Agentic AppSec Unleashed '26
by Checkmarx

Bring Visibility Across the AI Supply Chain

David Dewaele
Product Director SSCS, Checkmarx

Paul DeLaria
Principal Solution Architect - ISV Security, AWS

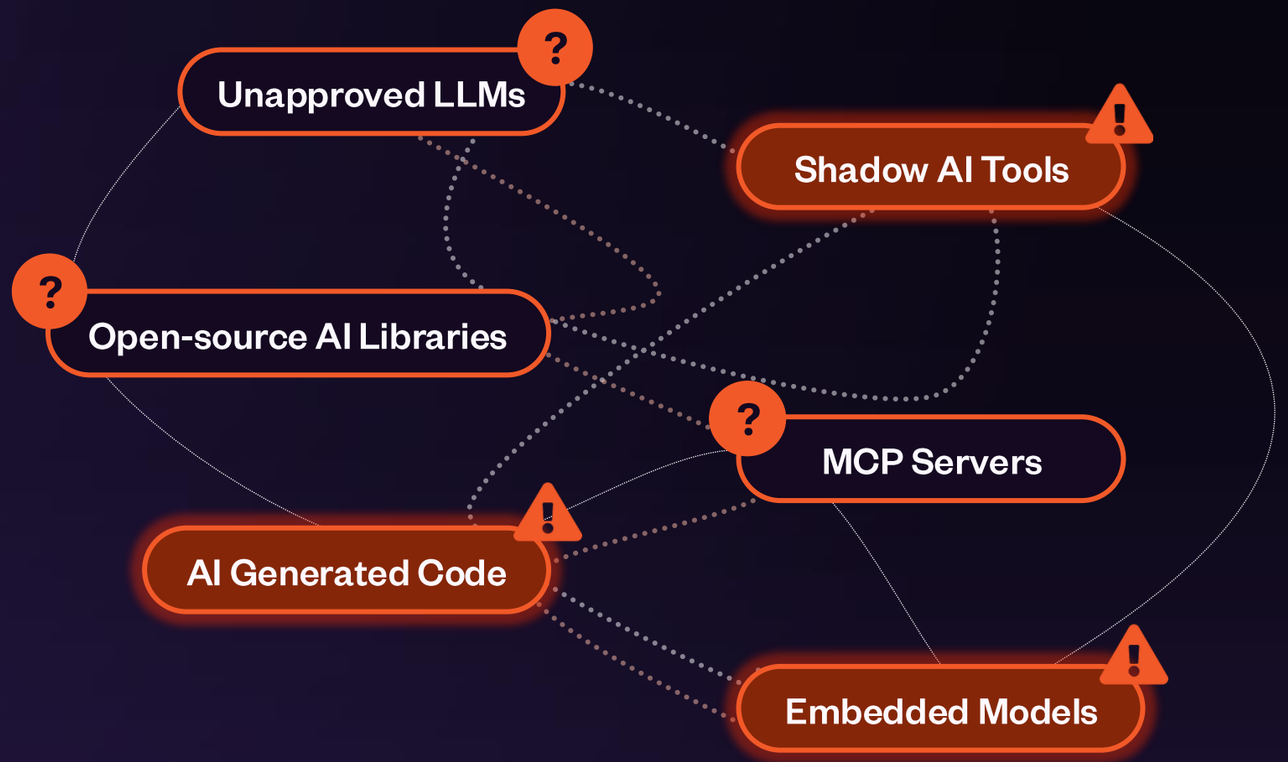
AI Supply Chain Security: It All Starts With Visibility

No centralized inventory of AI assets

Shadow AI tools spreading across teams

Unknown risk from models, MCP servers, and AI libraries

No policy enforcement or compliance reporting



AI Supply Chain Security: 3-Layer Approach

Visibility

Address shadow AI in Agentic Development Life Cycle

Identify AI assets across all repos/ applications

Provide Global Inventory of LLMs, MCPs, Agents, AI Libraries and SDKs

Risk Assessment

LLMs, MCPs, Agents

Deterministic Static analysis of model files, MCP configs before execution

Complements existing AppSec engines

Detects supply chain risks (insecure deserialization, dangerous loaders, tool poisoning, supply chain manipulation, etc.)

Governance and Reporting

Customize policies for allowed/blocked AI assets

Compliance mapping and governance (e.g OWASP Top 10 for LLM, EU AI act)

Generate and share AI-BOMs at repos/ applications/ global level

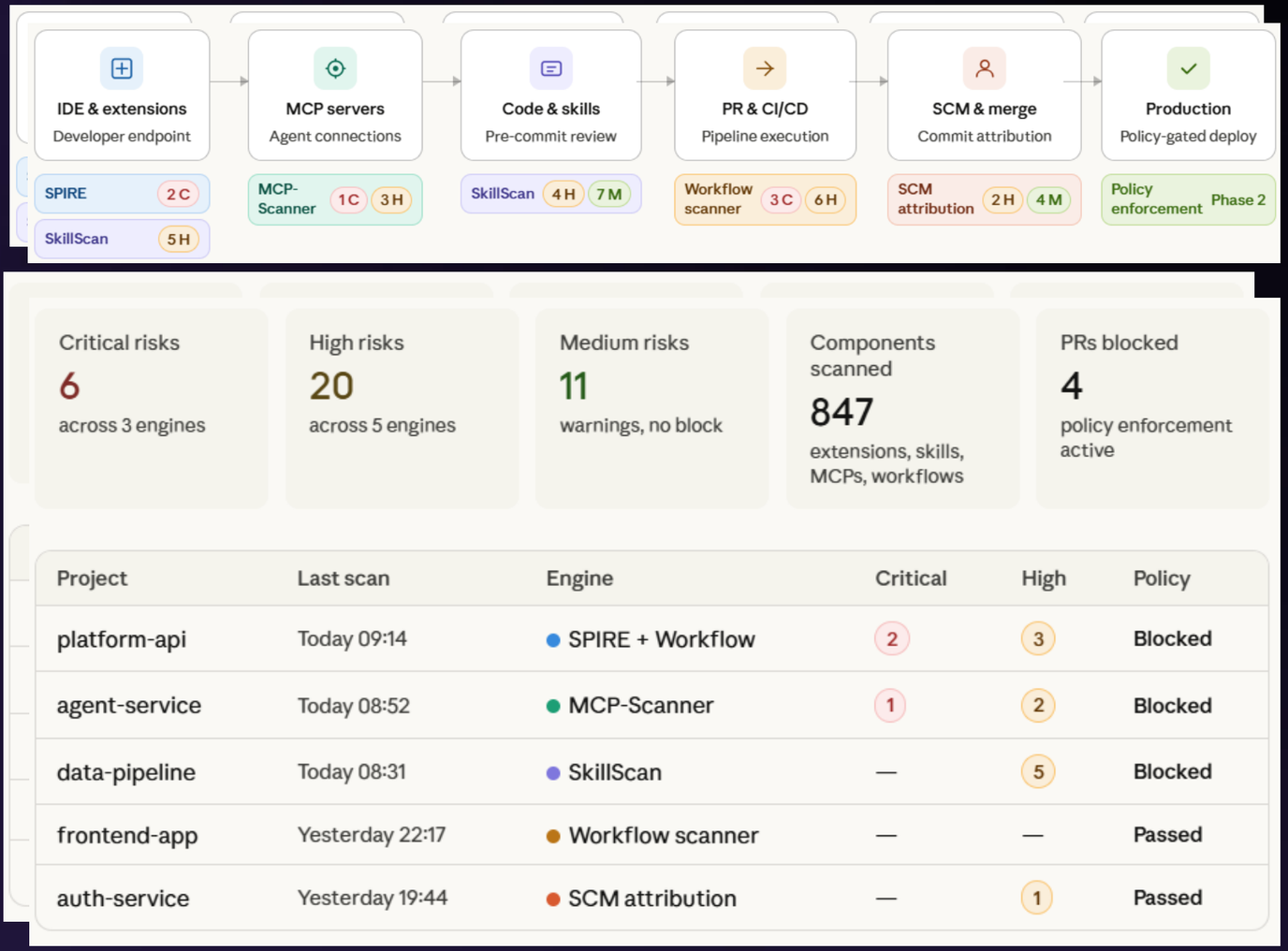
AI Supply Chain Security: AI Security

Securing LLMs	Securing MCPs	Securing Skills
Scans: ML-related files and artifacts (model weights, serialized files) in a project or remote repo (Git, Hugging Face)	**Scans:** MCP server source code, configuration files, and tool schemas	**Scans:** Agent skill files (Markdown-based instruction sets) to assess intent and behavior semantics
Covers: ML supply chain integrity, model-loading exploitation, malicious artifact injection	**Covers:** MCP supply chain integrity, configuration-level tampering, credential exposure, dependency trust	**Covers:** Prompt injection variants, agentic behavior manipulation, malicious persistence
Risks: Insecure deserialization, dangerous model loaders, shell execution, dynamic eval, suspicious pickle/Torch gadget patterns	**Risks:** Tool poisoning via hidden schema instructions, typosquatting, rug pulls, hardcoded API keys and MCP tokens, code-level injection flaws.	**Risks:** Privilege escalation via dangerous tool invocation, data exfiltration to external destinations, social engineering and output manipulation.

AI Supply Chain Security:
Beyond the Source Code

Securing IDE Extensions	Securing Developer Endpoints	Securing CI/CD
Scans: : IDE extensions from marketplaces, VSCode (Visual Studio Code) and OpenVSX (Open VSX Registry)	**Scans:** Inventory of developer endpoint components	**Scans:** Workflow yaml files
Covers: IDE supply chain integrity, malicious or typosquatted plugin detection, dependency vulnerability exposure within the development environment.	**Covers:** Local supply chain integrity, developer identity abuse, AI agent manipulation, credential exposure	**Covers:** CI/CD pipeline integrity, poisoned pipeline execution, supply chain attacks via compromised or weakly pinned GitHub Actions
Risks: Malicious plugins injecting unauthorized code or backdoors, outdated extensions with known CVEs, insecure IDE configurations exposing debug ports or access controls.	**Risks:** Malicious or typosquatted IDE extensions, poisoned MCP server schemas and configurations, hardcoded credentials in locally-installed tools, agent skill files attempting privilege escalation, etc.	**Risks:** Misuse of privileged triggers, expression injection, cache poisoning, artifact abuse, hardcoded secrets, unpinned actions, over-privileged permissions, and insecure unsafe command toggles.

Supply Chain Levels for Agentic Software Artifacts



Agentic AppSec Unleashed '26

by Checkmarx