

Midterm 2 - Part 1 - Reading Assignment

Software Defined Networking (SDN) is creating a trend and popular way of designing and managing networks. The characteristics of Software Defined Networking is broken up into two parts. The first part of the characteristic is that it has a control plane and decides how to handle the traffic from the data plane. The second part of characteristic of Software defined networking is consolidating the control plane so that the one software can control many aspects of the data plane. With this in mind, it helps network engineers to eventually have the perfect “plug and play” networking tool. Making an application like this would give the client/customer more control of their network because they would have their choice of how their network traffic.

The biggest technologies that help create a solid foundation for Software Defined Networking is OpenFlow and Network OSEs. The OpenFlow protocol much like the definition in the beginning makes Software Defined Networking appealing because it worked on top of existing networking infrastructure. Basically OpenFlow has the initial set of capabilities on switch was as easy as performing a firmware upgrade. In addition, vendors did not need to upgrade the hardware they already have to make their switches OpenFlow compatible. Software Defined Networking is not only applied at networks like businesses, schools, and corporations, but they are applied at data centers as well. The reason it be useful there because it would become cost efficient.

The reasons why OpenFlow became widely popular because it provided beneficial intellectual contributions. The first one is generalizing network devices and functions. It does this by defining forwarding behavior on traffic flows based on any set of thirteen different packet headers. Second contribution is the vision of a network operating system. This operating system would be a “node” operating system and that abstracts the installation of state in network switches from the logic and the applications that controls the behavior of the network. Last is distributed state management techniques. Since separating the control and data planes is challenging, the Onix controller. The controller incorporates previous distributed networks which helps with consistency and durability. In addition, Onix contains a transactional persistent database backed by a replicated state machine. This is for a slowly-changing network state, as well as a built in memory distributed hash table for rapidly-changing state with weaker consistency requirements.

Software Defined Networking has become popular over the years since the birth of the internet, however it is far from perfect. Though the hard work on OpenFlow and network operating systems creates the right balance between vision and pragmatism. The data plane right now is confined to primitive match action operations on packet-header fields. There needs to be more work and research for home networks, cellular, and WiFi networks as well. As a current or future network engineer, it is a great feature to have in your tool belt when developing and or managing a network. Since it is still very new, there can be new ways in designing Software Defined Networking applications with the current OpenFlow protocols at hand. The possibilities are endless.