Intermittent Data Synchronization Across Distributed Network Environments

Client
Robert Kausch, CACI International <rkausch@caci.com>

Background
CACI International, Inc. Produces networked devices that connect to cloud systems from remote locations. In a normal use case, when a device is connected to the cloud system, data is streamed in the form of individual messages from the cloud to the remote system, and a data cache is maintained locally on the device. Occasionally and unpredictably, this connection is interupted. When the connection is restored, data from the outage period needs to be synchronized to the remote device.

Project Goals and Requirements
The primary focus of this project is to develop a pair of services (one on each end of the transaction) to synchronize textual data between a cloud system and a remote device. The team members will investigate and experiment with various technologies, including but not limited to: Kafka, Syncthing, ActiveMQ, or similar technologies, and / or a custom implementation. Upon completion of the project, the team will produce an industry-relevant whitepaper, and demonstrate a pair of microservices through which data is synchronized across a network boundary, observing the following parameters:

Required Parameters:

1. Missing data needs to be synchronized in reverse time order, with the most recent data sent first, and the oldest synchronized data sent last.
2. During synchronization operations, data will be cached locally on the device and made available for use immediately upon receipt, instead of waiting for the entire synchronization operation to complete.
3. During synchronization operations, “current” data is being received alongside synchronization of “missing” data from the outage period. Both will be cached into the same local repository without interfering with each other.
4. Outage periods may contain significant quantities of data, both in size and time duration (up to two weeks).
5. Data may not be duplicated within synchronization operations.

Optional Parameters:

1. Connectivity may be lost during synchronization operations. Upon restoration of connectivity, the outage window may be treated as a single outage and a new synchronization operation initiated. Partially recieved messages should resume synchronziation from the last bit received.
2. Data is organized into “sets”, and the prioritization of a given set may be changed from the device. Highest priority sets should stream more quickly than lower priority sets, but no set should “starve” during synchronization.
3. Missing data can be synchronized in either reverse or normal time order, depending on user selection.

Suggested team size and location:
2-3 students. Work can be performed from a combination of our offices in the Denver Tech Center and remotely.

Skills / Experience for CSM Students
Students will utilize technologies common to our production environments, including Java 17, Docker, and Kubernetes, as well as using common DevSecOps processes to produce and continually deploy software. Students will be teamed with one or more mentors to help produce the project.

NDAs and Intellectual Property
Students will be required to sign a Non-Disclosure Agreement, and an IP Deed document with CACI International, Inc. All intellectual property developed as part of this project will be owned by CACI International, Inc.