保护数据方面的有效性。

预期结果：

通过采用高级加密方法并在传输、使用和存储过程中保护数据来增强数据库安全性。

2. 密钥管理生命周期测试:

测试要求:

检查加密密钥从创建到销毁的整个生命周期，以确保遵守的安全密钥管理实践。

测试方法:

测试密钥的分发、更新、撤销和销毁过程，评估其鲁棒性和对密钥管理标准的遵守情况。

预期结果:

密钥有一个安全密钥管理生命周期，有效保护加密密钥，降低未经授权访问的风险。

3. 细粒度IAM策略实施:

测试要求:

实施和测试细粒度的身份和访问管理（IAM）策略，这些策略为不同的用户角色和场景指定精确的权限。

测试方法:

进行基于场景的测试，以验证每个用户角色只能根据其定义的权限访问数据和执行操作。

预期结果:

对访问权限进行精细控制，确保用户只能执行授权的操作，从而增强数据安全性。

4. 定期安全和合规审计:

测试要求:

进行频繁且全面的安全审计，审计范围应超出标准检查，包括对国际标准和行业特定法规的合规性评估。

测试方法：

执行深入审计、漏洞评估和合规性检查，以确保符合相关的安全标准和法规。

预期结果：

安全状况持续改进，遵守行业最佳实践和合规要求。

5. 零信任架构（ZTA）评估：

测试要求：

评估零信任安全模型的实施情况，在这种模型中，信任永远不会隐式授予，而必须不断进行验证。

测试方法：

评估零信任环境中的向量数据库部署，验证是否根据身份和上下文应用访问控制。

预期结果：

通过零信任模型增强安全性，减少攻击面并确保严格的访问控制。

6. 实时监控和异常检测：

测试要求：实施并评估实时监控系统和异常检测算法，以实时识别和响应异常访问模式或潜在的安全威胁。

测试方法：通过模拟安全事件并监控其检测和响应，测试实时监控工具和异常检测算法的有效性。

预期结果：系统能及早发现并快速响应安全威胁，最大程度地减少潜在损害和数据

泄露。

7. 灾难恢复和备份测试：

测试要求：

确保存在健全的灾难恢复和数据备份流程。

测试方法：

测试灾难恢复和备份系统在发生数据泄露或数据丢失事件时，能否快速准确地恢复数据。

预期结果：

可靠高效的灾难恢复和数据备份流程，最大限度地减少数据丢失和停机时间。

8. 基于角色的访问控制技术：

测试要求：

评估向量数据库中的数据访问是否与基于角色的访问控制一致。

测试方法：

对数据访问的 **RBAC** 进行全面评估。使用不同的角色访问数据，并确保只有正确的数据才能被正确的角色访问。标记任何异常的数据访问。

预期结果：

确保当其他角色访问数据时，目标角色的数据不会被其他角色看到。

5.3使用 RAG（检索增强生成）的提示和知识检索

AI应用程序的“使用RAG（检索增强生成）进行提示和知识检索”阶段的测试规范

包括以下项目：

5.3.1 提示构造测试

1. 验证为RAG模型创建的提示是否有效地传达了预期的查询或命令。

测试要求：

确保为RAG模型构建的提示有效且准确地表示预期的查询或命令。

测试方法：

测试提示构建过程的清晰度、相关性和完整性。这涉及评估各种提示，以确保它们有效地将预期请求传达给 RAG 模型，并且模型的响应与提示的意图一致。这些测试可能包括用户场景模拟和提示可变性的自动化测试。

根据下游 AI 应用的不同，证明这一要求可能需要大量资源或专业知识。因此，组织可能需要一系列方法来证明这一要求。例如，有一些公共存储库¹²和第三方公司可以帮助证明要求

预期结果：

提示结构良好，明确无误，并能有效指导RAG模型提供相关且准确的响应。

2. 验证RAG模型的输出是否与用例和提供的提示相关

测试要求：

确保RAG模型的结果精确且相关

测试方法：

7.公共资源：<https://github.com/microsoft/promptbench>

8.公共资源：<https://github.com/promptfoo/promptfoo>

针对不同的用例测试不同的提示，并查看输出在清晰度和相关性方面是否与预期输出一致。根据下游AI应用的不同，证明这一要求可能需要大量资源或专业知识。

因此，组织可能需要一系列方法来证明这一要求。例如，有一些**公共资源**⁷⁸和第三方公司可以帮助证明要求。

预期结果：

生成式AI输出可能会提供与用例不相关的结果。确保输出一致有助于确保输出的可用性、公平性和相关性。

3. 提示注入测试

测试要求：

确保模型在响应各种精心设计的输入（包括查询和注入的上下文）时不会执行非法操作。

测试方法：

测试模型对一系列精心设计的、可能存在恶意输入的响应。这涉及模拟可能利用输入处理中的漏洞的场景。测试应包括直接和间接的提示注入，如《OWASP LLM 应用程序十大威胁》所述。

预期结果：

模型可始终安全地处理精心设计的恶意输入，不会执行恶意操作或显示易受攻击的行为。

4. 敏感信息泄漏测试

测试要求：

防止通过模型输出无意中共享机密数据。

测试方法：

评估模型的输出，查找可能泄露敏感或机密信息的实例。这包括通过提示工程、越狱以及各种策略和技术来测试可能触发此类泄露的场景。审查学术文献和公开的独立测试结果，评估信息泄漏情况。如果没有公开的独立测试，组织应考虑是否可以使用其他系统。

预期结果：

模型在其输出中始终避免泄露敏感或机密信息。

5. 防止领域受限时聊天机器人进行边界规避

测试要求：

实施多层方法，确保聊天机器人保持在其领域内。这种方法需要强大的故障安全机制、完善的防护措施、输出过滤和专门的算法。

测试方法：

通过提供与其指定领域无关的故意查询来测试聊天机器人，评估其识别和管理此类情况的能力。使用异常和噪声模拟真实世界的场景，确保聊天机器人提供准确可靠的信息。进行 **A/B** 测试，将聊天机器人的性能与对照组进行比较，从而深入了解其在特定领域内的有效性。

预期结果：

该聊天机器人表现出卓越的能力，能够识别其领域外的查询，礼貌地承认其不相关性或引导对话回到正轨。该系统在模拟的真实场景中可靠地提取了准确的领域特定数据，不受无关或嘈杂信息的影响。在 **A/B** 测试期间，聊天机器人的表现超出了预期，特别是在响应质量、用户满意度和指定领域内的相关性方面。聊天机器人严格遵守实施的故障安全机制、防护措施和专用算法，确保强大而安全的用户体验。用户对聊天机器人准确而有用的回复表示高度满意。

5.3.2 提示模板测试

提示模板是用于生成提示的预定义结构或指南，这些提示有助于从模型中获得特定类型的响应。这些模板旨在通过提供一致且优化的方式来表述查询或命令，从而简化与模型的交互，并确保模型尽可能准确地理解用户的意图。提示模板的设计可以显著影响模型响应的有效性和效率，使其对于需要可靠且上下文适当的输出的应用程序至关重要。

以下是提示模板所需的测试。

1. 使用模板进行访问控制测试

测试要求：

确保系统使用提示模板遵守总体访问控制策略，防止利用模板规避安全机制或访问控制。

测试方法：

进行系统级测试，以评估不同角色/用户如何访问提示模板，以及如何在系统安全和访问控制框架的上下文中使用提示模板。这涉及验证系统在允许访问特定模板或模板功能之前是否检查用户权限，尤其是那些可能触发敏感操作或访问特权信息的模板。测试应模拟各种用户角色尝试使用模板。它应调查他们是否能以应该受限制的方式使用模板，查看系统在处理模板之前是否正确执行访问控制。

预期结果：

系统确保所有与提示模板的交互都受到适当的访问控制。用户只能以与其权限一致的方式使用模板，无法使用模板绕过系统级访问限制。用户尝试访问或使用超出其授权级别的模板时会被拒绝，表明系统有效地执行了与提示模板相关的访问控制。

2. 模板的鲁棒性和清晰度测试

测试要求：

确保提示模板对于误解和误用具有鲁棒性，以免导致意外或不适当的输出。模板应清晰地引导用户，降低利用歧义或导致不良系统响应的输入的风险。

测试方法：

对模板进行全面的审查和用户测试（涵盖所有相关的用户角色），以评估其清晰度和可能被误解的可能性。这包括与各种用户一起评估模板，包括那些有意测试模板有效性边界的人。目标是识别并纠正任何可能被用户（有意或无意）利用来生成非预期、不适当或超出模板预期用途范围的响应的歧义或弱点。测试还应评估模板对输入格式和内容期望的指导，以确保用户了解如何提供导致所需类型响应的输入。

预期结果：

模板有效地引导用户提供与模板预期用途一致的输入，最大限度地降低误解或误用的风险。模板的设计和说明明确地减轻了潜在的对抗性操纵，确保系统的响应保持在预期和适当的范围内。用户输入和系统输出高度一致，反映了模板在以安全和预期的方式指导用户与系统交互方面的有效性。

3. RAG实现的上下文访问控制和响应过滤

测试要求：

实现动态访问控制。应用程序必须根据上下文（包括时间、位置、设备类型和网络安全状态）评估用户请求。应用程序必须根据上下文动态调整用户权限，例如限制在工作时间以外访问敏感数据，以及限制在不安全的网络上访问特定功能。

利用基于属性的访问控制（ABAC）。应用程序必须使用ABAC来根据各种属性（如用户角色和数据分类）管理用户访问。应用程序必须将ABAC与企业身份提供者和外部API集成，以实时检索用户属性。

确保数据集成和访问验证。应用程序必须安全地与外部系统集成，验证API密钥，并使用作用范围访问令牌限制对授权数据的访问。应用程序必须将从集成平台检索到的

访问权限与用户权限进行比较，以确保访问控制的一致性。

实现上下文响应过滤。应用程序必须实现基于用户上下文和权限过滤搜索结果的逻辑。应用程序必须根据用户的角色或上下文动态修改响应，以排除未经授权的数据。

测试方法：

(1) 代码审查：审查应用程序代码，确保存在动态访问控制、ABAC实现和数据访问验证的逻辑。

(2) 动态分析：必须使用安全测试工具在运行时动态分析应用程序的行为。应模拟具有不同上下文的用户请求，以验证访问控制和响应过滤是否按预期工作。

(3) 渗透测试：必须进行渗透测试，通过各种技术尝试未经授权访问敏感数据，以验证实现的访问控制是否阻止了未经授权的访问。

预期结果：

确保敏感信息始终受到保护，防止未经授权的访问和泄露。用户应只能访问其特定上下文和角色所需的数据，在保持运营效率的同时增强安全性。系统必须适应各种用户上下文，动态应用适当的访问控制和过滤器。系统必须遵守相关的数据保护法律和标准，最大限度地降低法律和财务风险。

5.3.2 外部 API 集成测试（函数调用、插件）

外部 API 集成是指将 LLM 应用程序与外部 API 连接的过程，以扩展其功能并从其他系统访问数据或服务。这使得 LLM 能够执行超出其固有知识和语言处理能力的任务。

为了确定外部 API 与 RAG 模型集成的可靠性和安全性，确保无缝连接、准确的数据交换和强大的安全措施，我们需要执行以下测试。

测试要求：

确保外部 API 与 RAG 和 LLM 模型的可靠和安全集成，包括管理用于访问 API 的

密钥。

测试方法：

对 API 连接、数据交换、错误处理 and 安全性进行测试。这包括测试函数调用的正确性、数据传输的准确性、错误和异常处理的稳健性，以及是否符合安全协议（如身份验证和数据加密）。

预期结果：

外部 API 与 RAG 和 LLM 模型安全集成，展示出可靠和安全的数据交换，并有效处理错误，而不会影响系统性能或安全。作为 API 的客户端或提供者，请参考第 5.4.1 节和第 5.8.4 节关于 API 安全的内容。

5.3.3 向量数据库检索测试

为保证 RAG 系统准确、高效、相关地从向量数据库中检索信息，确保及时做出正确的响应。

测试要求：

确保从向量数据库中准确高效地检索信息。

测试方法：

测试检索过程的相关性、准确性和速度。这涉及用各种输入查询向量数据库，并评估检索信息的相关性和正确性。组织还可以评估其他性能指标，如响应时间。

预期结果：

RAG 系统能够高效地从向量数据库中检索出相关且准确的信息，有助于对提示做出精确且信息丰富的响应。

5.4 提示执行/推理

人工智能应用程序中“提示执行/推理”阶段的测试规范，主要关注LLM API、缓存和验证机制，其测试结构如下：

5.4.1 LLM 应用 API 测试

如果您作为 LLM 应用提供商，有 API 提供给第三方使用，您需要根据以下测试规范进行测试。

1. 缓解访问控制失效：

身份验证要求： 正确实现OAuth 2.0、SAML 2.0和OpenID Connect等身份验证协议，并安全地处理API密钥和令牌。使用基于令牌的身份验证机制，如JSON Web Tokens (JWT)，在无状态环境中安全传递身份验证信息。

测试方法：

模拟各种身份验证场景，测试协议实现和密钥/令牌管理。如果使用JWT令牌，通过验证签名、检查颁发者和确保受众与预期接收者匹配来验证令牌的完整性。

预期结果：

成功的身份验证过程，以及对敏感凭证的安全、无泄漏的处理。

授权要求：

实施全面的访问控制，根据用户的角色和权限，管理并限制用户操作。这些措施包括防止权限提升以及强制执行基于策略的访问。授权矩阵必须以结构化和机器可读的格式记录，同时易于人类理解和更新。它还应该采用分层方法来定义各种授权组合，这些组合应该适用于应用程序的不同技术平台和架构框架。

测试方法：

验证基于角色的访问控制（RBAC）或基于属性的访问控制（ABAC）系统，确保权限分配和执行正确。组织必须创建广泛的集成测试，以验证授权矩阵对于被测应用程序

序的完整性和适用性。这些测试应直接使用形式化的矩阵作为输入。任何测试失败实例都必须突出显示违反的授权组合。

预期结果：

受控的访问措施应确保只有经授权的 API 用户/客户端可以根据允许的范围访问或修改数据，有效地防止未经授权的违规行为。

2. 防止加密失效：

测试要求：

对所有传输和静止的数据采用高级的加密技术，包括使用行业标准的加密协议和定期更新加密密钥。

测试方法：

采用已确立的加密标准，并遵循健全的密钥管理实践。

预期结果：

数据的强加密，显著降低未经授权的数据访问和泄露的风险。

3. 防止注入缺陷：

测试要求：

通过验证所有输入数据并使用安全的数据库访问方法，保护 API 免受 SQL、NoSQL 和命令注入攻击。请注意，这里的注入缺陷不是指提示注入。提示注入将在 5.3.1 中讨论。

测试方法：

实施预处理语句、存储过程和全面的输入验证。

预期结果：

有效地缓解注入漏洞（的风险），确保数据的完整性和安全性。

4. 不安全的设计对策：

测试要求:

以安全为先的思想开发 API，将安全措施融入设计中，并定期进行威胁建模和风险评估。

测试方法:

遵循“设计即安全”的原则，进行威胁建模，并在整个设计和开发过程中将安全检查点融入其中。

预期结果:

具有弹性的 API 架构，从设计阶段开始最小化安全风险和漏洞（的危害）。

5. 安全的配置管理:

测试要求:

系统地配置和定期审核所有安全设置，使所有系统和软件保持最新安全补丁。

测试方法:

使用自动化工具进行配置管理，并定期进行安全审计。

预期结果:

配置完善的 API 环境，最大限度地减少由于错误配置而导致的漏洞风险。

6. 处理易受攻击和过时的组件:

测试要求:

持续监控和更新所有的第三方库、API、框架和依赖项，以防止易受攻击的组件（造成的危害）。

测试方法:

定期修补和更新组件，使用漏洞扫描工具。

预期结果:

降低由于第三方组件中的漏洞而导致安全漏洞的风险。

7. 健全的身份识别和认证：

测试要求：

实施强身份验证系统，包括多因素身份验证和安全的密码策略，以抵抗诸如凭证填充和暴力破解等攻击。

测试方法：

部署多因素身份验证，强制施行安全的密码使用规范，并监控异常的认证尝试。

预期结果：

强化对未授权访问的防护。

8. 软件和数据完整性保证：

测试要求：

定期验证软件和 API 处理的数据的完整性，防止未经授权的代码更改和数据篡改。

测试方法：

施行软件完整性检查以及数据验证流程。

预期结果：

确保软件及数据完整且可信。

9. 有效地安全日志及监控：

测试要求：

实施健全的日志记录和监控系统，能够实时地检测、报警和响应可疑活动或安全漏洞。

测试方法：

建立全面的日志记录及持续监控机制，以发现异常活动或安全事件。

预期结果：

应用系统能及早发现并迅速响应潜在的安全问题。

10. 服务器端请求伪造（SSRF）防御：

测试要求：

严格验证所有用户提供的输入，特别是用于服务器端请求的 URL 或数据，以防范 SSRF 攻击。

测试方法：

实施严格的输入验证和清理程序，着重防范 SSRF 漏洞。

预期结果：

有效缓解 SSRF 风险，保护 API 免受未经授权的内部网络访问。

5.4.2 缓存和验证测试

评估缓存机制在提高响应时间方面的效率，以及验证过程在确保 LLM 响应准确性和适当性方面的彻底性。

测试要求：

验证缓存机制在提高响应时间方面的有效性，以及验证过程确保响应准确性的鲁棒性。

测试方法：

通过评估缓存系统对重复查询响应时间的影响进行测试。这包括评估缓存命中率、缓存中的数据完整性以及缓存更新的效率。对于验证测试，实施检查确保来自 LLM 的响应是准确的、相关的，并且没有错误或不当的内容。这可以涉及自动验证检查和人工审查过程。

预期结果：

缓存机制在不损害数据完整性的情况下，可显著提高了频繁查询的响应时间。验证过程有效地确保 LLM 响应的准确性和适当性，最大限度地减少错误和不适当的内容。

5.5 代理行为

AI 代理是一个复杂的软件系统，它可以根据预定义的目标或对特定输入的响应来自主执行任务。其架构的核心是不同的组件，包括通过指令或问题激活代理的提示机制；用于存储过去对话细节的记忆模块，以提供上下文相关的响应；以及一个单独的知识库，其中包含丰富的真实世界、最新的信息，代理可以使用这些信息来准确理解 and 与世界交互。此外，战略规划和反思模块包含用于决策的算法，使代理能够评估选项、预测结果，并通过一组工具相应地执行操作。

尽管 AI 代理技术发展迅速，但其开发的通用标准仍未定义，这反而促进了 AI 代理的持续创新。在这个不断发展的领域中，安全的重要性不言而喻。随着 AI 代理变得越来越复杂，并越来越多地融入日常生活的各个方面，确保它们能够抵御威胁和脆弱性是至关重要的。这凸显了在开发 AI 代理时采取强有力的安全措施的重要性，以维护其操作的可信和完整性。

AI 应用程序中“代理行为”的测试规范可详细说明如下，涵盖提示、记忆、知识、规划、行动和工具等各个方面：

5.5.1 提示响应测试

1. 确认 AI 代理能够有效且准确地解释提示，并提供连贯、相关且语境相符的响应

测试要求：

确保 AI 代理准确地解释和响应提示。

测试方法：

测试 AI 代理理解和响应各种提示的能力，评估响应的清晰度、相关性和适当性。这涉及评估系统的自然语言理解和生成能力。详情请参阅第 5.3.1 和 5.3.2 节。

预期结果： AI 代理始终如一地正确解释提示，并提供连贯、相关且上下文适当的

响应。

2. 确认 AI 代理能够被有效地控制，并且不会采取不允许的自主行动。

测试要求：

确保 AI 代理不采取可能被禁止的自主行动。在采取任何可能引起安全问题的行动时，它还会请求人类批准。

测试方法：

AI 代理通常具有较高的权限。测试 AI 代理访问和采取可能被禁止的自主行动的能力。确保代理不会访问位置、文件或采取可能对立或被对手利用的行动。还要确保 AI 代理在采取行动之前请求人类批准，如果人类不允许采取特定行动，代理就不会采取该行动。

预期结果：

AI 代理在采取任何行动之前始终请求人类批准，并按预期工作。

5.5.2 记忆利用测试

验证 AI 在响应提示和执行任务时使用记忆的熟练程度，确保准确回忆和应用先前获取的信息。

测试要求：

验证 AI 代理在响应提示和执行任务时能有效地使用其记忆的能力。

测试方法：

通过评估 AI 如何在其响应和行动中整合先前学习或提供的信息来测试其记忆回忆和利用能力。这可以包括测试在引用过去的交互或数据时的一致性和准确性。

预期结果：

AI 展示了有效的记忆使用,在其响应和决策中准确地回忆和利用相关的过去信息。

5.5.3 知识应用测试

确保 AI 能够有效地利用其知识库（在大多数情况下，知识库由向量数据库、图数据库，甚至 SQL/NOSQL 数据库组成）提供信息丰富、准确和全面的响应和行动。

测试要求：

确保 AI 能够在响应和行动时能有效地应用其知识库。

测试方法：

通过提供需要利用存储信息的场景或查询来评估 AI 对其知识库的使用。评估应侧重于知识的相关度、准确度和深度。

预期结果：

AI 有效地应用其知识库，提供准确和深入的响应和行动，这些响应和行动基于其之前积累的信息。

5.5.4 规划能力测试

评估 AI 在规划和执行复杂任务方面的熟练程度，重点关注其战略思维和解决问题的能力。

测试要求：

测试 AI 规划和执行复杂任务的能力。

测试方法：

通过呈现需要采取行动或决策步骤的任务或场景来评估 AI 的规划能力。这涉及评估 AI 的战略思维和解决问题的能力。

预期结果：

AI 展示出健全的规划能力，为各种场景制定和执行有效的战略或行动计划。

5.5.5 行动执行测试

确保 AI 在各种场景下有效且适当地执行操作的能力，重点关注准确性、及时性和适用性。

测试要求：

验证 AI 有效且适当地执行操作的能力。

测试方法：

在模拟环境中或通过预定义的任务测试 AI 的操作执行。重点应放在 AI 所采取行动的准确性、及时性和适当性上。

预期结果：

AI 始终如一地正确、高效且适当地执行操作，以响应给定的任务或提示。

5.5.6 工具利用测试

确认 AI 在整合和利用可用工具方面的有效性，从而提高其在任务执行和响应提示时的性能和能力。

测试要求：

确保 AI 能有效地利用可用工具增强其能力。

测试方法：

评估 AI 在执行任务或响应提示时整合和使用各种工具（如数据库、软件库或硬件设备）的能力。这包括测试 AI 能够利用这些工具来提高其性能或功能的能力。

预期结果：

AI 成功集成并利用各种工具，在其响应和行动中展示出增强的性能和能力。

5.5.7 过度代理测试

对代理执行的行动范围进行批判性评估和调节，确保它们是平衡的，不会导致意外或过度的结果。

测试要求：

分析并限制代理执行的行动范围，以防止意外后果。

测试方法：

- (1) 情景测试：制定涵盖各种决策情境的广泛测试场景，包括边缘案例和潜在的道德困境。评估 AI 代理在每个场景中的反应和行为，确保其符合人类价值观和预期目标。
- (2) 对抗测试：采用模糊测试、输入操作和故意破坏系统等技术，以识别 AI 代理决策过程中存在的漏洞、意外后果和潜在的故障模式。
- (3) 模拟测试：创建现实世界环境的详细模拟，在真实条件下测试 AI 代理的决策能力。监测代理的性能、适应性以及对预定义规则和约束的遵守情况。
- (4) 访问控制测试：实施并全面测试访问控制机制，确保只有授权用户可以参与交互或修改 AI 代理的决策过程。这包括测试适当的身份验证、授权和审计功能，以防止未经授权的访问或篡改。根据最小权限原则，给予 AI 代理对系统和数据的有限访问至关重要。这意味着授予代理执行其预期功能所需的最低访问级别，不多也不少。通过限制代理对敏感信息和关键系统的访问，我们可以减轻与受损或故障 AI 代理相关的潜在风险。这种有限访问方法应经过严格测试，以确保代理不能超出其预期权限或获得对受保护资源的未经授权的访问。应定期进行审计和审查，以验证随着 AI 代理的能力和部署环境随着时间的推移而发展，访问控制仍然有效并保持适当的范围。
- (5) 人机协同测试：让专家参与测试过程，以提供监督、指导和反馈，帮助确保代理的行为与人类判断一致，并可根据需要进行调整。

(6) 持续监控和评估：在部署后，对 AI 代理的决策过程，实施持续的监控和评估机制。定期评估代理的性能，并与已确立的指标、基准和人类反馈进行对比，以发现存在的偏差或需要改进的领域。

预期结果：

代理展示出平衡且受控的代理行为，不会有过度或意外的行动。

5.6 微调

AI 应用“微调”的测试规范，主要关注数据隐私检查、基础模型选择、模型部署和训练数据污染测试。该规范结构如下：

5.6.1 数据隐私检查测试

为了保证用于微调 AI 模型的数据严格遵循隐私和数据保护法规，应确保其来源合乎伦理道德并适当的匿名化。

测试要求：

确保用于微调的数据尊重隐私并符合相关的数据保护法规。

测试方法：

对微调过程中的数据收集、处理和存储实践进行全面审查。这包括验证是否遵守隐私法（如 GDPR 或 HIPAA），确保数据在需要进行匿名化，并检查在使用个人数据时是否具有合适的同意机制并明确用途。检查是否使用差分隐私（DP）来保护训练数据的隐私：DP 是一种在共享关于一组个人的信息时提供隐私的方法，通过描述组内的模式，同时保留关于特定个人的信息。这是通过对个人数据进行任意小的更改来实现的，这些更改不会改变感兴趣的统计数据。因此，数据不能用于推断任何个人的太多信息。如果使用 DP，请使用 NITS 的《评估差分隐私保证指南》进行评估，请参阅以下链接：

<https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-226.ipd.pdf>

预期结果：

微调过程中使用的数据完全符合隐私法规，这些数据被合理的匿名化或假名化，其来源合乎伦理道德，并获得了所有必要的许可。

5.6.2 用于微调的基础模型选择测试

确定所选基础模型是否与具体应用和微调要求（详见文档第 5.1 节）最佳匹配，确保其性能和适应性适合预期目的。

测试要求：

确认所选基础模型最适合特定应用和微调过程。另请参阅本文档中的第 5.1 节。

测试方法：

评估基础模型的性能、对目标领域的适用性及其有效集成新数据的能力。这可以包括根据特定性能指标对模型进行基准测试，并评估其对微调过程中引入的新数据的适应性。

预期结果：

所选的基础模型与微调目标高度兼容，在微调后展现出显著的性能提升，并适用于特定的应用领域。

5.6.3 用于微调的基础模型存储测试

确保微调模型正确存储在模型注册表中。根据微调程序适当地更新微调模型卡。

测试要求：

确认任何微调模型都以适当的访问权限正确存储。模型以正确的模型卡片适当存储。

测试方法：

根据模型微调所依据的数据评估微调模型的访问权限。确保对特定敏感度模型没有

权限的用户，在模型使用更高敏感度数据进行微调后，无法访问该模型。检查模型卡是否包含模型的正确数据，并保持最新。

预期结果：

所选的基础模型与微调目标高度兼容，在微调后展现出显著的性能提升，并适用于特定的应用领域。

5.6.4 训练数据污染测试

确保训练数据的完整性，检测并防止篡改数据、偏见数据或污染数据，从而保持模型的无偏性。

测试要求：

检测并防止训练数据被篡改或（注入）偏见。

测试方法：

检查训练数据的完整性，寻找篡改、注入偏见或其他形式的污染的迹象。

预期结果：

训练数据没有被篡改或注入偏见，确保模型的完整性和无偏性。

5.6.5 微调后的模型部署测试

验证微调后的模型在生产环境中高效、安全地运行，并能有效扩展，保持高性能和对抗安全威胁的鲁棒性。

测试要求：

确保微调后的模型在生产环境中高效、安全地运行，并不会泄露机密、敏感信息或专有数据。

测试方法：

经过微调的模型部署完成后，对其性能、可扩展性和安全性进行测试，并对输入请求进行妥善控制，使其包含有一些可能诱使模型暴露机密、敏感或专有数据的内容。这包括在真实环境中评估模型的响应准确性、延迟、处理高负载场景的能力以及对安全威胁的抵抗力。

预期结果：

经过微调的模型在生产环境中保持了高性能和准确性，在不同的负载条件下能够有效地扩展，并且对潜在的威胁表现出强大的抵抗能力。

5.7 响应处理

AI 应用“响应处理”的测试规范，主要关注基本事实检查、相关性检查、不良内容检查和伦理检查，具体如下：

5.7.1 基本事实检查测试

1. 事实检查测试

测试要求：

确保 AI 应用的响应内容基于现实且内容准确无误。

测试方法：

实施测试以验证响应的事实准确性。这涉及将人工智能的回应与可靠的数据源或既定事实进行交叉对比，特别是对于涉及事实信息陈述的回答。

预期结果：

AI 始终提供在事实上准确且可验证的回答，展现出基于现实内容的牢固基础。

2. 反馈回路机制测试：

测试要求：

建立并测试反馈系统，让用户或其他系统报告 AI 生成内容的问题，促进持续改进。

测试方法：

- (1) 评估反馈机制在收集用户或系统报告问题方面的有效性。
- (2) 测试分析和处理反馈推动改进的过程。
- (3) 评估人工智能应用程序对反馈的反应能力以及其迭代增强内容生成能力。

预期结果：

- (1) 确认反馈机制能有效收集和分析用户或系统报告的问题。
- (2) 确保 AI 应用对反馈具有响应性，并在生成的内容中展现出持续改进。
- (3) 识别并解决与反馈处理及改进流程相关的问题。

5.7.2 相关性检查测试

确认 AI 的回应始终与给定的提示或查询保持相关且在上下文中恰当。

测试要求： 验证 AI 的回应与给定的提示或查询的相关性。

测试方法：

通过将 AI 的回应与提示的上下文和内容进行比较来评估其相关性。这包括评估各种不同的提示，并确保 AI 的回应始终切题，且适合当前的查询。

预期结果：

AI 的回应始终与提示相关，展示了对上下文和查询具体要求的理解。

5.7.3 不良内容检查测试

确保 AI 的回应中不含任何有害、冒犯或不适宜的内容，维护高质量的对话水平和适当性。

测试要求：

确保 AI 的回应中不含任何有害、冒犯或不适宜的内容。

测试方法：

进行测试以识别和衡量 AI 回应中有害或不适宜语言的存在。这可能涉及使用预定义的有害内容标记进行自动化扫描，以及由人类评估员进行的手动审查。根据具体的下游 AI 应用，展示这一需求可能需要大量资源或特定的专业知识。因此，组织可能需要一系列的方法实施这一需求。例如，有公共资源和第三方公司可以协助实现这一需求。什么内容被视为有害、冒犯或不适宜是高度依赖于上下文的，并会根据具体的下游 AI 应用和运营环境而变化。评估人员在测试过程中应考虑并考虑到这一上下文背景。

预期结果：

AI 应用始终避免生成有害或不适宜的内容，保持高质量的对话水平和适当性。

5.7.4 伦理检查测试

确保 AI 的回应从道德上是健全的，没有有害的偏见或刻板印象，并且不支持不道德的行为，与既定的伦理指导方针保持一致。

测试要求：

验证 AI 的回应是否遵循了伦理指导方针，不传播有害的偏见或不道德的观点。

测试方法：

评估 AI 回应的道德完整性，检查是否存在偏见、刻板印象或推广不道德行为的情况。这可能涉及使用伦理指南或框架作为评估的基准。根据具体的下游 AI 应用，实现这一需求可能需要大量资源或特定的专业知识。因此，组织可能需要一系列的方法来证明这一需求。例如，有公共资源和第三方公司可以协助实施这一需求。什么被认为是不道德的是高度依赖于上下文的，并会根据具体的下游 AI 应用和运营环境而变化。评估人员在测试过程中应考虑并考虑到这一上下文背景。

预期结果：

AI 始终提供无有害偏见和刻板印象的回应，符合道德标准，且不促进不道德行为。

5.7.5 不安全输出处理测试

测试要求：

确保模型输出的安全处理，防止被恶意利用。

测试方法：

验证涉及模型输出处理的机制和流程，检查可能导致被利用的漏洞。

预期结果：

输出处理过程是安全的，有效地防止任何形式的利用。

5.7.6 后门攻击测试

测试要求：

测试人工智能系统对后门攻击的弹性，后门攻击涉及恶意训练的模型，这些模型在常规情况下表现正常，但在特定触发条件下会表现出有针对性的错误分类或行为。

测试方法：

实施测试，尝试在 AI 系统的训练或微调阶段引入后门触发器。评估模型在这些触发条件下的行为和输出，以检测任何针对性的错误分类或偏离预期性能的表现。评估设计用于检测和缓解后门攻击的防御措施和监控系统的有效性。

预期结果：

AI 系统展现出强大的抵御后门攻击的能力，在存在潜在触发器的情况下仍维持预期的性能和输出。

防御措施和监控系统能有效地检测并在特定条件下标记任何试图引入后门或模型出现可疑行为的尝试。系统能够在不损害整体功能、安全或完整性的前提下，承受或从后门攻击中恢复。

5.7.7 隐私和版权合规检查

测试要求：

确保 AI 系统的回应和输出遵守相关的隐私法规和版权法律，尊重用户隐私和知识产权。

测试方法：

评估 AI 系统处理用户数据和个人信息的方式，验证其是否符合如 GDPR、CCPA 或其他地区特有法律等适用的隐私规定。测试 AI 系统保护用户隐私的能力，即在其回

应和输出中匿名化或保护敏感信息。通过测试 AI 系统正确归属内容、避免剽窃以及在使用受版权保护材料时获取必要许可的能力，评估其对知识产权的尊重。运用内容来源与真实性联盟（C2PA）标准来验证 AI 系统中所使用数据的来源，确保遵守版权要求并实现正确的归属。

预期结果：

AI 系统一贯地展示了遵守相关隐私法规的能力，适当地处理和保护用户数据和个人信息在其回应和输出中。系统有效地匿名化或保护敏感的用户信息，确保在整个交互过程中隐私得到维护。AI 系统通过正确归属内容、避免剽窃以及在输出中使用受版权保护材料时获得所需许可，尊重了知识产权。系统的回应和输出免于隐私侵犯和版权侵权，降低了部署 AI 应用的组织面临的法律风险。审计和评估确认了 AI 系统遵守隐私和版权要求，向利益相关者和监管机构提供了保证。AI 系统展现了适应隐私法规和知识产权法律更新的能力，确保了持续的合规性。C2PA 标准成功实施，验证了 AI 系统中使用数据的来源，实现了正确的归属并符合版权要求。

5.7.8 妥善处理未知或不支持的查询

测试要求：

确保 AI 系统能够优雅地处理未知、不受支持或与使用场景无关的查询，向用户提供适当的反馈。

测试方法：

测试 AI 系统对超出其知识领域、不受支持或与预期使用案例无关的查询的响应。评估系统提供信息丰富且用户友好的反馈的能力，引导用户提出更合适的查询或指向相关资源。

预期结果：

AI 系统能够妥善地处理未知、不受支持或无关的查询，避免造成用户的困惑或给出误导性回复。系统向用户提供清晰和信息丰富的反馈，建议替代查询，提供指导，或在适当的时候将他们重定向到相关资源。

5.8 AI 应用运行时安全

以下是 AI 应用程序运行时安全的测试规范。

5.8.1 数据保护测试

为了保护敏感数据并维护隐私标准，必须实施严格措施以确保数据完整性和保密性。

测试要求：

确保数据的完整性和保密性。

测试方法：

实施加密效能、访问控制稳健性以及持续监控系统的测试。当采用新兴的隐私保护技术，如机密计算或其他隐私增强技术（PETs），如全同态加密（FHE），至关重要的是验证这些 PET 技术是否正确实施并按预期运行。PET 实施的恰当验证有助于确保正在处理的数据的保密性和完整性，以及隐私保护技术的有效性。如果没有彻底的验证，PET 解决方案可能无法提供预期的保护级别，从而有可能使敏感数据或计算暴露于未经授权的访问或篡改之下。

预期结果：

数据在静止状态和传输过程中完全加密，访问控制有效地阻止未经授权的访问，监控系统能及时检测并报告任何数据泄露或违规事件。

5.8.2 模型安全测试

以下测试规范旨在保护经过微调的 AI 模型免受对抗性攻击和未经授权的复制：

1. 模型水印：

测试要求：

在 AI 模型中实施水印技术，以在模型内部嵌入一个独特的标识符。此标识符应有

助于在模型被复制时识别模型的所有权和来源。

测试方法：

通过尝试复制模型并验证是否可以提取嵌入的标识符来测试水印过程的有效性。另外，评估集成水印时模型性能是否降级（如果有的话）。

预期结果：

通过水印成功识别模型的所有权和来源，从而阻止了未经授权的复制。如果存在模型性能降级，不应违反或损害预期用途或安全性或安全结果（例如医疗决策等）。

2. 访问控制和身份验证：

测试要求：

对访问模型实施严格的访问控制机制和身份验证协议。

测试方法：

测试用户身份验证过程、基于角色的访问控制，并监控访问日志以检测未经授权的访问尝试。

预期结果：

健全的访问控制，确保只有授权用户可以访问模型，未经授权的尝试能够被及时检测并阻止。

3. API安全与速率限制：

测试要求：

加强用于与模型交互的 API 的安全性。

测试方法：

进行全面的测试，以验证 API 端点的安全性，包括进行速率限制，以防止大规模下载或抓取模型数据。

预期结果：

具备有效限速的安全 API，以防范数据滥用和未经授权的访问。

4. 代码/参数混淆和加密：

测试要求：

采用代码/参数混淆和加密技术，使模型难以理解且难以复制。

测试方法：

测试代码混淆和加密的鲁棒性，以抵抗逆向工程尝试和未经授权的访问。

预期结果：

代码/参数难以被逆向，阻止对模型的复制企图。

5. 定期安全审计：

测试要求：

对托管模型的基础设施进行定期安全审计。

测试方法：

进行漏洞评估，检查可能允许未经授权访问或下载模型的安全弱点。

预期结果：

持续识别并修复漏洞，确保基础设施的安全性。

6. 入侵检测和异常监控：

测试要求：

实施入侵检测系统和异常监控工具，以识别可能表明试图窃取模型的可疑活动。

测试方法：

通过模拟入侵尝试测试入侵检测系统的有效性，并监控警报。

预期结果：

可早期检测到可疑活动，能对潜在的安全威胁做出及时响应。

7. 法律保护与合规性检查：

测试要求：

审查并测试是否符合版权、专利和商业机密等法律保护，这些保护为保护模型免遭盗用提供了法律依据。

测试方法：

进行法律和合规性检查，以确保遵守知识产权、数据保护法和任何应用程序 AI 合规要求。

预期结果：

法律保护到位，遵守相关法律，为模型保护提供法律依据。

5.8.3 基础设施安全测试

为了防止利用可能危及操作的漏洞，AI 应用的基础架构需要具备强大的安全性。

测试要求：

保障承载 AI 应用的基础架构的安全。

测试方法：

定期更新和修补系统，进行网络安全评估，并评估硬件安全性。频繁地进行漏洞扫描，以识别任何潜在的弱点或不必要的服务。利用强化验证技术，确保系统的安全性和稳健性。

预期结果：

基础设施对网络威胁表现出强大的防御能力，所有组件都已更新至最新安全补丁。

5.8.4 API 安全测试

应用程序编程接口（API）必须经过严格的测试，以验证身份验证、授权、速率限制和输入净化机制，如第 5.4.1 节所述，以实现与外部系统的安全集成。

测试要求：

确保通过 API 与外部系统进行安全交互。详情请参阅 5.4.1

测试方法：

测试身份验证、授权、速率限制和输入验证。

预期结果：

API 对未经授权的访问和滥用表现出强大的弹性，保持数据完整性和系统稳定性。

5.8.5 合规和审计追踪测试

遵守适用的法律和标准对于道德的 AI 应用至关重要，需要持续的合规性验证和详细的审计追踪确认一致性。

测试要求：

验证是否符合相关法律和标准，并保持维护有效的审计追踪。

测试方法：

执行定期合规检查和审计日志分析。

预期结果：

人工智能应用程序符合法律标准，审计跟踪能够准确追踪系统访问和变更记录。

5.8.6 实时监控和异常检测测试

对系统活动和模型性能中的异常进行主动监控对于早期检测影响安全或准确性的新出现问题至关重要。

测试要求：

检测并处理异常活动或模型性能的偏差。

测试方法：

实施并测试来自网络层、操作系统层和应用层的实时监控和异常检测系统。

预期结果：

系统能有效识别并发出潜在安全问题的警报，便于迅速响应。

5.8.7 配置与态势管理测试

为了确保安全基础设施中 SaaS 应用程序、身份和数据的完整性，通过安全态势管理（SSPM）解决方案进行配置和态势管理测试至关重要。SSPM 解决方案协助安全团队维护当前的监控和安全更新。通过建立安全基线，这些解决方案有助于监督配置设置，并在出现任何偏差时向安全团队发出警报，这是管理配置漂移和识别其他与配置相关的漏洞的关键。配置漂移——由于多种原因可能发生的对系统的未经授权的更改——对系

统完整性构成风险。手动的态势检查繁琐且容易出错。因此，采用集成了 AI 和自动化的 SSPM 解决方案进行持续的配置检查极为有益。这些高级工具能够在出现偏差时自动纠正配置或恢复到基线。

测试要求：

确保 SSPM 有效监控和维护 SaaS 应用程序、身份和数据的安全状态，并在配置漂移时及时发出警报。

测试方法：

通过 AI 驱动的自动化实现 SSPM 解决方案，以进行持续的配置验证和管理。定期评估这些工具在检测和纠正错误配置方面的有效性。

预期结果：

SSPM 解决方案应始终保持基线配置设置，自动检测和纠正配置偏差，并确保 IT 审计准备就绪。这些解决方案应提供全面的 SaaS 安全态势测量，并随着时间的推移启用风险报告，以确保持续符合安全标准。

5.8.8 事件响应计划测试

必须建立并通过对模拟事件的测试来完善事件响应计划，以便能够及时和有序地处理安全事件和其他危机。

测试要求：

制定有效的紧急事件响应计划。

测试方法：

进行响应演练、通信和影响评估测试，包括接收第三方报告。

预期结果：

事件响应协议能够迅速且有效地执行，最大限度地减少影响和恢复时间。

5.8.9 用户访问管理测试

限制用户权限的粒度访问控制和多因素身份验证系统为防止未经授权访问 AI 应用程序提供了关键的防线。

测试要求：

严格控制用户对人工智能应用程序的访问。

测试方法：

审核用户权限并测试多因素身份验证系统。

预期结果：

访问权限得到适当限制，身份验证机制可靠地防止未经授权的访问。

5.8.10 依赖和第三方组件安全测试

由于外部库和组件如果被破坏会带来严重的安全风险，因此在集成之前需要对源代码进行细致的验证和漏洞测试，并且还需要定期进行依赖安全检查。

测试要求：

确保外部库和组件的安全性。

测试方法：

执行源验证和漏洞扫描。

预期结果：

所有依赖项都来自受信任的来源，并且没有已知的漏洞，依赖项检查是一个持续的过程。

5.8.11 安全鲁棒性测试与验证

对 AI 应用进行模拟复杂的网络攻击，如渗透测试、漏洞扫描和道德黑客攻击，对于揭示弱点和加强防御以防止被利用是至关重要。

测试要求：

识别并缓解潜在的安全漏洞风险。

测试方法：

进行渗透测试、漏洞评估和模拟黑客攻击测试。

预期结果：

AI 应用展现对现实世界攻击的强大防御能力，且漏洞能够被迅速识别并解决。

5.8.12 可用性测试

为了确保在高需求下的可用性和可靠性，人工智能系统必须在模拟的高流量场景下展示出在逼近基础设施负载极限时的弹性性能。

测试要求：

评估模型在高负载场景下的韧性和性能。

测试方法：

将模型置于高负载场景下，评估其性能和处理大流量而不中断服务的能力。

预期结果：

即使在高负载条件下，模型也能保持功能和性能，防止拒绝服务（DoS）事件的发生。

5.8.13 侦察防护测试

保护人工智能应用程序的敏感细节免受外部发现是至关重要的。必须通过审计和模拟攻击来发现并解决可能允许未经授权侦察的漏洞。

测试要求：

进行模拟和审计以识别外部实体可能在运行时用来收集关于 AI 应用敏感信息的方法。

测试方法：

模拟侦察技术，评估 AI 应用对信息收集的易感性。审计 AI 应用屏蔽敏感细节和防止未经授权数据披露的能力。

预期结果：

识别与信息披露相关的漏洞，并确认 AI 应用对侦察防护的能力。

5.8.14 持久性缓解测试

为了防止对手获得并保持对人工智能系统的隐蔽访问以进行利用，必须通过严格的持续安全测试和补救措施，不断识别并消除与持久性相关的漏洞。

测试要求： 定期扫描并消除可能允许攻击者在运行时对 AI 应用保持持久访问的漏洞。

测试方法：

定期进行漏洞扫描和评估，识别和修复与持久性相关的漏洞。

预期结果：

检测并缓解可能使攻击者在运行时实现持久访问的漏洞。

5.8.15 权限提升防御测试

防止在运行时未经授权提升人工智能系统中的用户权限对于维护访问控制完整性至关重要，需要对权限提升攻击场景进行严格的测试。

测试要求：

评估系统在运行时防止未经授权提升用户权限的能力。

测试方法：

测试权限提升场景以评估系统防御。

预期结果：

确认 AI 应用在运行时有效防止了未经授权的权限提升。

5.8.16 防御规避检测测试

为了维护健全的安全标准，人工智能系统必须具备检测并应对在实时操作中试图绕过或禁用关键安全防护的尝试。

测试要求：

测试系统在运行时检测并应对试图规避现有安全机制的能力。

测试方法：

模拟规避尝试并评估系统检测并应对它们的能力。

预期结果：

验证系统在运行时能够有效检测并应对规避尝试。

5.8.17 发现抗性测试

保护人工智能系统专有细节和敏感功能免遭未经授权的访问和信息泄露，在运行时需要进行严格的测试以确保防护效果。

测试要求：

进行评估，以确保内部系统细节和功能在运行时不易被未经授权的用户发现。

测试方法：

测试信息泄露和未经授权的发现尝试。

预期结果：

验证内部系统详细信息在运行时得到充分保护，防止未经授权的发现。

5.8.18 数据采集防护测试

对保障措施严格评估必须确认人工智能系统能够防止未经授权的数据收集和实时操作中的数据泄露，以保护隐私并保持数据控制。

测试要求：

验证系统在运行时具有足够的措施来防止未经授权的数据收集和泄漏。

测试方法：

测试数据采集保障措施并评估数据处理实践。

预期结果：

数据采集受到控制，并在运行时受到保护，防止未经授权的访问或泄露。

5.9 附加测试规范

除了上述针对 AI 应用堆栈的测试规范外，以下附加测试规范对于 AI 安全性也非常重要。

5.9.1 供应链漏洞测试

以下是用于识别和缓解 LLM 应用程序生命周期中漏洞的测试规范。

1. 第三方组件评估

测试要求:

评估应用在供应链中使用的所有第三方组件、库和依赖项，以识别漏洞或安全弱项。

测试方法:

- (1) 对供应链中的第三方组件进行全面检查，重点关注它们的安全状况。
- (2) 使用自动化漏洞扫描工具识别供应链中第三方软件组件的已知漏洞。
- (3) 对供应链中的第三方代码进行手动代码审查和分析，发现那些自动化工具可能无法检测到的安全漏洞。

预期结果:

提供一份详细报告，列出供应链中第三方组件的漏洞和安全弱点，并提出缓解措施。

2. 代码审查和分析

测试要求:

执行详细的软件物料清单（**SBOM**）分析，以加强安全措施。这包括对供应链中集成的第三方代码的详细检查，是识别和缓解潜在风险的关键步骤。

测试方法:

审查第三方对供应链中使用的第三方代码的静态和动态代码评估结果，并进行 **SBOM** 分析以进行漏洞审查、第三方依赖项审查、软件成分分析（**SCA**）和许可证审查。使用静态代码分析工具自动检测能指示第三方代码中安全漏洞的代码模式。如果发现漏洞，应在修复后进行后续 **SCA** 和漏洞评估，以验证已识别的问题是否已得到妥善解决。

预期结果:

提供一份报告，突出显示供应链中使用的第三方代码中的安全漏洞，并提供关于如何修复这些问题的指导。

3. 动态应用安全测试（**DAST**）供应链集成测试

测试要求:

在运行时使用动态测试技术评估供应链中与第三方组件和服务的集成的安全性。

测试方法：

使用自动化动态测试工具与应用程序及其集成（包括第三方组件）进行交互，以识别漏洞。测试供应链集成中的输入验证问题、访问控制问题和身份验证弱点等问题。

预期结果：

检测到运行时供应链集成中的安全漏洞，包括静态代码分析中可能未明显的漏洞。

4. 软件成分分析（SCA 供应链重点）**测试要求：**

利用软件组成分析工具识别和跟踪供应链中使用的开源组件。

测试方法：

使用 SCA 工具创建供应链中使用的所有开源组件和库的清单。将清单与已知漏洞数据库进行对比，以识别供应链中有安全问题的组件。

预期结果：

供应链中开源组件的完整清单和有已知漏洞的组件列表，并提供更新建议。

5. 威胁建模**测试要求：**

进行威胁建模演练，识别 LLM 应用程序供应链中特定的潜在威胁和攻击向量，重点是第三方代码。

测试方法：

与利益相关者合作创建威胁模型，记录供应链中潜在的安全风险和攻击场景。评估供应链中第三方组件的安全状况并识别安全漏洞。

预期结果：

威胁模型清晰地理解与第三方代码相关的供应链特定安全风险，并制定应对这些风险的路线图。

6. 供应链验证（第三方组件信任）**测试要求：**

验证供应链中从供应商或第三方来源接收的软件组件和更新的真实性和完整性。

测试方法：

实施安全更新机制，例如数字签名或校验和，以确保供应链中的第三方组件和更新在传输过程中未被篡改。

验证供应商和第三方来源的身份和安全实践，以建立供应链信任。

预期结果：

对供应链中从供应商或第三方来源接收的组件和更新的真实性和完整性充满信心。

7. 集成安全测试

测试要求： 评估与外部系统和服务的集成的安全性，强调 LLM 应用与供应链中第三方实体之间的安全数据交换。

测试方法：

对供应链中的集成点进行渗透测试和漏洞扫描，以识别潜在弱点。验证与供应链中第三方实体交换的数据是否已加密并安全传输。

预期结果： 确保与外部系统和供应链中的第三方实体的安全集成，防止数据泄露和未经授权的访问。

8. 建立和维护开源 AI 应用的社区信任

测试要求：

开源 AI 应用的社区信任

测试方法：

促进开源 AI 应用的透明和积极的社区审查过程。鼓励社区参与验证数据使用和伦理实践。保持对安全最佳实践/补丁的更新。尽可能发布完整或部分评估报告，例如红队测试结果或产品评论。

预期结果：

开源 AI 应用程序的社区信任和满意度水平较高，以积极的社区反馈和参与度为标志，如代码库克隆、分支、关注者和星标等是一些客观衡量标准。为模型发布者设立开放的标准，并期望它们达到同行水平。

5.9.2 安全的 AI 应用开发过程

以下是针对安全的 AI 应用开发过程的测试规范。

1. AI 开发安全测试

测试要求：

通过系统性测试评估 AI 开发生命周期的安全性。

测试方法：

检查是否实施了安全的 SDLC 实践，确保在整个开发过程中优先考虑并跟踪明确的安全需求。确保开发团队接受了适当的安全培训，并了解安全编码实践和常见漏洞。验证是否存在考虑安全方面的正式同行评审过程。检查是否使用静态分析工具扫描代码库中的漏洞，并审查和解决扫描结果。确保在设计阶段进行威胁建模，以识别和缓解潜在的安全风险。确保团队意识到安全风险，定期进行安全测试，遵循安全配置实践，制定应急响应计划，并根据经验教训不断改进其安全实践。

测试数据处理实践，确保数据安全和隐私得到维护。评估算法实现的安全性和潜在漏洞。

预期结果：

识别和缓解 AI 开发过程中与安全相关的问题。

2. AI 需求验证测试

测试要求：

评估需求规划是否足以确保 AI 应用程序符合指定的基准、道德准则和合规标准。

测试方法：

- (1) 测试 AI 应用程序功能是否符合指定的基准和需求。
- (2) 评估 AI 应用程序行为中对伦理指南和伦理考虑的遵守情况。何为不道德高度依赖于具体的下游 AI 应用程序和操作环境。在测试过程中，评估人员应考虑

和考虑上下文因素。

(3) 验证是否符合相关的监管和合规标准。

预期结果：

确认 AI 应用程序的需求得到满足，伦理指南得到遵守，并符合合规标准。

3. AI 应用程序开发完整性测试

测试要求：

检查 AI 应用开发过程是否符合安全和伦理指南，特别是在持续学习环境中。

测试方法：

测试 AI 应用程序开发实践的完整性，包括持续学习过程。评估在 AI 应用程序开发中实施的安全措施和道德考虑。不道德行为高度依赖于上下文，并会根据具体的下游 AI 应用程序和操作环境而变化。对此要求的评估人员应在测试时思考和关联到上下文。

预期结果：

确保 AI 应用开发的完整性，特别是在持续学习场景中，重点关注安全和伦理。

5.9.3 AI 应用治理测试

以下是针对 AI 应用治理的验证和测试规范。

1. 培训与意识评估测试

测试要求：

评估培训项目在培育个人关于 AI 特定风险和安全实践方面的有效性。

测试方法：

- (1) 评估与 AI 风险和安全实践相关的培训内容和交付方式。
- (2) 测试参与者在现实世界 AI 场景中的知识保留和应用能力。
- (3) 收集受训人员对培训过程中存在的弱点或问题的反馈。

预期结果：

确认培训项目在提升 AI 风险意识和促进安全实践方面的有效性。

2. AI 网络安全管理评估测试

测试要求：

评估为管理 AI 网络安全项目而实施的策略和实践。

测试方法：

- (1) 评估针对 AI 应用的网络安全策略和协议。
- (2) 测试识别和缓解 AI 相关网络威胁的实践有效性。

预期结果：

确保策略和实践有效地管理 AI 网络安全，降低网络威胁的风险。

3. 领导与治理测试

测试要求：

测试高层领导在指导 AI 安全策略方面的有效性。

测试方法：

评估领导层对 AI 安全治理的参与度和承诺。测试高层决策与 AI 安全目标和道德考虑的契合度。

预期结果：

确认高层领导和治理在塑造 AI 安全策略方面的有效性。

4. AI 项目管理审计测试

测试要求：

从项目启动到部署，全面审查和测试 AI 项目管理的安全性和道德监督。

测试方法：

- (1) 审计 AI 项目管理实践，确保从始至终都考虑安全性和道德因素。
- (2) 评估项目管理团队在识别和缓解 AI 应用潜在风险方面的能力。
- (3) 检查项目文档、流程和政策是否符合最佳安全实践。

预期结果：

确保 AI 项目管理从始至终都受到全面的安全和道德监督。

5. AI 过程审计测试

测试要求：

定期或连续地审查 AI 过程，以确保从项目开始到部署期间有全面的安全和道德监督。

测试方法：

审计 AI 过程，以确保从项目开始就整合了安全和道德考量，并且随着应用程序的变化而持续进行。评估运行中 AI 环境的变化，以发现可能导致不符合安全和道德准则的问题。

预期结果：

确保随着 AI 应用程序的变化，AI 过程会持续更新，重点是整个生命周期的 AI 服务、安全和道德监督。

6. 算法偏见和公平性测试

测试要求：

评估 AI 算法和模型以识别潜在的偏见，并确保公平和非歧视性的结果。

测试方法：

使用各种数据集和情境来测试 AI 算法和模型，识别出特定群体或个人的潜在偏见或不公平对待。对用于减少算法偏见和促进公平性的过程和技术进行评估。

预期结果：

确认 AI 算法和模型没有重大偏见，并提供公平和非歧视性的结果。

7. AI 伦理审查测试

测试要求：

评估 AI 伦理审查过程的有效性，以确保与伦理原则和准则的一致性。

测试方法：

审计 AI 应用程序对伦理影响审查的流程和程序。测试实施的道德准则和解决道德关切及困境的决策过程。

预期结果：

确保 AI 伦理审查过程中在坚持道德原则和准则方面的全面有效。

8. AI 风险评估和缓解测试

测试要求：

评定 AI 应用程序相关的潜在风险识别、评估和缓解的过程和实践。

测试方法：

测试为 AI 应用程序实施的风险评估方法和程序。评估为解决已识别风险而实施的风险缓解策略和控制的有效性。

预期结果：

确认 AI 风险评估和缓解过程是健全的，能够有效管理潜在风险。

9. AI 事件响应和恢复测试

测试要求：

评估 AI 对相关事件或中断事件的响应和恢复计划的准备情况和有效性。

测试方法：

在模拟的 AI 相关事件情景中测试事件响应和恢复程序。评估事件检测、遏制和恢复策略的有效性，也包括通信和报告协议。

预期结果：

确保组织能够及时有效地应对 AI 相关事件或中断，并从中恢复过来。

10. AI 相关的法律法规和行业标准的合规性

测试要求：

确保遵守与 AI 应用程序相关的法律法规和行业标准。

测试方法：

评估组织的流程和控制，以监控和遵守适用的 AI 相关法律法规和行业标准。测试合规要求的实施以及合规监控和报告机制的有效性。

预期结果：

确认组织遵守与 AI 相关的法律法规和行业标准。

11. AI 数据治理

测试要求：

对管理和治理 AI 应用程序中所用数据的流程和控制进行评估，包括数据质量、隐私和安全性。

测试方法：

测试与 AI 应用程序相关的数据治理实践，包括数据获取、预处理、存储和访问控制。评估实施的措施，以确保数据质量、保护敏感数据和维护数据完整性。

预期结果：

确保 AI 应用程序中使用的数据得到有效管理，确保数据质量、隐私和安全。

12. AI 模型生命周期管理

测试要求：

评估管理微调 AI 模型生命周期的做法和流程，包括开发、部署、监控和更新。

测试方法：

测试微调 AI 模型的开发、验证、部署以及持续监控和维护的程序和控制。评估基于性能、准确性和潜在风险更新和淘汰 AI 模型的过程。

预期结果：

确认 AI 模型在其生命周期中得到有效管理，确保适当的开发、部署、监控和更新流程。

5.9.4 安全模型共享和部署

如果将微调模型与第三方共享，请执行以下测试。

1. 安全模型共享和部署测试

测试要求：

强制执行并测试用于安全地共享和部署 AI 模型的严格协议，特别是对于高风险应用，以确保模型的完整性和安全性保持不变。确保正确共享模型卡片包含正确的详细信息。

测试方法：

测试模型部署过程是否符合安全协议，包括代码签名、加密、模型水印和访问控制。评估模型共享机制，以验证只有授权实体可以访问和部署模型。进行渗透测试以识别模型部署所在基础设施中的漏洞。评估模型部署对整体系统安全态势的影响。

预期结果：

验证 AI 模型是否完整的和保持原有安全措施不变的方式进行了安全部署。确保只有授权实体可以共享和部署模型。识别并缓解部署基础设施中的漏洞。确认模型部署不会危及整体系统的安全。

2. 安全模型版本控制和回滚测试

测试要求：

实施并测试 AI 模型的版本控制机制，以便在出现问题或漏洞时允许安全回滚。确保模型卡片在模型之间继承并更新。

测试方法：

- (1) 评估版本系统维护 AI 模型历史版本的能力。
- (2) 测试回滚过程，以确保在识别出问题时可以安全启动它。
- (3) 进行模型回滚模拟，以验证它们在恢复系统完整性方面的有效性。

预期结果：

- (1) 确认 AI 模型版本得到安全管理，允许安全回滚。
- (2) 验证回滚过程可以安全启动。
- (3) 确保模型回滚有效恢复系统完整性。

3. 安全模型监控和告警测试

测试要求：

为部署的 AI 模型建立持续的监控和告警机制，以检测异常和安全威胁。确保在模型更改时及时进行模型卡片更新。

测试方法：

评估监控系统在识别模型异常行为方面的有效性，并在模型发生变化时更新模型卡片。测试管理流程变更或告警机制，以确保及时通知安全事件。进行安全威胁模拟，

以评估监控系统的反应。

预期结果：

验证持续监控可检测到模型异常行为。确认告警机制可及时通知安全事件。确保监控系统有效响应安全威胁。

4. 安全模型访问控制和权限测试

测试要求：

强制执行对 AI 模型部署和使用的限制性访问控制和权限。

测试方法：

评估 AI 模型部署和采取访问的控制措施。测试权限，以确保只有授权实体可以与模型互动。

预期结果：

验证 AI 模型访问受到控制，只有授权实体拥有适当的权限。

5. 安全模型修补和更新测试

测试要求：

实施并测试已部署 AI 模型的安全修补程序和更新程序。

测试方法：

测试修补过程以验证其安全性和完整性。、评估模型更新对系统安全的影响。

预期结果：

确保修补和更新 AI 模型的过程安全进行，不会损害系统安全。

5.9.5 决策透明度

以下是关于“决策透明度”的测试规范，以确保全面评估。

1. 决策透明度测试

测试要求：

提供并评估那些能够洞察 AI 决策过程的机制，特别是在欧盟 AI 法案定义的高风险应用中，以维持问责制和信任。模型卡片、数据卡片或 AI 应用卡片对于透明度至关重要。

测试方法：

- (1) 评估向授权用户提供决策洞察力的机制和数据的可用性及可访问性。
- (2) 确保存在适当的模型卡片、数据卡片和 AI 应用卡片，并且数据是实时最新的。
- (3) 测试 AI 决策的解释机制的有效性，包括提供决策理由。
- (4) 评估关于 AI 决策所提供信息的可理解性和透明度。
- (5) 进行决策场景模拟，以验证决策解释的准确性和一致性。

预期结果：

- (1) 验证提供洞察 AI 决策的机制是可用的且可访问的。
- (2) 确认所提供的决策解释在维护问责制和信任方面是有效的。
- (3) 保证决策理由是清晰、易懂且透明的。

2. 决策模型审计与验证测试

测试要求：

实施并测试 AI 决策模型的审计和验证过程，以确保其准确性和公平性。确保模型、数据或应用卡片反映相同的内容。

测试方法：

- (1) 评估针对决策模型的审计机制，以检测偏见和不公平性。
- (2) 测试验证程序，确保决策模型符合道德准则和合规标准，同时确保没有未检测到的、影响应用结果的偏见。
- (3) 进行有偏见的决策场景模拟，以评估审计和验证的有效性。

预期结果：

- (1) 确认审计和验证过程能够识别和纠正决策模型中的偏见和不公平性。
- (2) 验证决策模型符合道德准则和标准。

3. 用户反馈整合与测试

测试要求：

纳入用户反馈机制，并评估其在提高决策透明度和公平性方面的有效性。

测试方法：

- (1) 评估用户反馈收集机制。
- (2) 测试将用户反馈融入决策过程的效果。
- (3) 评估用户反馈对决策透明度和公平性的影响。

预期结果：

确保用户反馈机制在提升决策透明度和公平性方面是有效的。

4. 伦理决策影响评估测试

测试要求：

在高风险应用中对评估 AI 决策伦理影响的机制实施和测试。

测试方法：

- (1) 评估伦理影响评估机制的有效性。
- (2) 从伦理角度评估决策结果。

预期结果：

验证伦理影响评估机制能够有效地评估 AI 决策的伦理影响。

5. 问责机制测试

测试要求：

实施并测试确保 AI 决策问责机制，包括跟踪和报告。

测试方法：

- (1) 评估现有的问责机制，包括跟踪和报告流程。
- (2) 测试这些机制在使责任方对 AI 决策负责方面的有效性。

预期结果：

确认问责机制能够有效地跟踪和报告 AI 决策，确保责任方被问责。