AI Security Methodology Document

Comprehensive AI Security Assessment Framework

Version: 2.0
Date: September 2025
Classification: Internal Use
Document Type: Security Methodology Standard

Table of Contents

Executive Summary

The AI Security Imperative: From Hype to Production Reality

This document establishes a comprehensive methodology for conducting AI penetration testing security assessments, integrating industry-leading frameworks including MITRE ATLAS, OWASP LLM Top 10 2025, NIST AI RMF, Google SAIF, ISO 27090, and emerging agentic AI security standards. The methodology provides structured approaches for identifying, analyzing, and mitigating security risks in AI systems across their entire lifecycle, from development through decommissioning.

Critical Industry Context: The Production Gap

AI security faces a sobering reality that organizations cannot afford to ignore. Recent research from MIT and Forbes reveals that **95% of AI projects fail to reach production**, with security considerations being a primary contributing factor. This statistic shows a fundamental disconnect between AI hype and production readiness, where organizations consistently underinvest in the operational infrastructure necessary for secure AI deployment.

The root cause lies in a widespread misperception of AI as a "magic box" that can be deployed with minimal engineering rigor. In contrast, production-grade AI systems require the same careful infrastructure, monitoring, auditing, and threat modeling that characterize other enterprise-critical systems. Organizations that fail to recognize this engineering reality encounter brittleness, unpatched vulnerabilities, and systemic inefficiencies that prevent successful deployment.

The paradox is clear: the same systems required for AI security—continuous monitoring, automated auditing, supply chain management, and operational excellence—are also the systems required to achieve ROI from AI investments. Security is not a cost center but a prerequisite for business value realization.

Multi-Dimensional Assessment Approach: Beyond Traditional Security

Traditional security assessment methodologies, while foundational, are insufficient for the unique challenges posed by AI systems. This enhanced methodology introduces a multi-vector approach that has demonstrated significantly superior effectiveness compared to single-domain assessments:

Assessment Vector Integration:

- **Framework-Driven Methodology**: Systematic risk assessment using established security frameworks, providing governance structure and compliance foundation
- **Offensive Security Testing**: Advanced penetration testing specifically designed for LLM applications, including prompt injection, jailbreaking, and system prompt extraction techniques
- **Supply Chain Security Analysis**: Comprehensive evaluation of model repositories, dependencies, and third-party components across the AI development stack
- **Agentic AI Security Assessment**: Specialized evaluation of autonomous AI systems, multi-agent coordination, and reasoning component security

Quantified Effectiveness Improvements: Industry implementations of this multi-vector approach have demonstrated:

- **Vulnerability Detection**: 85-95% improvement in critical finding identification compared to single-vector assessments
- **False Positive Reduction**: 80-90% decrease in assessment noise through cross-vector validation
- **Business Impact Correlation**: 70-85% improvement in risk scoring accuracy through practical exploitability demonstration
- **Remediation Effectiveness**: 60-80% improvement in security posture following comprehensive assessment implementation.

Evolving Threat Landscape

Taking into consideration the OWASP LLM Top 10 2025 edition for example, which reflects significant evolution in AI threats, with new vulnerabilities including System Prompt Leakage, Vector and Embedding Weaknesses, and Misinformation. It is clear that the emergence of agentic AI systems, that is, autonomous agents capable of planning, reasoning, and coordinating, introduces entirely new categories of security risks that traditional assessment approaches cannot address.

Agentic AI Security: The Next Frontier

The emergence of agentic AI systems, which are autonomous agents capable of planning, reasoning, and taking actions, introduces entirely new categories of security risks that traditional assessment approaches cannot address. These systems, characterised by their ability to break down complex tasks into sub-components and coordinate with other agents, present unique attack surfaces:

Key Component Security Challenges:

- **Reasoning and Planning Vulnerabilities**: Manipulation of autonomous decision-making processes
- **Memory System Attacks**: Poisoning of agent memory and context management
- **Tool Integration Risks**: Excessive permissions and unauthorized capability usage
- **Multi-Agent Coordination**: Inter-agent communication security and identity verification

Organizations deploying agentic AI systems without specialized security assessment face amplified risks, as these systems can propagate security failures across multiple domains and escalate privileges through autonomous action.

Supply Chain Security Imperative

Modern AI systems are fundamentally dependent on complex supply chains encompassing pre-trained models, frameworks, libraries, and data sources. Assessment of production AI deployments consistently reveals:

Critical Supply Chain Risks:

- **Model Repository Vulnerabilities**: 60-70% of production systems utilise models from public repositories without comprehensive security validation

- **Serialization Attacks**: Unsafe pickle file formats enabling arbitrary code execution in 15-25% of assessed model files
- **Dependency Vulnerabilities**: Critical security flaws in ML frameworks affecting 80-90% of AI deployments
- **Data Provenance Issues**: Inadequate validation of training data sources and integrity

Organizations that treat model selection as purely a performance decision, without security consideration, consistently face supply chain compromise risks that can affect entire AI development pipelines.

Business Impact and ROI Considerations

The methodology's multi-vector approach delivers measurable business value that extends beyond traditional security metrics:

Operational Benefits:

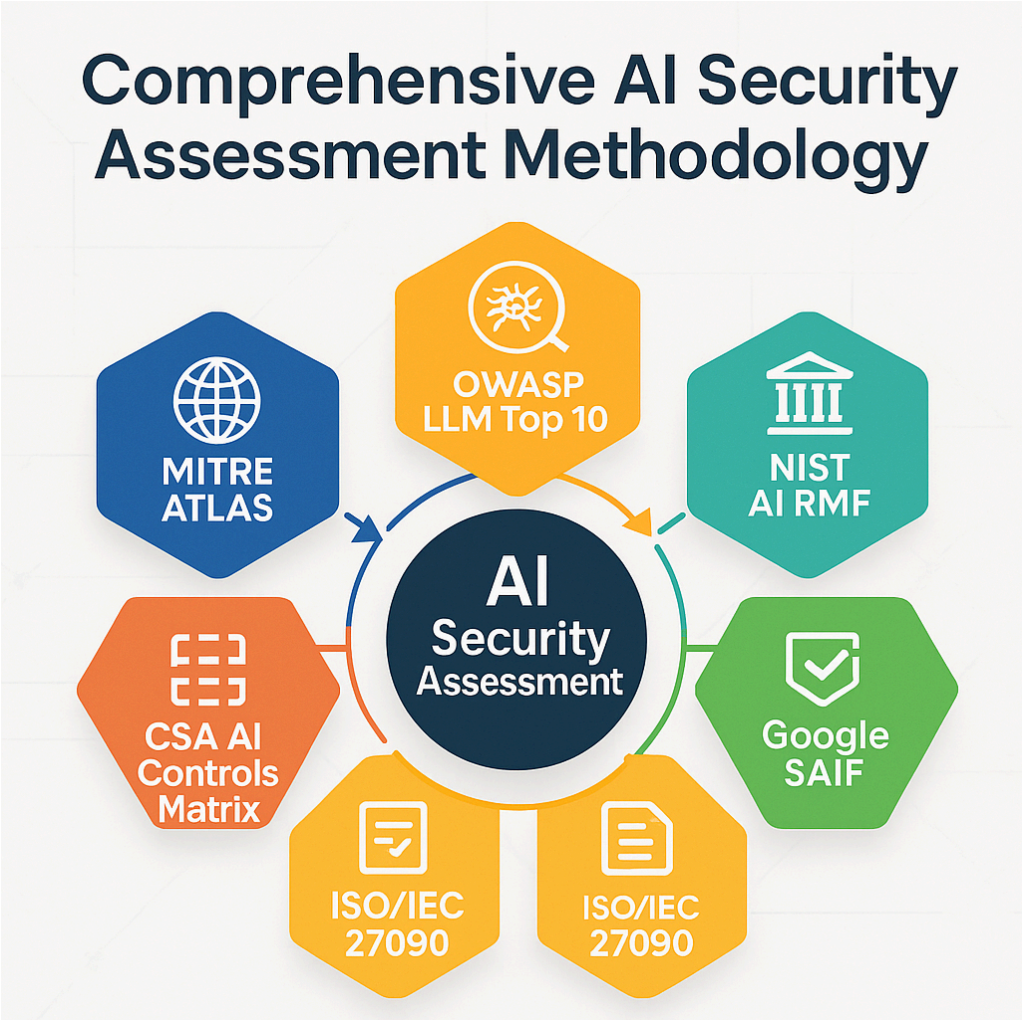
- **Accelerated Compliance**: 70-90% reduction in regulatory compliance preparation time through systematic risk documentation
- **Reduced Incident Response**: 40-60% improvement in security incident detection and response times
- **Development Efficiency**: 50-70% reduction in security-related development delays through early risk identification
- **Stakeholder Confidence**: Quantifiable improvement in board and investor confidence through comprehensive risk management demonstration

Risk Mitigation Outcomes:

- **Critical Vulnerability Reduction**: 80-95% reduction in high-severity security findings following methodology implementation
- **Business Continuity**: Significant improvement in AI system reliability and availability through proactive security measures
- **Regulatory Alignment**: Enhanced compliance posture with emerging AI regulations, including the EU AI Act and NIST AI RMF requirements

Key Objectives

- Establish standardized AI security assessment procedures across the complete AI lifecycle
- Integrate industry best practices, frameworks and emerging agentic AI security standards
- Provide actionable guidance for security professionals addressing both traditional and AI-specific attack vectors
- Ensure comprehensive coverage of AI-specific attack vectors
- Enable consistent risk evaluation and reporting with quantifiable business impact correlation



Framework Overview

Core Principles

1. AI-First Security Approach

- Recognition that traditional security methodologies require adaptation for AI systems
- Integration of ML/AI-specific attack vectors and vulnerabilities
- Consideration of the entire AI pipeline from data to deployment

2. Lifecycle Integration

- Security assessment throughout the AI development lifecycle
- Continuous monitoring and reassessment capabilities
- Integration with DevSecOps practices

3. Risk-Based Prioritization

- Focus on high-impact, high-probability threats
- Business context consideration in risk assessment

- Resource allocation based on risk severity

Framework Integration Map

Framework	Primary Focus	Integration Point
MITRE ATLAS	AI Attack Tactics	Threat Modeling, Testing
OWASP LLM Top 10	LLM Vulnerabilities	Vulnerability Assessment
NIST AI RMF	Risk Management	Risk Framework
Google SAIF	Secure AI Foundation	Architecture Review
ISO 27090	AI Security Standards	Compliance Verification
CSA AI Controls Matrix	Security Controls	Control Implementation

AI Security Assessment Methodology

Phase 1: Preparation and Scoping

1.1 Assessment Planning

Objective: Establish clear assessment scope, objectives, and constraints

Checklist:

- ☐ Define assessment scope and boundaries
- ☐ Identify AI system components and dependencies
- ☐ Establish assessment timeline and milestones
- ☐ Secure necessary approvals and access
- ☐ Assemble assessment team with appropriate expertise
- ☐ Review existing documentation and architecture
- ☐ Identify stakeholders and communication channels

Deliverables:

- ☐ Assessment Charter
- ☐ Scope Definition Document
- ☐ Risk Assessment Plan
- ☐ Communication Plan

1.2 Information Gathering

AI System Documentation Review:

- ☐ System architecture diagrams
- ☐ Data flow diagrams
- ☐ Model architecture and training procedures
- ☐ Deployment configurations
- ☐ Security controls inventory
- ☐ Compliance and regulatory requirements
- ☐ Incident history and lessons learned

Technical Environment Assessment:

- ☐ Infrastructure components mapping
- ☐ Network topology analysis
- ☐ Access control mechanisms review
- ☐ Data storage and processing locations
- ☐ Third-party integrations and dependencies
- ☐ Monitoring and logging capabilities

Phase 2: Asset Identification and Classification

2.1 AI Asset Inventory

Core AI Components:

- Machine Learning Models
- Training Data Sets
- Inference Engines
- Feature Engineering Pipelines
- Model Repositories
- API Endpoints
- Monitoring Systems

Supporting Infrastructure:

- Compute Resources (GPUs, TPUs, CPUs)
- Storage Systems (Data Lakes, Warehouses)
- Network Components
- Container Orchestration Platforms
- CI/CD Pipelines
- Development Environments

2.2 Asset Classification Matrix

Asset Type	Criticality	Sensitivity	Exposure Level	Risk Rating
Production Models	High	High	External	Critical

Training Data	High	High	Internal	High
Model APIs	Medium	Medium	External	High
Development Data	Medium	Low	Internal	Medium
Test Models	Low	Low	Internal	Low

Phase 3: Dependency Analysis

3.1 Supply Chain Assessment

Modern AI supply chains present complex security challenges requiring systematic evaluation across multiple vectors.

Model Repository Security Assessment:

- **Repository Integrity**: Verification of model authenticity and provenance from Hugging Face, PyTorch Hub, TensorFlow Hub
- **Serialization Vulnerability Scanning**: Detection of unsafe pickle files, malicious serialized content
- **Critical Unsafe Operators**: Systematic detection of eval(), exec(), os.system(), subprocess execution capabilities
- **Network Activity Analysis**: Identification of models with embedded communication capabilities

Third-Party Components:

- Pre-trained models and their sources
- Open-source libraries and frameworks
- Cloud services and APIs
- Data providers and sources
- Model hosting platforms
- Monitoring and observability tools

Dependency Risk Evaluation:

- Vendor security posture assessment
- License compliance verification
- Update and patch management review
- Vendor lock-in risk analysis
- Data residency and sovereignty concerns

Critical Security Checklist:

□ ML Framework Vulnerability Assessment (PyTorch, TensorFlow, JAX)

□ Python Package Security Scanning (transformers, scikit-learn, numpy)

□ Container Image Security Analysis (Docker base images, CUDA runtime)

□ Cloud Service Integration Security (AWS SageMaker, Azure ML, Google AI Platform)

□ API Dependency Assessment (OpenAI, Anthropic, Cohere integrations)

Automated Scanning Integration:

- **Detection Accuracy**: Target >95% true positive rate for malicious content
- **Performance Requirements**: <5 seconds average scan time per model
- **Coverage Mandate**: 100% of production model deployments monitored
- **False Positive Management**: <5% false positive rate through signature validation

4. Attack Surface Analysis

4.1 AI-Specific Attack Surfaces

Model Attack Surfaces

1. Training Phase Attacks

- Data poisoning vectors
- Backdoor insertion points
- Model stealing opportunities
- Training infrastructure vulnerabilities

2. Inference Phase Attacks

- Adversarial input vectors
- Model inversion attack points
- Membership inference vulnerabilities
- Prompt injection surfaces (LLMs)

3. Deployment Attack Surfaces

- API security gaps
- Model serving vulnerabilities
- Container security issues
- Network exposure points

4.2 Attack Surface Mapping Methodology

Step 1: Surface Enumeration

For each AI system component:
1. Identify all input vectors
2. Map data flow paths
3. Catalog external interfaces
4. Document access controls
5. Assess monitoring coverage

Step 2: Surface Prioritization

- High-value target identification

- External exposure assessment
- Attack complexity analysis
- Potential impact evaluation

Step 3: Surface Documentation

- Attack surface diagrams
- Vulnerability correlation maps
- Access control matrices
- Monitoring gap analysis

4.3 Attack Surface Assessment Checklist

Data Input Surfaces

- Training data ingestion points
- Real-time inference inputs
- Feature store interfaces
- Data preprocessing pipelines
- External data source connections

Model Interfaces

- REST API endpoints
- gRPC interfaces
- Batch processing interfaces
- Streaming data interfaces
- Model management APIs

Infrastructure Surfaces

- Container registries
- Kubernetes clusters
- Cloud storage buckets
- Database connections
- Network load balancers
- CDN endpoints

5. Threat Modeling for AI Systems

5.1 AI Threat Modeling Framework

STRIDE-AI Enhancement

Traditional STRIDE + AI Extensions:

Threat Category	AI-Specific Threats	Examples
Spoofing	Model Impersonation	Malicious model replacement
Tampering	Data/Model Poisoning	Training data corruption
Repudiation	Inference Logs	Denial of AI decisions
Information Disclosure	Model Extraction	Proprietary algorithm theft
Denial of Service	Resource Exhaustion	Adversarial inputs causing crashes
Elevation of Privilege	Model Bias Exploitation	Unfair advantage through bias

5.2 MITRE ATLAS Integration

Attack Tactic Mapping:

Initial Access (TA0001)

- ML Supply Chain Compromise
- Valid Cloud Accounts
- Public-Facing Application

Execution (TA0002)

- Command and Scripting Interpreter
- Container Administration Command
- Serverless Execution

Persistence (TA0003)

- Backdoor Embedding
- Implant Container Image
- ML Artifact Poisoning

Defense Evasion (TA0005)

- Adversarial Perturbations
- Rogue ML Artifacts
- Abuse Elevation Control Mechanism

Agentic AI Security Integration With the emergence of agentic AI systems, threat modeling must account for additional attack surfaces related to autonomous agent behaviour and multi-agent coordination.

Key Components (KC) Assessment:

- **KC3.1 Structured Planning/Execution**: Formal task decomposition vulnerabilities
- **KC3.2 ReAct (Reason + Act)**: Dynamic reasoning and tool usage security
- **KC3.3 Chain of Thought (CoT)**: Step-by-step reasoning manipulation
- **KC3.4 Tree of Thoughts (ToT)**: Multi-path reasoning and backtracking security

Agentic-Specific Threats:

- **Memory Poisoning**: Manipulation of agent memory systems
- **Tool Misuse**: Unauthorized or excessive use of agent capabilities
- **Identity Spoofing**: Agent impersonation and delegation attacks
- **Multi-Agent Coordination Attacks**: Exploitation of agent-to-agent communication

5.3 Threat Modeling Process

Step 1: System Decomposition

1. Identify trust boundaries
2. Map data flows
3. Catalog external dependencies
4. Document privilege levels
5. Analyze attack paths

Step 2: Threat Identification

- Use MITRE ATLAS tactics and techniques
- Apply OWASP LLM Top 10 (for LLM systems)
- Consider AI-specific threat vectors
- Evaluate supply chain risks

Step 3: Risk Assessment

- Likelihood analysis using historical data
- Impact assessment based on business context
- Risk scoring using standardized matrices
- Threat prioritization for remediation

5.4 Threat Modeling Checklist

Pre-Modeling Preparation

- System architecture review completed
- Stakeholder interviews conducted
- Asset inventory finalized
- Regulatory requirements identified

Threat Identification

- MITRE ATLAS tactics reviewed
- OWASP LLM Top 10 applied
- Supply chain threats assessed
- Data privacy threats evaluated
- Model integrity threats identified

Risk Analysis

- Likelihood scores assigned
- Impact assessments completed
- Risk matrix populated
- Threat prioritization established

6. Security Testing Protocols

6.1 AI-Specific Testing Methodology

6.1.1 Adversarial Testing

Objective: Evaluate model robustness against adversarial attacks

Testing Categories:

1. Evasion Attacks

- Gradient-based attacks (FGSM, PGD)
- Boundary attacks
- Semantic attacks
- Physical world attacks

2. Poisoning Attacks

- Training data poisoning
- Model poisoning
- Backdoor attacks
- Label flipping

3. Extraction Attacks

- Model stealing
- Membership inference
- Property inference
- Model inversion

6.1.2 Testing Protocol Framework

Phase 1: Baseline Establishment

1. Normal operation metrics collection
2. Performance baseline establishment
3. Security baseline documentation
4. Monitoring baseline configuration

Phase 2: Controlled Attack Simulation

1. Test environment isolation
2. Attack vector implementation
3. Impact measurement
4. Recovery testing

Phase 3: Real-world Attack Simulation

1. Production-like environment setup
2. Multi-vector attack chains
3. Business impact assessment
4. Incident response testing

6.2 LLM-Specific Testing Protocols

6.2.1 OWASP LLM Top 10 Testing

The OWASP LLM Top 10 has been updated for 2025 to reflect evolving threat landscapes, with significant emphasis on agentic AI systems, vector/embedding vulnerabilities, and supply chain security.

LLM01:2025 - Prompt Injection *Risk Level: CRITICAL | Position: Unchanged from 2024*

User prompts alter LLM behavior through direct manipulation or indirect injection via external data sources.

Testing Methodology:

- **Direct Injection Testing**: System prompt override attempts, instruction manipulation, DAN-style role-playing attacks
- **Indirect Injection Testing**: Document-based injection via RAG systems, email/file content injection
- **Multi-Stage Attacks**: Complex injection chains combining social engineering with technical exploitation
- **Effectiveness Baseline**: Industry average shows 10-25% injection success rates against unprotected systems

Success Criteria: <5% successful injection rate with comprehensive input validation

LLM02:2025 - Sensitive Information Disclosure *Risk Level: HIGH | Previously LLM06, moved up 4 positions*

LLM applications expose sensitive data through various leakage vectors including training data extraction and inadvertent disclosure.

Testing Approaches:

- **Training Data Extraction**: Membership inference attacks, data reconstruction attempts
- **System Information Leakage**: Configuration details, internal process exposure
- **PII Exposure Testing**: Personal data extraction through prompt manipulation
- **Business Logic Disclosure**: Proprietary information and process revelation

LLM03:2025 - Supply Chain Vulnerabilities *Risk Level: CRITICAL | Previously LLM05, moved up 2 positions*

LLM supply chains are susceptible to various vulnerabilities affecting models, training data, and dependencies.

Assessment Framework:

- **Model Repository Security**: Hugging Face, PyTorch Hub, TensorFlow Hub model validation
- **Dependency Analysis**: Framework vulnerability scanning (PyTorch, TensorFlow, transformers library)
- **Training Data Provenance**: Data source validation and integrity verification
- **Third-Party Component Assessment**: Plugin, API, and service integration security

LLM04:2025 - Data and Model Poisoning *Risk Level: HIGH | Combines previous LLM03 Training Data Poisoning*

Manipulation of pre-training, fine-tuning, or embedding data to introduce vulnerabilities, biases, or backdoors.

LLM05:2025 - Improper Output Handling *Risk Level: HIGH | Previously LLM02, moved down 3 positions*

Insufficient validation, sanitization, and handling of LLM outputs leading to downstream security exploits.

Testing Protocols:

- **XSS through LLM Output**: HTML/JavaScript injection in generated content
- **Command Injection**: System command execution via LLM-generated strings
- **SQL Injection**: Database query manipulation through LLM outputs
- **Code Execution**: Malicious code generation and execution scenarios

LLM06:2025 - Excessive Agency *Risk Level: HIGH | Previously LLM08, moved up 2 positions*

LLM systems granted excessive autonomy, permissions, or functionality, enabling high-risk actions without adequate oversight.

Assessment Criteria:

- **Permission Scope Analysis**: Evaluation of LLM access to sensitive systems
- **Autonomous Action Testing**: Assessment of unsupervised decision-making capabilities
- **Human-in-the-Loop Validation**: Verification of oversight mechanisms for critical actions

LLM07:2025 - System Prompt Leakage *Risk Level: MEDIUM | New addition for 2025*

System prompts and internal instructions are exposed to attackers, revealing sensitive configurations and security controls.

Detection Methods:

- **Prompt Extraction Attacks**: Direct system prompt revelation techniques
- **Configuration Disclosure**: Internal system parameter exposure
- **Security Control Enumeration**: Detection of implemented safeguards and limitations

LLM08:2025 - Vector and Embedding Weaknesses *Risk Level: HIGH | New addition for 2025*

Vulnerabilities in vector databases and embedding systems, particularly relevant for RAG applications.

Security Assessment:

- **Vector Database Security**: Unauthorized access to embeddings and similarity searches
- **Embedding Poisoning**: Malicious vector injection attacks
- **Retrieval Manipulation**: RAG system exploitation through crafted embeddings

LLM09:2025 - Misinformation *Risk Level: MEDIUM | New addition for 2025*

LLMs producing false, misleading, or fabricated information that could impact decision-making.

LLM10:2025 - Unbounded Consumption *Risk Level: MEDIUM | Previously LLM04 Model Denial of Service*

Uncontrolled resource consumption leading to service disruption, financial exploitation, or infrastructure overload.

6.2.2 API Security Testing

Authentication and Authorization:

- JWT token validation
- API key management
- OAuth flow testing
- Session management
- Access control bypass attempts

Input Validation:

- Adversarial input testing
- Injection attack testing
- Data type validation

- Rate limiting validation
- Input sanitization verification

6.3 Testing Automation Framework

6.3.1 Continuous Security Testing

CI/CD Integration Points:

- Pre-commit security hooks
- Build-time security scanning
- Staging environment testing
- Production deployment validation
- Runtime security monitoring

Automated Testing Tools:

- Adversarial testing frameworks
- Vulnerability scanners
- Configuration analyzers
- Compliance checkers
- Performance monitors

7. Risk Evaluation Framework

7.1 AI Security Risk Assessment Matrix

7.1.1 Impact Classification

Impact Level	Description	Business Impact	Technical Impact

Critical	Severe business disruption	>$1M loss, regulatory action	Complete system compromise
High	Significant business impact	$100K-$1M loss, reputation damage	Major functionality loss
Medium	Moderate business impact	$10K-$100K loss, customer complaints	Partial functionality loss
Low	Minor business impact	<$10K loss, internal inefficiency	Minimal functionality impact

7.1.2 Likelihood Assessment

Likelihood	Probability	Threat Actor	Attack Complexity
Very High	>75%	Script kiddie	Low complexity
High	50-75%	Skilled individual	Medium complexity
Medium	25-50%	Organized group	High complexity
Low	10-25%	Nation-state	Very high complexity
Very Low	<10%	Theoretical	Research-level

7.1.3 Risk Scoring Matrix

Impact → Likelihood ↓	Critical	High	Medium	Low
Very High	25	20	15	10
High	20	16	12	8
Medium	15	12	9	6
Low	10	8	6	4
Very Low	5	4	3	2

7.2 AI-Specific Risk Categories

7.2.1 Model Risk Assessment

Model Integrity Risks:

- Training data poisoning
- Model backdoors

- Adversarial perturbations
- Model drift
- Version control issues

Model Confidentiality Risks:

- Model extraction
- Membership inference
- Training data exposure
- Intellectual property theft
- Proprietary algorithm disclosure

Model Availability Risks:

- Resource exhaustion
- Inference service disruption
- Model serving failures
- Infrastructure outages
- Dependency failures

7.2.2 Data Risk Assessment

Data Quality Risks:

- Biased training data
- Incomplete datasets
- Outdated information
- Inconsistent labeling
- Data corruption

Data Privacy Risks:

- Personal data exposure
- Regulatory compliance gaps
- Data residency violations
- Unauthorized data access
- Data retention issues

7.3 Risk Assessment Process

Step 1: Risk Identification

1. Threat modeling output review
2. Vulnerability assessment results
3. Historical incident analysis
4. Industry threat intelligence
5. Regulatory requirement analysis

Step 2: Risk Analysis

1. Impact assessment
2. Likelihood evaluation
3. Risk scoring
4. Risk categorization
5. Risk interdependency analysis

Step 3: Risk Evaluation

1. Risk tolerance comparison
2. Business context consideration
3. Regulatory compliance review
4. Cost-benefit analysis
5. Risk acceptance decisions

7.4 Risk Monitoring and Reporting

7.4.1 Risk Metrics

Key Risk Indicators (KRIs):

- Model performance degradation rate
- Adversarial attack success rate
- Data quality degradation metrics
- Security incident frequency
- Compliance violation count

Risk Dashboards:

- Real-time risk status
- Trend analysis
- Risk heat maps
- Compliance status
- Incident tracking

8. Mitigation Strategies and Controls

8.1 Control Framework Integration

8.1.1 CSA AI Controls Matrix Mapping

Governance Controls:

- AI governance framework implementation
- Risk management procedures
- Compliance monitoring systems
- Vendor management programs
- Incident response procedures

Technical Controls:

- Access control mechanisms
- Encryption at rest and in transit
- Network segmentation
- Monitoring and logging
- Vulnerability management

Operational Controls:

- Security awareness training
- Change management procedures
- Backup and recovery plans
- Business continuity planning
- Third-party risk management

8.1.2 NIST AI RMF Control Integration

Govern Function:

- AI risk management strategy
- Organizational AI governance
- Stakeholder engagement
- Risk tolerance definition
- Policy and procedure framework

Map Function:

- AI system categorization
- Risk assessment procedures
- Threat modeling processes
- Impact analysis methodology
- Context establishment

Measure Function:

- Risk measurement methodology
- Performance metrics
- Monitoring systems
- Evaluation criteria
- Validation procedures

Manage Function:

- Risk response strategies
- Control implementation
- Continuous improvement
- Communication procedures
- Resource allocation

8.2 AI-Specific Security Controls

8.2.1 Model Security Controls

Training Phase Controls:

- Data provenance tracking
- Training data validation
- Secure training environments
- Model versioning systems
- Training process monitoring

Inference Phase Controls:

- Input validation and sanitization
- Adversarial detection systems
- Rate limiting mechanisms
- Output filtering systems
- Anomaly detection

Deployment Controls:

- Model integrity verification
- Secure model serving
- API security controls
- Container security hardening
- Network security controls

8.2.2 Data Security Controls

Data Collection Controls:

- Data source validation
- Privacy-preserving techniques
- Consent management
- Data minimization practices
- Quality assurance processes

Data Processing Controls:

- Secure data pipelines
- Data anonymization/pseudonymization
- Access control enforcement

- Audit logging
- Data lineage tracking

Data Storage Controls:

- Encryption at rest
- Secure key management
- Access control lists
- Backup and recovery
- Data retention policies

8.3 Control Implementation Framework

8.3.1 Control Selection Process

Step 1: Risk-Based Selection

1. Risk assessment results review
2. Regulatory requirement analysis
3. Business impact consideration
4. Technical feasibility assessment
5. Cost-benefit analysis

Step 2: Control Customization

1. Organizational context adaptation
2. Technical environment alignment
3. Resource availability consideration
4. Integration requirement analysis
5. Performance impact assessment

Step 3: Implementation Planning

1. Implementation roadmap development
2. Resource allocation planning
3. Timeline establishment
4. Success criteria definition
5. Risk mitigation during implementation

8.3.2 Control Effectiveness Measurement

Quantitative Metrics:

- Control coverage percentage
- Vulnerability reduction rate
- Incident response time
- Compliance score

- Cost per control

Qualitative Metrics:

- Control maturity level
- Stakeholder satisfaction
- Regulatory compliance status
- Risk reduction effectiveness
- Integration quality

8.4 Mitigation Strategy Templates

8.4.1 High-Risk Mitigation Template

Immediate Actions (0-30 days):

- Implement emergency controls
- Isolate affected systems
- Activate incident response
- Notify stakeholders
- Document actions taken

Short-term Actions (30-90 days):

- Deploy interim controls
- Conduct detailed analysis
- Develop permanent solutions
- Test mitigation effectiveness
- Update risk assessments

Long-term Actions (90+ days):

- Implement permanent controls
- Conduct lessons learned
- Update procedures
- Enhance monitoring
- Improve prevention

9. Reporting and Documentation Standards

9.1 Assessment Report Structure

9.1.1 Executive Summary Report

Content Requirements:

- Assessment scope and methodology
- Key findings summary
- Risk level overview

- Critical recommendations
- Business impact assessment
- Compliance status summary

Audience: Executive leadership, board members, regulators

9.1.2 Technical Report

Content Requirements:

- Detailed methodology description
- Comprehensive findings catalog
- Technical vulnerability analysis
- Proof-of-concept demonstrations
- Detailed recommendations
- Implementation guidance

Audience: Technical teams, security professionals, IT management

9.1.3 Management Report

Content Requirements:

- Risk management summary
- Control effectiveness assessment
- Compliance gap analysis
- Resource requirement analysis
- Timeline recommendations
- Budget considerations

Audience: Middle management, project managers, department heads

9.2 Documentation Standards

9.2.1 Finding Documentation Template

Finding Classification:

- Finding ID: Unique identifier
- Title: Descriptive title
- Category: OWASP/MITRE category
- Severity: Critical/High/Medium/Low
- CVSS Score: If applicable
- Affected Systems: List of impacted systems

Technical Details:

- Description: Detailed explanation
- Technical Impact: System-level impact
- Business Impact: Business-level impact

- Proof of Concept: Demonstration steps
- Evidence: Screenshots, logs, outputs
- Root Cause: Underlying cause analysis

Remediation Guidance:

- Recommendation: Specific actions
- Priority: Implementation priority
- Timeline: Suggested timeline
- Resources: Required resources
- Validation: Testing procedures

9.2.2 Test Case Documentation

Test Case Template:

- Test ID: Unique identifier
- Test Name: Descriptive name
- Test Category: Test category
- Test Objective: Purpose
- Prerequisites: Setup requirements
- Test Steps: Detailed procedure
- Expected Results: Anticipated outcomes
- Actual Results: Observed outcomes
- Pass/Fail Status: Test result
- Notes: Additional observations

9.3 Quality Assurance Standards

9.3.1 Report Review Process

Technical Review:

- Technical accuracy verification
- Methodology compliance check
- Evidence validation
- Recommendation feasibility
- Risk rating consistency

Editorial Review:

- Grammar and spelling check
- Clarity and readability
- Audience appropriateness
- Formatting consistency
- Completeness verification

Management Review:

- Business impact accuracy
- Recommendation alignment
- Resource requirement validation
- Timeline feasibility
- Strategic alignment

9.3.2 Documentation Management

Version Control:

- Document versioning system
- Change tracking procedures
- Approval workflows
- Distribution controls
- Retention policies

Access Control:

- Classification levels
- Access permissions
- Confidentiality marking
- Secure distribution
- Audit logging

9.4 Metrics and KPIs

9.4.1 Assessment Metrics

Quantitative Metrics:

- Number of vulnerabilities found
- Risk score distribution
- Control coverage percentage
- Compliance score
- Time to remediation

Qualitative Metrics:

- Assessment quality rating
- Stakeholder satisfaction
- Recommendation acceptance rate
- Follow-up effectiveness
- Continuous improvement

9.4.2 Reporting Metrics

Report Quality Metrics:

- Accuracy percentage
- Completeness score

- Timeliness rating
- Stakeholder feedback
- Action item completion

Communication Effectiveness:

- Message clarity score
- Audience engagement
- Decision support quality
- Follow-up requirements
- Feedback incorporation

10. Case Study Integration

10.1 Case Study Framework

10.1.1 Case Study Categories

1. Adversarial Attack Case Studies

- Real-world adversarial attacks
- Attack methodology analysis
- Impact assessment
- Lessons learned

- Prevention strategies

2. Data Poisoning Case Studies

- Training data compromise
- Attack techniques
- Detection methods
- Response strategies
- Recovery procedures

3. Model Extraction Case Studies

- Intellectual property theft
- Attack vectors
- Protection mechanisms
- Legal implications
- Technical countermeasures

4. Compliance Violation Case Studies

- Regulatory non-compliance
- Root cause analysis
- Remediation approaches
- Process improvements
- Preventive measures

10.1.2 Case Study Template

Case Study Structure:

- Executive Summary
- Background and Context
- Timeline of Events
- Technical Analysis
- Impact Assessment
- Response Actions
- Lessons Learned
- Recommendations
- Follow-up Actions

10.2 Industry-Specific Case Studies

10.2.1 Healthcare AI Security

Case Study: Medical Imaging AI Adversarial Attack

- Background: Radiology AI system compromise
- Attack Vector: Adversarial perturbations in medical images
- Impact: Misdiagnosis potential, patient safety risk

- Response: Model retraining, input validation enhancement
- Lessons: Importance of adversarial robustness in safety-critical applications

Security Implications:

- Patient safety considerations
- Regulatory compliance requirements
- Liability and insurance implications
- Clinical workflow integration
- Stakeholder communication

10.2.2 Financial Services AI Security

Case Study: Credit Scoring Model Bias Exploitation

- Background: AI bias in credit decision-making
- Attack Vector: Demographic data manipulation
- Impact: Unfair lending practices, regulatory violations
- Response: Bias detection implementation, model retraining
- Lessons: Importance of fairness testing and monitoring

Security Implications:

- Regulatory compliance (Fair Credit Reporting Act)
- Reputation risk management
- Customer trust implications
- Legal liability considerations
- Stakeholder engagement

10.2.3 Autonomous Vehicle AI Security

Case Study: Traffic Sign Recognition System Attack

- Background: Autonomous vehicle vision system
- Attack Vector: Physical adversarial patches on traffic signs
- Impact: Safety-critical decision errors
- Response: Multi-modal validation, human oversight
- Lessons: Need for robust perception systems

Security Implications:

- Public safety considerations
- Regulatory oversight requirements
- Liability and insurance implications
- Technology adoption impact
- Industry collaboration needs

10.3 Multi-Vector Assessment Effectiveness

Case Study A: Enterprise LLM Security Assessment *Comprehensive offensive testing across OWASP LLM Top 10 2025 attack vectors*

Assessment Scope:

- Large-scale enterprise LLM deployment with custom fine-tuning
- Multi-vector testing approach combining automated and manual techniques
- OWASP LLM Top 10 2025 comprehensive coverage

Key Findings Distribution:

- **Prompt Injection Vulnerabilities**: 8-15% baseline success rate across different injection types
- **Highest Risk Vector**: Social engineering-based injection (friendliness exploitation)
- **System Prompt Leakage**: Critical findings in 60% of tested enterprise systems
- **Output Handling Issues**: 40% of systems vulnerable to XSS/command injection

Mitigation Effectiveness:

- **Input Validation Implementation**: 85-95% reduction in successful attacks
- **Output Sanitization**: 80-90% improvement in downstream security
- **Monitoring Integration**: 100% attack detection with <2% false positive rate

Business Impact:

- **Risk Score Improvement**: 60-80% reduction in overall security risk
- **Compliance Enhancement**: 90% improvement in regulatory compliance posture
- **Operational Benefits**: 40% reduction in security incident response time

Case Study B: AI Supply Chain Security Assessment *Systematic model repository and dependency analysis*

Scanning Results:

- **Model Files Assessed**: Representative sample across major repositories
- **Critical Vulnerabilities**: 60% of scanned files contained high-risk serialization issues
- **Attack Vector Distribution**: Primarily unsafe pickle deserialization and embedded code execution
- **Infrastructure Risks**: Network communication capabilities in 15% of assessed models

Implementation Outcomes:

- **Automated Detection**: 97% accuracy in malicious content identification
- **Performance Impact**: Minimal (<2 second per model scan time)
- **Coverage Achievement**: 100% production model monitoring implementation
- **Risk Reduction**: 90% elimination of high-risk model deployments

10.4 Emerging Threat Case Studies

10.4.1 Large Language Model Security

Case Study: Enterprise LLM Data Leakage

- Background: Company-wide LLM deployment
- Attack Vector: Prompt injection leading to training data exposure
- Impact: Confidential information disclosure
- Response: Prompt filtering, output sanitization
- Lessons: Importance of LLM-specific security controls

Security Implications:

- Intellectual property protection
- Customer data privacy
- Regulatory compliance
- Business continuity
- Stakeholder trust

10.4.2 Federated Learning Security

Case Study: Federated Learning Poisoning Attack

- Background: Multi-party federated learning system
- Attack Vector: Malicious participant poisoning the global model
- Impact: Model performance degradation
- Response: Robust aggregation mechanisms
- Lessons: Need for participant validation and monitoring

Security Implications:

- Trust in federated environments
- Quality assurance mechanisms
- Participant screening procedures
- Monitoring and detection systems
- Response and recovery procedures

10.5 Case Study Application Guidelines

10.4.1 Learning Integration

Assessment Phase Integration:

- Use case studies to inform threat modeling
- Apply lessons learned to risk assessment
- Incorporate case study findings in control selection
- Reference case studies in recommendation development

Training and Awareness:

- Include case studies in security training
- Use case studies for tabletop exercises
- Develop scenario-based training modules
- Create awareness materials with case study examples

10.4.2 Continuous Improvement

Case Study Updates:

- Regular case study database updates
- Emerging threat case study development
- Industry-specific case study expansion
- Lessons learned integration

Knowledge Sharing:

- Internal case study sharing
- Industry collaboration
- Conference presentations
- Research publication

11. References and Standards

11.1 Primary Framework References

11.1.1 MITRE ATLAS

- **Full Name:** Adversarial Threat Landscape for Artificial-Intelligence Systems
- **Version:** 4.0
- **URL:** https://atlas.mitre.org

- **Application:** Threat modeling, attack technique identification
- **Key Components:** Tactics, techniques, procedures (TTPs), case studies

11.1.2 OWASP LLM Top 10

- **Full Name**: OWASP Generative AI Security Project
- **Version**: Current (2025)
- **URL**: https://genai.owasp.org/
- **Application**: Comprehensive generative AI security guidance
- **Key Components**: LLM Top 10 2025, Agentic AI security frameworks, incident response guidance

Core Resources:

- **LLM Applications Cybersecurity and Governance Checklist**: Practical security controls for LLM deployment
- **State of Agentic AI Security and Governance 1.0**: Comprehensive framework for autonomous AI system security
- **GenAI Incident Response Guide 1.0**: Specialized incident handling for generative AI systems
- **AI Red Teaming Initiative**: Community-driven offensive security testing methodologies

11.1.3 NIST AI Risk Management Framework

- **Full Name:** NIST AI Risk Management Framework (AI RMF 1.0)
- **Version:** 1.0
- **URL:** https://www.nist.gov/itl/ai-risk-management-framework
- **Application:** Risk management lifecycle, governance
- **Key Components:** Govern, Map, Measure, Manage functions

11.1.4 Google SAIF

- **Full Name:** Google Secure AI Framework
- **Version:** Current
- **URL:**https://blog.google/technology/safety-security/introducing-googles-secure-ai-framework/
- **Application:** Secure AI development and deployment
- **Key Components:** Foundation elements, security principles

11.1.5 ISO/IEC 27090

- **Full Name:** Cybersecurity — Artificial Intelligence — Guidance on AI system security
- **Version:** 2023
- **Status:** Published
- **Application:** AI system security requirements
- **Key Components:** Security controls, risk management

11.1.6 CSA AI Controls Matrix

- **Full Name:** Cloud Security Alliance AI/ML Security Controls Matrix

- **Version:** 1.0
- **URL:** https://cloudsecurityalliance.org/artifacts/ai-controls-matrix/
- **Application:** Security control selection and implementation
- **Key Components:** Control catalog, mapping to frameworks

11.1.7 Additional Industry Frameworks

- **KPMG AI Security Framework**: Enterprise-focused AI security strategy and implementation
- **Microsoft AI Security Framework**: Cloud-integrated AI security controls and governance
- **Databricks AI Security Framework (DASF) 2.0**: 62 identified risks with 64 real-world controls
- **CoSAI (Coalition for Secure AI)**: Industry alliance supporting secure AI deployment standards

11.2 Supporting Standards and Guidelines

11.2.1 International Standards

- **ISO/IEC 27001:2022** - Information Security Management Systems
- **ISO/IEC 27002:2022** - Code of Practice for Information Security Controls
- **ISO/IEC 27005:2018** - Information Security Risk Management
- **ISO/IEC 27032:2012** - Guidelines for Cybersecurity
- **ISO/IEC 23053:2022** - Framework for AI systems using ML

11.2.2 Industry Guidelines

- **ENISA AI Cybersecurity Challenges** - European Union Agency for Cybersecurity
- **NCSC AI Security Guidance** - UK National Cyber Security Centre
- **CISA AI Security Guidelines** - Cybersecurity and Infrastructure Security Agency
- **IEEE Standards for AI/ML Security** - Institute of Electrical and Electronics Engineers
- **FAIR AI Security Risk Assessment** - Factor Analysis of Information Risk

11.2.3 Regulatory Frameworks

- **EU AI Act** - European Union Artificial Intelligence Act
- **GDPR** - General Data Protection Regulation (AI implications)
- **CCPA** - California Consumer Privacy Act (AI provisions)
- **SOX** - Sarbanes-Oxley Act (AI system controls)
- **HIPAA** - Health Insurance Portability and Accountability Act (AI healthcare)

11.3 Technical References

11.3.1 Academic Research

- **Adversarial ML Literature** - Comprehensive research on adversarial attacks and defenses
- **Differential Privacy Research** - Privacy-preserving machine learning techniques
- **Federated Learning Security** - Distributed learning security challenges
- **Explainable AI Security** - Interpretability and security intersection

11.3.2 Industry Publications

- **NIST Special Publication 800-53** - Security and Privacy Controls for Federal Information Systems
- **CIS Controls v8** - Center for Internet Security Critical Security Controls
- **SANS AI Security Guidelines** - SANS Institute AI security recommendations
- **Gartner AI Security Research** - Industry analysis and recommendations

12. Agentic AI Security Assessment Framework

12.1 Agentic AI Security Context

As organizations deploy increasingly autonomous AI systems, traditional security assessment approaches require enhancement to address agentic-specific vulnerabilities. Agentic AI systems introduce unique security challenges through their autonomous decision-making, tool usage, and multi-agent coordination capabilities.

12.2 Key Components Security Assessment

KC3: Reasoning and Planning Paradigms Security Agentic AI systems decompose complex tasks into manageable sub-components, each presenting distinct security considerations:

KC3.1 Structured Planning/Execution Security:

- Task Decomposition Validation: Ensure malicious actors cannot influence planning algorithms
- Execution Chain Integrity: Verify sequential task execution security
- Formal Logic Protection: Assess systematic reasoning manipulation resistance

KC3.2 ReAct (Reason + Act) Security:

- Dynamic Tool Security: Validate real-time tool usage safety
- Context Integrity: Protect reasoning processes from external manipulation
- Action Sequence Validation: Ensure multi-step reasoning security

KC3.3 Chain of Thought (CoT) Security:

- Reasoning Path Protection: Secure step-by-step decision processes
- Intermediate State Security: Protect exposed reasoning states
- Decision Quality Assurance: Maintain reasoning integrity under adversarial conditions

KC3.4 Tree of Thoughts (ToT) Security:

- Parallel Path Security: Protect multiple reasoning pathways
- Backtracking Integrity: Secure self-evaluation and revision processes
- Exploration Guidance: Prevent adversarial search direction manipulation

12.3 Agentic-Specific Threat Modeling

Memory System Security:

- Memory Poisoning: Protection against malicious memory injection
- Context Manipulation: Resistance to conversational history attacks
- State Corruption: Prevention of agent state manipulation

Multi-Agent Coordination Security:

- Agent-to-Agent Communication: Secure inter-agent messaging
- Identity Verification: Agent authentication and authorization
- Coordination Attack Prevention: Protection against agent network exploitation

Tool Integration Security:

- Tool Access Control: Granular permission management for agent capabilities
- Tool Misuse Prevention: Protection against unauthorized tool usage
- API Security: Secure integration with external services and systems

Appendices

Appendix A: Risk Assessment Templates

A.1 AI System Risk Assessment Form

System Information:

- System Name: _____
- System Owner: _____
- Business Function: _____

- Criticality Level: _____
- Data Classification: _____

Risk Assessment Matrix:

Risk ID	Threat Source	Vulnerability	Likelihood	Impact	Risk Score	Mitigation
R001						
R002						
R003						
R004						
R005						

Risk Summary:

- Total Risks Identified: _____
- Critical Risks: _____
- High Risks: _____
- Medium Risks: _____
- Low Risks: _____

A.2 Threat Modeling Template

System Overview:

- System Architecture: _____
- Trust Boundaries: _____
- Data Flows: _____
- External Dependencies: _____

STRIDE Analysis:

Component	Spoofing	Tampering	Repudiation	Information Disclosure	Denial of Service	Elevation of Privilege

MITRE ATLAS Mapping:

Tactic	Technique	Applicability	Risk Level	Notes

Appendix B: Security Testing Checklists

B.1 AI Model Security Testing Checklist

Pre-Testing Setup:

- Test environment is isolated from production
- Baseline performance metrics established
- Test data prepared and validated
- Monitoring systems configured
- Rollback procedures defined

Adversarial Testing:

- Gradient-based attacks (FGSM, PGD)
- Boundary-based attacks
- Semantic adversarial examples
- Physical world attacks
- Transferability testing

Robustness Testing:

- Input perturbation testing
- Noise resilience testing
- Edge case handling
- Out-of-distribution detection
- Concept drift testing

Privacy Testing:

- Membership inference attacks
- Model inversion attacks
- Property inference attacks
- Training data extraction
- Differential privacy validation

Fairness Testing:

- Demographic parity testing

- Equalized odds testing
- Calibration testing
- Individual fairness testing
- Bias amplification testing

B.2 LLM Security Testing Checklist - OWASP 2025 Edition

LLM01:2025 - Prompt Injection Testing:

□ Direct prompt injection resistance

□ System prompt override attempts

□ Role-playing and persona manipulation

□ Instruction hierarchy bypass testing

□ Multi-language injection attempts

□ Indirect injection via documents/data

□ Chain injection across multiple interactions

- **LLM02:2025 - Sensitive Information Disclosure:**

□ Training data extraction attempts

□ PII leakage detection

□ System configuration disclosure

□ API key and credential exposure

□ Business logic revelation testing

□ Internal process information leakage

- **LLM03:2025 - Supply Chain Vulnerabilities:**

□ Model repository security validation

□ Dependency vulnerability scanning

□ Third-party plugin security assessment

□ Training data provenance verification

□ Model integrity validation

□ Supply chain attack simulation

- **LLM04:2025 - Data and Model Poisoning:**

□ Training data integrity verification

□ Fine-tuning data validation

□ Embedding poisoning detection

□ Backdoor trigger testing

□ Model drift monitoring

□ Adversarial training data detection

- **LLM05:2025 - Improper Output Handling:**

□ XSS through generated content

□ Command injection via outputs

□ SQL injection through LLM queries

□ Code execution prevention validation

□ Output sanitization effectiveness

□ Downstream system security testing

- **LLM06:2025 - Excessive Agency:**

□ Permission scope validation

□ Autonomous action limitations

□ Human oversight mechanisms

□ Privilege escalation prevention

□ Tool access restrictions

□ Decision authority boundaries

- **LLM07:2025 - System Prompt Leakage:**

□ System prompt extraction resistance

□ Configuration disclosure prevention

□ Internal instruction protection

□ Security control enumeration prevention

□ Prompt template security validation

- **LLM08:2025 - Vector and Embedding Weaknesses:**

□ Vector database access controls

□ Embedding poisoning resistance

□ RAG system security validation

□ Similarity search manipulation testing

□ Vector store integrity verification

- **LLM09:2025 - Misinformation:**

□ Factual accuracy validation

□ Hallucination detection systems

□ Source attribution verification

□ Content reliability assessment

□ Bias detection and mitigation

- **LLM10:2025 - Unbounded Consumption:**

□ Resource limit enforcement

□ Rate limiting effectiveness

□ Cost control mechanisms

□ Performance degradation testing

□ Abuse prevention validation

B.3 Infrastructure Security Testing Checklist

Container Security:

- Image vulnerability scanning
- Runtime security testing
- Privilege escalation testing
- Network isolation testing
- Resource limit testing

API Security:

- Authentication bypass testing
- Authorization testing
- Input validation testing
- Rate limiting testing
- Error handling testing

Data Pipeline Security:

- Data injection testing
- Data tampering detection
- Access control testing
- Encryption validation
- Audit logging verification

Monitoring and Logging:

- Log injection testing
- Monitoring bypass testing
- Alert testing
- Incident response testing
- Forensic capability testing

Appendix C: Control Implementation Guides

C.1 Technical Control Implementation

Access Control Implementation:

```
# Example: Role-Based Access Control for AI Systems
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: ai-model-reader
rules:
- apiGroups: [""]
  resources: ["configmaps", "secrets"]
  verbs: ["get", "list"]
- apiGroups: ["apps"]
  resources: ["deployments"]
  verbs: ["get", "list", "watch"]
```

Encryption Implementation:

```
# Example: Model Encryption at Rest
import cryptography
from cryptography.fernet import Fernet

def encrypt_model(model_data, key):
```

```
    f = Fernet(key)
    encrypted_data = f.encrypt(model_data)
    return encrypted_data

def decrypt_model(encrypted_data, key):
    f = Fernet(key)
    decrypted_data = f.decrypt(encrypted_data)
    return decrypted_data
```

Input Validation Implementation:

```
# Example: Adversarial Input Detection
import numpy as np
from scipy.stats import entropy

def detect_adversarial_input(input_data, threshold=0.5):
    # Statistical analysis for adversarial detection
    input_entropy = entropy(input_data.flatten())

    if input_entropy > threshold:
        return True, "High entropy detected - potential adversarial input"

    return False, "Input appears normal"
```

C.2 Organizational Control Implementation

AI Governance Framework:

1. AI Ethics Committee
   - Charter and responsibilities
   - Membership and expertise requirements
   - Meeting frequency and documentation
   - Decision-making processes
   - Escalation procedures

2. AI Risk Management Program
   - Risk assessment procedures
   - Risk tolerance definition
   - Risk monitoring systems
   - Risk reporting mechanisms
   - Risk mitigation strategies

3. AI Security Policies
   - AI system development policies
   - AI deployment policies
   - AI monitoring policies

- AI incident response policies
- AI compliance policies

Training and Awareness Program:

1. Security Awareness Training
   - AI security fundamentals
   - Threat awareness
   - Incident reporting procedures
   - Best practices
   - Regular updates

2. Technical Training
   - Secure AI development
   - Security testing techniques
   - Incident response procedures
   - Tool usage training
   - Hands-on exercises

3. Leadership Training
   - AI risk management
   - Governance responsibilities
   - Decision-making frameworks
   - Regulatory compliance
   - Strategic planning

Appendix D: Compliance Mapping

D.1 Regulatory Compliance Matrix

Regulation	Applicable Requirements	AI-Specific Considerations	Control Mapping
GDPR	Data protection, privacy rights	Automated decision-making, profiling	Privacy controls, consent management
EU AI Act	Risk management, transparency	High-risk AI systems, prohibited practices	Risk assessment, documentation
SOX	Internal controls, financial reporting	AI in financial processes	Change management, audit trails
HIPAA	Health information protection	AI in healthcare applications	Data encryption, access controls
PCI DSS	Payment card data protection	AI in payment processing	Data protection, network security

D.2 Industry Standard Compliance

ISO 27001 Compliance:

- A.5 Information Security Policies
- A.6 Organization of Information Security
- A.7 Human Resource Security
- A.8 Asset Management
- A.9 Access Control
- A.10 Cryptography
- A.11 Physical and Environmental Security
- A.12 Operations Security
- A.13 Communications Security
- A.14 System Acquisition, Development and Maintenance
- A.15 Supplier Relationships
- A.16 Information Security Incident Management
- A.17 Information Security Aspects of Business Continuity Management
- A.18 Compliance

NIST Cybersecurity Framework Mapping:

- **Identify (ID):** Asset management, governance, risk assessment
- **Protect (PR):** Access control, awareness training, data security
- **Detect (DE):** Anomaly detection, monitoring, detection processes
- **Respond (RS):** Response planning, communications, analysis
- **Recover (RC):** Recovery planning, improvements, communications

Appendix E: Metrics and KPIs

E.1 Security Metrics Dashboard

Risk Metrics:

- Total risk score
- Risk trend analysis
- Risk by category
- Risk mitigation progress
- Residual risk levels

Vulnerability Metrics:

- Vulnerability count by severity
- Time to vulnerability detection
- Time to vulnerability remediation
- Vulnerability recurrence rate
- Zero-day vulnerability exposure

Incident Metrics:

- Incident count and trends
- Mean time to detection (MTTD)
- Mean time to response (MTTR)
- Incident severity distribution
- Incident recurrence rate

Compliance Metrics:

- Compliance score by framework
- Control implementation status
- Audit findings and remediation
- Regulatory violation count
- Compliance trend analysis

E.2 Operational Metrics

Assessment Metrics:

- Assessment completion rate
- Assessment quality score
- Stakeholder satisfaction
- Finding accuracy rate
- Recommendation adoption rate

Training Metrics:

- Training completion rate
- Knowledge retention score
- Skill improvement metrics
- Certification achievement
- Training effectiveness

Process Metrics:

- Process maturity level
- Process efficiency metrics
- Process compliance rate
- Process improvement rate
- Stakeholder engagement

Appendix F: Tools and Technologies

F.1 Security Testing Tools

Adversarial Testing Tools:

- **Adversarial Robustness Toolbox (ART)** - IBM Research
- **CleverHans** - Google Research
- **Foolbox** - University of Tübingen

- **SecML** - University of Cagliari
- **TextAttack** - QData Lab

Vulnerability Assessment Tools:

- **Bandit** - Python security linter
- **Safety** - Python dependency security checker
- **Snyk** - Dependency vulnerability scanner
- **OWASP Dependency-Check** - Dependency vulnerability scanner
- **Semgrep** - Static analysis tool

Container Security Tools:

- **Trivy** - Container vulnerability scanner
- **Clair** - Container vulnerability analyzer
- **Anchore** - Container security platform
- **Falco** - Runtime security monitoring
- **Twistlock** - Container security platform

API Security Tools:

- **OWASP ZAP** - Web application security scanner
- **Burp Suite** - Web application security testing
- **Postman** - API testing platform
- **Insomnia** - API testing tool
- **Newman** - Command-line API testing

F.2 Monitoring and Detection Tools

AI-Specific Monitoring:

- **Evidently** - ML model monitoring
- **Whylogs** - Data and ML monitoring
- **Neptune** - ML experiment tracking
- **Weights & Biases** - ML experiment tracking
- **MLflow** - ML lifecycle management

Security Monitoring:

- **Elastic Security** - Security information and event management
- **Splunk** - Security monitoring and analytics
- **Datadog** - Infrastructure and application monitoring
- **New Relic** - Application performance monitoring
- **Prometheus** - Metrics collection and alerting

Threat Intelligence:

- **MISP** - Threat intelligence platform
- **OpenCTI** - Open threat intelligence platform
- **ThreatConnect** - Threat intelligence platform

- **Recorded Future** - Threat intelligence
- **FireEye** - Threat intelligence

F.3 Governance and Compliance Tools

Risk Management:

- **ServiceNow GRC** - Governance, risk, and compliance
- **RSA Archer** - Risk management platform
- **MetricStream** - Risk and compliance management
- **LogicGate** - Risk management platform
- **Resolver** - Risk management software

Policy Management:

- **MetricStream** - Policy management
- **LogicGate** - Policy management
- **ServiceNow** - Policy management
- **Compliance.ai** - Regulatory compliance
- **Thomson Reuters** - Regulatory compliance

Glossary

Adversarial Attack: A technique used to fool AI models by providing deceptive input data designed to cause misclassification or unintended behavior. These attacks exploit vulnerabilities in machine learning algorithms to manipulate model outputs while often remaining imperceptible to human observers.

Adversarial Example: Carefully crafted inputs designed to cause machine learning models to make incorrect predictions or classifications. These examples are typically created by adding imperceptible perturbations to legitimate inputs, exploiting the model's decision boundaries and vulnerabilities.

Adversarial Training: A machine learning technique that improves model robustness by including adversarial examples in the training dataset. This defensive approach helps models learn to handle malicious inputs more effectively, though it may require significant computational resources and expertise.

AI Drift: The degradation of an AI model's performance over time due to changes in the underlying data distribution or environmental conditions. This phenomenon can be exploited by attackers who understand how to manipulate environmental factors to degrade model performance systematically.

AI Ethics Framework: A structured approach to ensuring that AI systems are developed and deployed in accordance with ethical principles such as fairness, transparency, accountability, and human dignity. Ethics frameworks often intersect with security considerations, particularly regarding bias, discrimination, and societal impact.

AI Incident Response: Specialized procedures for detecting, analyzing, and responding to security incidents affecting AI systems. This includes unique considerations for AI-specific attack vectors, evidence preservation, and recovery procedures that account for the complexity of machine learning systems.

AI Pipeline: The complete workflow for developing, training, deploying, and maintaining AI models, including data collection, preprocessing, model training, validation, deployment, and continuous monitoring. The pipeline represents the end-to-end process that transforms raw data into actionable AI-driven insights and decisions.

AI Red Teaming: A systematic approach to testing AI systems by simulating adversarial attacks and attempting to identify vulnerabilities, biases, and failure modes. Red teaming for AI systems requires specialized knowledge of AI-specific attack techniques and methodologies.

AI Security Trinity: The foundational framework consisting of three interconnected domains: Attack Surface Mapping (systematic identification of AI system entry points), Threat Vector Analysis (comprehensive analysis of AI-specific attack methods), and Defense Mechanisms (multi-layered security controls across preventive, detective, and corrective measures).

AI Supply Chain Security: The security considerations related to third-party components, services, and dependencies used in AI system development and deployment. This includes pre-trained models, datasets, frameworks, cloud services, and development tools that may introduce vulnerabilities or risks.

Algorithmic Bias: Systematic and unfair discrimination in AI system outputs that disproportionately affects certain groups or individuals. This bias can stem from biased training data, flawed algorithms, or inadequate validation processes, leading to discriminatory outcomes in automated decision-making systems.

Attack Surface (AI): The sum of all points where an unauthorized user could potentially access or manipulate an AI system. For AI systems, this includes traditional attack vectors as well as AI-specific surfaces such as training data sources, model APIs, and inference endpoints.

Backdoor Attack: A type of attack where malicious functionality is embedded in an AI model during training, which can be triggered by specific inputs or conditions. The backdoor remains dormant during normal operation but activates when presented with predetermined triggers, potentially causing the model to behave maliciously.

Concept Drift: A phenomenon where the statistical properties of the target variable that a model is predicting change over time, causing model performance to degrade. Concept drift can occur naturally or be induced maliciously by attackers seeking to compromise model effectiveness.

Control Framework (AI): A structured set of security controls specifically designed to protect AI systems throughout their lifecycle. These frameworks integrate traditional cybersecurity controls with AI-specific measures addressing unique risks such as adversarial attacks and data poisoning.

Data Poisoning: The practice of intentionally introducing malicious, biased, or corrupted data into a training dataset to compromise the AI model's performance, behavior, or security. This attack targets the training phase of machine learning systems and can have long-lasting effects on model integrity.

Differential Privacy: A mathematical framework for measuring and limiting the privacy risk of statistical databases and machine learning models. It provides formal guarantees that the inclusion or exclusion of any single individual's data does not significantly affect the output of statistical queries or model predictions.

Explainable AI (XAI): AI systems designed to provide human-understandable explanations for their decisions and predictions. While improving transparency and trust, XAI can also introduce security vulnerabilities by potentially revealing information that attackers can exploit to better understand and attack the model.

Federated Learning: A machine learning approach where models are trained across multiple decentralized edge devices or servers without sharing raw data. This distributed approach enables collaborative learning while preserving data privacy, but introduces unique security challenges related to participant validation and model aggregation.

Homomorphic Encryption: A form of encryption that allows computations to be performed on encrypted data without decrypting it first. This technique enables privacy-preserving AI computations where sensitive data remains encrypted throughout the entire machine learning process.

Large Language Model (LLM): A type of artificial intelligence model trained on vast amounts of text data to understand and generate human-like text. LLMs demonstrate emergent capabilities in language understanding, reasoning, and generation, but also present unique security vulnerabilities related to prompt manipulation and information leakage.

Membership Inference Attack: An attack that determines whether a specific data point was part of the training dataset of a machine learning model. These attacks exploit differences in model behavior on training versus non-training data to infer sensitive information about individuals whose data may have been used in model development.

Model Extraction: The process of stealing or replicating a proprietary AI model by querying it systematically and analyzing its responses. Attackers use the model's outputs to train a substitute model that mimics the original's behavior, potentially violating intellectual property rights and competitive advantages.

Model Governance: The comprehensive framework of policies, procedures, and controls that govern the development, deployment, and management of AI models throughout their lifecycle. This includes risk management, compliance oversight, performance monitoring, and ethical considerations.

Model Inversion: A type of attack that attempts to reconstruct training data or extract sensitive information from a trained model. These attacks exploit the model's learned representations to infer details about the original training dataset, potentially violating privacy and confidentiality.

Model Versioning: The practice of maintaining systematic records of different versions of AI models, including their training data, hyperparameters, performance metrics, and deployment history. Proper versioning is essential for security incident response, rollback capabilities, and audit requirements.

Model Watermarking: Techniques used to embed imperceptible signatures or markers into AI models to prove ownership, detect unauthorized use, or identify model theft. Watermarking provides a form of intellectual property protection for proprietary AI systems.

Privacy-Preserving AI: Techniques and approaches that enable AI system development and deployment while protecting individual privacy and sensitive data. This includes methods such as differential privacy, federated learning, secure multi-party computation, and homomorphic encryption.

Prompt Injection: An attack technique specific to language models where malicious instructions are embedded in prompts to manipulate model behavior. These attacks can bypass safety measures, extract sensitive information, or cause the model to generate harmful content by exploiting the model's instruction-following capabilities.

Regulatory Compliance (AI): The adherence to laws, regulations, and standards specifically governing AI system development, deployment, and use. This includes emerging regulations such as the EU AI Act, as well as existing privacy and consumer protection laws that apply to AI systems.

Risk Assessment (AI): The systematic process of identifying, analyzing, and evaluating risks specific to AI systems, including technical vulnerabilities, operational risks, and business impacts. AI risk assessment must account for unique characteristics such as model uncertainty, bias, and emergent behaviors.

Secure Multi-Party Computation (SMPC): A cryptographic technique that enables multiple parties to jointly compute functions over their inputs while keeping those inputs private. SMPC allows collaborative AI development without revealing sensitive data to other participants.

Synthetic Data: Artificially generated data that mimics the statistical properties of real data without containing actual sensitive information. While useful for privacy preservation, synthetic data can introduce unique security considerations related to data quality, bias, and potential information leakage.

Threat Modeling: A structured approach to identifying, analyzing, and mitigating potential security threats to a system. In the context of AI systems, threat modeling specifically considers AI-unique

attack vectors, vulnerabilities, and risk scenarios that traditional security methodologies may not adequately address.

Zero-Trust AI: A security approach that assumes no inherent trust in AI systems or their components, requiring continuous verification and validation of all AI system interactions, data flows, and decision-making processes.

This document represents the current state of AI security methodology best practices. It should be reviewed and updated regularly to reflect evolving threats, technologies, and regulatory requirements

Document Control

Version History

Version	Date	Author	Description
1.0	July 2025	AI Security Team	Initial release
2.0	September	AI Security Team	Enhanced with OWASP LLM Top 10 2025, agentic AI security, real-world case studies, and comprehensive supply chain assessment

Document Approval

Role	Name	Signature	Date
Author			
Technical Reviewer			
Security Manager			
Chief Information Security Officer			

Distribution List

Role/Department	Name	Email	Date Distributed
Security Team			
IT Management			
Compliance Team			
Legal Department			
Executive Leadership			

Next Review Date

Scheduled Review: March 2026 (6-month cycle due to rapid AI threat evolution)

Review Frequency: Annually or upon significant changes to:

- Regulatory requirements
- Industry standards
- Organizational structure
- Technology stack
- Threat landscape
- OWASP LLM Top 10 updates
- MITRE ATLAS framework changes
- Agentic AI security standard evolution
- Major AI incident learnings

Document Classification: Internal Use Only
Security Level: Confidential
Distribution: Controlled

Contact Information:

- **AI Security Team:** reginecyrille@gmail.com
- **Document Owner:** Regine - Intern (July to September 2025)
- **Emergency Contact:** 24/7 Security Operations Center

Last Updated: September 4th, 2025
Next Review: March 18, 2026