FogBus External tests report

# Benchmarks used for evaluation:

1. Operation limited benchmarks – These send real oximeter like requests to the FogBus framework but only 50 number of such requests. The requests are sent after an interval of 5 seconds, 1 second and 0 seconds.
2. Time limited benchmarks – These send real oximeter like requests to the FogBus framework but are limited by time = 300 seconds. Each request is sent after an interval of 5 seconds, 1 second and 0 seconds.

# Cloud Tests

This series of tests compares what happens when the cloud is not integrated in the Fog domain. To perform this test, we assign thresholds to the workload limit that each Fog Node and Cloud can take (Master is only for work allocation among different workers and not for computation). As we could only send requests from limited users and at limited rate, we set the thresholds to low values to model a similar situation of load congestion in the real case. The worker cpu limit was set to 10% and cloud to 5%. If worker has load > 10% it sends the task to cloud, else sends it to worker. If cloud gets the request but has load > 5% then the task is dropped and is informed to the user. If cloud is not present, clearly the drop rate would be higher for compute intensive tasks. Though the tasks that can be completed with cloud compared to those without cloud would have high latency, but the drop rate would be significantly lower. Different load scenarios were created using the time limited benchmarks. To maintain consistency, only 1 worker node was used in this series of tests.

Clearly the drop percentage is much lower in the Cloud Enabled case and the difference in drop percentage increases as load increases. Thus, FogBus is more robust compared to those that do not allow integration with Cloud services for computation, and hence better than [10], [14], [16], [17], [12], [19], [21].

# Distributed processing and Decentralized allocation Tests

FogBus allows fail safe operation using continuous heartbeats from the worker nodes. If a worker node fails, then the work is assigned to another worker. Also, to prevent bursts of operations to a worker node, it sends the task to the worker with minimum load.

We see that for different loads to the framework, if we have a load balanced and heartbeat-based work allocation then the drop percentage is significantly lower compared to a policy that distributes work randomly to one of the nodes.

Moreover, if the master fails, FogBus can shift the master privileges to another worker and reconfigure all workers. This allows a failsafe mechanism to escape cases when the Master node breaks down. We see below that in a centralized system with 1 master, if a master node fails then all requests are dropped, but in a decentralized system like FogBus, the drop rate is near 0. Thus FogBus is better than [15], [18], [11], [12], [22].

# Data integrity test

Blockchain allows FogBus to store data in a completely reliable and hack-proof way. If data is changed at any node using fraudulent means, the FogBus system does not send further requests to that node till it is restarted (as that node may be compromised). Also, the FogBus framework keeps distributed copies of the chain and hence periodically compares the latest hash values of all chains to determine the compromised node. It also, automatically copies the majority chain to the compromised nodes to resume processing. This makes FogBus framework resilient to data tampering and hence better than [13], [20].