Bachelor Theses - Serverless Computing

Serverless computing is a new paradigm for building distributed applications. A serverless application relies entirely on a platform for executing and scaling functions. Serverless computing may support Fog Computing in practice. In Fog computing, applications can improve latency, responsiveness, and bandwidth by offloading work earlier in the processing pipeline. Currently, no serverless framework allowing the utilization of the different heterogeneous resources in Fog exists.

To further the development of a Serverless-Fog-Environment, we want to address the following challenges through Bachelor Theses:

1. Function distribution in Fog
   Provisioning of function containers and consistently updating running containers in a Fog environment is a challenge due to the high degree of distribution and heterogeneity of the infrastructure. How can function code and metadata be distributed across the fog?
   
   **Topic & Goal:** The objective is to investigate methods that can be used to transport serverless-function-code between heterogeneously distributed nodes. Of interest are distributed file systems like IPFS and software like git.

2. Distributed Monitoring of Fog-Components
   Monitoring of a distributed infrastructure of short-lived applications can be challenging as devices are limited in resources and connectivity. Can techniques originally developed for microservice management, such as Jaeger and Zipkin, be extended to function in a “Serverless Fog Environment”?
   
   **Topic & Goal:** The objective is to develop a distributed monitoring system for fog components. This monitoring system can extend existing solutions or be built from scratch.

3. Evaluate the usability of FaaS-built web services
   FaaS promises to reduce operational overhead and cost; we want to evaluate this claim for a typical web service. Therefore, an experimental setup using a web service deployed on a VM and deployed in Lambda is to be used. Benchmarking the cost and performance for different access patterns can then be used to verify that claim.
   
   **Topic & Goal:** The objective is to build a subset of the “Sock Shop” infrastructure on AWS lambda and then test both a VM deployment and a Lambda deployment with different traffic patterns and evaluate the results.

Interested in other Serverless Topics? Let me know!

Contact: Sebastian Werner
werner@tu-berlin.de

Our Mission:
Our lectures cover fundamental methods and techniques in the areas of service computing, cloud computing, and enterprise computing. We like to engage students in hands-on building of distributed information systems and to take an interdisciplinary approach to evaluating such systems. Through a close mentoring of students, especially in our seminars, we aim to introduce students to our ongoing research and to excite them to do future studies and research with us.