Cryptography and Network Security

Team Name – Coders

Members

Aditya Narayan Bhandari(23BCE1128)

Tarun Chebolu(23BCE1016)

Scenario and problem statement

In a “Puzzle Room” environment, two cooperative agents must jointly solve a coordination puzzle under partial observability: Agent A (Observer) can see the full map and hidden trap indicators but cannot interact with switches, while Agent B (Actuator) can press switches and move blocks but only has local vision, so success requires exchanging observations, delegating subtasks, and confirming synchronized actions such as pressing two switches within a time window to open a door or moving blocks in a precise order to avoid traps. To make the communication *agent-native* (not generic chat), the agents use an A2A-inspired, task-oriented protocol: they discover each other’s capabilities via an Agent Card, coordinate through a task lifecycle using structured messages composed of parts, and produce structured results as artifacts that can be updated as new evidence is found during solving. A third simulated entity, Eve (Network Adversary), sits on the network path and can eavesdrop, replay recorded traffic, and optionally attempt a man-in-the-middle proxy attack, so the protocol must provide confidentiality, integrity/authenticity, and freshness guarantees. The project goal is to design and implement this A2A-style agent-to-agent protocol and secure it by performing an authenticated key exchange to derive per-session keys, then encrypting/authenticating every protocol message while enforcing replay protection via sequence numbers/nonces, and finally demonstrating security empirically by showing that Eve can only log ciphertext, that replayed messages are rejected, and that MITM attempts fail authentication or trigger an explicit abort.

Abstract

This project designs and implements a secure agent-to-agent communication protocol for a cooperative “Puzzle Room” environment in which an Observer agent with global knowledge must coordinate with an Actuator agent that can interact with the environment but has only local visibility. The protocol is inspired by the Agent2Agent (A2A) model by making communication task-centric rather than chat-centric: agents discover peer capabilities via an Agent Card, delegate work using a task lifecycle, exchange structured messages composed of parts, and produce structured outputs as artifacts suitable for incremental puzzle-solving updates. To evaluate security in a realistic adversarial setting, a third simulated network adversary (Eve) is placed on the communication path with capabilities to eavesdrop, replay traffic, and attempt man-in-the-middle interference. The protocol therefore incorporates an authenticated key exchange (e.g., using a Noise handshake pattern) to establish session keys and uses authenticated encryption (AEAD) plus freshness mechanisms (sequence numbers/nonces) to provide confidentiality, integrity, and replay resistance. The implementation is validated through experiments showing that Eve can observe only ciphertext, replay attempts are detected and rejected, and active interception fails authentication or triggers a safe abort, while the two cooperative agents retain effective coordination through A2A-like task delegation and artifact exchange.

Proposed Solution

The system will implement an A2A-inspired coordination layer where each agent publishes an Agent Card for capability discovery, then collaborates through a task-centric workflow (create task, stream task events, and return structured artifacts/messages composed of parts) so puzzle-solving coordination is expressed as explicit tasks and verifiable outputs. Over this layer, the agents will establish a secure channel using an authenticated key exchange (implemented via a Noise handshake pattern) to derive per-session symmetric keys, after which every protocol message (task creation, task updates, and artifact chunks) is protected using AEAD encryption with integrity checks and enforced freshness using sequence numbers to block replay and reordering. A third simulated adversary (Eve) will be placed on the network path to capture, replay, and attempt interception of traffic, and the implementation will demonstrate that Eve can only observe ciphertext, that replayed messages are rejected by the freshness policy, and that interception attempts fail authentication or cause an explicit abort, while the two cooperative agents still solve the puzzle efficiently using the task/artifact abstraction.