Experimental Research with IP-over-P2P Overlay Virtual Networks

Renato J. Figueiredo

***Abstract*—Network virtualization has been recognized as a key technology underpinning cloud computing environments. At the infrastructure level, software-defined networks (SDNs) allow improved operations and management in large multi-tenant data centers, and network function virtualization (NFV) leverage virtual machine technologies to deploy network services. In addition, several use cases (such as multi-cloud platform-as-a-service systems) call for network overlays that can be deployed at Internet endpoints, without requiring access to core network infrastructure. The open-source IP-over-P2P project pursues research and development of "user-defined networks", where virtual IP routers and network interfaces run at endpoints – including mobile devices, sensors, desktops, and cloud virtual machine instances – and where P2P links provides a basis for efficient tunneling with end-to-end privacy and authentication. This position paper elaborates on experimental requirements that are borne out of this project, how they may be addressed by NSFCloud, and help support experiments with other classes of P2P systems.**

I. Introduction

A key characteristic of cloud computing environments is the ability to provision isolated resources dynamically and with the flexibility to accommodate user-specified environments. Machine virtualization for commodity servers provided the key enabling technology; subsequently it has become clear that the role of virtualization transcends the virtual machine and applies to other major computer subsystems (storage, networking). In particular, network virtualization became an increasingly relevant research topic, with efforts on software-defined networks and network function virtualization focusing on various aspects of the problem. The IP-over-P2P project led by the University of Florida focuses on one modality of network virtualization – peer-to-peer overlay virtual networks, which provide virtualization capabilities by running virtual routers at edge devices, not relying on any changes to Internet infrastructure. This approach has several applications in the broad context of cloud computing, and the design and implementation of such overlays requires significant experimental support. This position paper overviews the technology behind IPOP and highlights experimental needs from this overlay, which can be generalized to other classes of P2P systems that may benefit from NSFCloud.

This material is based upon work supported in part by the National Science Foundation under Grants No. 1339737, 1265341, and 1234983. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Renato J. Figueiredo is with the Advanced Computing and Information Systems Lab at the Electrical and Computer Engineering Department, University of Florida, FL, 32611, USA. (e-mail: renato@acis.ufl.edu).

II. IP-over-P2P Virtual Network Overlay

IPOP [2] is an open-source user-centric software virtual network allowing end users to define and create their own virtual private networks (VPNs). In its most recent design [1], IPOP virtual networks provide end-to-end IP tunneling over "TinCan" links setup and managed through a control API to create various software-defined VPN overlays. In contrast to typical VPN technologies, which require complex setup, management and a gateway to relay VPN traffic, IPOP is straightforward to configure, because: a) it runs on existing Internet infrastructure, without requiring any specialized networking infrastructure, b) it uses online social network (OSN) infrastructures to allow users to define their own networks, and c) it seamlessly traverses firewalls, NATs, and create VPN tunnels in a peer-to-peer fashion, avoiding the need for centralized VPN gateways.

IPOP has been used in scenarios including virtual clusters across multiple clouds (e.g. ConPaaS project, www.conpaas.eu), trusted collaborative environments linking private clouds (e.g. in PRAGMA, www.pragma-grid.net), mobile device offloading to clouds, next-generation P2P-based social networking applications, as well as in easy to setup VPNs for small businesses and to connect friends and family.

III. Experimental Needs

The IPOP project has both research and software development thrusts. Research efforts use IPOP software as a basis for quantitative experiments in realistic environments, and research ideas that are tested and provide compelling use cases feed into the requirements and features of the software. The software itself is written in C++ (datapath) and Python (controller), and is deployed as user-level processes that perform packet capture, processing, and overlay routing. IPOP relies on virtual network interfaces (e.g. a tap device) to integrate with existing operating systems. Our experiments (and other researchers who use IPOP) thus benefit from the deployment of large numbers of operating system instances that can run both a virtual network interface, and the IPOP software – including (emulated) mobile devices with Android.

Experiments with overlay networks at scale: Experimental research in this area involves the ability to deploy software-defined networks, possibly including hardware-supported SDN switches, as well as end-to-end user-level overlay virtual networks connecting mobile devices to cloud and personal devices. In addition, the ability to perform unit, integration and regression test requirements for the open-source software that we release to the community as part of a funded SI2/SSE project can benefit from the NSFCloud infrastructure; in the future, this can provide a basis for investigation of software engineering research questions.

Thus, the ability to programmatically deploy and inject faults (e.g. churn) in very large-scale overlays is key to evaluating the performance of device-to-device social virtual private networks at scale. Since end-to-end experiments require us to deploy a large number of overlay network endpoints, each with a virtual network interface and running applications in an isolated environment, it is crucial in our experiments to be able to scale to large number of O/S instances. Full-system virtualization is often too coarse-grained for these experiments – by using container-based environments (e.g. Docker/LXC) to time-share multi-core, large-memory systems, our experience is that it is feasible to run overlay network scalability experiments with an order of 50 containers per VM per core in current FutureGrid resources. We also perform research that investigates nested virtualization environments for PaaS environments.

With higher-capacity NSFCloud resources, we anticipate to be possible to conduct integrated, end-to-end experiments involving emulated overlays of tens of thousands of endpoints, and compute/storage cloud nodes for offloading, which is currently not possible using available institutional infrastructure. NSFCloud may assist in providing support to these experiments by enabling its users to deploy and manage large numbers of light-weight containers – in addition to VMs – and configure software-defined networks.

Experiments with mobile devices, Internet-of-Things: IPOP has been demonstrated to run in mobile and embedded devices, opening up interesting applications in virtual networks that can reach end users and sensors. Mobile computing devices, such as smartphones and tablets, are the primary systems by which a growing number of Internet users interact with networking services. The diminishing size and battery requirements of mobile devices restrict the scope of computations possible on such devices and motivate approaches that support the selective offloading of computations to remote resources. There are a variety of resources available to potentially host offloaded computations – including cloud-provisioned resources, as well as devices near the user and within a user's personal or social network. A key challenge lies in architecting an offloading framework that enables applications to seamlessly discover available services, effectively and securely communicate with them, while presenting APIs that hide the complexities associated with managing the interactions with a remote device and run-time systems that handle the decision-making involved in selecting tasks for off-loading to minimize power consumption while maximizing performance.

These challenges are being addressed by UF researchers in the context of a funded NSF/NSF-C collaboration with researchers at Peking, Nanjing and Fudan Universities (China) and Northwestern University. This research effort considers mobile devices connected to cloud computing back-ends (as well as opportunistic resources that belong to the mobile user, or peers in their social network) via software-defined ad-hoc overlay virtual networks that can be used to autonomously create trusted virtual network links for resource discovery and remote task execution. On top of the virtual network communication fabric, middleware at the mobile platform and the cloud execution back-end applies techniques to automatically partition applications developed by our collaborators, allowing off-loading run-time decisions that can be done in a transparent manner and guided by scheduling heuristics including machine learning.

Even more constrained than mobile devices are small computer systems such as Raspberry Pi and Edison, which can be used in sensors and in the Internet-of-Things. There are several new research questions that arise in this context which require new infrastructure for experimental research.

One important aspect of this research considers implications of off-loading from the perspective of the mobile device user. Computation off-loading allows a mobile device to leverage accelerator devices not present locally in the platform, such as CPUs and/or high-performance GPUs. Fundamental trade-offs arise with respect performance, energy consumption, and cost as off-loading decisions are made when multiple devices are potential targets for off-loading (e.g. a high-performance server provisioned by a cloud infrastructure, or a desktop with reduced capabilities but accessible through a lower-latency local-area network and at negligible cost).

The NSFCloud experimental infrastructure may provide capabilities that allow our research to consider a variety of heterogeneous computational resources (including GPUs, low-power servers, high-performance servers). It would be very helpful to have access to actual Android-based mobile devices – as well as Raspberry Pi and/or Edison – or, alternatively, the ability to emulate these platforms within NSFCloud.

REFERENCES

[1] Pierre St Juste, Kyuho Jeong, Heungsik Eom, Corey Baker, Renato Figueiredo, 'TinCan: User-Defined P2P Virtual Network Overlays for Ad-hoc Collaboration', ICST Transactions on Collaborative Computing, 2014

[2] Ganguly, Arijit , A. Aagrawal, P. O. Boykin, Renato Figueiredo, 'IP over P2P: Enabling Self-configuring Virtual IP Networks for Grid Computing'. In Proc. 20th IEEE International Parallel and Distributed Processing Symposium (IPDPS), 04/2006.