NETRONOME

WHITE PAPER

Agilio® OVS Software Architecture

FOR SERVER-BASED NETWORKING

THERE IS CONSTANT PRESSURE TO IMPROVE SERVER-BASED NETWORKING PERFORMANCE DUE TO THE INCREASED USE OF SERVER AND NETWORK VIRTUALIZATION IN MODERN DATA CENTERS.

NETRONOME'S AGILIO SOFTWARE IS DEDICATED TO OFFLOADING AND ACCELERATING SERVER-BASED NETWORKING.

CONTENTS

INTRODUCTION

Server-based networking is gaining a lot of attention recently due to its importance in data center functionality and performance. It serves as the conduit for Virtual Machines (VMs) and applications to the data center network. There is constant pressure to improve server-based networking performance due to the increased use of server and network virtualization in modern data centers.

Netronome's Agilio® software is dedicated to offloading and accelerating server-based networking. Agilio software and the Agilio family of SmartNICs aim to be a drop-in accelerator for Open vSwitch (OVS). Use cases include compute nodes for IaaS or SaaS, Network Functions Virtualization (NFV), and non-virtualized service nodes, among others. In these use cases it is common to have a large number of network overlays and/or security policies that are enforced on the server. Agilio software focuses on OVS offload because it is the defacto standard for implementing overlays and network-level security policies among many data center operators.

The Agilio software is derived from the OVS code base and preserves all compatible interfaces. The user experience is identical to a stock OVS instantiation from a configuration and management perspective, meaning there is no difference in interaction between an Agil-

io-accelerated OVS and a non-accelerated OVS with exception to system performance. The remaining sections of this whitepaper go into the details of how the Agilio software interacts with OVS on the host and OVS on the Agilio SmartNIC.

AGILIO SOFTWARE ARCHITECTURE

THE CORE OF THE AGILIO SOFTWARE WHICH PROVIDES ALL OF THE SYSTEM-LEVEL BENEFITS, IS THE ACCELERATED DATAPATH RUNNING ON THE AGILIO SMARTNIC.

The core of the Agilio software which provides all of the system-level benefits is the accelerated datapath running on the Agilio adapter. Once a flow is offloaded to the Agilio SmartNIC the expected OVS match and action processing are completely contained to the SmartNIC. This offload accelerates complex packet processing and preserves CPU cycles, allowing more VMs to be deployed on the server.

Several components make up the accelerated datapath, the most critical being the exact match flow tracker, repeated hash tables, and PCIe drivers. The exact match flow tracker is what provides the bulk of the acceleration inside of the system. This table is a cache of the flow entries (or microflow) being processed by the system. It is described in more detail in an upcoming section. The repeated hashing table on the Agilio SmartNIC is merely a synchronized copy of the table in the OVS kernel datapath. Maintenance operations (e.g. timeout processing) on this table are the responsibility of the OVS kernel datapath. The OVS kernel datapath interacts with the Agilio software to, for example, remove flow entries, aggregate statistics, or obtain timestamp data.

The PCIe drivers are responsible for packet transmit and receive to/from the host and VMs. Agilio's PCIe drivers allow the server adapter SR-IOV Virtual Functions (VFs) to be used in conjunction with the OVS datapath, something that is not possible without a SmartNIC. Each VF on the SmartNIC is represented as a virtual port on the vSwitch. For example, when a packet is received on a physical network port and processed by the Agilio SmartNIC, an action for that packet may be to transmit on a virtual port where that virtual port is mapped to a VF. This would allow a VM to receive traffic on a VF. The opposite traffic direction applies as well. When a VM wants to send a packet, it transmits the packet on a VF. When a packet is received on a VF and therefore a virtual port on the Agilio accelerated vSwitch, the flow lookups and actions are executed on the packet (e.g. Entunnel in VXLAN).

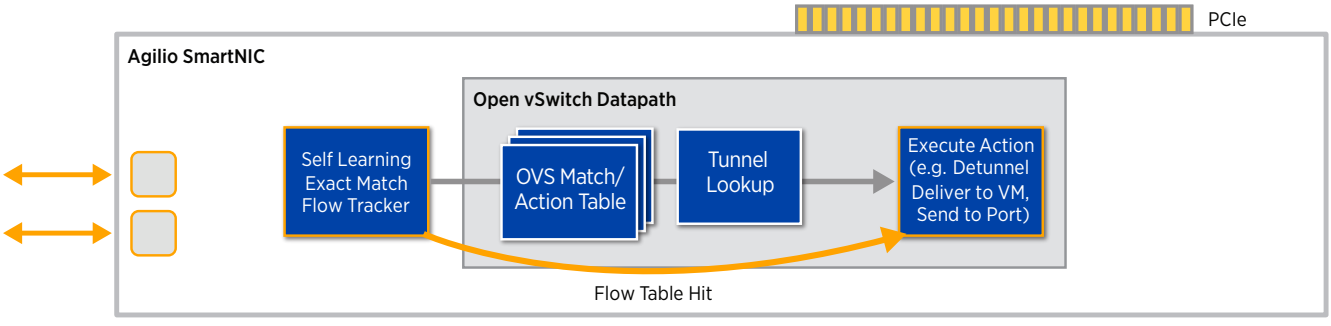

Figure 1. Agilio Accelerated Datapath Workflow

The Agilio accelerated datapath receives its configuration from the kernel-based OVS datapath. The OVS kernel datapath in turn receives its configuration from the OVS user space agent. These software components remain unchanged within the Agilio software that provides

key benefits such as backwards compatibility, full OVS feature support, and ease of use. The interaction between the Agilio accelerated datapath and the host-based OVS software are described in the next section.

CONFIGURATION INTERFACES AND HOOKS FOR TRANSPARENT OFFLOAD

Before describing the details of how Agilio offload operates, it is important to understand the inner-workings of OVS at a high level. OVS contains a user space-based agent and a kernel-based datapath. The user space agent is responsible for switch configuration and flow table population. It is broken up into three fundamental components: ovs-vswitchd, ovsdb-server, and a configuration state database. The user space agent can accept configuration via a local command line (e.g. ovs-vsctl) and from a remote controller. When using a remote controller it is common to use OVSDB to interact with ovsdb-server for vSwitch configuration. To program flow entries into ovs-vswitchd from a remote controller, Openflow is a commonly used protocol. The actual forwarding and data plane processing of flows is done in the kernel-based datapath via a series of repeated hashing tables. The kernel-based datapath acts as a local cache for the vSwitch. As the user space agent processes flows, they are cached in the kernel datapath such that subsequent packets of flow can take the kernel "fast path".

The kernel-based datapath in OVS is where the Agilio offload hooks are inserted. The adapter-based or kernel-based datapaths will not process new flows entering the system because the flow entries have not yet been learned. As a result, the user space agent processes new flows. The user space agent updates the repeated hashing tables in the OVS kernel datapath. The Agilio offload hooks in the kernel become aware of the update and ensures that the same flow entry is applied to the Agilio repeated hashing tables on the server adapter by sending a control message to the accelerated datapath over PCIe. At this point, the next packet in the flow will be processed on the Agilio SmartNIC.

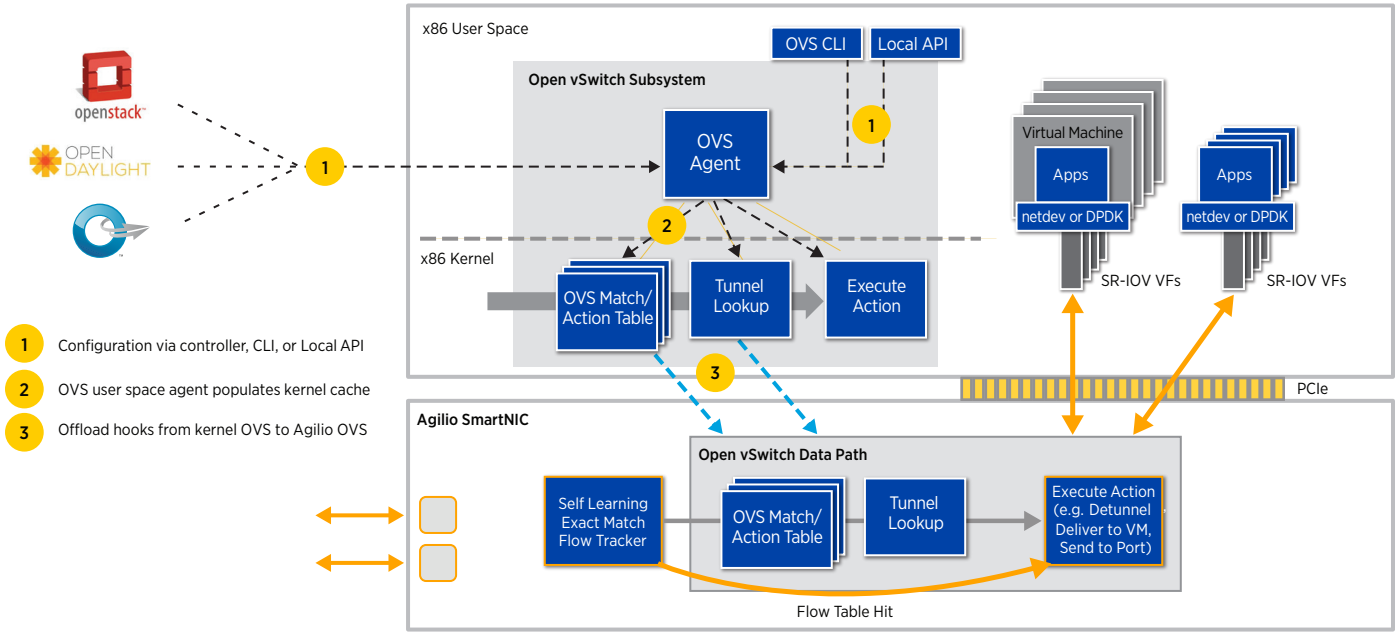

Figure 2. Configuration of OVS and Agilio Datapaths

ACCELERATION THROUGH EXACT MATCH FLOW TRACKER

THE AGILIO SOFTWARE ON THE SMARTNIC INCLUDES AN EXACT MATCH FLOW TRACKER THAT TRACKS EACH FLOW (OR MICROFLOW) PASSING THROUGH THE SYSTEM.

The Agilio software on the SmartNIC includes an exact match flow tracker that tracks each flow (or microflow) passing through the system. The ingress port as well as all packet header fields supported by the system identifies the flow. The exact match flow tracker improves performance relative to the repeated hashing tables because only one flow lookup is required, whereas a number of repeated hashing table lookups, each with a different mask, are typically required. At a high level, the exact match flow tracker could therefore be considered a L1 cache, the repeated hashing tables on the Agilio SmartNIC being a L2 cache, the OVS kernel datapath being a L3 cache, and the OVS user space components supplying the item being cached.

When a packet enters the system, a lookup against the exact match flow tracker is executed. Should an entry not be found, an entry is added. For new flows, or until the policy associated with the flow becomes known, the repeated hashing tables on the Agilio SmartNIC are used. Should a match be found as a result of repeated hashing, the policy associated with the exact match flow entry will be updated and all remaining packets of the flow will be processed using the exact match flow entry. Should repeated hashing not result in a match, the packet is sent to the x86 for OVS kernel and possibly user space processing.

This following diagram illustrates the flow learning hierarchy within Agilio when the SmartNIC receives packets:

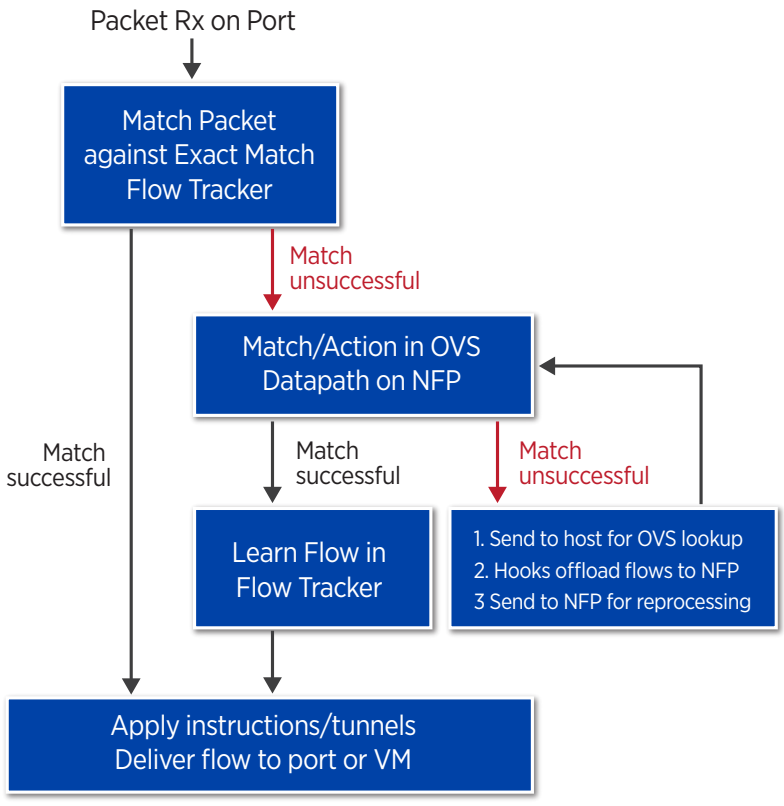

Figure 3. Flow Diagram Illustrating How Flows are Programmed Into the Datapath

TUNNEL DECAPSULATION AND ENCAPSULATION

Tunnel decapsulation involves removing the tunnel header for encapsulations such as VXLAN, NVGRE or GENEVE. This is typical for packets received on the adapter and destined to VMs in a virtualized network environment. The Agilio accelerated datapath includes a component that handles detunneling on ingress of a physical network port. Once the tunneled packet is decapsulated, the remainder of the classification and action pipeline operates as described in the previous sections. The only difference is that the ingress port for the packet appears to be the logical port associated with the tunnel and not the physical network port.

Entunneling is performed if the egress port set for the flow action is a logical port associated with a tunnel. This results in appropriate L2/L3 headers and tunnel specific headers being prepended to the original frame. This is typical for packets sourced from a VM and to be transmitted on the network, but require some level of tunneling for a virtualized network environment. To enable acceleration on the Agilio SmartNIC, Linux kernel data structures associated with network interfaces and tunnels are continually synchronized with corresponding data structures in the Agilio software.

DETAILS OF FALLBACK TRAFFIC PATH

The aim for the Agilio software is to have the majority of server traffic (or ideally, all of server traffic) processed on the SmartNIC. However, there are some conditions in which traffic is not handled on the SmartNIC and is therefore forwarded to the host. This is referred to as the fallback path. When traffic takes the fallback path, packets are processed by the host-based OVS: first the OVS kernel datapath, then if there is no flow match the user space agent processes the flow and caches the entry in the kernel datapath.

From a system design perspective, this is beneficial for two primary reasons. First by using the host to learn flows, the system can inherit the OVS control and management plane configuration methods which preserves pre-existing network orchestration and controller setup. And second, any traffic that cannot be processed by the Agilio SmartNIC will by default be processed by the host, ensuring that all network traffic types are supported.

There are a number of scenarios by which the fallback traffic path can be used. They are as follows:

1. Traffic for which the exact match flow tracker and/or repeated hashing table entries on the Agilio SmartNIC have not been created yet.
2. Traffic destined to a MAC / IP address which needs to be handled on the x86, for example, control plane traffic.
3. Traffic which the Agilio SmartNIC cannot process either due to classification or action processing, for example, a custom or proprietary protocol unknown by the Agilio software.

FUTURE OFFLOAD METHODS UNDER RESEARCH

The current Agilio software architecture leverages the OVS kernel datapath. This datapath uses Linux kernel netdevs for packet I/O. The use of OVS entirely in user space via the DPDK Poll Mode Driver (PMD) is a growing deployment method. Netronome is currently investigating how offload hooks can be applied to the user space-based flow cache within OVS. In this

case, the Agilio-accelerated flow table sequence and interaction with host-based OVS would remain the same, the only difference being that OVS flow cache tables in the host reside in user space rather than the Linux kernel. Netronome expects to release the user space offload capability in the near future.

THE AGILIO SOFTWARE OFFERS THE ABILITY TO ACCELERATE THE OVS DATAPATH BY USING A SMARTNIC IN A TRANSPARENT WAY THAT PRESERVES PRE-EXISTING CONFIGURATION INTERFACES.

An additional offload method that is gaining interest is explicit offload, where the initial classification rules are programmed directly to the Agilio-accelerated datapath on the server adapter. This method eliminates the need for the fallback traffic path in the case of new flow arrival. The fallback path is still desired for exception or control plane traffic in this case. An immediate benefit for this style of offload is a drastically improved flow setup rate, which is important in all use cases but more so in NFV and service node use cases. Netronome is also investigating how this capability can be incorporated into the Agilio software.

CONCLUSION

The Agilio software offers the ability to accelerate the OVS datapath by using a SmartNIC in a transparent way that preserves pre-existing configuration interfaces. By inheriting the OVS control plane, the same controller and orchestration systems can be used to program accelerated and non-accelerated systems in an identical manner. The only difference being system performance. Once a flow is offloaded to the Agilio SmartNIC, the expected OVS match and action processing are completely contained to the SmartNIC. The vSwitch performance is drastically improved and VMs can leverage the SR-IOV interface from the SmartNIC for increased packet I/O rates. As a result, the Agilio offload simultaneously increases the servers PPS and bandwidth capacity while preserving several CPU cycles, allowing more VMs to be deployed on the server.

NETRONOME

Netronome Systems, Inc.
2903 Bunker Hill Lane, Suite 150 Santa Clara, CA 95054
Tel: 408.496.0022 | Fax: 408.586.0002
www.netronome.com