

FlowQoS: Not Every Flow is Born the Same

Dhruv Sharma
UC San Diego, CA
dhsharma@cs.ucsd.edu

Robert Jenkins
UC San Diego, CA
rfjenkins@ucsd.edu

Frederik Nygaard
UC San Diego, CA
freddyny@ucsd.edu

Feichao Qian
UC San Diego, CA
feqian@ucsd.edu

Abstract—ith majority of the world’s data and computation handled by cloud-based systems, cloud management stacks such as Apache’s CloudStack, VMware’s vSphere and OpenStack have become an increasingly important component in cloud software. However, like every other complex distributed system, these cloud stacks are susceptible to faults, whose root cause is often hard to diagnose. We present Hansel, a system that leverages non-intrusive network monitoring to expedite root cause analysis of such faults manifesting in OpenStack operations. Hansel is fast and accurate, and precise even under conditions of stress. ith majority of the world’s data and computation handled by cloud-based systems, cloud management stacks such as Apache’s CloudStack, VMware’s vSphere and OpenStack have become an increasingly important component in cloud software. However, like every other complex distributed system, these cloud stacks are susceptible to faults, whose root cause is often hard to diagnose. We present Hansel, a system that leverages non-intrusive network monitoring to expedite root cause analysis of such faults manifesting in OpenStack operations. Hansel is fast and accurate, and precise even under conditions of stress. W

I. INTRODUCTION

With majority of the world’s data and computation handled by cloud-based systems, cloud management stacks (CMSes), such as Apache’s CloudStack [?], VMware’s vSphere [?] and OpenStack [1], have become an increasingly important component in the cloud software. These CMSes are complex, large scale distributed systems enabling orchestration of tasks, including virtual machine (VM) spawning, termination, migration, etc. However, like every other distributed system, these CMSes are susceptible to a variety of complex faults, most of which are often hard to diagnose even for skilled developers/operators. For example, several faults [?], [?], [?] in Rackspace’s cloud offering based atop OpenStack took several hours or even days to resolve. In this paper, we focus on the problem of automatically determining the cause of such hard to diagnose faults manifesting in OpenStack operations.

II. RELATED WORKS

III. MOTIVATION

IV. PROBLEM

V. DESIGN

VI. IMPLEMENTATION

VII. EVALUATION

VIII. DISCUSSION

IX. CONCLUSION

REFERENCES

[1] OpenStack. <https://www.openstack.org/>.