Poster: EdgeKeeper – Resilient and Lightweight Coordination for Mobile Edge Computing Systems

S. Bhunia, R. Stoleru, A. Haroon, M. Sagor, A. Altaweel, M. Chao, M. Maurice†, R. Blalock†
Department of Computer Science and Engineering, Texas A&M University
†National Institute of Standards and Technology (NIST)
{sbhunia, stoleru, amran.haroon, msagor, altaweelala1983, chaomengyuan}@tamu.edu
†{maxwell.maurice, roger.blalock}@nist.gov

ABSTRACT

Mobile Edge Computing (MEC) is gaining significant interest from first responders and tactical teams. Typical cloud-based coordination (e.g., service discovery and coordination, device naming, authentication) does not work in MEC due to high user mobility. We design and implement EdgeKeeper to provide cloud-like service coordination to distributed edge computing applications for MEC systems. It maintains an edge cluster among devices and intelligently stores data on a group of replicas to guard against node failures/disconnections. We provide a full-system implementation of EdgeKeeper for Android and Linux platforms and evaluate it with MEC applications in a real-world wide-area search and rescue operation conducted by first responders.

1 Introduction

In disaster response mission the responders are assisted by handheld devices, on-body cameras, and other sensors. As the cellular wireless infrastructure is usually unavailable, these teams carry a deployable system, typically providing a diverse hardware configuration with 4G LTE, WiFi, or WiFi Direct. This enables mobile devices to provide distributed computing resources. Traditionally, popular mobile applications offload processing-intensive jobs to remote cloud servers. In the absence of cloud connectivity, forming an edge becomes an emerging necessity [1]. In our proposed design, as can be seen in Figure 1, multiple mobile nodes form an edge network where mobile devices can offload tasks to nearby mobile devices as well as the cloud server when it is available.

Similar to Apache Hadoop ecosystem, we developed the DistressNet-NG ecosystem, mainly targeting edge networks formed by handheld devices and deployable manpacks carried by first responders. EdgeKeeper runs on all the devices in the background and provides resilient coordination to client applications. It provides a comprehensive API for client applications that includes device naming, application coordination, metadata storage, authentication and edge status monitoring. It hides all the complexity of edge coordination from applications. It is implemented on Linux and Android platforms, and the source code is available on Github. Several tests during real-world deployment of first-responders prove its effectiveness.

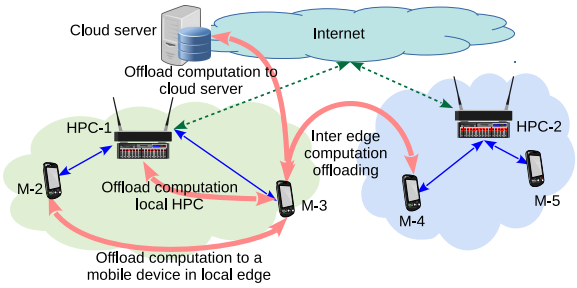

Figure 1: Mobile edge network architecture

2 EdgeKeeper Design

EdgeKeeper adopts the master-replica concept and replicates data over multiple devices to tackle link failure. The role of replica and client is chosen dynamically, depending on the network status.

Identity, device naming and authentication: EdgeKeeper uses Global Naming Service (GNS) [2], which employs multiple name servers to deal with high name resolution rates across the globe. Each name record is associated with a primary key: a globally unique identifier (GUID). EdgeKeeper uses a local cache mechanism to store the name records at the edge and it also lazily updates the GNS server when the Internet is available. EdgeKeeper uses X509 certificates for node authentication and a certifying authority (CA) at each organization creates client certificates and signs them.

Service Discovery and Coordination: GUID records are used for service discovery. A device offering some service adds the service name and the role (e.g., server, client, etc.) in the GUID record to be discovered by other nodes. Any node who wants to find a list of nodes offering a particular service will query to retrieve a list of GUIDs.

Resilient MetaData Storage: EdgeKeeper provides resilient metadata storage to client applications. EdgeKeeper clusters dynamically chose the replica nodes (maintaining consensus) based on the availability of devices.