Jul 01, 2025 Version 1

A2A Protocol V.1

DOI
https://dx.doi.org/10.17504/protocols.io.8epv5k7r6v1b/v1

Paola Di Maio[1]

[1]CSKRNS

Paola Di Maio
CSKRNS

DOI: https://dx.doi.org/10.17504/protocols.io.8epv5k7r6v1b/v1

Protocol Citation: Paola Di Maio 2025. A2A Protocol. **protocols.io** https://dx.doi.org/10.17504/protocols.io.8epv5k7r6v1b/v1

Protocol status: Working
We use this protocol and it's working

Created: July 01, 2025

Last Modified: July 01, 2025

Protocol Integer ID: 221353

Keywords: a2a protocol, agent protocol, a2a, protocol, master agent, agent, multiple specialized agent, autonomous ai agent, ai, interoperable way, delegate task, complex workflow

Abstract

A2A (Agent-to-Agent Protocol) is a framework that allows autonomous AI agents to communicate, coordinate, and delegate tasks among themselves in a secure, modular, and interoperable way. It is designed to support complex workflows where multiple specialized agents work together, often under the orchestration of a master agent.

1 Agent Registration & Discovery

2 Agents register their capabilities and presence in a shared directory or registry.

3 Agents can query this registry to discover other agents suitable for collaboration or specific tasks.

4 Authentication & Authorization

5 Agents authenticate themselves to the directory and to each other to establish trust.

6 Authorization checks ensure only permitted agents can access certain functions or data.

7 Negotiation & Task Delegation

8 An orchestrating (master) agent identifies a complex task and breaks it into subtasks.

9 The orchestrator negotiates and delegates subtasks to specialized agents based on their capabilities and availability.

10 Secure Messaging & Communication

11 Agents communicate using a standardized, encrypted protocol.

12 Messages include commands, status updates, data payloads, or requests for clarification.

13 Subtask Execution

14 Specialized agents execute their assigned subtasks, possibly using their own tools or APIs.

15 Agents may use additional protocols (like MCP) to interface with external systems as needed.

16 Progress Reporting & Feedback

17 Agents report progress, results, or errors back to the orchestrator.

18 The orchestrator may adjust assignments or parameters based on feedback.

19 Result Aggregation

20 The orchestrator collects and validates all results from the specialized agents.

21 Results are aggregated and post-processed to produce the final output.

22 Completion & Logging

23 The orchestrator confirms task completion.

24 All agents log their actions and results for auditing and future optimization.