Cisco Virtual Routers, CSR 1000V and ISRv

The Impact of Configuration Changes
On Throughput Performance
An Independent Assessment

DR161111E
November 2016

Miercom
www.miercom.com

Contents

1 - Executive Summary

Miercom was engaged by Cisco Systems to independently validate the performance of its software-based Cloud Services Router 1000V, designed to run in a third-party-managed cloud environment and allow an enterprise to extend its network security and services into the cloud. CSR 1000V also allows service and cloud providers to offer innovative cloud-based networking services. In addition to testing CSR 1000V, we also tested a variant form of the IOS XE router software – the Virtual Integrated Services Router, or ISRv, in an NFVIS (Network Functions Virtualization Infrastructure Software) environment running on an Enterprise Network Compute System (ENCS) Server.

The testing assessed the impact of various configuration settings and deployment options on the throughput performance of the CSR 1000V package in public and private cloud environments.

Key Findings and Observations:

- We found that the CSR 1000V virtual router delivers the same IOS XE features as advanced Cisco physical routers, like the ASR 1000, except for support for some outdated legacy technologies – like Frame Relay, ATM and ISDN.
- Using just one or two virtual CPUs (vCPUs) per virtual machine (VM), CSR 1000V delivers outstanding throughput performance – up to physical limit of 20 Gbps on x86 Server with two 10 GE ports NIC and up to 5 Gbps on Amazon Web Services.
- The latest CSR 1000V software we tested, IOS XE release 16.3.1a, delivers up to 40 percent better throughput than the previous version 3.17.
- Benefits to using CSR 1000V became apparent in all scenarios tested – including quick implementation, flexibility, hypervisor agnostic, and IOS XE’s familiar configuration and diagnostics tools.
- Unlike classic routers, where hardware and software is optimized out of the box, a CSR 1000V setup has to be configured for optimal performance on several levels – the BIOS, the host operating system, and the Hypervisor/virtual machine settings. CSR 1000V supports major IO technologies like SR-IOV, fd.io VPP, OVS-DPDK as vSwitches in order to increase network performance.
- Horizontal scaling – load-balancing traffic across multiple VMs, each with two vCPUs – can deliver double the throughput of a single VM with up to eight vCPUs.

Based on the results of the testing, we proudly award the ***Miercom Performance Verified Certification*** to Cisco’s Cloud Services Router 1000V.

Robert Smithers
CEO
Miercom

2 - Product Overview

More and more enterprises are deploying their IT applications in the cloud – virtualized data centers built and managed by third-party service providers. Since the enterprise doesn't own the cloud resources, or even its connectivity with the cloud, a way was needed to allow enterprises to deploy their own router functionality within the cloud. For this specific environment – control of a single-tenant router in a multi-tenant cloud – Cisco now offers the Cloud Services Router 1000V.

The Cisco CSR 1000V is called a virtual-form-factor router. It is a set of specialized licensed software that runs on selected x86 hardware platforms. The hardware/software combination fully emulates an advanced Cisco router. As the name implies, the Cloud Service Router 1000V, which loads and runs with Cisco IOS XE within the cloud, provides enterprise connectivity, network services and network security. These include: WAN gateway, routing; addressing DNS, DHCP, and NAT; secure VPN gateway and IPsec; MPLS support; firewall/ACLs; and high-availability configurations. In short, the package lets the enterprise fully extend its network configuration into the cloud.

Cloud providers themselves can also offer enterprise-class networking services to their tenants or customers with the Cisco Cloud Service Router package. CSR 1000V lets service providers support various scenarios, including virtual Provider Edge (vPE), virtual Customer Premises Equipment (vCPE), which lets carriers deploy or add customer services remotely, virtual Broadband Network Gateway (vBNG) and virtual Route Reflector (vRR).

Since the software-based router runs in the cloud, the enterprise customer can gain the same level of privacy and security for its cloud deployment as it does for its own premises and private network. Without it, enterprises usually must route – or backhaul – all of their user connections to the cloud through an enterprise data center, adding cost and complexity and using up considerable additional resources and bandwidth.

Hardware Requirements

The Cisco CSR 1000V router consists of a set of downloadable software modules. The same software runs on any of the supported cloud hardware platforms and Hypervisor environments. The minimum requirements for this software and the underlying IOS XE operating system are: 1 vCPUs, 4 GB RAM and 8 GB hard-drive space.

As discussed in this report, performance and throughput can vary significantly depending on the platform and how it is configured. There are some 100 parameters among the BIOS settings, the operating system and Hypervisor configuration that can affect the performance of this software-based router. Cisco offers detailed instructions on how to properly configure each supported platform and Hypervisor to maximize performance and throughput.

The below table lists the current hardware platforms that have been thoroughly tested and fully support the CSR 1000V.

Vendor	Servers Tested for Compatibility
Cisco	All Cisco UCS servers, including the UCS B230 M2, UCS C220 M3, UCS C210 M2, UCS C200 M2, UCS B22 M3, UCS C240
HP	HP ProLiant DL180G6
Dell	Dell R720 with Xeon E5-2660

Note: The minimum clock rate supported is 1.9 GHz.

Supported OS and Hypervisor Environments

Current IOS XE software releases for the Cisco CSR 1000V are IOS XE 3.17 and Denali 16.3.1a. Generally, Cisco CSR 1000V is supported on selected Type 1 (native, bare metal) Hypervisors. CSR 1000V is not supported on Type 2 (hosted) Hypervisors, including VMware Fusion, VMware Player or Virtual Box.

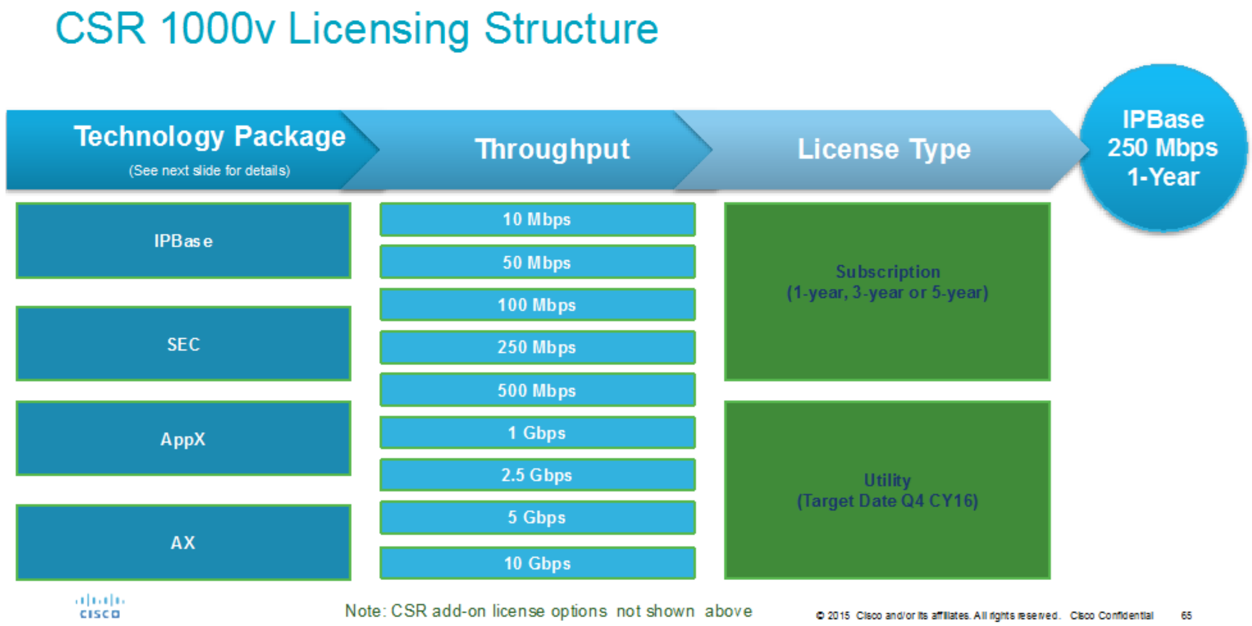
The below table summarizes supported Hypervisors.

Hypervisor environments supported by Cisco CSR 1000V

Hypervisor	Version(s)
VMware ESXi	Server 6.0 rebuild 2 recommended; Server 5.5 supported
KVM (Kernel-based Virtual Machine)	RHEL (Red Hat Enterprise Linux) 7.2 recommended; 7.1 supported
Citrix XenServer	6.5 recommended
Microsoft Hyper-V	Windows Server 2012-R2 Hyper-V Manager 6.3 recommended
AWS (Amazon Web Services)	C4 and T2 instance types recommended
Microsoft Azure	Standard D2 and Standard D3 recommended Standard D4 supported

For more detail on Hypervisor support see:
http://www.cisco.com/c/en/us/td/docs/routers/csr1000/software/configuration/csr1000Vswcfg/csroverview.html#22356

Licensing



The CSR 1000V is licensed based on a combination of throughput and feature set. The package can be leased for a term of 1, 3 or 5 years or on a permanent, perpetual license.

Four packages are offered, each with a range of maximum throughputs. They are:

- **IP Base**: includes routing, addressing, basic security ACLs, high availability and full management. Maximum supported throughput levels for the IP Base package are: 10, 25, 50, 100, 250, or 500 Mbps; or 1, 2.5, 5, or 10 Gbps
- **Security**: Includes all the IP Base features plus advanced security – including zone-based firewall, IPsec and route-based VPNs. Maximum supported throughput levels for the Security package are: 10, 25, 50, 100, 250, or 500 Mbps; or 1, 2.5 or 5 Gbps
- **AppX**: Includes all the Security features plus advanced networking, application experience, and hybrid-cloud connectivity features. Maximum supported throughput levels for the Appx package are: 10, 25, 50, 100, 250, or 500 Mbps; or 1, 2.5 or 5 Gbps
- **AX**: Includes all available features. Maximum supported throughput levels for the AX package are: 10, 25, 50, 100, 250, or 500 Mbps; or 1, or 2.5 Gbps.

The license is rate-limited to the maximum throughout. Evaluation licenses are also available.

Quick Deployment: ISRv on ENCS

In addition, Cisco offers the same CSR 1000V software code specially packaged on an Enterprise Network Compute System (ENCS) platform 5400. We conducted some throughput testing on this platform, what Cisco calls the Integrated Services Virtual Router (ISRv). The ISRv is used as a component of Cisco's Network Function Virtualization Infrastructure Software (NFVIS) solution, which distills Enterprise network functions into software, making it possible to deploy services quickly: The Hypervisor is pre-loaded. Then the configuration is pushed on via the GUI interface. The platform is powered up and can begin working immediately.

On the ISRv we were able to achieve wire-speed throughput (4 Gbps, completely filling two Gigabit Ethernet links in both directions) on the ENCS platform 5400 Series Server with four virtual CPUs (vCPUs), and 2.5 Gbps using two vCPUs. Most testing for this report, however, was conducted with the same IOS XE software (CSR 1000V) running on a Cisco UCS C240 M4.

3 - Test Bed: How We Did It

The purpose of this testing was to determine the performance characteristics of the CSR 1000V in both a private-cloud (x86 Server) and a public-cloud environment (Amazon Web Services)for various customer environments and operational settings. Performance was compared while running:

- Cisco Express Forwarding (CEF) – a basic IPv4 forwarding used as a baseline for performance characterization
- Different IO technologies: fd.io Vector Packet Processing (VPP), OVS-DPDK and Single Root I/O Virtualization (SR-IOV)
- Internet Protocol Security (IPsec) tunneling
- Single and multiple virtual CPUs (vCPUs) on a single VM
- Virtual Broadband Network Gateway (vBNG) with 500 Kbps and 1 Mbps per PPP Session
- Virtual Integrated Services Router (ISRv) running as a part of Network Functions Virtualization Infrastructure Software on ENCS 5400 Series Server
- Horizontal Scaling, running the same software on multiple VMs.

The Private Cloud environment we created consisted of several CSR 1000V nodes installed on kernel-based virtual machines (KVMs). A KVM is a virtualization infrastructure for the Linux kernel that turns it into a Hypervisor. It was merged into the Linux kernel mainline in version 2.6.20, released in 2007. KVM runs on and requires a processor with hardware virtualization extensions.

During the initial tests, just basic fine tuning was performed. It was not the goal of the tests to achieve the absolute best performance of a single VM, which would have entailed top speed CPU, special engineering images with tweaked buffer sizes, special code changes, etc. Rather, we took publicly available, stable images and current versions of all software to characterize real-world performance for various customer operational environments.

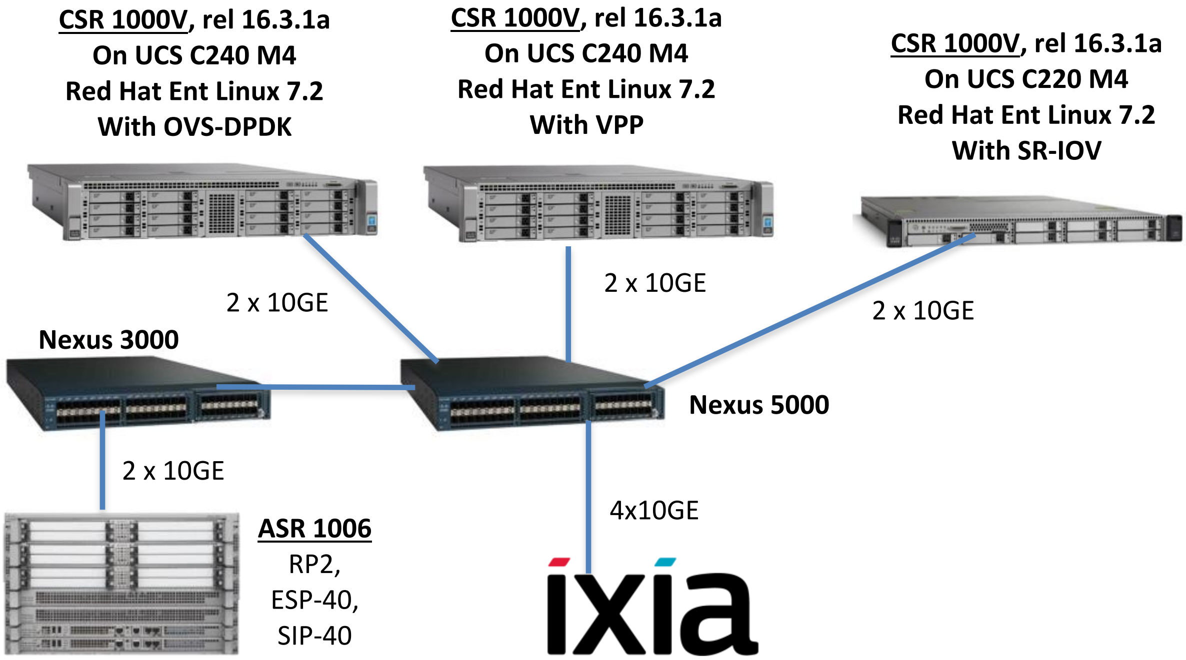
The tests performed here were designed to give an initial indication of the performance that can be achieved for some representative configurations. Where further optimization is needed, details are provided in the Cisco online documentation. Users can also contact Cisco support directly.

Most testing was done with the IOS XE 16.3.1a version of the CSR 1000V software. A few tests were run with a forthcoming, pre-released software version, to see how performance compared.

In planning the tests, we identified the primary uses for the CSR 1000V package. These include: as a highly secure VPN gateway, a Multi-Protocol Label-Switched (MPLS) WAN termination, a data-center network extension, a control and traffic redirector, and service provider roles including virtual Provider Edge (vPE), virtual Customer Premises Equipment (vCPE), which lets carriers deploy or add customer services remotely, virtual Broadband Network Gateway (vBNG) and virtual Route Reflector (vRR).

The diagram below depicts the private cloud environment we created for the testing

Private Cloud Test Bed: Featuring KVMs and Ixia Traffic Generator



This test bed was used for all test cases on the Private Cloud. Each CSR1000V was configured for a different test in order to save configuration time.

UCS (Unified Computing System) server details

Server Platforms	Two Cisco C240 M4 and one C220 M4 Servers with identical hardware configuration
Processor	2 x Intel® Xeon Processor E5-2699 v3 @ 2.30GHz with 18 cores each; 36 CPUs total
Memory	328 GB RAM Total
NIC used	1 x Intel® X520-DA2 Adapter (Total: 2 or 4 x 10-GbE ports), pinned to the same NUMA Socket as CSR 1000V VM and VPP/OVS-DPDK
BIOS	C220M4.2.0.10c.0.03232016081

Hypervisor and software environment

Host OS	Red Hat Enterprise Linux Server release 7.2 (Maipo) LSB Version: core-4.1-amd64:core-4.1-noarch
CSR 1000V version	IOS XE RLS **16.3.1a** Image
Hypervisor	Compiled against library: libvirt 1.2.17 Using API: QEMU 1.2.17 Running Hypervisor: QEMU 2.3.0

CPU speed. We ran most tests at the nominal CPU speed, 2.3 GHz, and saw better performance with intel_ state=disabled. Later, we ran a few additional tests on an x86 server with a 2.6 GHz CPU and saw some performance improvement with the faster CPU. All our test configurations used CPU pinning to designate CPUs for command and data and 1 GB hugepages to better handle the data operations.

IO Test Software: The performance application, OVS-DPDK (Intel's Data Plane Development Kit) was installed. DPDK support, which speeds performance and reduces latency, has been available in OVS since version 2.2. Denali IOS XE Software 16.3 images and later have DPDK integrated.

BIOS settings were changed for particular tests using command: host# **show bios/advanced**

These BIOS set-up parameters: Enhanced Intel Speedstep Tec: Enabled

Intel Turbo Boost Technology: Enabled

Hyper-threading: Disabled

Max license used: A Cisco-internal CSR 1000V License for 200 Gbps max throughput was used in order to avoid license enforcement drops and constrain throughput. This is the max-size license, only available for internal use, to allow "unlimited" bandwidth for our testing so that a limited license would not constrain the bandwidth.

Nexus switches: No special features were enabled on either the Nexus 5000 or 3000 switches in the test bed. Both were simply Layer-2 switches. The Nexus 5000, which ran version 7.0(5) software, provided 10 Gbps wire-speed capacity from the Ixia tester to the UCS servers. The Nexus 3000, which ran version 6.0(2) software, provided 10 Gbps wire-speed capacity to the ASR 1006 router and, via a trunk port, to the Nexus 5000.

Ixia tester: For test-traffic generation and to calculate throughput, an Ixia N2X server, running the IxServer 6.80.1100.12 application, was used for all the Cisco internal-cloud, KVM-based testing, along with Ixia client software version 7.40.1075.27. Much of the throughput testing and calculation was done using the RFC 2544 Quick Test on the Ixia.

VPP: v16.06 Release as rpm package with two worker threads was used. Worker threads were pinned to the same socket, where CSR1k vCPUs and 10 GE NIC interfaces were assigned. VPP was used as pure layer 2 switch, no additional features like routing, no modifications or patches to VPP was done. Rx and Tx Buffer size was set to 1024.

OVS-DPDK: 2.5.0 Release as rpm package with two PMD (Poll Mode Driver) threads was used. PMD threads were pinned to the same socket, where CSR1k vCPUs and dpdk 10 GE NIC interfaces were assigned. OVS-DPDK was used as pure layer 2 switch, no additional features, no modifications or patches to OVS-DPDK was done. Buffer size was set to 1024. The goal was to have the same configuration parameters as VPP.

NDR vs PDR: The Packet Drop Rate – PDR – is a setting for the percent of packets that can be dropped due to internal router, vSwitch (VPP/OVS-DKPD), Hypervisor or other issues. There are several likely places where packet drops can occur, and drops can happen simultaneously in more than one place.

For most of our throughput tests we applied a 0.01 percent Packet Drop Rate.

Alternately, the CSR 1000V can be set to a No Drop Rate – NDR. This is the maximum rate that the router will process packets with none dropped. Generally, limiting throughput to NDR will improve individual packet reliability, but at the expense of overall performance.

Identifying Bottlenecks: in complex virtual scenarios several bottlenecks are possible: CSR 1000V VM itself, virtual Switch, Hypervisor, Network Card, public cloud limitations, etc. Most critical / common bottlenecks are CPU and IO Limits. As a generic rule, if we see CPU utilization below 30%, then we consider IO or other factors to be the bottle neck.

RFC 2544: The industry-standard Request For Comments (RFC) 2544 was used for all the tests in this report as the basis for calculating bi-directional throughput. This standard has long been used to calculate throughput over a link connecting two devices.

In our case, the RFC 2544 throughput test determines the maximum rate at which the Cisco CSR 1000V can send and receive traffic over a connecting link. Layer-2 frames of a specific size are sent at a specified rate, which is then continually stepped up in subsequent iterations, using a binary search algorithm, until the maximum rate at which the switch forwards data without losing frames is determined. Most of our tests were conducted twice, with two different packet sizes: the IMIX, and maximum-sized, 1,518-byte.

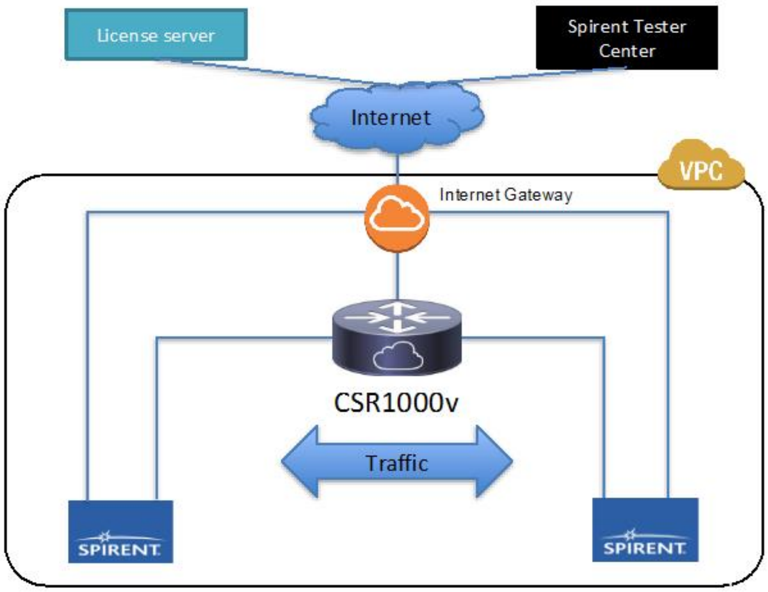
We set up IP forwarding, using Cisco Expedited Forwarding (CEF), and obtained baseline throughput results.

Test Traffic: As our results confirmed, maximum throughput is achieved with large frame sizes. This is because packet overhead is minimized when sending maximum-sized frames (i.e., 1518 bytes). The amount of bandwidth that is consumed by overhead, which reduces throughput, is considerable when there are many small frames (i.e., 64 bytes).

We also ran all tests using bi-directional **IMIX** traffic, which better represents realistic customer loads. IMIX traffic generation is performed the same way by both Ixia and Spirent test systems. IMIX consists of three packet sizes: 64, 576, and 1,500 bytes, with a distribution weight of 7, 4 and 1, respectively. In other words, for every large (1,500-byte) packet sent, seven small (64-byte) and four medium-sized (576-byte) are sent.

100 hosts per Interface: In cases where the test called for more realistic “user” traffic, the test traffic was set up to simulate 100 hosts.

Public Cloud Test Bed (AWS): CSR1000V and Spirent Traffic Generator



CSR 1000V VM details

AWS Instance Type	C4.large (Moderate Network) HVM with Enhanced Networking, 2 hyper-threads (one core) of an E5-2666 v3 @2.9GHz C4.8xlarge (10G Network) HVM with Enhanced Networking, 2xE5-2666 v3 @2.9GHz*, Placement Groups were not used.
IOS XE Image	IOS-XE 16.3.1a Image
Hyper threading	On

*C4.8xlarge has 36 vCPU, but CSR only uses up to 8 vCPU (physical cores, not hyperthreads)

Spirent VM details

AWS Instance type	C4.8xlarge (10G Network) HVM with Enhanced Networking, 36xE5-2666 v3@2.9GHz
Version	Spirent Anywhere 4.63 on Ubuntu 14.04
VPC	Same availability zone in a single VPC

Spirent tester: For the test bed we created for the Amazon Web Services (AWS) environment, a different tester, from Spirent Communications, was used. The Ixia and Spirent testers are functionally similar, and all of the throughput tests run by both systems were based on the same RFC 2544 standard. We installed Spirent tester on C4.8xlarge instance that provides 10G Network capability to avoid tester network limits.

Test Cases

The results of our testing are detailed in six sections in this report:

- IPv4 Forwarding as Baseline for performance characterization
- IPv4 Forwarding with Features Added. Use case: feature-rich Enterprise Router
- IPsec-encrypted tunneling. Use case: secure IPSec Gateway
- Virtual Broadband Network Gateway (vBNG). Use case: Service Provider Access Aggregation
- ISRv on Network Functions Virtualization Infrastructure Software (NFVIS)
- Horizontal scaling across multiple virtual machines for better performance

We tested several use cases on private and public clouds. In order to present our results in the optimal form, we will describe private cloud results first, followed by AWS public cloud section.

4 – Private Cloud (KVM) Test Results

4.1 – IPv4 Forwarding

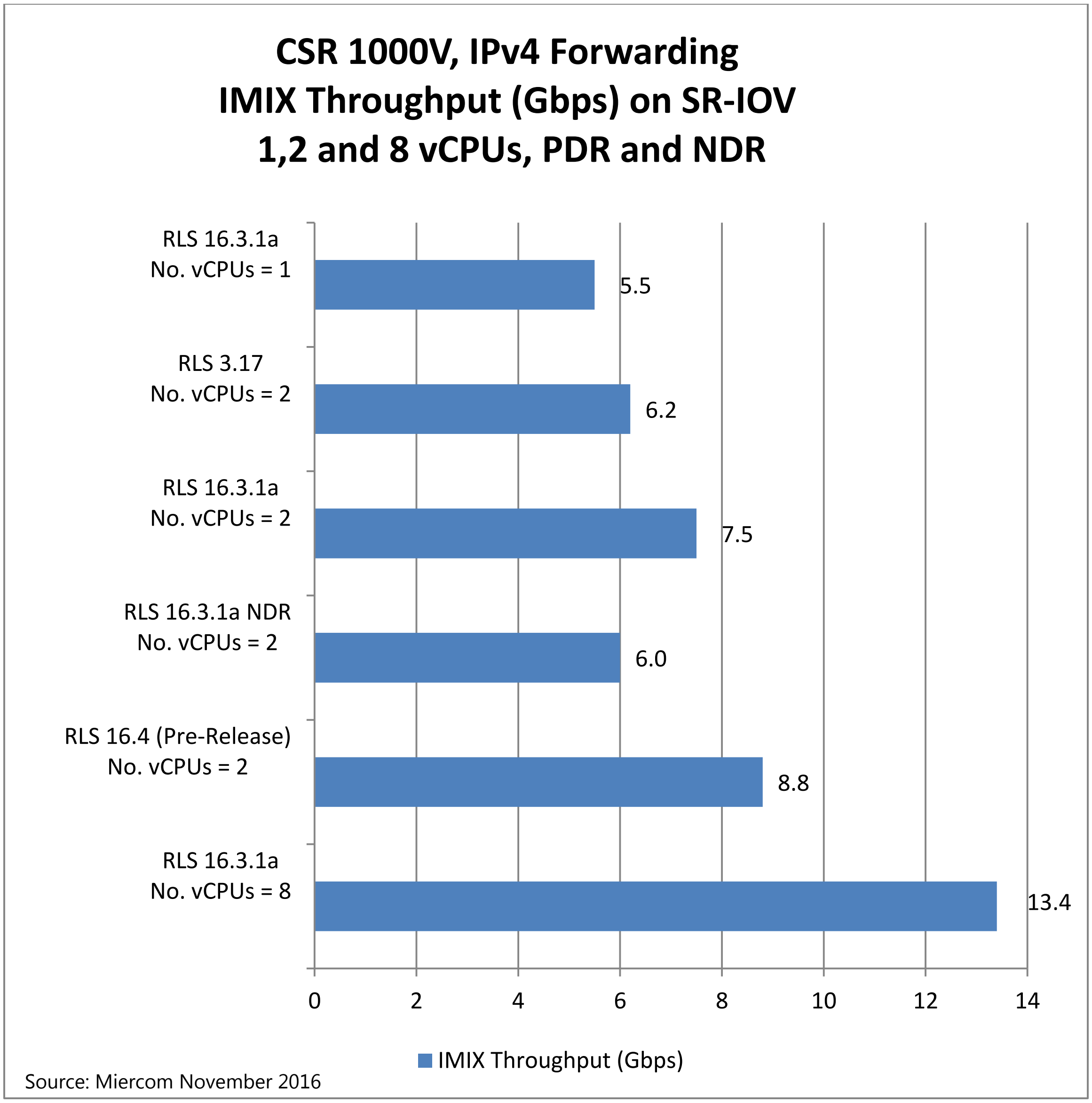
We used basic IPv4 packet forwarding for initial benchmarking. This provided a baseline to use for comparing the results of the different test configurations.

For all the tests in this first group, the CSR 1000V was first configured for basic IP forwarding.

The first set of tests – with CSR 1000V running basic IPv4 forwarding – examined the impact of various configuration and deployment settings, including:

- How performance scales with additional (up to eight) virtual CPUs (vCPUs) on the same virtual machine (VM),
- How performance is impacted by setting NDR (No Drop Rate) or PDR (Partial Drop Rate)
- How performance of the current CSR 1000V IOS XE Release 16.3.1a compares with the previous release (3.17) and a pre-release version of the next version (16.4)

The optimum throughput performances for this first set of variables are shown in the following chart, which shows the results from the tests for IMIX traffic.

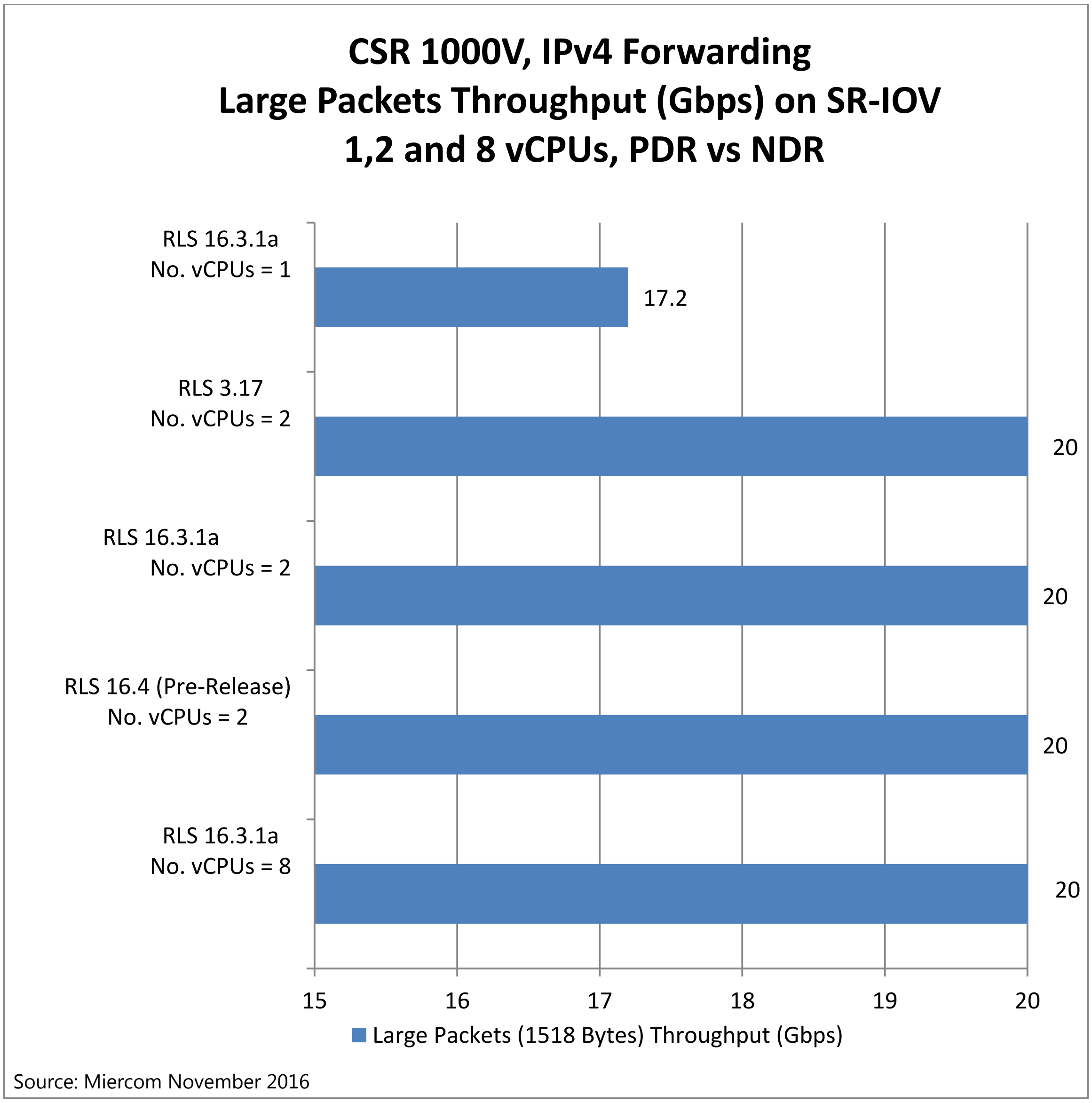


We also saw performance improvement by increasing the number of vCPUs. Where a single vCPU generated 5.5 Gbps of throughput, 13.4 Gbps could be achieved with eight vCPUs.

Please note, that RLS 3.17 does not have DPDK integrated. RLS 16.3.1a and following releases do have DPDK integrated – that's why we saw performance improvement from 6,2 Gbps to 7,5 Gbps for 2 vCPU VM.

Note that we did an extra test run on a pre-release version of CSR 1000V, RLS 16.4, with two vCPUs. Achieving an 8.8 Gbps throughput rate shows the upward performance improvement trend continuing.

Our throughput results show that setting to NDR reduces throughput. With two vCPUs and a PDR setting of 0.01 percent, the CSR 1000V achieved 7.5 Gbps of throughput. However, by setting to a No Drop Rate, throughput dropped to 6.0 Gbps. The same version of CSR 1000V software, RLS 16.3.1a, and configuration was used in the PDR versus NDR tests.



I/O Methods

We then tested different I/O methods to determine their effect on throughput. So using the same configuration – basic IPv4 forwarding with the same features enabled – we ran tests with three common I/O methods:

- SR-IOV: Single Root I/O Virtualization (SR-IOV). SR-IOV to VM is a Direct Memory Access (DMA) of a packet to VM memory. CPU cycles are used only to process packets, so CPU utilization is reduced.
- fd.io VPP: Vector Packet Processing (VPP) as vSwitch. VPP takes packets in groups, creates a routing vector for the group of packets and then processes them all at once. No routing functions on VPP were used in the testes, just standard switching.
- OVS-DPDK: Open vSwitch with DPDK

Selected Settings

We then varied certain key configuration parameters to see the effect on throughput. These included:

- Whether or not to enable the SpeedStep and Turbo Boost BIOS settings
- Whether a higher CPU clock speed impacts throughput.

With and without SpeedStep and Turbo Boost. In this environment we ran CSR 1000V and IOS XE on a KVM (kernel-based virtual machine. Cisco Expedited Forwarding was enabled, but no additional features were configured or enabled. The I/O method used was SR-IOV (Single Root I/O Virtualization) and a single virtual CPU was configured. An IMIX throughput test was run, yielding 4.0 Gbps of throughput.

Then two BIOS system options were enabled: **SpeedStep**, an advanced Intel BIOS setting, and **Turbo Boost**, an Intel setting that overclocks the CPU when heavily loaded.

Then the IMIX traffic test was re-run. With these two settings enabled, throughput increased to 5.2 Gbps, with all other settings and configuration parameters the same (see table). The conclusion: Users can achieve a 30-percent performance boost by turning on SpeedStep and Turbo Boost in the BIOS. These settings improve the CPU speed, but they also then slow down the CPUs to conserve power if extra CPU bandwidth is not needed. This results in less predictable results with wider variation between test runs. For all other testing we ran with these speed-boost and power-conservation features turned off.

	Without SpeedStep and Turbo Boost	With SpeedStep and Turbo Boost	Percent Increase
KVM Throughput	4 Gbps	5.2 Gbps	+ 30 Percent
Number of vCPUs	1	1	

CPU speed increase. We then ran IMIX throughput tests on the Private Cloud with a CPU running at the base 2.3-GHz speed, and again with a different CPU (on a different x86 server) running at the higher 3.2-GHz clock speed, a 37.5-percent increase. All other parameters and settings were the same, and the two BIOS settings – SpeedStep and Turbo Boost – were again disabled.

As the below table shows, throughput increased by 30 percent, from 4.0 to 5.2 Gbps, with the higher CPU speed. The conclusion: Apply a higher-speed CPU setting when you can.

	2.3-GHz CPU	3.2-GHz CPU	Percent Increase
KVM Throughput	4.0 Gbps	5.2 Gbps	**+ 30 Percent**
Number of vCPUs	1	1	
Processor speed	2.3 GHz	3.2 GHz	**+ 37.5 Percent**

4.2 - IPv4 Forwarding, with Features Added

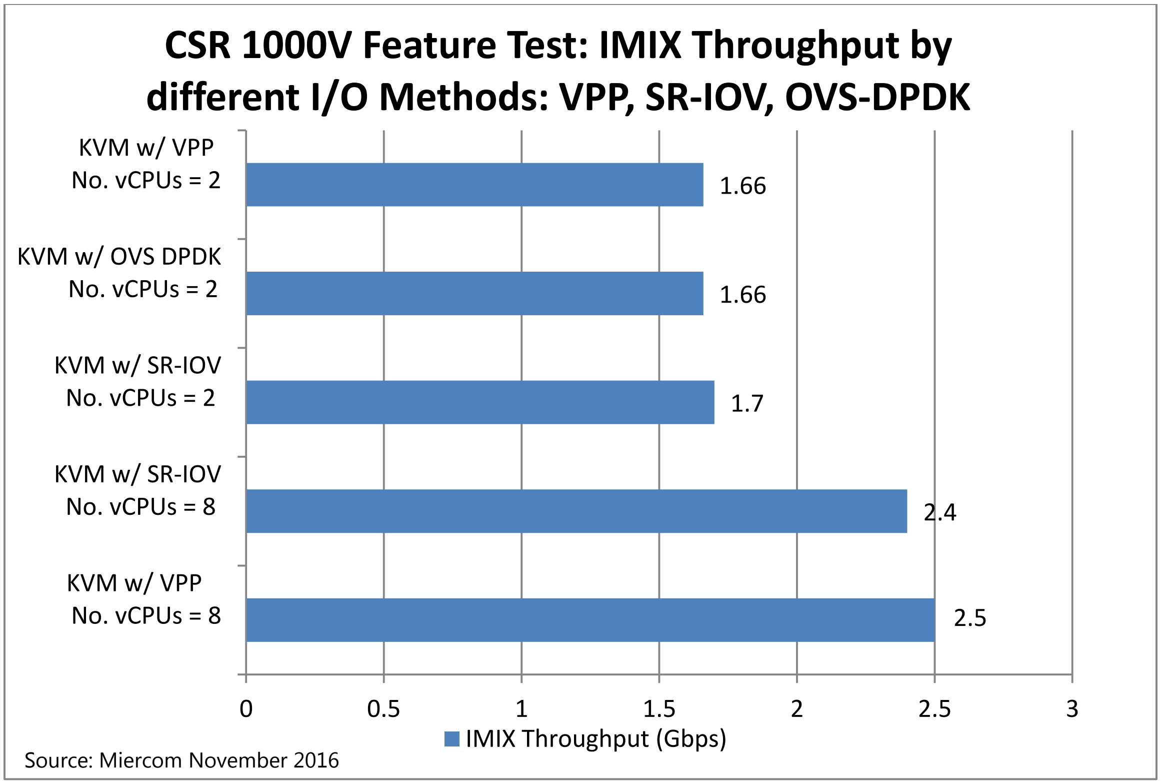
No generic feature combination can cover all use cases: The CSR 1000V running with IOS XE supports more than 3,000 different features. Notwithstanding, we distilled a set of five diverse features, which are popularly employed by customers:

- IPsec Gateway – where the CSR 1000V is set to support encrypted connections
- Hierarchical QoS – where a policy is defined to prioritize traffic
- NBAR (Cisco's Network Based Application Recognition) – which entails a deep packet inspection
- Zone-based Firewall – where the router inspects traffic, though only unidirectionally to avoid firewall blocking
- NAT/PAT – where 100 different port addresses are translated.

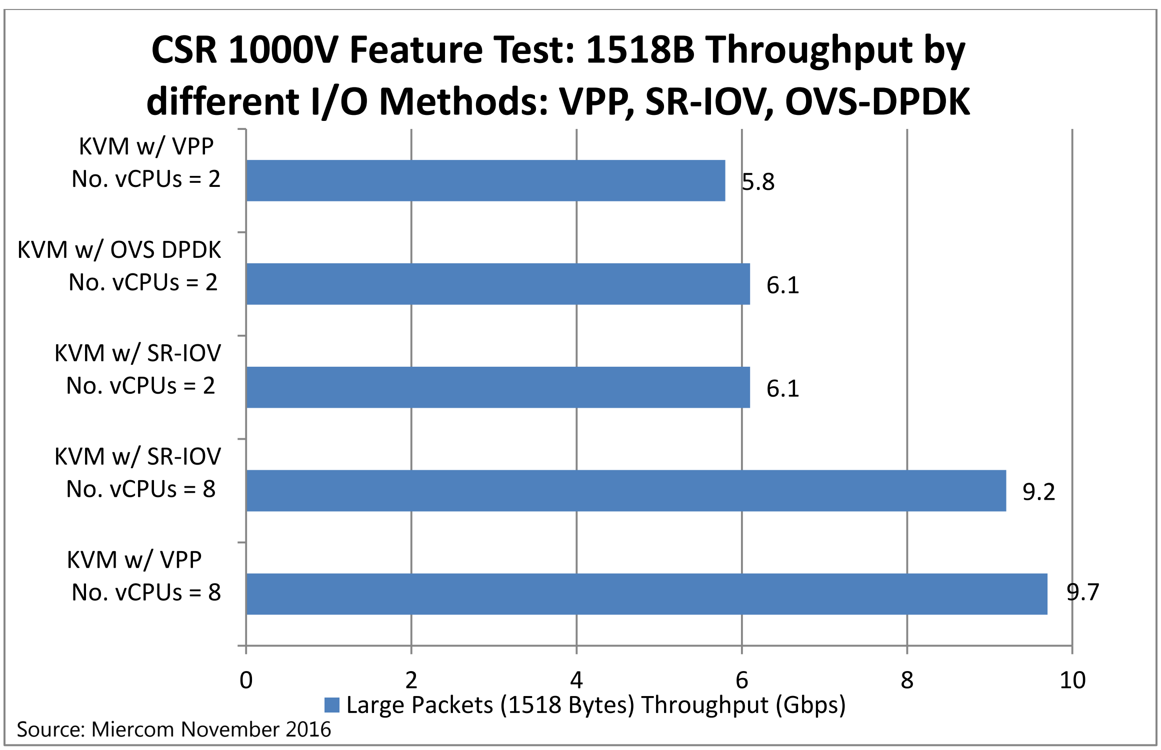
We created a configuration with these features – one that would not cause router drops due to licensing or security issues – and loaded it into the CSR 1000V. The traffic generated for the tests in this section was designed to use the feature pathways at a basic level.

The next chart shows the IMIX throughput results for the different I/O technologies. In the private cloud (the KVM system), tests were done with two vCPUs and RLS 16.3.1a. The configuration using OVS-DPDK achieved the same throughput as the one with VPP I/O, 1.66 Gbps. The configuration with SR-IOV fared slightly better with 1.7 Gbps of throughput.

With eight vCPUs in the private-cloud configuration, VPP achieved 2.5 Gbps of throughput, just slightly better than SR-IOV, which managed just 2.4 Gbps.



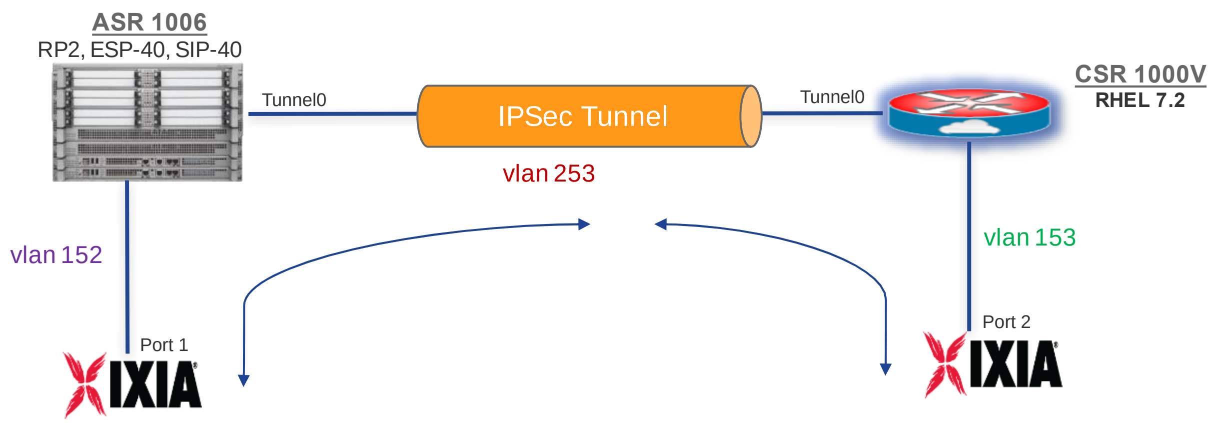
The same tests were done with large packets (1518 Bytes).



4.3 - IPsec (IP Security) Encrypted Tunneling

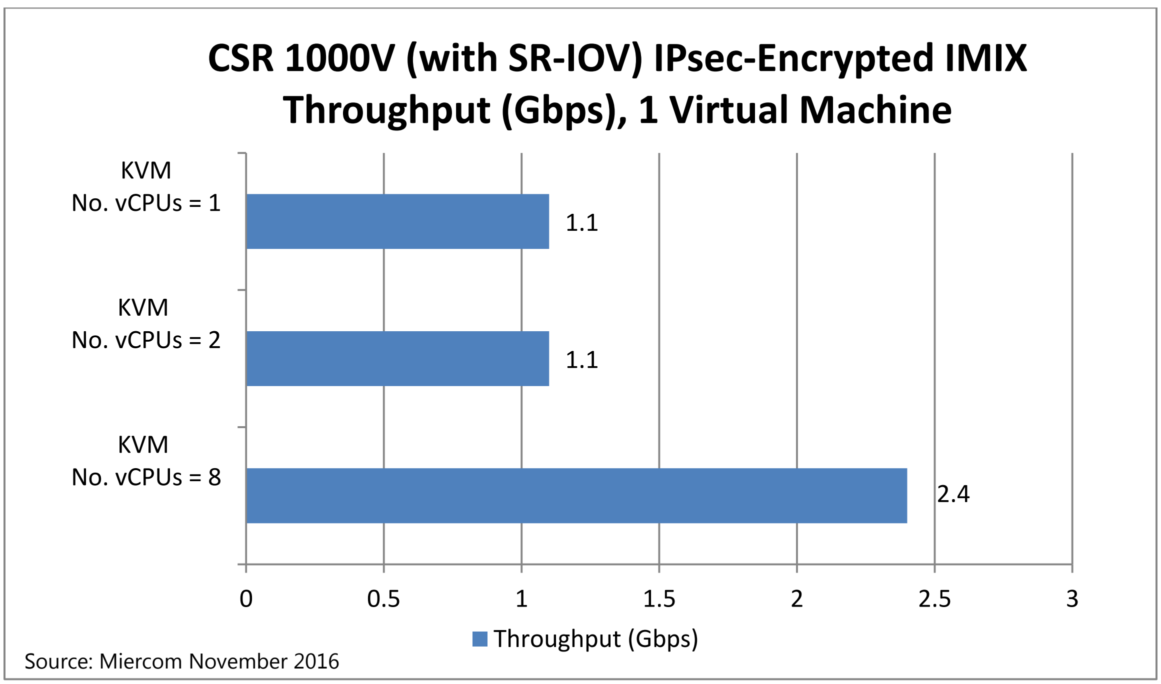
For the next round of testing, the test bed was modified (see the diagram below), this time so that two CSR 1000V KVM nodes were linked over an IP-security (IPsec) tunnel. IPsec processing – where connections are certificate-validated and packets are encrypted – is extremely processor intensive. The following encryption set was used: aes encryption, esp-aes esp-sha-hmac.

Private Cloud IPsec Test Bed



The CSR 1000V ran as a single virtual machine (VM), initially with a single vCPU configured. The I/O method used was SR-IOV (Single Root I/O Virtualization). RFC 2544 Throughput tests were then run.

The results for IMIX traffic are shown in the following chart.



In the private cloud, the throughput with one or two vCPUs configured was the same – 1.1 Gbps. It is because there is little to no control plane processing and in a 1vCPU the control plane and data plane share a single core while in a 2vCPU they have separate cores. With a 1vCPU, when the control plane is not busy then the data plane processing gets all of the cores cycles so it gets ~100% of one core. With a 2vCPU config the data plane still only gets ~100% of one core (independent of the control plane being busy or not). That's why we saw this.

Throughput did increase when up to eight vCPUs were configured. We clocked 2.4 Gbps of throughput with eight vCPUs configured per CSR 1000V KVM in the private-cloud test bed.

4.4 - Virtual Broadband Network Gateway (vBNG)

CSR 1000V supports up to 8,000 dual-stack PPP/IP (Point-to-Point Protocol over IP) sessions. Such connections are key for customer access to broadband services, which the CSR 1000V supports as a virtual Broadband Network Gateway (vBNG).

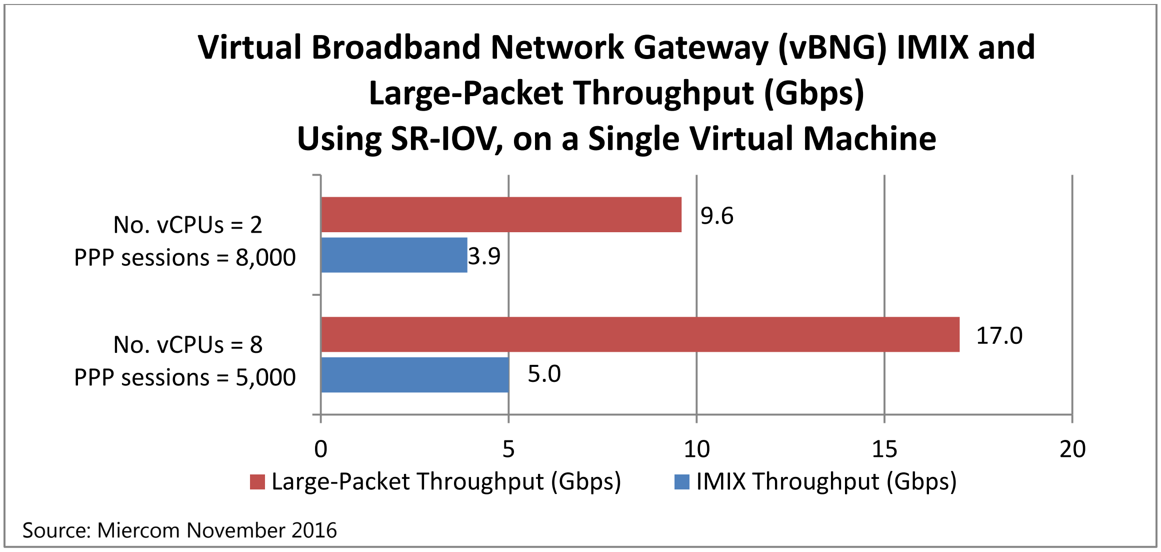
Our testing of the CSR 1000V as a vBNG had the IOS XE 16.3.1a software running on a KVM in a simulated cloud environment. We used the SR-IOV (Single-Root I/O Virtualization) I/O method. Generally, modern I/O technologies like SR-IOV or VPP (Vector Packet Processing) can boost the throughput of a CSR 1000V, as much as doubling it from a typical 2.5 Gbps up to 5 Gbps.

Service providers typically calculate with average bandwidth per household. Even if a household has a connection up to 50 or 100 Mbps, average bandwidth, which is actually consumed is a fraction of it. Typical numbers for average bandwidth per household may vary per region and country, we used two different profiles: 500 kbps and 1 Mbps as average bandwidth.

In order to cover both profiles above, we tested two different configurations – one CSR 1000V VM with:

1. Two virtual CPUs, 8,000 PPP sessions with 500 Kbps per PPP Session
2. Eight vCPUs, 5,000 PPP sessions with 1 Mbps per PPP Session

We tested for RFC 2544-based throughput, where test traffic is sent via PPP sessions and back again. We used local user authentication, in order to exclude RADIUS (an external authentication server) as a bottleneck.
The results, shown in the chart, are the throughputs for IMIX and large-packet test runs. Both test profiles (8,000 PPP Sessions with 500 Kbps and 5,000 Sessions with 1 Mbps) were successfully tested with the IMIX traffic. Large packets (1518 Bytes) shown as expected much better performance. Readers should note that as additional functions are added (such as RADIUS verification and other features), throughput will be subsequently impacted.



4.5 - Integrated Services Virtual Router (ISRv)

In addition to testing CSR 1000V in numerous cases, we also tested a variant form of the IOS XE router software – the Integrated Services Virtual Router, or ISRv, in an NFVIS (Network Functions Virtualization Infrastructure Software) environment.

Generally, NFVIS provides the Linux-based virtualization layer that allows you to easily add VNFs to your network. An integrated hypervisor lets you create and run network functions as virtual appliances using a graphical user interface. Programmable, open APIs allow enhanced applications, such as the ESA app to work in the virtual branch.

The Cisco ISRv is a virtual form-factor Cisco IOS-XE router that delivers comprehensive WAN gateway and network services functions into virtual environments. Using familiar, industry-leading Cisco IOS
XE Software networking capabilities (same features present on Cisco 4000 series, ISR and ASR 1000 series physical routers), the Cisco ISRv enables enterprises to deliver WAN services to their remote locations using Cisco network functions virtualization (NFV) technology.

As underlying hardware in this particular test, we used an Enterprise Network Compute System (ENCS) 5400 Series rather than a UCS C240 (see picture below). The Cisco 5400 ENCS is a line of compute appliances designed for the Cisco Enterprise Network Functions Virtualization (ENFV) solution. It delivers a new standard of software-defined flexibility, and performance, and offers a lower total cost of ownership (TCO). It is a hybrid platform that combines the best attributes of a traditional router and a traditional server, and offers the same functionality with a smaller infrastructure footprint.

The ENCS 5400 system had two gigabit ethernet ports and delivered 2.5 Gbps with two vCPUs, and wire speed (4.0 Gbps) with four vCPUs.

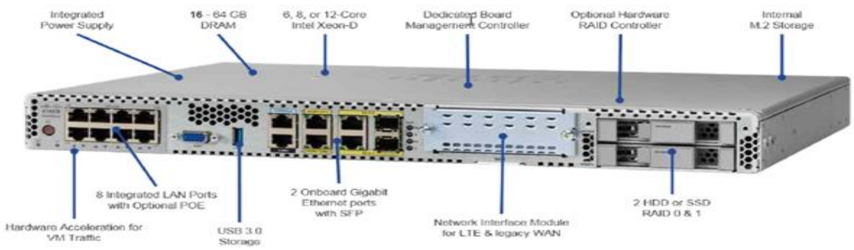
IMIX Throughput for Network Functions Virtualization Infrastructure Software (NFVIS)

With ISRv (Integrated Services Virtual Router) on ENCS (a router form-factor server)

No. of virtual CPUs	Throughput (Gbps)
2	2.5
4	4.0*

* Represents full bi-directional line rate for the two Gigabit Ethernet links

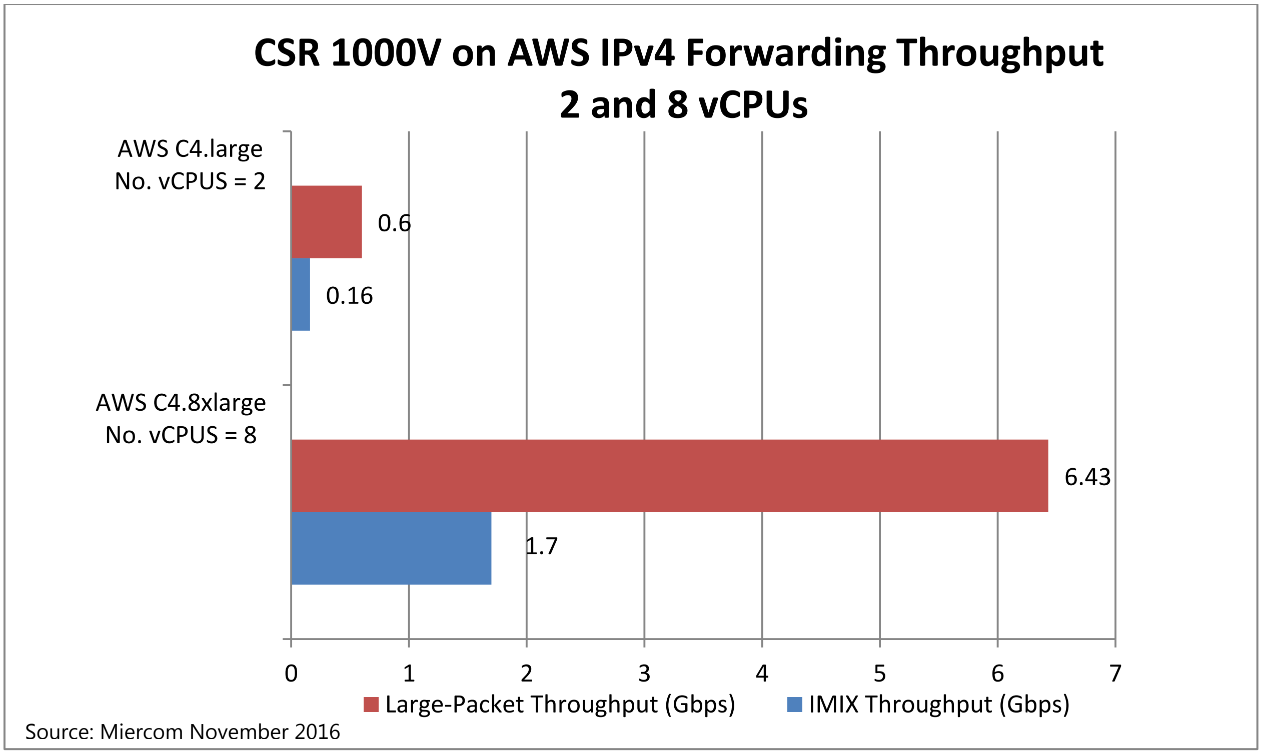
ENCS 5400 Series



5 – Public Cloud (AWS) Test Results

5.1 – IPv4 Forwarding

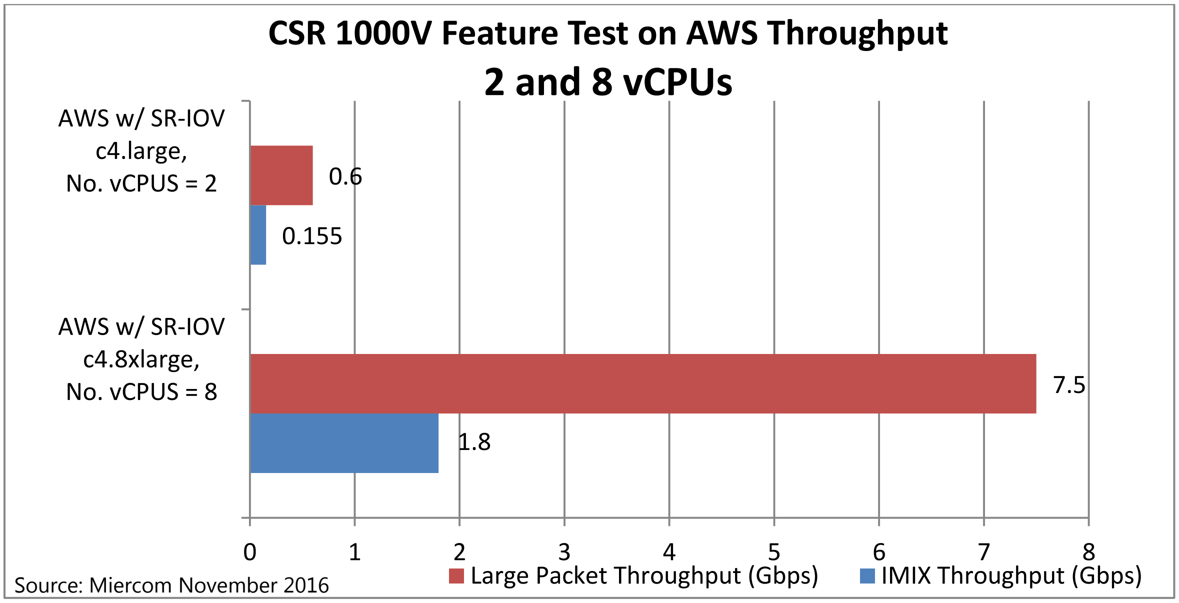
In the case of AWS, increasing from two to eight vCPUs achieved a tenfold increase in throughput – from 0.16 Gbps to 1.7 Gbps with IMIX traffic, from 0.6 Gbps to 6.43 Gbps with large packet (1518 Bytes) traffic. This looks like a side effect of instance size combined with hyper-threading. In a C4.large we have 1 physical CPU (pCPU) running 2 hyper-threads. In a C4.8xlarge we have get 8 pCPU each running single threaded. This means we get 8x the number of pCPU resources and we get the further advantage that the control plane isn't sharing a pCPU's pipeline with the data plane.



5.2 - IPv4 Forwarding, with Features Added

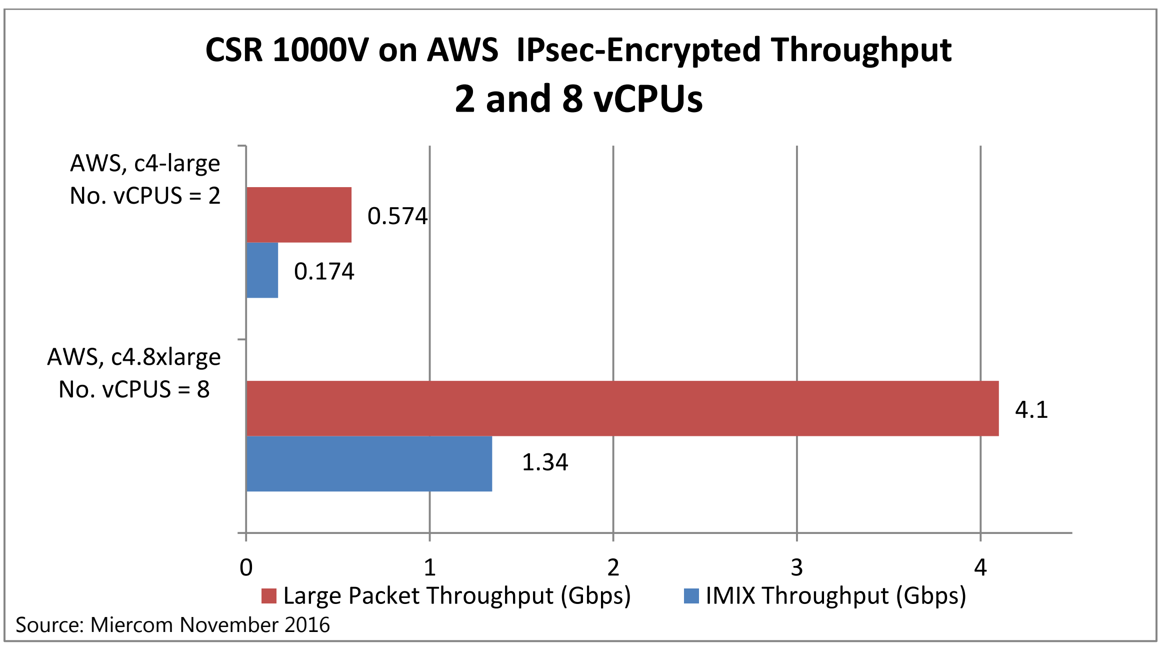
The same feature set with exactly the same configuration as we used for private cloud (KVM) tests was used for Amazon Web Services Tests.

The diagram below shows our finding for CSR 10000v feature tests on AWS.



5.3 - IPsec (IP Security) Encrypted Tunneling

Again, the same IPSec tests were run on private (KVM) and public (AWS)cloud. The results for IMIX and large packets are shown in the chart below.



With full IPsec processing and encryption applied to all traffic through the IPsec tunnel, bi-directionally, 0.174 Gbps with IMIX traffic and 0.574 Gbps with large packet (1418 Bytes) traffic throughput was achieved in the AWS environment with a single vCPU. With eight vCPUs configured throughput climbed to 1.34 Gbps with IMIX traffic and 4.1 Gbps with large packet (1418 Bytes) traffic, about an eight-fold increase.

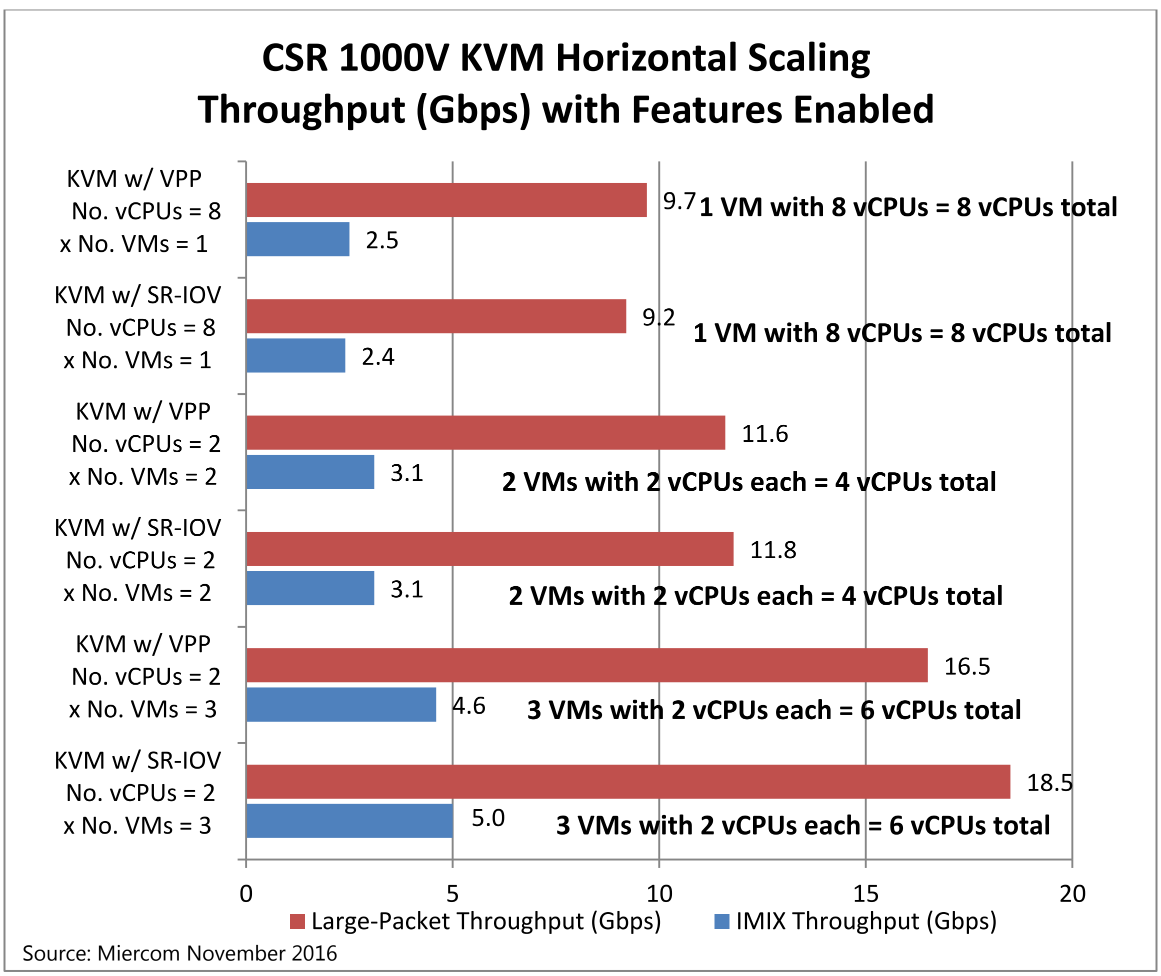
6 - Horizontal Scaling Across Multiple Virtual Machines (VMs)

Another throughput performance boost we tested was horizontal scaling – where two Virtual Machines (VMs) are used instead of one, with two vCPUs per VM. This can work particularly well when load balancing between the VMs is a feature of the configuration.

These tests were conducted with features enabled – the same set of features applied in the IPv4 Tests with Features battery of tests. As before, the same features enabled in the configuration created were:

- IPsec Gateway
- Hierarchical QoS
- NBAR deep packet inspection
- NAT/PAT – 100 different port translations
- Zone-based Firewall (inspection only, of unidirectional traffic)

It turns out that using two VM's, each with two vCPUs, yields better throughput than a single VM with eight vCPUs. If the user can do load balancing, then it is faster – and likely more economical – to use half the vCPUs, and two VMs, for the same or better performance. In the following diagram that the quoted performance numbers are NET performance numbers (not per VM but the sum of all VM's for that particular run).



7 - About "Miercom Performance Verified" Testing

This report was sponsored by Cisco Systems, Inc. The data was obtained completely and independently by Miercom engineers and lab-test staff as part of our Performance Verified assessment. Testing such as this is based on a methodology that is jointly co-developed with the sponsoring vendor. The test cases are designed to focus on specific claims of the sponsoring vendor, and either validate or repudiate those claims. The results are presented in a report such as this one, independently published by Miercom.

8 - About Miercom

Miercom has published hundreds of network-product-comparison analyses in leading trade periodicals and other publications. Miercom's reputation as the leading, independent product test center is undisputed.

Private test services available from Miercom include competitive product analyses, as well as individual product evaluations. Miercom features comprehensive certification and test programs including: Certified Interoperable, Certified Reliable, Certified Secure and Certified Green. Products may also be evaluated under the Performance Verified program, the industry's most thorough and trusted assessment for product usability and performance.

9 - Use of This Report

Every effort was made to ensure the accuracy of the data contained in this report but errors and/or oversights can occur. The information documented in this report may also rely on various test tools, the accuracy of which is beyond our control. Furthermore, the document relies on certain representations by the vendors that were reasonably verified by Miercom but beyond our control to verify to 100 percent certainty.

This document is provided "as is," by Miercom and gives no warranty, representation or undertaking, whether express or implied, and accepts no legal responsibility, whether direct or indirect, for the accuracy, completeness, usefulness or suitability of any information contained in this report.

No part of any document may be reproduced, in whole or in part, without the specific written permission of Miercom or Cisco Systems, Inc. All trademarks used in the document are owned by their respective owners. You agree not to use any trademark in or as the whole or part of your own trademarks in connection with any activities, products or services which are not ours, or in a manner which may be confusing, misleading or deceptive or in a manner that disparages us or our information, projects or developments.