Enhancing Datacenter Network Security and Scalability with Trusted End Host Monitors

Alan Shieh†, Srikanth Kandula, Albert Greenberg

ashieh@cs.cornell.edu, {srikanth, albert}@microsoft.com

Although datacenters dedicated for cloud computing services are becoming increasingly prevalent, the current datacenter network security architecture is poorly suited for this application. Policy enforcement is smeared between the network and end hosts, increasing cost and complexity while reducing flexibility and security. Enforcement is typically done at network chokepoints, which inherently see high traffic levels from aggregate traffic; packet filters and deep packet inspection engines that can operate at these data rates require expensive, custom hardware.

Elasticity to tenant demand and hosting untrusted tenant applications are central features of cloud computing that present further challenges. As tenant VMs and applications are spun up or migrated, the common infrastructure needs to be reconfigured, reducing performance and pulling the network devices into the trusted computing base (TCB). The security implications are difficult to understand, since correctness now depends on topology and low-level, non end-to-end techniques such as L2 steering and VLANs. Buggy tenant applications are attractive targets for compromise, since nodes in the cloud have a fat pipe with which to attack other other tenants; such exposure poses a barrier to hosting critical customer infrastructure in cloud datacenters.

By providing a trusted enforcement mechanism at the end hosts, our architecture facilitates shifting policy enforcement from the network to end hosts. We assume that every end host contains commodity trusted hardware, such as the Trusted Platform Module (TPM) and a trusted NIC, such as the one in Intel Active Management Technology (AMT) which is currently available as a configurable option on many of the newer desktop and server motherboards from Intel. We also assume a virtualized software stack that is strongly isolated from tenant-provided software. Given the wide use of virtualization in cloud datacenters (e.g., AWS, Azure) and the high degree of control and homogeneity in datacenters, these assumptions are quite close to current practice. Enforcement mechanisms, such as packet filters and stateful firewalls, and monitoring mechanisms, such as traffic sampling and intrusion detection, are handled by these trusted layers, and hence guaranteed to run even when the tenant software is malicious or compromised.

While some security policies may need to be enforced in the network, a substantial fraction of the policies used in practice can indeed be refactored to be implemented at just the end hosts as we show below. For such policies, end host enforcement provides several benefits. Data rates at the end hosts are low and CPU cores aplenty so that there are more per-packet resources to enforce a policy at the end hosts. Further, application-level semantic information is still available (e.g., TCP state, data beore IPSec) which lets richer policies be enforced at lower cost. Since the in-network processing requirements are reduced, the network can be built from less expensive switches, routers, and middleboxes. We use a few concrete examples to better illustrate these benefits:

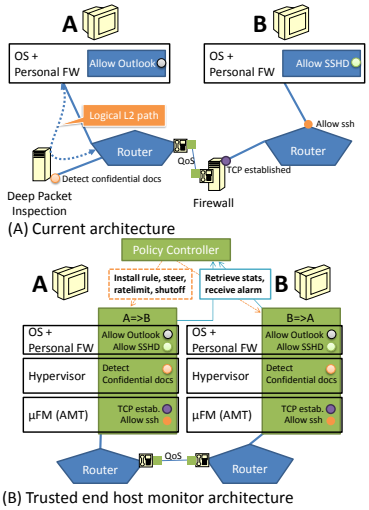

Figure 1: **Datacenter network architecture.**

- *End to end access policies.* Access control in datacenters are often expressed in group- and role-based policy languages. For instance, each tenant exposes only limited external access and restrict the internal access between front-end, back-end, and application servers. Implementing such policies in end hosts eases migration, since little to no network reconfiguration is required, and allows efficient and accurate enforcement of policies based on local information, such as process identities.
- *Analysis of encrypted traffic.* Though encryption enhances confidentiality, it can degrade security if attack payloads are hidden from intrusion protection systems. Trusted end host monitors can inspect network traffic without increasing data exposure, since the traffic is already available at the end host in plaintext.

†Student author. Work done at Microsoft Research.