CS540 - Paper Review Report # 16

Hailu Belay Kahsay - 20155624

Title: Fog Computing and Its Role in the Internet of Things

Author: Flavio Bonomi, Rodolfo Milito, Jiang Zhu, Sateesh Addepalli

In the paper, it is indicated that cloud computing model is an efficient alternative to owning and managing private data centers for customers facing Web applications and batch processing because it frees the enterprise and the end users from the specification of many details. But this becomes an issue for latency-sensitive applications as nodes need to meet delay requirements and IoTs require mobility support and geo-distribution in addition to location awareness and low Latency.

The authors argue that Fog Computing(cloud close to the ground), is a highly virtualized platform that provides compute, storage, and networking services between end devices and traditional Cloud Computing Data Centers, typically, but not exclusively located at the edge of network, meets the above IoT requirements. Fog computing characteristics: Low latency and location awareness, Widespread geographical distribution, Mobility, Very large number of nodes, Predominant role of wireless access, Strong presence of streaming and real time applications, Interoperability and federation, Support for on-line analytic and interplay with the Cloud, and Heterogeneity, makes it the appropriate platform for a number of critical Internet of Things (IoT) services and applications namely, Connected Vehicle, Smart Grid , Smart Cities, and, in general, Wireless Sensors and Actuators Networks (WSANs).

Considering three scenarios of interest: Connected Vehicle, Smart Grid, and Wireless Sensor and Actuator Networks, the role the Fog plays is demonstrated. In the connected vehicle scenario, fog computing characteristics such as: infotainment, safety, trac support, and analytics: geo-distribution (throughout cities and along roads), mobility and location awareness, low latency, heterogeneity, and support for real-time interactions attributes of fog computing are shown. But in case of Wireless Sensors and Actuators Networks, proximity and location awareness, geo-distribution and hierarchical organization characteristics of the Fog make it the suitable platform to support both energy-constrained WSNs and WSANs.

Finally, the authors pointed out some: [1] Architecture of this massive infrastructure of compute, storage, and networking devices, [2] Orchestration and resource management of the Fog nodes, [3] Innovative services and applications to be supported by the Fog, as future works.

Discussion Points:

- Flexibility and Scalability issues?
  - SDN: How to cooperate different controllers, where to place controllers in fog network, how to distributed SDN
  - Network function virtualization (NFV): by virtualizing gateways, switches, load balancers, firewalls and intrusion detection devices and placing those instances on fog nodes.
    - performance of virtualized network appliances: throughput or latency of virtualized network appliances (middlebox) in fog network, and how to achieve efficient instantiation, placement and migration of virtual appliances in a dynamic network.
- QoS issues: Connectivity, Reliability, Capacity and Delay
- Security and Privacy

Title: A Hierarchical Edge Cloud Architecture for Mobile Computing

Author: Liang Tong, Yong Li and Wei Gao

With the increasing number and complexity of mobile applications the ability of the mobile devices to provide the services with the required or better performance becomes an issue. To significantly improve the performance of mobile devices we can leverage cloud computing and migrating mobile workloads for remote execution at the cloud(execute mobile applications remotely). Such remote executions benefit a large varieties of mobile applications, such as gesture recognition, voice control, recognition assistance and mobile gaming.

The authors argue that modern cloud computing services such as, Amazon EC2 and Microsoft Azure, are solely hosted by data centers and are incapable of efficiently executing mobile applications due to long network transmission latency and the overhead for global VM provisioning and management (may add overload to data centers during peak hours).

To address the above mentioned challenges and to improve the efficiency of cloud resource utilization by organizing the edge cloud servers into a hierarchical architecture, the authors proposed to deploy cloud servers at the network edge and design the edge cloud as a tree hierarchy of geo-distributed servers, so as to efficiently utilize the cloud resources to serve the peak loads from mobile users. That is, Instead of serving mobile users directly using a flat collection of edge cloud servers, the proposed idea opportunistically aggregate and serve the peak loads that exceed the capacities of lower tiers of edge cloud servers to other servers at higher tiers in the edge cloud hierarchy. It is observed that this approach is able to serve larger amounts of peak loads with the same amount of computational capacities being provisioned at the edge.

As per the formal analysis and systematic experiment conducted to compare the performance and resource utilization efficiency between the flat and hierarchical designs of the edge cloud, it is shown that advantage hierarchical edge cloud architecture performs better than the flat one. The authors further develop workload placement algorithms, that decides which edge cloud servers mobile programs are placed on and how much computational capacity is provisioned to execute each program, to ensure efficient utilization of cloud resources.

To address the major challenge of the workload placement algorithm,the difficulty of appropriately balancing between the computation and communication delays in the edge cloud, and get the optimal solution, by first considering a scenario which has only one server at each tier of the edge cloud, the authors aggregate the decisions of workload placement at different edge cloud servers together.

Finally, the performance of the proposed hierarchical edge cloud architecture is evaluated by formal analysis, small-scale system experimentation, and large scale trace-based simulations.

Discussion points:

- Privacy and security?
- Concurrency and Synchronization?