Motivation

Modern military rescue operations often rely on networks that are:

- Actors-based Cloud Safety
  - Actors = encapsulated state + behavior + thread of control + mailbox
  - Encapsulation
  - Mobility
  - Asynchronous message-passing
    - Uniform, high-level primitive for both data exchange and synchronization
    - Autonomous, concurrent actors
    - Inherently concurrent model of programming
    - No shared state
    - No data races

- Advantages
  - Modular and hence extensible
  - Multi-platform access
  - Advanced security technologies
  - Low latency and high efficiency

- Challenges
  - Fast-response and secured structure
  - Lower data processing latency
  - Large-scale efficiency
  - Garbage collection and smarter algorithms

Research Plan

- Build an easy-access as well as message-safe cloud computing platform
- Extend current local Actor framework (ActorFoundry)
- Provide more APIs for future use

Local Actors
- Nodes in cluster are individual actors in charge of different jobs, such as receiving GPS data, receiving location data, etc.

Central Server
- One Central Node is in charge of all data communication and security checks, etc.

Actors Network
- Build a message-safe and cross-platform cloud computing framework using ActorFoundry

Implementing Secured Cloud Using ActorFoundry

- Figure 1: Actors are reactive entities. In response to a message, an actor can (1) change its internal state, (2) send messages to peer actors, (3) create new actors, and/or (4) migrate to another computing host.

- Figure 2: Actor Name provides location transparency NameService, TransportLayer (TCP, UDP)

Fundamental Challenges

- Advantages
  - Modular and hence extensible
  - Multi-platform access
  - Advanced security technologies
  - Low latency and high efficiency

- Challenges
  - Fast-response and secured structure
  - Lower data processing latency
  - Large-scale efficiency
  - Garbage collection and smarter algorithms

Acknowledgments

