ClawGuard Shield

AI Agent Security Compliance Report

10

Risk Score (0-10)

Severity: CRITICAL

Organization: CrystalDBA

Report Generated: 2026-04-02 21:26 UTC

Findings: 30 threat(s) detected

Scan Time: 24560ms

Scanner: ClawGuard Shield v0.5.0 (50 patterns, 7 categories)

EU AI Act Compliance Reference Included (Enforcement: 02 August 2026)

1. Executive Summary

The scan detected 30 security threat(s) with an overall risk score of 10/10 (CRITICAL). This indicates critical security concerns that require attention. The scan completed in 24560ms using deterministic pattern matching across 42 attack vectors.

Scan Statistics

Total Findings	**30**
Risk Score	**10 / 10**
Overall Severity	**CRITICAL**
Scan Duration	**24560ms**
Patterns Checked	**42**
Attack Categories	**5**
Detection Method	**Deterministic Pattern Matching**
False Positive Rate	**0%**

Severity Breakdown

2. Detailed Findings

Data Exfiltration (10 finding(s))

Techniques to steal sensitive data through the AI agent by embedding hidden requests, markdown image injections, or encoded payloads that transmit data to external servers.

MEDIUM 70% **Email Address (LLM06)**

Line: 21

Match: jssmith@crystal.cloud

Recommendation: Email address detected. OWASP LLM06: Sensitive Information Disclosure. DSGVO -- personal identifier.

HIGH 85% **Database Connection String**

Line: 14

Match: postgres://user:password@host:port/database.

Recommendation: Database connection string with potential credentials detected.

MEDIUM 65% **German Phone Number (LLM06)**

Line: 102

Match: 036854775807

Recommendation: German phone number detected. OWASP LLM06: Sensitive Information Disclosure.

HIGH 70% **Database Connection String**

Line: 42

Match: postgres://user:password@host:port/dbname

Recommendation: Database connection string with potential credentials detected.

HIGH 65% **Password in Cleartext**

Line: 46

Match: password=xxx)

Recommendation: Cleartext password detected. Never store or transmit passwords in plain text.

HIGH 80% **Password in Cleartext**

Line: 48

Match: password=)([^\s&;

Recommendation: Cleartext password detected. Never store or transmit passwords in plain text.

LOW 50% **IP Address in Sensitive Context (LLM06)**

Line: 54

Match: 1.11.6.11

Recommendation: IP address detected. OWASP LLM06. DSGVO -- IP addresses are personal data under EU law.

MEDIUM 65% **German Phone Number (LLM06)**

Line: 7

Match: 068979676331

Recommendation: German phone number detected. OWASP LLM06: Sensitive Information Disclosure.

LOW 50% **IP Address in Sensitive Context (LLM06)**

Line: 154

Match: 172.17.0.1

Recommendation: IP address detected. OWASP LLM06. DSGVO -- IP addresses are personal data under EU law.

HIGH 85% **Database Connection String**

Line: 650

Match: postgres://user:password@localhost:5432/dbname"

Recommendation: Database connection string with potential credentials detected.

Code Obfuscation (1 finding(s))

Obfuscated code patterns (Base64, hex encoding, string concatenation, dynamic imports) used to hide malicious payloads from static analysis.

HIGH 80% **Suspicious open() in Agent Input**

Line: 394

Match: open(file_path)

Recommendation: File open() call detected in agent input. In an agent context, direct file operations are suspicious and should be reviewed.

Output Injection (8 finding(s))

Attempts to inject malicious content (XSS, SQL injection) into AI agent outputs. OWASP LLM02: Insecure Output Handling.

HIGH 85% **Command Injection in Output (LLM05)**

Line: 262

Match: `n\" + txet_ffid + "n\`

Recommendation: Command injection via backticks or $() subshell. If output is rendered in shell context, this executes. OWASP LLM05.

HIGH 80% **SQL Injection Fragment (LLM02)**

Line: 550

Match: ' or '

Recommendation: SQL injection fragment detected. OWASP LLM02: Insecure Output Handling.

HIGH 80% **SQL Injection Fragment (LLM02)**

Line: 104

Match: ' or '

Recommendation: SQL injection fragment detected. OWASP LLM02: Insecure Output Handling.

HIGH 80% **SQL Injection Fragment (LLM02)**

Line: 192

Match: ' or '

Recommendation: SQL injection fragment detected. OWASP LLM02: Insecure Output Handling.

HIGH 80% **Command Injection in Output (LLM05)**

Line: 237

Match: `Connect more tools`

Recommendation: Command injection via backticks or $() subshell. If output is rendered in shell context, this executes. OWASP LLM05.

HIGH 80% **Command Injection in Output (LLM05)**

Line: 345

Match: `get_object_details`

Recommendation: Command injection via backticks or $() subshell. If output is rendered in shell context, this executes. OWASP LLM05.

HIGH 85% **Command Injection in Output (LLM05)**

Line: 348

Match: ` gnisu emit noitucexe latot no desab seireuq LQS tsewols eht stropeR | `

Recommendation: Command injection via backticks or $() subshell. If output is rendered in shell context, this executes. OWASP LLM05.

HIGH 80% **SQL Injection Fragment (LLM02)**

Line: 600

Match: ; DROP

Recommendation: SQL injection fragment detected. OWASP LLM02: Insecure Output Handling.

Dangerous Command (7 finding(s))

MEDIUM 65% **Package / Dependency Install**

Line: 232

Match: yum install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

MEDIUM 65% **Package / Dependency Install**

Line: 233

Match: brew install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

MEDIUM 65% **Package / Dependency Install**

Line: 242

Match: yum install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

MEDIUM 65% **Package / Dependency Install**

Line: 243

Match: brew install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

MEDIUM 65% **Package / Dependency Install**

Line: 105

Match: pip install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

CRITICAL 99% **Remote Code Execution**

Line: 631

Match: curl -sSL https://astral.sh/uv/install.sh | sh

Recommendation: CRITICAL: Pipe-to-shell pattern detected. This downloads and executes remote code without inspection.

MEDIUM 65% **Package / Dependency Install**

Line: 644

Match: pip install

Recommendation: Software installation command detected. Verify the package source for supply-chain safety.

Shell Injection (1 finding(s))

CRITICAL 95% **Backtick Command Substitution**

Line: 153

Match: `host.docker.internal`

Recommendation: CRITICAL: Backtick command substitution detected with a shell command.

Social Engineering (2 finding(s))

Manipulation tactics exploiting the AI agent's helpfulness bias. These attacks use emotional manipulation, authority claims, or urgency to bypass safety guidelines.

HIGH 80% **Approval Bypass (LLM08)**

Line: 614

Match: auto-run

Recommendation: Attempt to bypass human approval for agent actions. OWASP LLM08: Excessive Agency risk.

HIGH 80% **Approval Bypass (LLM08)**

Line: 615

Match: auto-run

Recommendation: Attempt to bypass human approval for agent actions. OWASP LLM08: Excessive Agency risk.

Supply Chain (1 finding(s))

CRITICAL 99% **Curl Pipe to Shell (LLM03)**

Line: 631

Match: curl -sSL https://astral.sh/uv/install.sh | sh

Recommendation: CRITICAL: Piping remote content directly to shell. Classic supply-chain attack vector. OWASP LLM03.

3. Remediation Priorities

The following remediation steps are ordered by severity. Address CRITICAL and HIGH findings immediately before deploying the AI agent to production.

1. CRITICAL **Backtick Command Substitution**
   CRITICAL: Backtick command substitution detected with a shell command.
2. CRITICAL **Remote Code Execution**
   CRITICAL: Pipe-to-shell pattern detected. This downloads and executes remote code without inspection.
3. CRITICAL **Curl Pipe to Shell (LLM03)**
   CRITICAL: Piping remote content directly to shell. Classic supply-chain attack vector. OWASP LLM03.

4. HIGH **Database Connection String**
Database connection string with potential credentials detected.

5. HIGH **Suspicious open() in Agent Input**
File open() call detected in agent input. In an agent context, direct file operations are suspicious and should be reviewed.

6. HIGH **Command Injection in Output (LLM05)**
Command injection via backticks or $() subshell. If output is rendered in shell context, this executes. OWASP LLM05.

7. HIGH **SQL Injection Fragment (LLM02)**
SQL injection fragment detected. OWASP LLM02: Insecure Output Handling.

8. HIGH **Password in Cleartext**
Cleartext password detected. Never store or transmit passwords in plain text.

9. HIGH **Approval Bypass (LLM08)**
Attempt to bypass human approval for agent actions. OWASP LLM08: Excessive Agency risk.

10. MEDIUM **Email Address (LLM06)**
Email address detected. OWASP LLM06: Sensitive Information Disclosure. DSGVO -- personal identifier.

11. MEDIUM **German Phone Number (LLM06)**
German phone number detected. OWASP LLM06: Sensitive Information Disclosure.

12. MEDIUM **Package / Dependency Install**
Software installation command detected. Verify the package source for supply-chain safety.

13. LOW **IP Address in Sensitive Context (LLM06)**
IP address detected. OWASP LLM06. DSGVO -- IP addresses are personal data under EU law.

4. EU AI Act Compliance Reference

The EU Artificial Intelligence Act (Regulation 2024/1689) establishes a comprehensive framework for AI systems in the European Union. Full enforcement begins 02 August 2026. The following articles are directly relevant to AI agent security scanning.

Article 9 - Risk Management System

Requirement:

High-risk AI systems require a risk management system throughout the system's lifecycle, including identification and analysis of known and foreseeable risks.

How this scan addresses it:

Prompt injection scanning directly addresses the requirement to identify and mitigate foreseeable security risks in AI systems.

Article 15 - Accuracy, Robustness and Cybersecurity

Requirement:

High-risk AI systems shall be resilient against attempts by unauthorized third parties to alter their use, outputs or performance by exploiting system vulnerabilities.

How this scan addresses it:

Documented prompt injection testing demonstrates compliance with cybersecurity robustness requirements.

Article 17 - Quality Management System

Requirement:

Providers of high-risk AI systems shall put a quality management system in place that ensures compliance, including procedures for data management, risk management, and post-market monitoring.

How this scan addresses it:

Regular security scanning with documented reports forms part of the required quality management system.

Article 61 - Post-Market Monitoring

Requirement:

Providers shall establish and document a post-market monitoring system proportionate to the nature of the AI system.

How this scan addresses it:

Continuous security scanning and compliance reporting satisfies post-market monitoring obligations for AI security.

Important Note

This report provides a security assessment of AI agent inputs. It does not constitute legal advice. EU AI Act compliance requires a comprehensive risk management approach. Consult qualified legal counsel for full regulatory compliance.

5. Methodology

Detection Engine

ClawGuard uses deterministic regex-based pattern matching - not LLM-based detection. This eliminates the fundamental vulnerability of using an LLM to detect attacks against LLMs.

Pattern Coverage

50 attack patterns across 7 categories: Prompt Injection, System Prompt Extraction, Data Exfiltration, Social Engineering, and Context Manipulation.

Performance

All scans complete in under 10ms with zero external API calls. The scanner operates fully offline.

False Positive Rate

Tested against real-world benign content: 0% false positive rate. Patterns are tuned for high precision.

Multilingual Support

Patterns include English and German variants for key attack types.