JUNIPER NETWORKS | Engineering Simplicity

# vSRX Deployment Guide for Microsoft Azure Cloud

Published
2020-12-28

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net



*vSRX Deployment Guide for Microsoft Azure Cloud*


The information in this document is current as of the date on the title page.

## YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

# About This Guide

Use this guide to install the vSRX Virtual Firewall in the Microsoft Azure Cloud. This guide also includes basic vSRX configuration and management procedures.

After completing the installation and basic configuration procedures covered in this guide, refer to the Junos OS documentation for information about further software configuration.

1
CHAPTER

# Overview

# vSRX Overview

**SUMMARY**

In this topic you learn about vSRX architecture and its benefits.

**IN THIS SECTION**

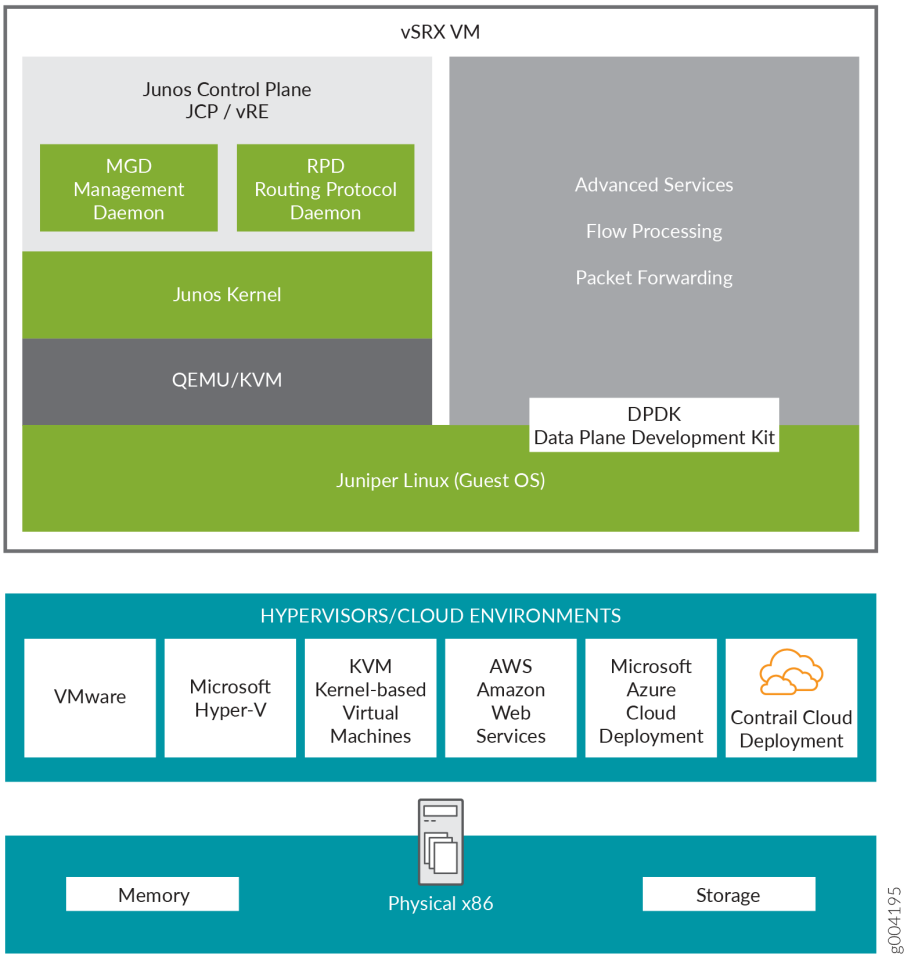


vSRX is a virtual security appliance that provides security and networking services at the perimeter or edge in virtualized private or public *cloud* environments. vSRX runs as a virtual machine (*VM*) on a standard x86 server. vSRX is built on the Junos operating system (Junos OS) and delivers networking and security features similar to those available on the software releases for the SRX Series Services Gateways.

The vSRX provides you with a complete Next-Generation Firewall (NGFW) solution, including core firewall, VPN, NAT, advanced Layer 4 through Layer 7 security services such as Application Security, intrusion detection and prevention (IPS), and UTM features including Enhanced Web Filtering and Anti-Virus. Combined with Sky ATP, the vSRX offers a cloud-based advanced anti-malware service with dynamic analysis to protect against sophisticated malware, and provides built-in machine learning to improve verdict efficacy and decrease time to remediation.

Figure 1 on page 3 shows the high-level architecture.

**Figure 1: vSRX Architecture**




vSRX includes the Junos control plane (JCP) and the packet forwarding engine (PFE) components that make up the data plane. vSRX uses one virtual CPU (vCPU) for the JCP and at least one vCPU for the PFE. Starting in Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, multi-core vSRX supports scaling vCPUs and virtual RAM (vRAM). Additional vCPUs are applied to the data plane to increase performance.

Junos OS Release 18.4R1 supports a new software architecture vSRX 3.0 that removes dual OS and nested virtualization requirement of existing vSRX architecture.

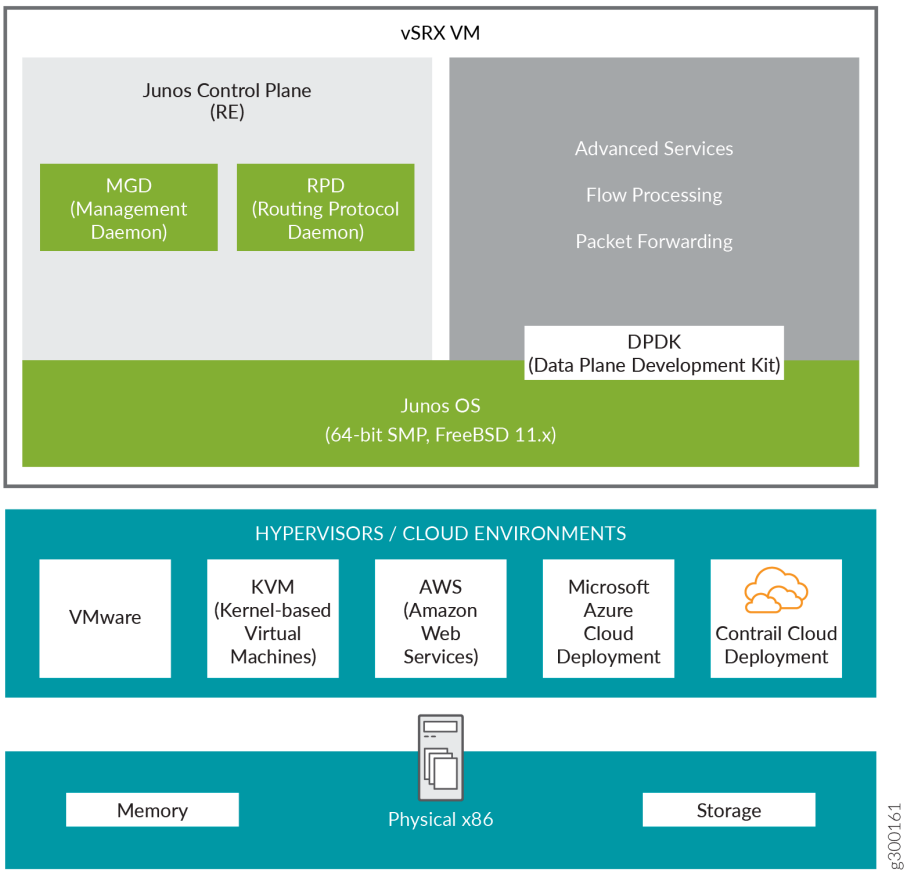
In vSRX 3.0 architecture, FreeBSD 11.x is used as the guest OS and the Routing Engine and Packet Forwarding Engine runs on FreeBSD 11.x as single virtual machine for improved performance and scalability. vSRX 3.0 uses DPDK to process the data packets in the data plane. A direct Junos upgrade from vSRX to vSRX 3.0 software is not supported.

vSRX 3.0 has the following enhancements compared to vSRX:

- Removed the restriction of requiring nested VM support in hypervisors.
- Removed the restriction of requiring ports connected to control plane to have Promiscuous mode enabled.
- Improved boot time and enhanced responsiveness of the control plane during management operations.
- Improved live migration.

Figure 2 on page 4 shows the high-level architecture for vSRX 3.0

**Figure 2: vSRX 3.0 Architecture**



## Benefits

vSRX on standard x86 servers enables you to quickly introduce new services, deliver customized services to customers, and scale security services based on dynamic needs. vSRX is ideal for public, private, and hybrid cloud environments.

Some of the key benefits of vSRX in a virtualized private or public cloud multitenant environment include:

- *Stateful firewall* protection at the tenant edge
- Faster deployment of virtual firewalls into new sites
- Ability to run on top of various hypervisors and public cloud infrastructures
- Full routing, *VPN*, core security, and networking capabilities
- Application security features (including IPS and App-Secure)
- Content security features (including Anti Virus, Web Filtering, Anti Spam, and Content Filtering)
- Centralized management with Junos Space Security Director and local management with J-Web Interface
- Juniper Networks Sky Advanced Threat Prevention (Sky ATP) integration

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D70 | Starting in Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, multi-core vSRX supports scaling vCPUs and virtual RAM (vRAM). Additional vCPUs are applied to the data plane to increase performance. |

# Understand vSRX with Microsoft Azure Cloud

**IN THIS SECTION**

This section presents an overview of vSRX as deployed in the Microsoft Azure cloud.

## vSRX with Microsoft Azure

Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to the Microsoft Azure Cloud. Microsoft Azure is Microsoft's application platform for the public cloud. It is an open, flexible, enterprise-grade cloud computing platform for building, deploying, and managing applications and services through a global network of Microsoft-managed data centers. It provides Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) services. You place your virtual machines (VMs) onto Azure virtual networks, where the distributed and virtual networks in Azure help ensure that your private network traffic is logically isolated from traffic on other Azure virtual networks.

The Azure WALinuxAgent performs the provisioning job for the vSRX instances. When a new vSRX instance is deployed, the continued increasing size of the waagent log file might cause the vSRX to stop. If the vSRX is still operating, then delete the **/var/log/waagent.log** directly or run the **clear log waagent.log all** command to clear the log file.

Or you can run the **set groups azure-provision system syslog file waagent.log archive size 1m** and **set groups azure-provision system syslog file waagent.log archive files 10** commands to prevent the growing of the waagent logs. These configurations will cause the rotation of log of waagent with the size bigger than 1MB and set a maximum of 10 backups.

You can add a vSRX virtual security appliance to provide networking security features as an application instance within an Azure virtual network. The vSRX protects the workloads that run within the virtual network on the Microsoft Azure Cloud.

You can deploy the vSRX VM in Azure using the following deployment methods:

- Azure Marketplace—Deploy the vSRX VM from the Azure Marketplace. The Azure Marketplace provides you with different methods to deploy a vSRX VM in your virtual network. You can choose a customized solution template offered by Juniper Networks to automate the vSRX VM deployment based on specific use cases (for example, a security gateway). A solution template automates the dependencies associated with specific deployment use cases, such as VM settings, virtual network settings (such as multiple subsets for the management interface (fxp0) and two revenue (data) interfaces), and so on. Or, you can select the vSRX VM image and define the deployment settings and dependencies based on your specific networking requirements. Starting in Junos OS Release 15.1X49-D91 for vSRX, you can deploy the vSRX to Microsoft Azure Cloud from the Azure Marketplace.

  Azure Marketplace also enables you to discover and subscribe to software that supports regulated workloads through Azure Marketplace for Azure Government Cloud (US).

- Azure CLI—Deploy the vSRX VM from the Azure CLI. You can customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud. To help automate and simplify the deployment of the vSRX VM in the Microsoft Azure virtual network, Juniper Networks provides a series of scripts, Azure Resource Manager (ARM) templates and parameter files, and configuration files in a GitHub repository.

  **NOTE**: Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to Microsoft Azure Cloud from the Azure CLI.

In Microsoft Azure, you can host servers and services on the cloud as a pay-as-you-go (PAYG) or bring-your-own-license (BYOL) service.

**NOTE**: vSRX PAYG images do not require any Juniper Networks licenses.

Starting in Junos OS Release 15.1X49-D120, vSRX on Microsoft Azure Cloud supports the vSRX Premium-Next Generation Firewall with Anti-Virus Protection bundle for PAYG, available as 1-hour or 1-year subscriptions. This bundle includes:

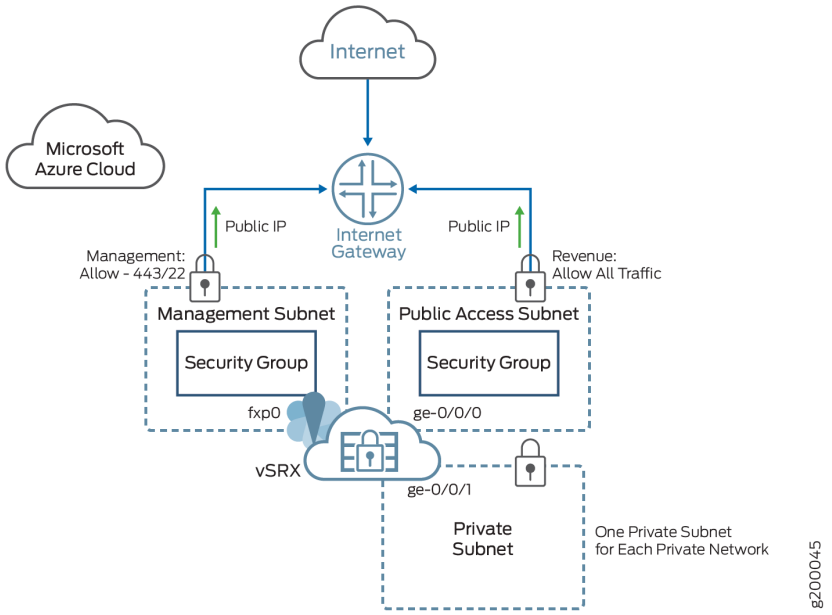
- Standard (STD) features of core security, including core firewall, IPsec VPN, NAT, CoS, and routing services.
- Advanced Layer 4 through 7 security services such as AppSecure features of AppID, AppFW, AppQoS, and AppTrack, IPS and rich routing capabilities, including the UTM antivirus feature.

Figure 3 on page 8 illustrates the deployment of a vSRX in Microsoft Azure.

In the Microsoft Azure, public subnets have access to the Internet gateway, but private subnets do not. vSRX requires two public subnets and one or more private subnets for each individual instance group. The public subnets consist of one for the management interface (fxp0) and one for a revenue (data)

interface. The private subnets, connected to the other vSRX interfaces, ensure that all traffic between applications on the private subnets and the Internet must pass through the vSRX instance.

**Figure 3: vSRX Deployed to Microsoft Azure**




For a glossary of Microsoft Azure terms see Microsoft Azure glossary.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D91 | Starting in Junos OS Release 15.1X49-D91 for vSRX, you can deploy the vSRX to Microsoft Azure Cloud from the Azure Marketplace. |
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to the Microsoft Azure Cloud. |
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to Microsoft Azure Cloud from the Azure CLI. |
| 15.1X49-D120 | Starting in Junos OS Release 15.1X49-D120, vSRX on Microsoft Azure Cloud supports the vSRX Premium-Next Generation Firewall with Anti-Virus Protection bundle for PAYG, available as 1-hour or 1-year subscriptions. |

RELATED DOCUMENTATION

- Microsoft Azure
- Azure Virtual Networks
- Microsoft Azure portal overview

# Requirements for vSRX on Microsoft Azure

IN THIS SECTION



This section presents an overview of requirements for deploying a vSRX instance on Microsoft Azure Cloud.

## System Requirements for vSRX on Microsoft Azure Cloud

Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to the Microsoft Azure Cloud. Microsoft Azure supports a wide variety of sizes and options for deployed Azure virtual machines (VMs).

For the vSRX deployment in Microsoft Azure, we recommend DSv2-series VMs. The DSv2-series VMs provided from Microsoft Azure use Premium Storage(SSD) and are ideal for applications that demand faster CPUs and better local disk performance, or have higher memory demands. Of the available DSv2-series VMs, we recommend that you select Standard_DS3_v2, Standard_DS4_v2, or Standard_DS5_v2 for the vSRX VM deployment in Microsoft Azure. For more details, see DSv2-series.

Table 1 on page 10 lists the properties of the Standard_DS3_v2 VM available in Microsoft Azure.

**Table 1: Properties of the Standard_DS3_v2 VM in Microsoft Azure**

| Component | Specification |
|---|---|
| Size | Standard_DS3_v2 |
| CPU cores | 4 |
| Memory | 14 GiB |
| Maximum number of data disks | 16 |
| Maximum cached and local disk storage throughput: IOPS/MBps (cache size in GB) | 16,000/128 (172) |
| Maximum uncached disk throughput: IOPS/MBps | 12,800/192 |
| Max NICs/Expected network bandwidth (Mbps) | 4/3000 |

Table 2 on page 10 lists the properties of the Standard_DS4_v2 VM available in Microsoft Azure.

**Table 2: Properties of the Standard_DS4_v2 VM in Microsoft Azure**

| Component | Specification |
|---|---|
| Size | Standard DS4_v2 |
| CPU cores | 8 |
| Memory | 28 GiB |
| Maximum number of data disks | 32 |
| Temp storage (SSD) GiB | 56 |

**Table 2: Properties of the Standard_DS4_v2 VM in Microsoft Azure *(Continued)***

| Component | Specification |
|---|---|
| Max cached and temp storage throughput: IOPS/ MBps (cache size in GiB) | 32000/256 (344) |
| Max uncached disk throughput: IOPS/MBps | 25600/384 |
| Max NICs/Expected network bandwidth (Mbps) | 8/6000 |

**NOTE**: The vSRX does not provide support for a high availability configuration in Microsoft Azure. In addition, the vSRX does not support Layer 2 transparent mode in Microsoft Azure.

Table 3 on page 11 lists the properties of the Standard_DS5_v2 VM available in Microsoft Azure.

**Table 3: Properties of the Standard_DS5_v2 VM in Microsoft Azure**

| Component | Specification |
|---|---|
| Size | Standard DS5_v2 |
| CPU cores | 16 |
| Memory | 56 GiB |
| Maximum number of data disks | 64 |
| Temp storage (SSD) GiB | 112 |
| Max cached and temp storage throughput: IOPS/ MBps (cache size in GiB) | 64000/512 (688) |

**Table 3: Properties of the Standard_DS5_v2 VM in Microsoft Azure *(Continued)***

| Component | Specification |
|---|---|
| Max uncached disk throughput: IOPS/MBps | 51200/768 |
| Max NICs/Expected network bandwidth (Mbps) | 8/12000 |

## Network Requirements for vSRX on Microsoft Azure Cloud

When you deploy a vSRX VM in a Microsoft Azure virtual network, note the following specifics of the deployment configuration:

- A dual public IP network configuration is a requirement for vSRX VM network connectivity; the vSRX VM requires two public subnets and one or more private subnets for each instance group.
- The public subnets required by the vSRX VM consist of one subnet for the out-of-band management interface (fxp0) for management access and another for the two revenue (data) interfaces. By default, one interface is assigned to the untrust security zone and the other to the trust security zone on the vSRX VM.
- In the Microsoft Azure deployment of the vSRX VM, the vSRX supports the management interface (fxp0) and the two revenue (data) interfaces (port ge-0/0/0 and ge-0/0/1), which includes public IP address mapping and data traffic forwarding to and from the vSRX VM.

## Microsoft Azure Instances and vSRX Instance Types

Microsoft Azure instance types supported for vSRX are listed in Table 4 on page 13.

**Table 4: Supported Microsoft Azure Instance Types for vSRX**

| Instance Type | vSRX Type | vCPUs | Memory in Instance Type (GB) | RSS Type |
|---|---|---|---|---|
| Standard_DS3_v2 | VSRX-4CPU-14G memory | 4 | 14 | HWRSS |
| Standard_DS4_v2 | VSRX-8CPU-28G memory | 8 | 28 | HWRSS |
| Standard_DS5_v2 | VSRX-16CPU-56G memory | 16 | 56 | HWRSS |

## Interface Mapping for vSRX on Microsoft Azure

Table 5 on page 13 lists the vSRX and Microsoft Azure interface names. The first network interface is used for the out-of-band management (fxp0) for vSRX.

**Table 5: vSRX and Microsoft Azure Interface Names**

| Interface Number | vSRX Interface | Microsoft Azure Interface |
|---|---|---|
| 1 | fxp0 | eth0 |
| 2 | ge-0/0/0 | eth1 |
| 3 | ge-0/0/1 | eth2 |

We recommend putting revenue interfaces in routing instances as a best practice to avoid asymmetric traffic/routing, because fxp0 is part of the default (inet.0) table by default. With fxp0 as part of the default routing table, there might be two default routes needed: one for the fxp0 interface for external management access, and the other for the revenue interfaces for traffic access. Putting the revenue

interfaces in a separate routing instance avoids this situation of two default routes in a single routing instance. Ensure that interfaces belonging to the same security zone are in the same routing instance.

## vSRX Default Settings on Microsoft Azure

vSRX requires the following basic configuration settings:

- Interfaces must be assigned IP addresses.
- Interfaces must be bound to zones.
- Policies must be configured between zones to permit or deny traffic.

Table 6 on page 14 lists the factory-default settings for security policies on the vSRX

**Table 6: Factory-Default Settings for Security Policies**

| Source Zone | Destination Zone | Policy Action |
|---|---|---|
| trust | untrust | permit |
| trust | trust | permit |

**CAUTION**: Do not use the **load factory-default** command on the vSRX instance in Microsoft Azure. The factory-default configuration removes the “azure provision” preconfiguration. This group contains critical system-level settings and route information for the vSRX. A misconfiguration in the group “azure-provision” may result in the possible loss of connectivity to vSRX from Microsoft Azure. If you must revert to factory default, ensure that you first manually reconfigure the Microsoft Azure preconfiguration statements before you commit the configuration; otherwise, you will lose access to the vSRX instance.

We strongly recommend that when you commit a configuration, perform an explicit **commit confirmed** to avoid the possibility of losing connectivity to vSRX. Once you have verified that the change works correctly, you can keep the new configuration active by entering the **commit** command within 10 minutes. Without the timely second confirm, configuration changes will be rolled back. See "Configure vSRX Using the CLI" on page 92 for preconfiguration details.

## Best Practices for Improving vSRX Performance

Review the following deployment practices to improve vSRX performance:

- Disable the source/destination check for all vSRX interfaces.
- Limit public key access permissions to 400 for key pairs.
- Ensure that there are no contradictions between Microsoft Azure security groups and your vSRX configuration.
- Use vSRX NAT to protect your instances from direct Internet traffic.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX to the Microsoft Azure Cloud. |

RELATED DOCUMENTATION

- KB Article - Interface must be in the same routing instance as the other interfaces in the zone
- Windows virtual machines in Azure

# Junos OS Features Supported on vSRX

**SUMMARY**

This topic provides details of the Junos OS features supported and not supported on vSRX.

**IN THIS SECTION**

## SRX Series Features Supported on vSRX

vSRX inherits most of the branch SRX Series features with the following considerations shown in Table 7 on page 16.

To determine the Junos OS features supported on vSRX, use the Juniper Networks Feature Explorer, a Web-based application that helps you to explore and compare Junos OS feature information to find the right software release and hardware platform for your network. Find Feature Explorer at: Feature Explorer: vSRX .

**Table 7: vSRX Feature Considerations**

| Feature | Description |
|---|---|
| IDP | The IDP feature is subscription based and must be purchased. After purchase, you can activate the IDP feature with the license key.<br><br>For SRX Series IDP configuration details, see:<br><br>Understanding Intrusion Detection and Prevention for SRX Series |

**Table 7: vSRX Feature Considerations *(Continued)***

| Feature | Description | |
|---|---|---|
| IPSec VPNs | Starting in Junos OS Release 19.3R1, vSRX supports the following authentication algorithms and encryption algorithms:<br>• Authentication algorithm: hmac-sha1-96 and HMAC-SHA-256-128 authentication<br>• Encryption algorithm: aes-128-cbc<br>Starting in Junos OS Release 20.3R1, vSRX supports 10,000 IPsec VPN tunnels.<br>To support the increased number of IPsec VPN tunnels, a minimum of 19 vCPUs are required. Out of the 19 vCPUs, 3 vCPUs must be dedicated to RE.<br>You must run the **request system software add optional://junos-ike.tgz** command the first time you wish to enable increased IPsec tunnel capacity. For subsequent software upgrades of the instance, the junos-ike package is upgraded automatically from the new Junos OS releases installed in the instance. If chassis cluster is enabled then run this command on both the nodes.<br>You can configure the number of vCPUs allocated to Junos Routing Engine using the **set security forwarding-options resource-manager cpu re <value>**.<br>**NOTE**: 64 G memory is required to support 10000 tunnels in PMI mode.<br>[See show security ipsec security-associations, show security ike tunnel-map, and show security ipsec tunnel-distribution.] | |
| IPsec VPN - Tunnel Scaling on vSRX | Types of Tunnels | Number of tunnels supported |
| | Site-Site VPN tunnels | 2000 |
| | AutoVPN tunnels | 10,000 |
| | IKE SA (Site-to-site) | 2000 |
| | IKE SA (AutoVPN) | 10,000 |

**Table 7: vSRX Feature Considerations *(Continued)***

| Feature | Description | |
|---|---|---|
| | IKE SA (Site-to-site + AutoVPN) | 10,000 |
| | IPSec SA pairs (Site-to-site) | 10,000<br>With 2000 IKE SAs, we can have 10,000 IPSec SA. |
| | IPSec SA pairs (AutoVPN) | 10,000 |
| | Site-to-site + AutoVPN IPSec SA pairs | 2000 Site-to-site 8000 AutoVPN |
| | Site-to-site + AutoVPN tunnels | 2000 Site-to-site 8000 AutoVPN |
| ISSU | ISSU is not supported. | |
| Logical Systems | Starting in Junos OS Release 20.1R1, you can configure logical systems and tenant systems on vSRX and vSRX 3.0 instances.<br>With Junos OS, you can partition a single security device into multiple logical devices that can perform independent tasks.<br>Each logical system has its own discrete administrative domain, logical interfaces, routing instances, security firewall and other security features.<br>See Logical Systems Overview. | |

**Table 7: vSRX Feature Considerations *(Continued)***

| Feature | Description |
|---|---|
| PowerMode IPsec | Starting in Junos OS Release 20.1R1, vSRX 3.0 instances support PowerMode IPsec that provides IPsec performance improvements using Vector Packet Processing (VPP) and Intel AES-NI instructions. PowerMode IPsec is a small software block inside the SRX PFE (SRX Packet Forwarding Engine) that is activated when PowerMode is enabled.<br><br>Supported Features in PowerMode IPsec<br>• IPsec functionality<br>• Traffic selectors<br>• Secure tunnel interface (st0)<br>• All control plane IKE functionality<br>• Auto VPN with traffic selector<br>• Auto VPN with routing protocol<br>• IPv6<br>• Stateful Layer 4 firewall<br>• High-Availability<br>• NAT-T<br><br>Non-Supported Features in PowerMode IPsec<br>• NAT<br>• IPsec in IPsec<br>• GTP/SCTP firewall<br>• Application firewall/AppSecure<br>• QoS<br>• Nested tunnel<br>• Screen |

**Table 7: vSRX Feature Considerations *(Continued)***

| Feature | Description |
|---|---|
| | • Multicast<br>• Host traffic |
| Tenant Systems | Starting in Junos OS Release 20.1R1, you can configure tenant systems on vSRX and vSRX 3.0 instances.<br>A tenant system provides logical partitioning of the SRX device into multiple domains similar to logical systems and provides high scalability.<br>See Tenant Systems Overview. |
| Transparent mode | The known behaviors for transparent mode support on vSRX are:<br>• The default MAC learning table size is restricted to 16,383 entries.<br>For information about configuring transparent mode for vSRX, see Layer 2 Bridging and Transparent Mode Overview. |

**Table 7: vSRX Feature Considerations *(Continued)***

| Feature | Description |
|---|---|
| UTM | • The UTM feature is subscription based and must be purchased. After purchase, you can activate the UTM feature with the license key.<br>• Starting in Junos OS Release 19.4R1, vSRX 3.0 instances support the Avira scan engine, which is an on-device antivirus scanning engine. See On-Device Antivirus Scan Engine.<br>• For SRX Series UTM configuration details, see Unified Threat Management Overview.<br>• For SRX Series UTM antispam configuration details, see Antispam Filtering Overview.<br>• **Advanced resource management (vSRX 3.0)**—Starting in Junos OS Release 19.4R1, vSRX 3.0 manages the additional system resource requirements for UTM-and IDP-specific services by reallocating CPU cores and extra memory. These values for memory and CPU cores are not user configured. Previously, system resources such as memory and CPU cores were fixed.<br>You can view the allocated CPU and memory for advance security services on vSRX 3.0 instance by using the **show security forward-options resource-manager settings** command. To view the flow session scaling, use the **show security monitoring** command.<br>[See show security monitoring and show security forward-options resource-manager settings.] |

Some Junos OS software features require a license to activate the feature. To understand more about vSRX Licenses, see, Licenses for vSRX. Please refer to the Licensing Guide for general information about License Management. Please refer to the product Data Sheets for further details, or contact your Juniper Account Team or Juniper Partner.

## SRX Series Features Not Supported on vSRX

vSRX inherits many features from the SRX Series device product line. Table 8 on page 22 lists SRX Series features that are not applicable in a virtualized environment, that are not currently supported, or that have qualified support on vSRX.

**Table 8: SRX Series Features Not Supported on vSRX**

| SRX Series Feature | vSRX Notes |
|---|---|
| **Application Layer Gateways** | |
| Avaya H.323 | Not supported |
| **Authentication with IC Series devices** | |
| Layer 2 enforcement in UAC deployments | Not supported<br><br>**NOTE**: UAC-IDP and UAC-UTM also are not supported. |
| **Chassis cluster support**<br>**NOTE**: Support for chassis clustering to provide network node redundancy is only available on a vSRX deployment in Contrail, VMware, KVM, and Windows Hyper-V Server 2016. | |
| Chassis cluster for VirtIO driver | Only supported with KVM<br><br>**NOTE**: The link status of VirtIO interfaces is always reported as UP, so a vSRX chassis cluster cannot receive link up and link down messages from VirtIO interfaces. |
| Dual control links | Not supported |
| In-band and low-impact cluster upgrades | Not supported |
| LAG and LACP (Layer 2 and Layer 3) | Not supported |
| Layer 2 Ethernet switching | Not supported |
| Low-latency firewall | Not supported |
| **Class of service** | |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| High-priority queue on SPC | Not supported |
| Tunnels | Only GRE and IP-IP tunnels supported<br>**NOTE**: A vSRX VM deployed on Microsoft Azure Cloud does not support GRE and multicast. |
| **Data plane security log messages (stream mode)** | |
| TLS protocol | Not supported |
| **Diagnostic tools** | |
| Flow monitoring cflowd version 9 | Not supported |
| Ping Ethernet (CFM) | Not supported |
| Traceroute Ethernet (CFM) | Not supported |
| **DNS proxy** | |
| Dynamic DNS | Not supported |
| **Ethernet link aggregation** | |
| LACP in standalone or chassis cluster mode | Not supported |
| Layer 3 LAG on routed ports | Not supported |
| Static LAG in standalone or chassis cluster mode | Not supported |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| **Ethernet link fault management** | |
| **Physical interface (encapsulations)**<br>• **ethernet-ccc**<br>• **ethernet-tcc**<br>• **extended-vlan-ccc**<br>• **extended-vlan-tcc** | Not supported |
| **Interface family**<br>• **ccc, tcc**<br>• **ethernet-switching** | Not supported |
| **Flow-based and packet-based processing** | |
| End-to-end packet debugging | Not supported |
| Network processor bundling | |
| Services offloading | |
| **Interfaces** | |
| Aggregated Ethernet interface | Not supported |
| IEEE 802.1X dynamic VLAN assignment | Not supported |
| IEEE 802.1X MAC bypass | Not supported |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| IEEE 802.1X port-based authentication control with multisupplicant support | Not supported |
| Interleaving using MLFR | Not supported |
| PoE | Not supported |
| PPP interface | Not supported |
| PPPoE-based radio-to-router protocol | Not supported |
| PPPoE interface<br><br>**NOTE**: Starting in Junos OS Release 15.1X49-D100 and Junos OS Release 17.4R1, the vSRX supports Point-to-Point Protocol over Ethernet (PPPoE) interface. | Not supported |
| Promiscuous mode on interfaces | Only supported if enabled on the hypervisor |
| **IPSec and VPNs** | |
| Acadia - Clientless VPN | Not supported |
| DVPN | Not supported |
| Hardware IPsec (bulk crypto) Cavium/RMI | Not supported |
| IPsec tunnel termination in routing instances | Supported on virtual router only |
| Multicast for AutoVPN | Not supported |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| **IPv6 support** | |
| DS-Lite concentrator (also called Address Family Transition Router [AFTR]) | Not supported |
| DS-Lite initiator (aka B4) | Not supported |
| **J-Web** | |
| Enhanced routing configuration | Not supported |
| New Setup wizard (for new configurations) | Not supported |
| PPPoE wizard | Not supported |
| Remote VPN wizard | Not supported |
| Rescue link on dashboard | Not supported |
| UTM configuration for Kaspersky antivirus and the default Web filtering profile | Not supported |
| **Log file formats for system (control plane) logs** | |
| Binary format (binary) | Not supported |
| WELF | Not supported |
| **Miscellaneous** | |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| GPRS<br><br>**NOTE**: Starting in Junos OS Release 15.1X49-D70 and Junos OS Release 17.3R1, vSRX supports GPRS. | Not supported |
| Hardware acceleration | Not supported |
| Logical systems | Not supported |
| Outbound SSH | Not supported |
| Remote instance access | Not supported |
| USB modem | Not supported |
| Wireless LAN | Not supported |
| **MPLS** | |
| Crcuit cross-connect (CCC) and translational cross-connect (TCC) | Not supported |
| Layer 2 VPNs for Ethernet connections | Only if promiscuous mode is enabled on the hypervisor |
| **Network Address Translation** | |
| Maximize persistent NAT bindings | Not supported |
| **Packet capture** | |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| Packet capture | Only supported on physical interfaces and tunnel interfaces, such as *gr*, *ip*, and *st0*. Packet capture is not supported on redundant Ethernet interfaces (*reth*). |
| **Routing** | |
| BGP extensions for IPv6 | Not supported |
| BGP Flowspec | Not supported |
| BGP route reflector | Not supported |
| CRTP | Not supported |
| **Switching** | |
| Layer 3 Q-in-Q VLAN tagging | Not supported |
| **Transparent mode** | |
| UTM | Not supported |
| **Unified threat management** | |
| Express AV | Not supported |
| Kaspersky AV | Not supported |
| **Upgrading and rebooting** | |

**Table 8: SRX Series Features Not Supported on vSRX *(Continued)***

| SRX Series Feature | vSRX Notes |
|---|---|
| Autorecovery | Not supported |
| Boot instance configuration | Not supported |
| Boot instance recovery | Not supported |
| Dual-root partitioning | Not supported |
| OS rollback | Not supported |
| **User interfaces** | |
| NSM | Not supported |
| SRC application | Not supported |
| Junos Space Virtual Director | Only supported with VMware |

2

CHAPTER

# Deploying vSRX from the Azure Portal

# Before You Deploy vSRX from the Azure Portal

You can deploy a vSRX virtual security appliance and its advanced security features in your virtual network directly from the Azure portal. This method provides a browser-based user interface for creating and configuring virtual machines and all related resources.

The Azure Marketplace provides you with different methods to deploy a vSRX virtual machine (VM) in a virtual network. You can choose a customized solution template offered by Juniper Networks in the Azure Marketplace to automate the vSRX deployment based on a specific use case (for example, a security gateway).

Solution templates allow the bundling of multiple Azure services and a software image into a template that enables you to quickly deploy a preconfigured solution. You access vSRX solution templates from the Azure Marketplace to simplify the end-to-end configuration steps involved in deploying a vSRX VM in your Azure virtual network. A solution template automates the dependencies associated with specific deployment use cases, such as VM settings, virtual network settings (such as multiple subsets for the management interface (fxp0) and two revenue (data) interfaces), and so on.

A vSRX solution template is based on a custom Microsoft Azure Resource Manager (ARM) template. The ARM template consists of JavaScript Object Notation (JSON) expressions that construct specific values for your vSRX deployment. To integrate with the Azure portal, each solution template uses **mainTemplate.json** and **createUiDefinition.json** files to define the components of the customized solution template for vSRX VM deployment.

You also have the option to select the vSRX image from Azure Marketplace and customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud. This deployment approach might be required if you have a vSRX VM deployment scenario that is outside of the use cases offered in the vSRX VM solution templates available from Juniper Networks.

Before you deploy the vSRX virtual security appliance from the Azure Marketplace:

- Review the requirements for deploying a vSRX VM in Microsoft Azure Cloud in "Requirements for vSRX on Microsoft Azure" on page 9.
- Obtain an account for and a subscription to Microsoft Azure (see Microsoft Azure).
- Use your Microsoft account username and password to log into the Microsoft Azure portal.
- Purchase a vSRX license or request an evaluation license. Licenses can be procured from the Juniper Networks License Management System (LMS).
- Ensure that your Azure subscription includes the following for your vSRX VM:
  - Resource group, as described in "Create a Resource Group" on page 32.
  - Storage account, as described in "Create a Storage Account" on page 36.

- Virtual network, as described in "Create a Virtual Network" on page 42.

RELATED DOCUMENTATION

Microsoft Azure portal

Microsoft Azure portal overview

# Create a Resource Group

A resource group contains the resources required to successfully deploy a vSRX VM in Azure. It is a container that holds related resources for an Azure solution. In Azure, you logically group related resources such as storage accounts, virtual networks, and virtual machines (VMs) to deploy, manage, and maintain them as a single entity.

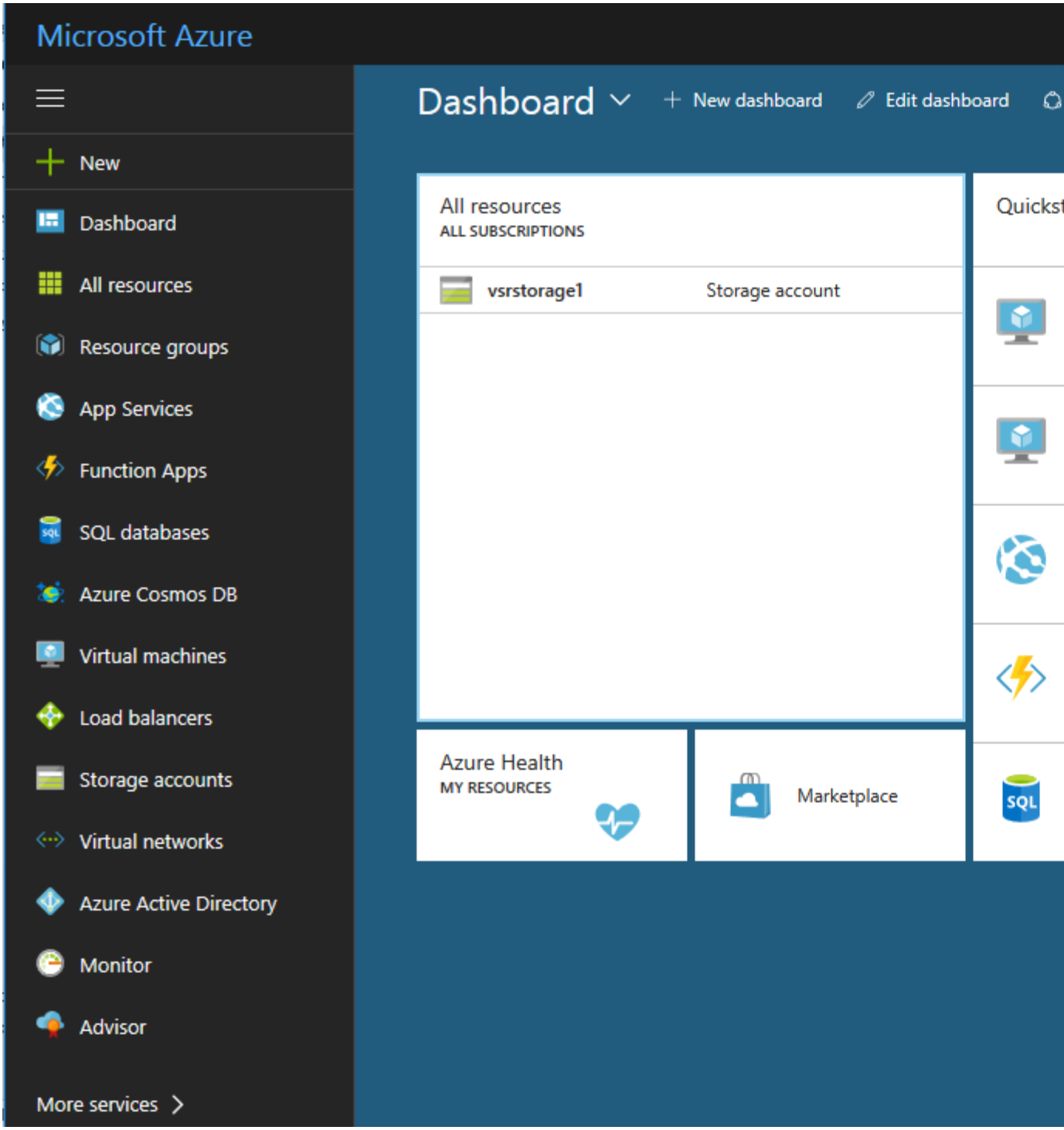
If you do not have an existing resource group in your subscription, then follow the steps outlined in this procedure.

To create a resource group in Azure:

1. Log in to the Microsoft Azure portal using your Microsoft account username and password. The Dashboard appears in the Azure portal (see Figure 4 on page 33). You see a unified dashboard for all

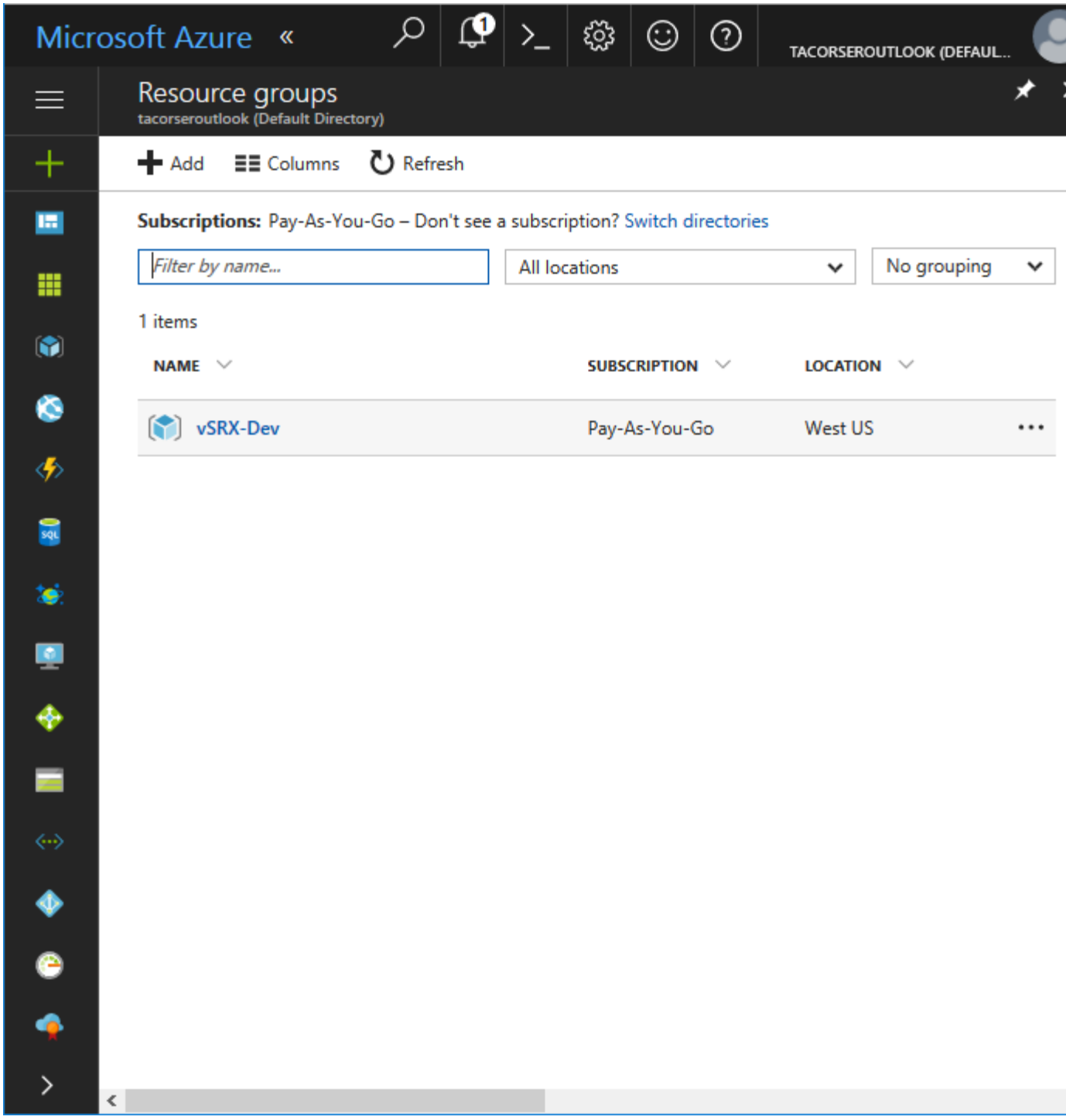
your assets in Azure. Verify that the dashboard includes all subscriptions to which you currently have access, and all resource groups and associated resources.

**Figure 4: Microsoft Azure Portal Dashboard**



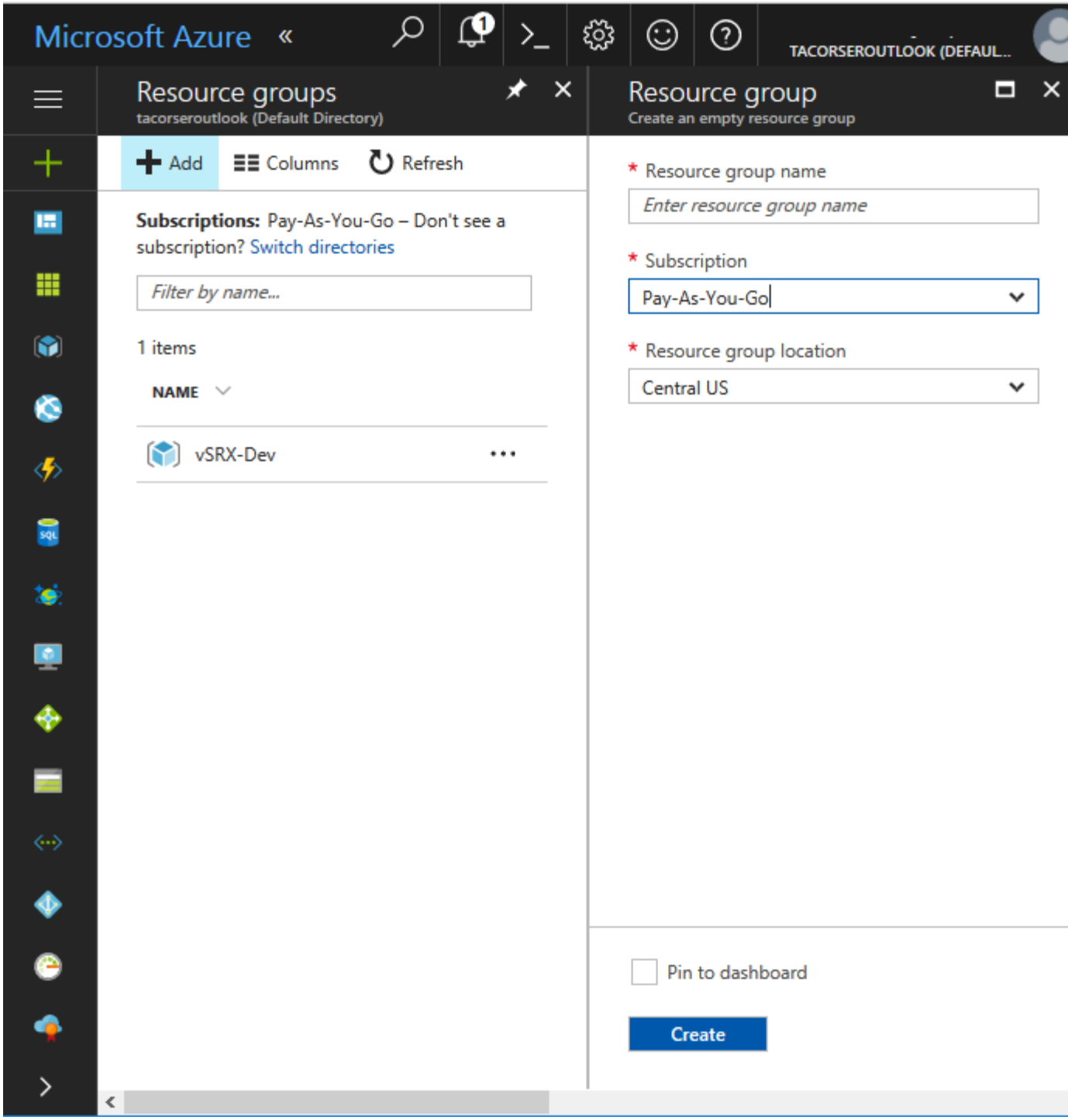
2. Click **Resource groups** from the menu of services to access the Resource Groups blade (see Figure 5 on page 34). You will see all the resource groups in your subscription listed in the blade.

**Figure 5: Resource Groups**



3. click **Add (+)** to create a new resource group. The Create Resource Group blade appears (see Figure 6 on page 35).

**Figure 6: Creating a Resource Group**



4. Provide the following information for the new resource group.

| Parameter | Description |
|---|---|
| Resource Group Name | Enter a unique name for your new resource group. A resource group name can include alphanumeric characters, periods (.), underscores (_), hyphens (-), and parenthesis (), but the name cannot end with a period. |
| Subscription | Select your Microsoft Azure subscription. |
| Resource Group Location | Select the location of the Microsoft Azure data center from which you intend to deploy the vSRX VM. Specify a location where the majority of your resources will reside. Typically, select the location that is closest to your physical location. |

5. Click **Create**. The resource group might take a few seconds to create. Once it is created, you see the resource group on the Azure portal dashboard.

RELATED DOCUMENTATION

- Azure Resource Manager overview
- Deploy resources with Resource Manager templates and Azure portal
- Manage Azure resources through portal

# Create a Storage Account

An Azure storage account provides a unique namespace to store and access your Azure storage data objects. All objects in a storage account are billed together as a group. By default, the data in your account is available only to the account owner.

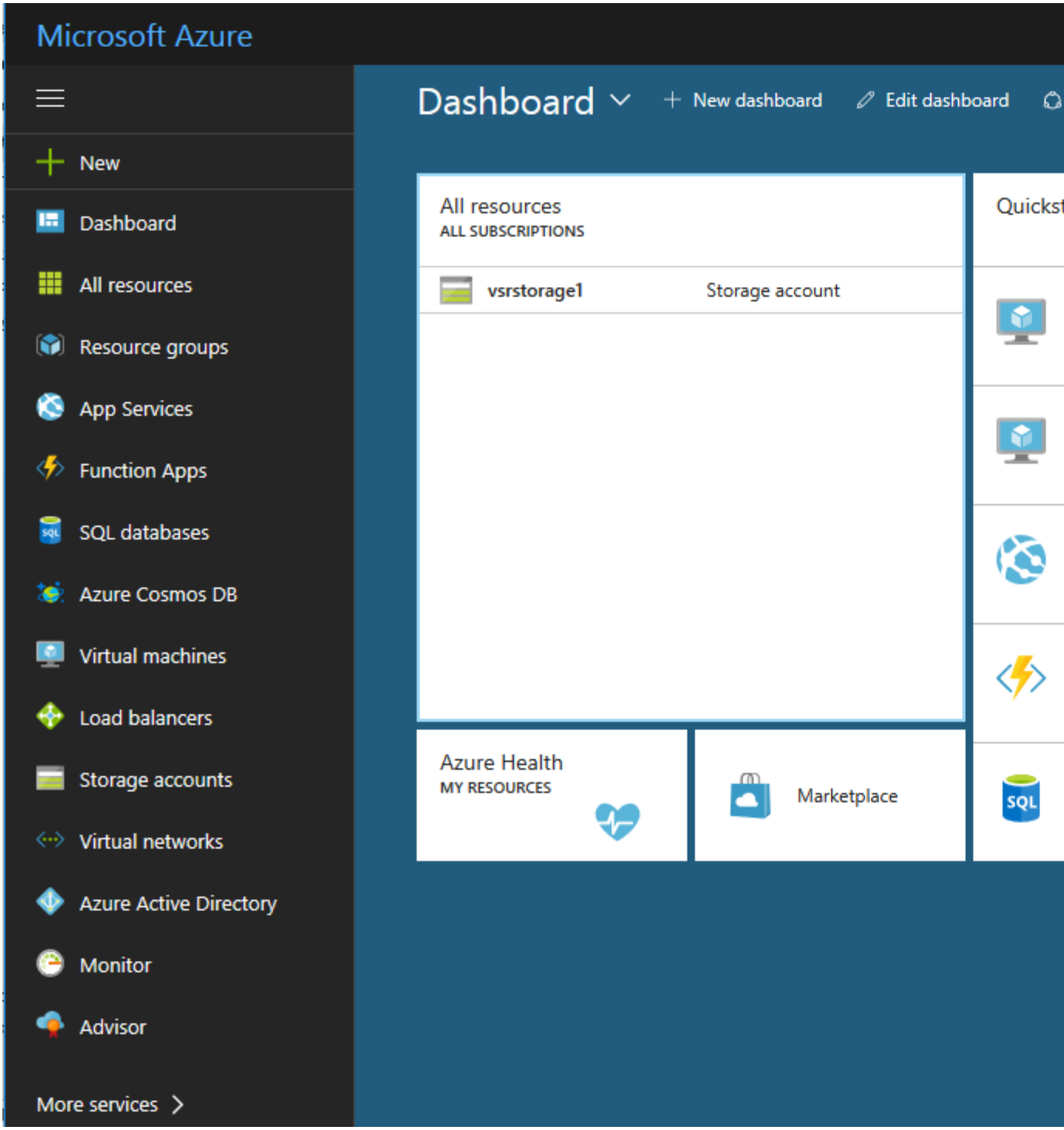
If you do not have an existing storage account in your subscription, follow the steps outlined in this procedure.

To create a storage account in Azure:

1. Log in to the Microsoft Azure portal using your Microsoft account username and password. The Dashboard appears in the Azure portal (see Figure 7 on page 38). You see a unified dashboard for all

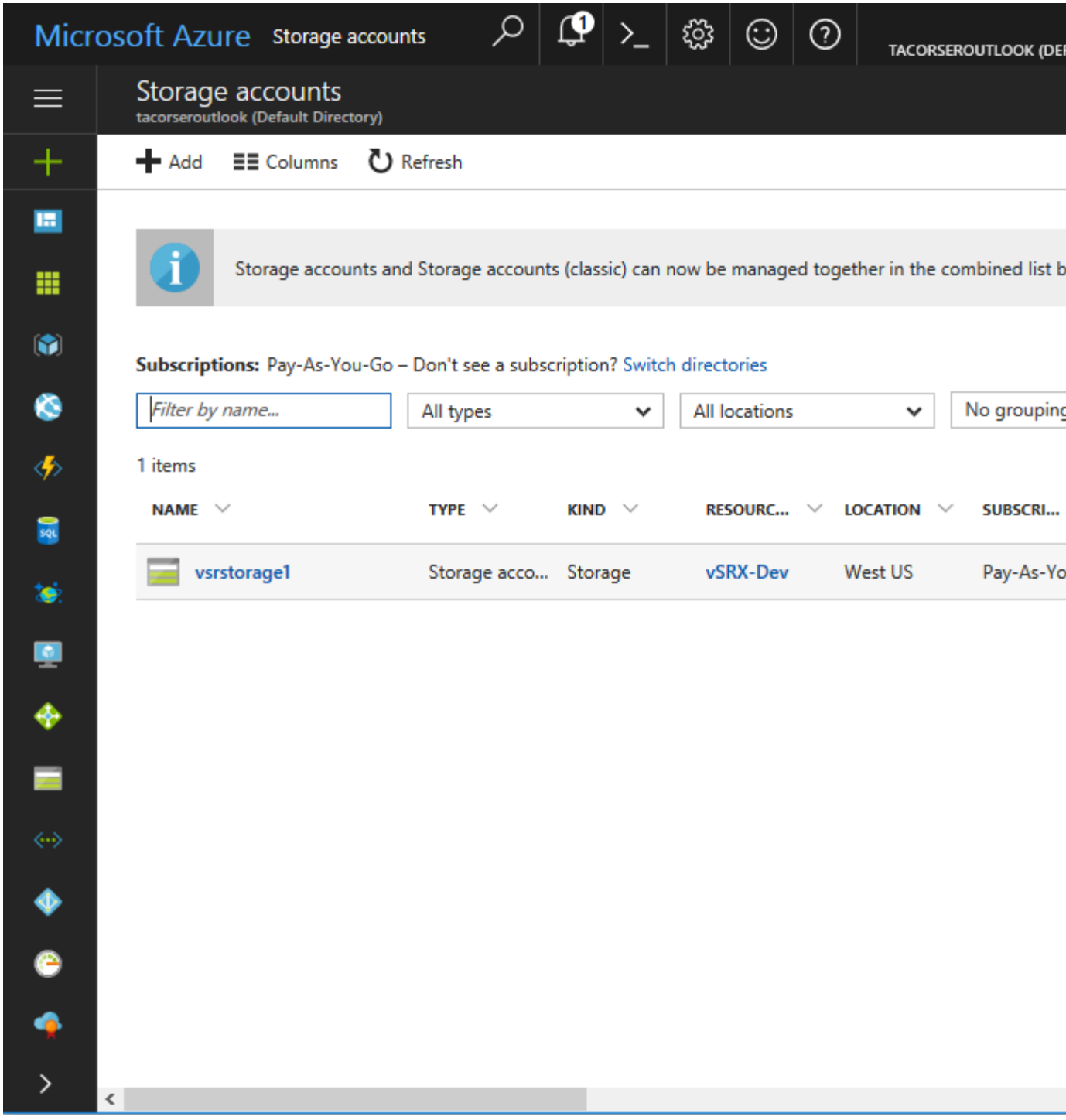
your assets in Azure. Verify that the dashboard includes all subscriptions to which you currently have access, and all resource groups and associated resources.

**Figure 7: Microsoft Azure Portal Dashboard**



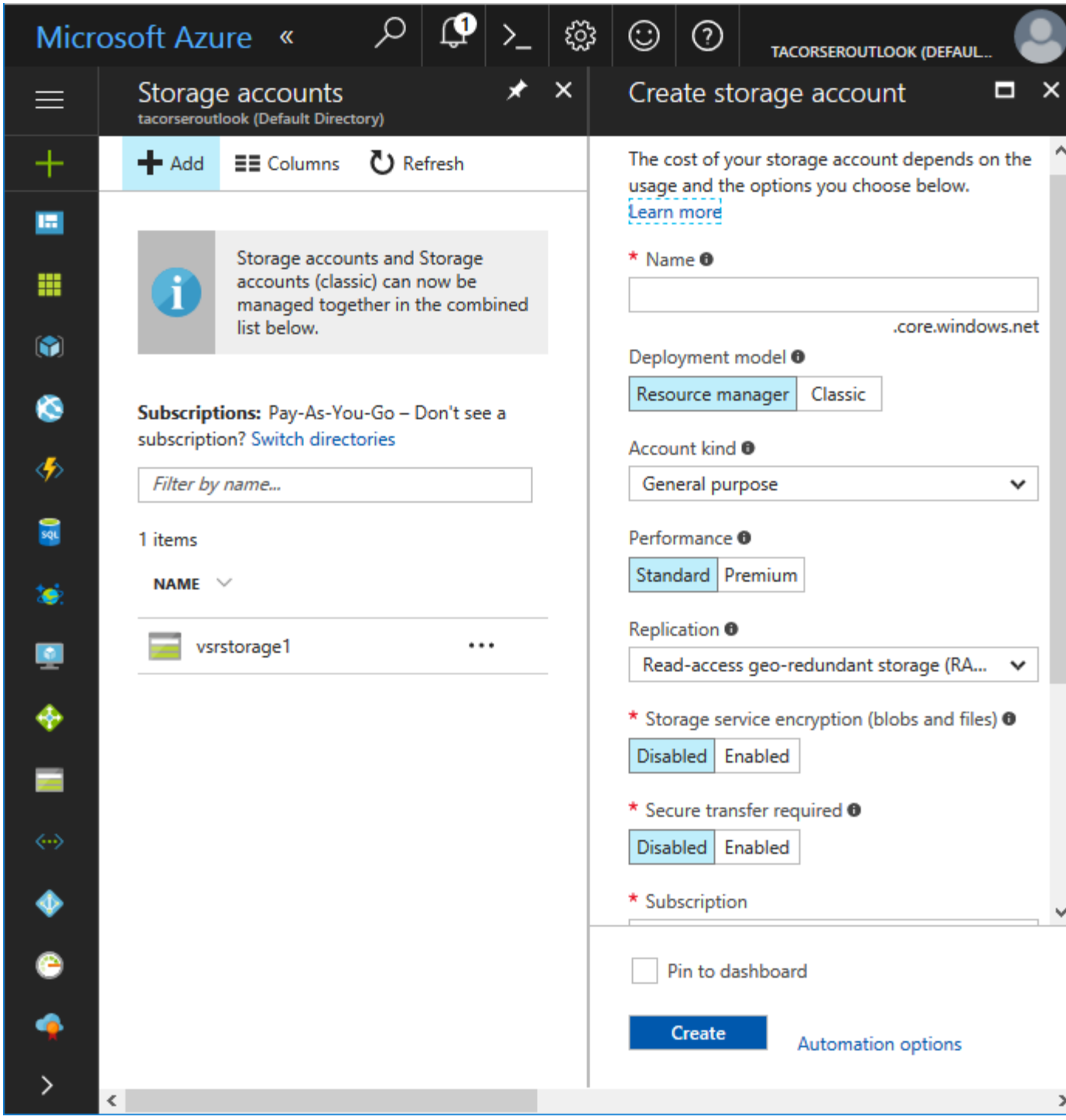
2. Click **Storage Accounts** from the menu of services to access the Storage Accounts blade (see Figure 8 on page 39).

**Figure 8: Azure Portal Storage Accounts**



3. Click **Add (+)** to create a new storage account. The Create Storage Account blade appears (see Figure 9 on page 40).

**Figure 9: Creating a Storage Account**



4. Provide the following information for the new storage account.

| Parameter | Description |
|---|---|
| Name | Enter a unique name for your new storage account. A storage account name can contain only lowercase letters and numbers, and must be between 3 and 24 characters. |
| Deployment Model | Select **Resource Manager** as the deployment model. |
| Account Kind | Select the type of storage account: **General purpose** or **Blob storage**. The default is **General purpose**.<br>• If General Purpose was selected, then specify the performance tier: **Standard** or **Premium**. The default is Standard.<br>• If Blob storage was selected, then specify the access tier: **Hot** or **Cool**. The default is Hot. |
| Performance | Select the type of performance: **Standard** or **Premium**. The default is **Standard**. |
| Replication | Select the replication option for the storage account: **Locally redundant storage (LRS)**, **Geo-redundant storage (GRS)**, **Read-access geo-redundant storage (RA-GRS)**, or **Zone-redundant storage (ZRS)**. The default is RA-GRS. |
| Storage Service Encryption | Enable or disable this option to protect your data at rest. Azure Storage encrypts data as written in an Azure datacenter, and decrypts that data once it is accessed. The default is Disabled. |
| Secure Transfer Required | Enable or disable this option to enhance the security of your storage account by allowing requests to the storage account by HTTPS only. The default is Disabled. |
| Subscription | Select your Microsoft Azure subscription. |

*(Continued)*

| Parameter | Description |
|---|---|
| Resource Group | Select an existing resource group or create a new one (see "Create a Resource Group" on page 32). |
| Location | Select the Azure data center geographic region in which you are deploying the vSRX VM. Typically, select the location that is closest to your physical location. |

5. Click **Create**. The storage account might take a few seconds to create. Once it is created, you see the storage account on the Azure portal dashboard.

RELATED DOCUMENTATION

- Introduction to Microsoft Azure Storage
- About Azure storage accounts

# Create a Virtual Network

The Azure Virtual Network service enables you to securely connect Azure resources to each other with virtual networks. A virtual network is a representation of your own network in the cloud. It is a logical isolation of the Azure cloud dedicated to your subscription. You can also connect virtual networks to your on-premises network.

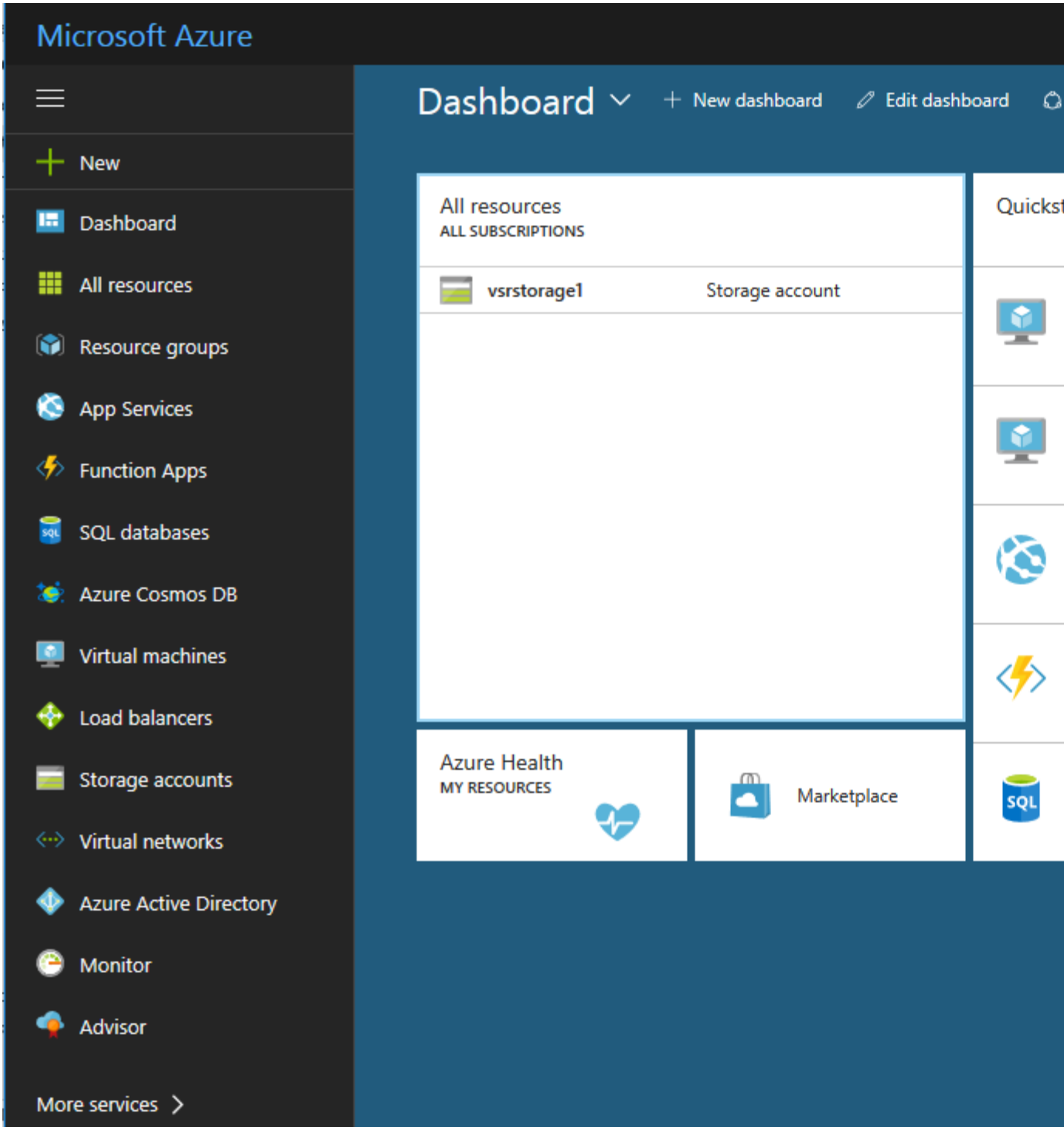
If you do not have an existing Azure virtual network, follow the steps outlined in this procedure.

To create an Azure virtual network:

1. Log in to the Microsoft Azure portal using your Microsoft account user name and password. The Dashboard appears in the Azure portal (see Figure 10 on page 43). You will see a unified dashboard

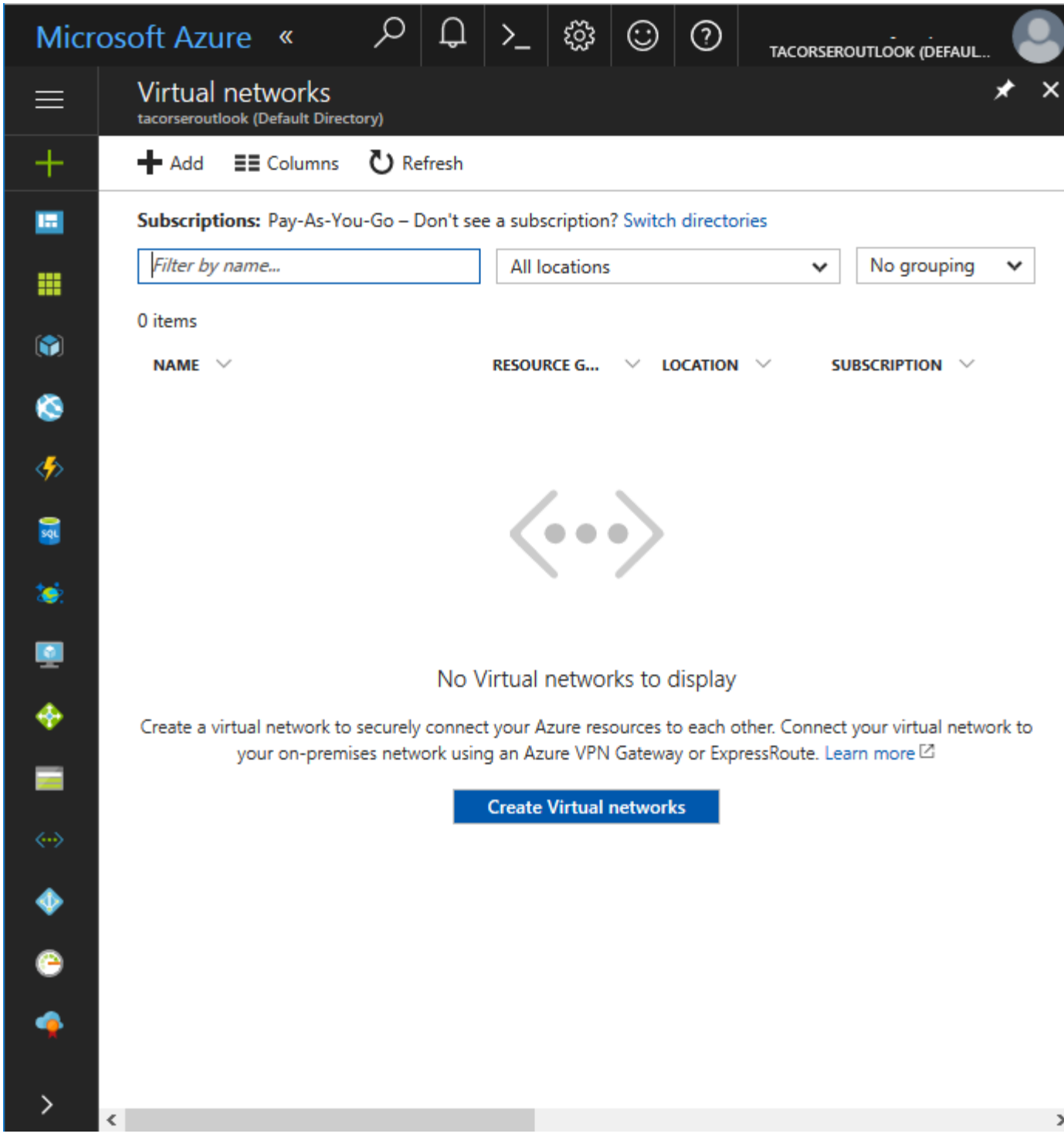
for all your assets in Azure. Verify that the dashboard includes all subscriptions to which you currently have access, and all resource groups and associated resources.

**Figure 10: Microsoft Azure Portal Dashboard**



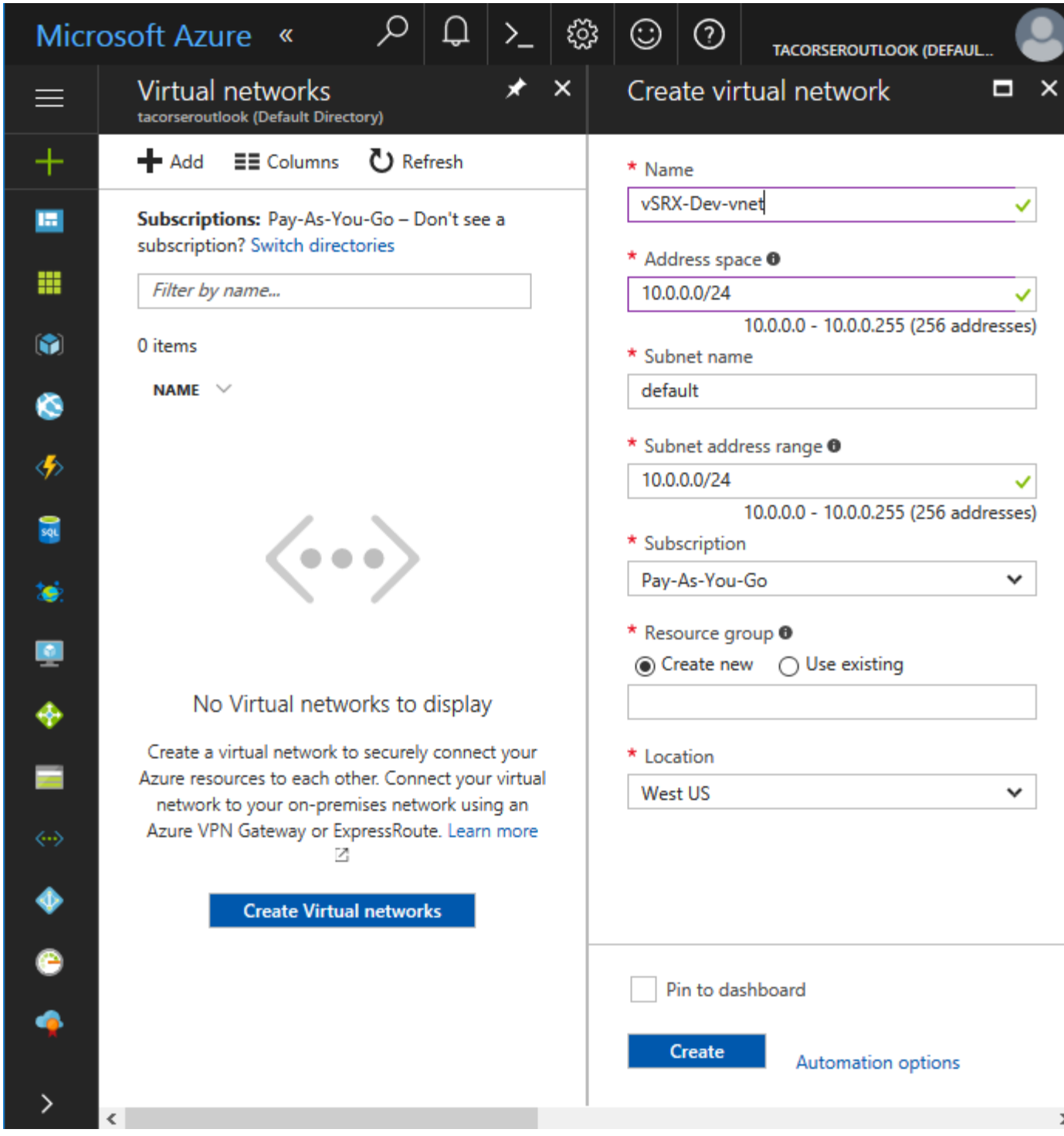
2. Click **Virtual Networks** from the menu of services to access the Virtual Networks blade (see Figure 11 on page 44).

**Figure 11: Azure Portal Virtual Networks**



3. Click **Add (+)** to create a new virtual network. The Create Virtual Network blade appears (see Figure 12 on page 45).

**Figure 12: Creating a Virtual Network**



4. Provide the following information for the new virtual network.

| Parameter | Description |
|---|---|
| Name | Enter a unique name for your new virtual network. The virtual network name must begin with a letter or number, end with a letter, number, or underscore, and the name may contain only letters, numbers, underscore, periods, or hyphens. |
| Address Space | Enter the virtual network's address range in CIDR notation. By default, the address range is 10.0.0.0/24.<br><br>**NOTE**: Ensure that the address space does not overlap with an existing network. |
| Subnet name | Enter a unique name for the subnet of the Azure virtual network. The subnet name must begin with a letter or number, end with a letter, number, or underscore, and the name may contain only letters, numbers, underscore, periods, or hyphens. |
| Subnet Address Range | Enter a network subnet address range in CIDR notation. It must be contained by the address space of the virtual network, as defined in the Address Space field. Subnet address ranges cannot overlap one another. By default, the address range is 10.0.0.0/24.<br><br>The subnet is a range of IP addresses in your virtual network to isolate VMs. Public subnets have access to the Internet gateway, but private subnets do not.<br><br>**NOTE**: The address range of a subnet that is already in use cannot be edited. |
| Subscription | Select your Microsoft Azure subscription. |
| Resource Group | Select an existing resource group or create a new one (see "Create a Resource Group" on page 32). |

*(Continued)*

| Parameter | Description |
|---|---|
| Location | Select the Azure data center geographic region in which you are deploying the vSRX VM. Typically, select the location that is closest to your physical location. |

5. Click **Create**. The virtual network might take a few seconds to create. Once it is created, you will see the virtual network on the Azure portal dashboard.

RELATED DOCUMENTATION

- Virtual networks and Windows virtual machines in Azure
- Create a virtual network
- Create, change, or delete network interfaces
- Create a VM (Classic) with multiple NICs

# Deploy the vSRX Image from Azure Marketplace

IN THIS SECTION



Starting in Junos OS Release 15.1X49-D91 for vSRX, you can deploy the vSRX virtual security appliance in your Azure virtual network by selecting the vSRX image from Azure Marketplace and customizing the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud.

This deployment approach might be needed if you have a vSRX VM deployment scenario that is outside of the use cases offered in the vSRX VM solution templates available from Juniper Networks.

**NOTE**: Be sure you have an account for and a subscription to Microsoft Azure before deploying the vSRX to Azure (see Microsoft Azure).

If you do not have an Azure subscription, then you can create a free account before you begin. See the Microsoft Azure website for more details.

Use the following procedures to deploy and configure a vSRX VM into an Azure virtual network from the Azure portal.

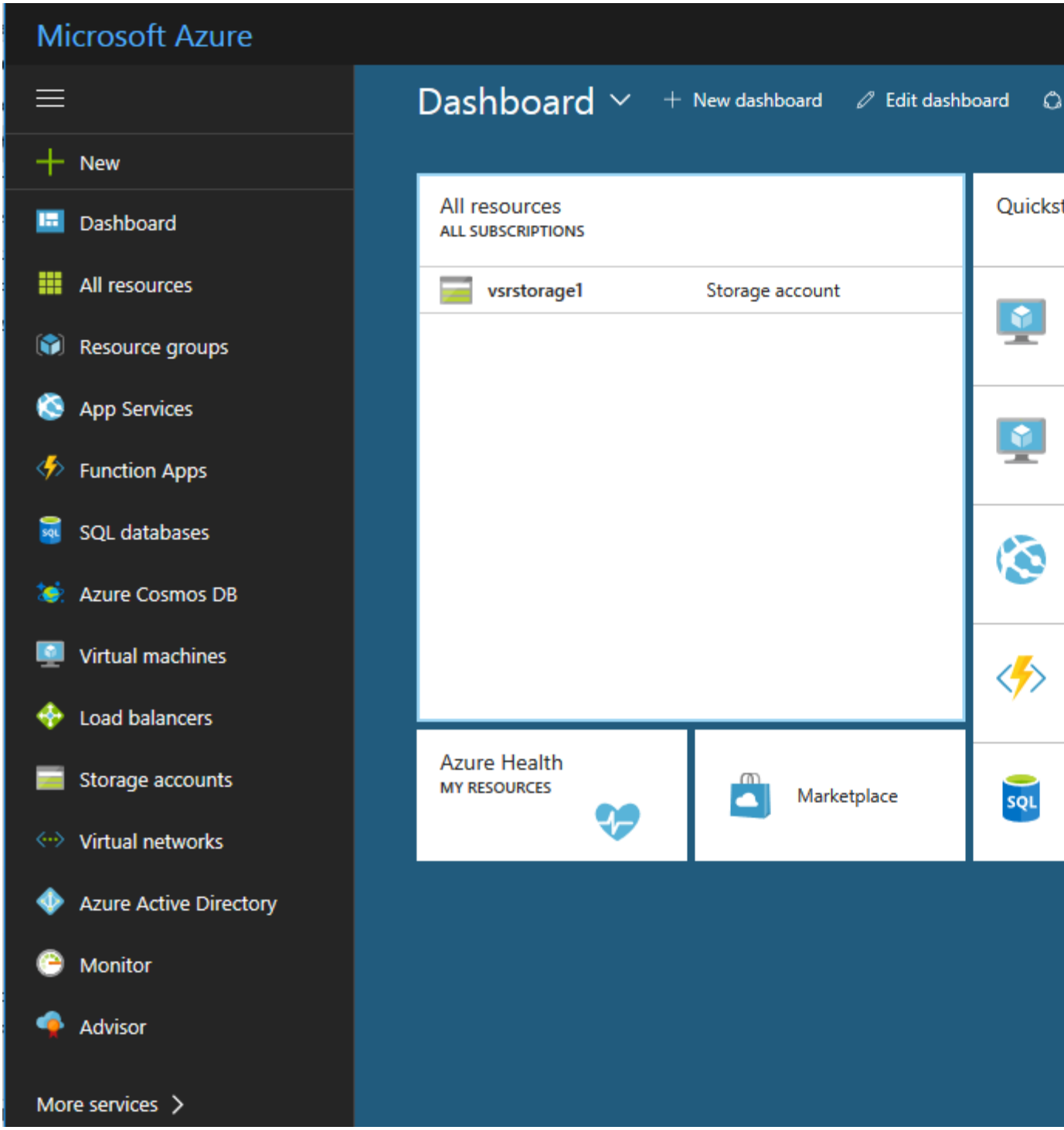
## Deploy the vSRX Image

To deploy and configure a vSRX VM into an Azure virtual network using the vSRX image from Azure Marketplace:

1. Log in to the Microsoft Azure portal using your Microsoft account user name and password. The Dashboard appears in the Azure portal (see Figure 13 on page 49). You will see a unified dashboard

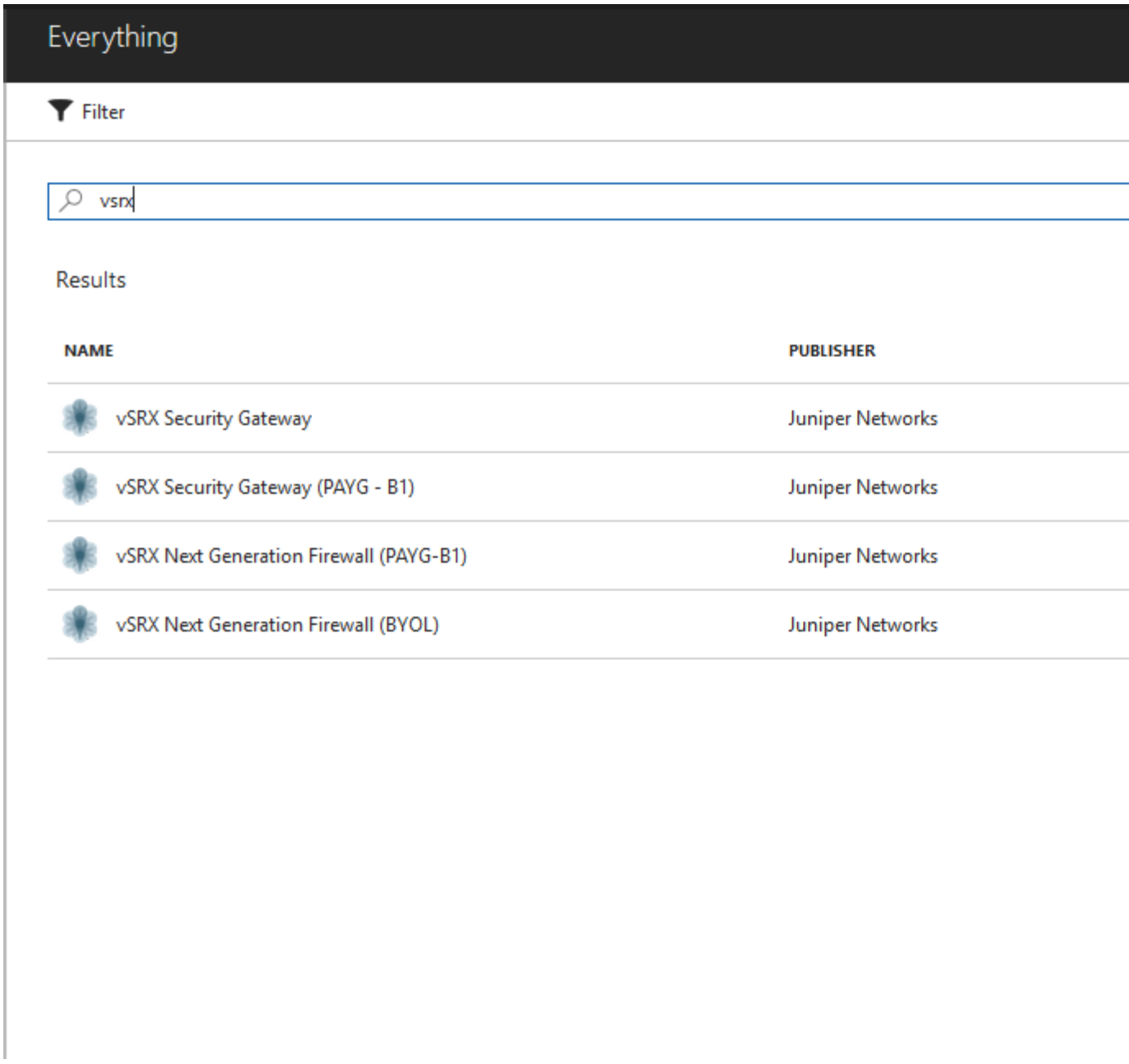
for all your assets in Azure. Verify that the dashboard includes all subscriptions to which you currently have access, and all resource groups and associated resources.

**Figure 13: Microsoft Azure Portal Dashboard**



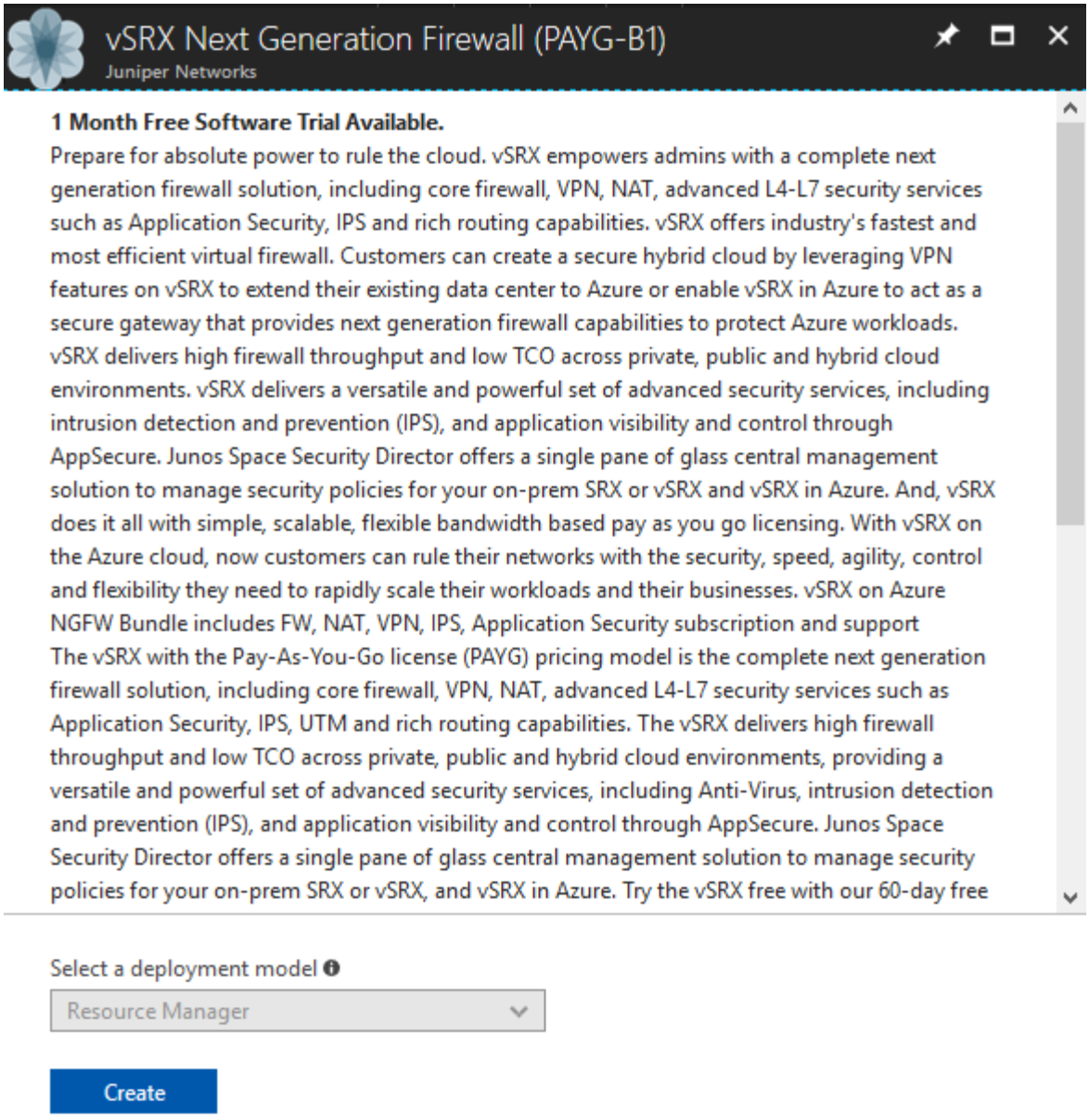
2. Click **Marketplace** from the dashboard to access the Azure Marketplace, and then click **Everything** (or click **New > Everything**). Enter **vsrx** to search for the available Juniper Networks vSRX VM images in the Azure Marketplace (see Figure 14 on page 50). The vSRX image is available as a pay-as-you-go (PAYG) or bring-your-own-license (BYOL) service.

**Figure 14: Locating the vSRX VM Image in the Azure Marketplace**



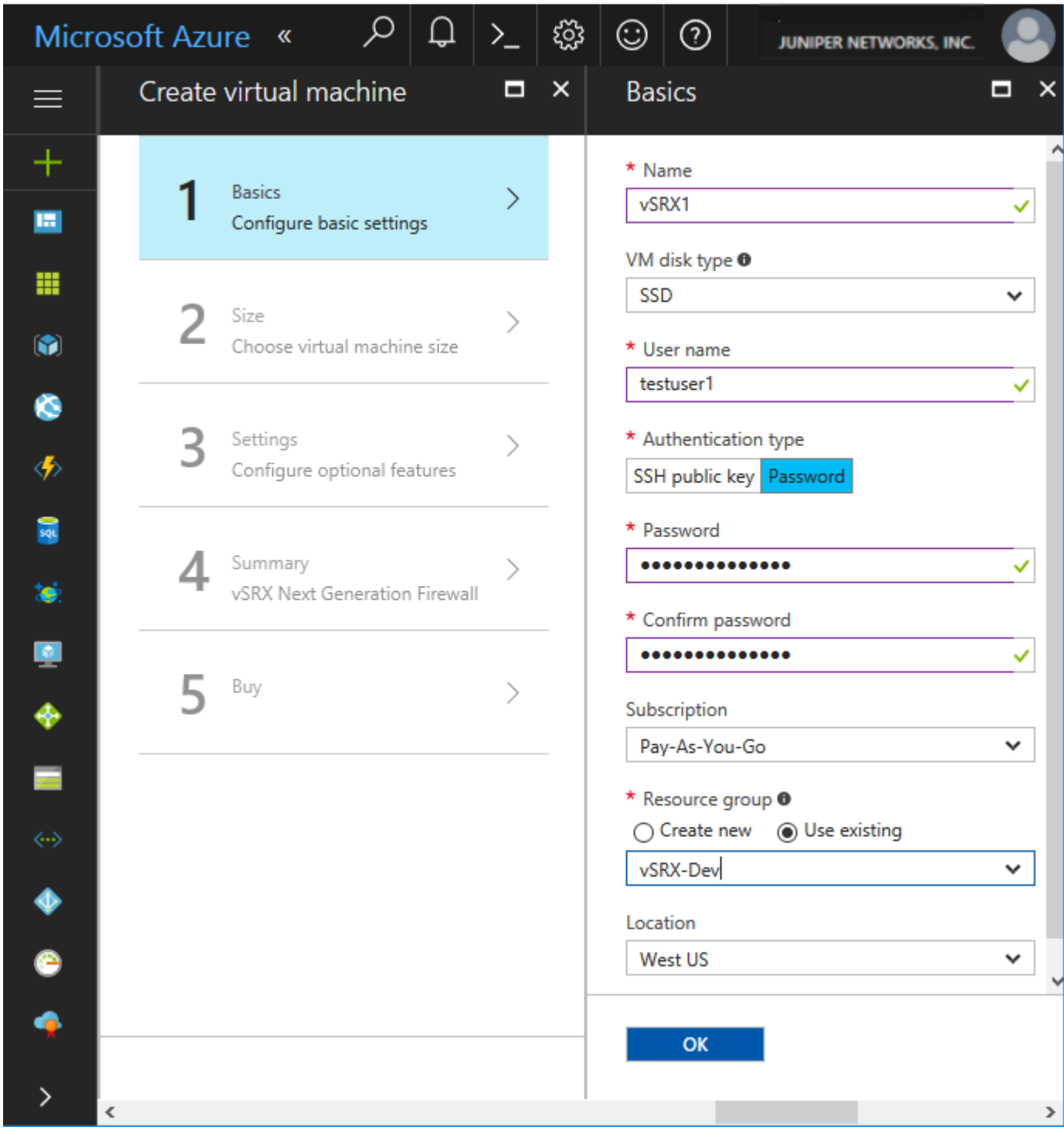
3. Select the vSRX VM image from the list and then click **Create** to initiate the vSRX VM deployment process (see Figure 15 on page 51).

**Figure 15: Initiating vSRX VM Deployment**



4. From the Create Virtual Machine blade, **1 Basics**, configure the following parameters (see Figure 16 on page 52).

**Figure 16: Create Virtual Machine - Basics**



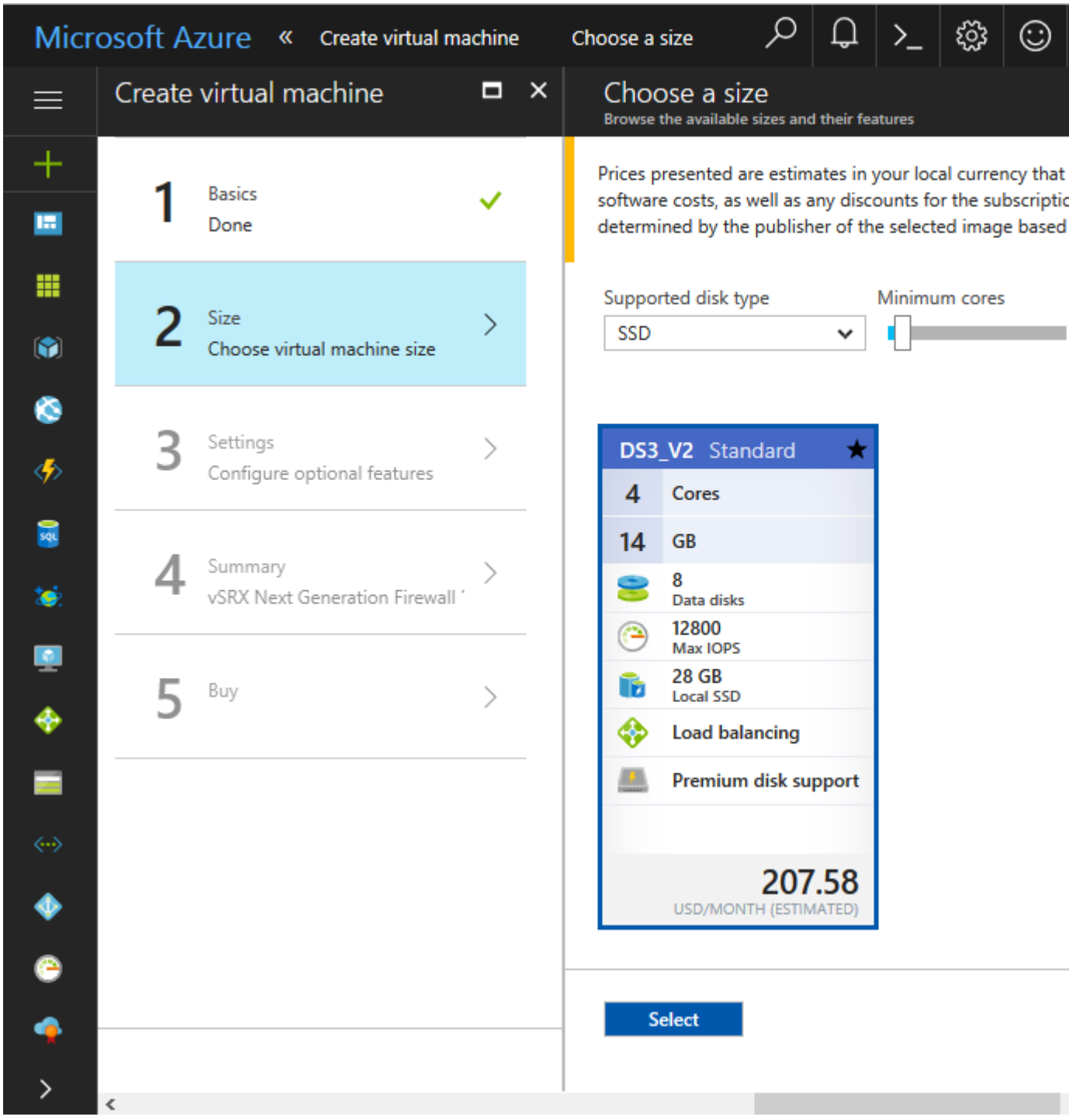
| Parameter | Description |
|---|---|
| Name | Specify a name for your vSRX VM. Your vSRX VM name cannot contain non-ASCII or special characters. |
| VM Disk Type | Specify the disk type to use for the vSRX VM: **SSD** or **HDD**. The default is **SSD**. |
| User name | Enter a username to access the vSRX VM. The username cannot contain uppercase characters, special characters, or start with a "$" or "-" character. |
| Authentication type | Select the required method of authentication to access the vSRX VM: **Password** or **SSH public key**. Select Password as type of authentication and then enter (and confirm) your password.<br>**NOTE**: In Junos OS Release 15.1X49-D91 for vSRX, SSH public key is not a supported authentication method. You will need to specify a password to log in to the vSRX VM.<br>Starting in Junos OS Release 15.1X49-D110 for vSRX, SSH public key is a supported authentication method. |
| Password | Enter an appropriate root password used to access the vSRX VM. |
| Subscription | Select your Microsoft Azure subscription. |
| Resource Group | Select an existing resource group or create a new one (see "Create a Resource Group" on page 32). |
| Location | Select the Azure geographic region in which you are deploying the vSRX VM. |

Click **OK**.

5. From the Create Virtual Machine blade, **2 Size**, select **DS3_v2 Standard** as the vSRX VM size (see Figure 17 on page 54). Click **Select**.

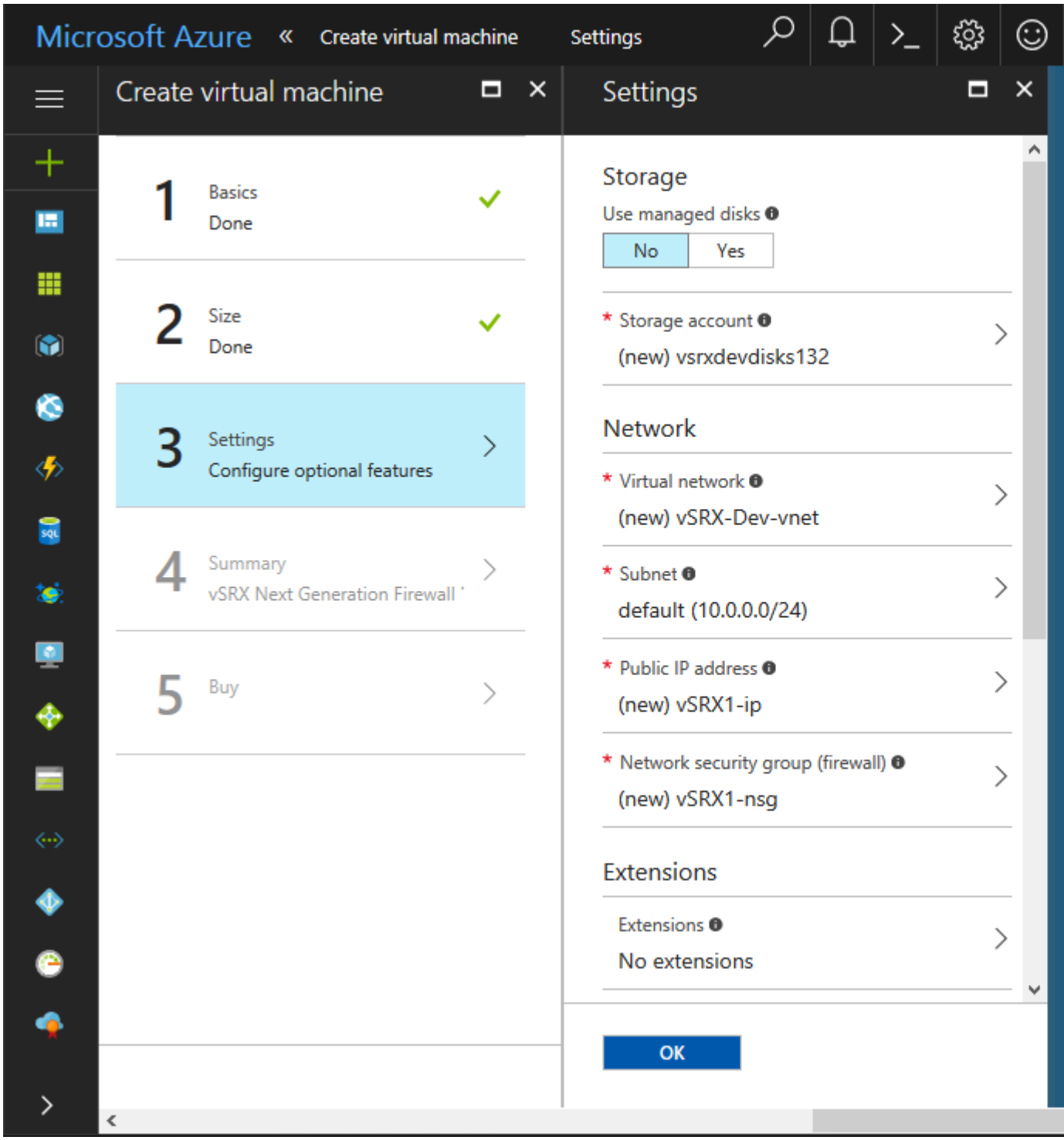
DS3_v2 Standard is used for a vSRX VM deployment. See "Requirements for vSRX on Microsoft Azure" on page 9 for the recommended system requirements for a vSRX instance in Microsoft Azure.

**Figure 17: Create Virtual Machine - Choose a Size**



6. From the Create Virtual Machine blade, **3 Settings**, configure the following parameters to define the storage, networking, and monitoring settings for the vSRX VM (see Figure 18 on page 56). Click **OK** when completed.

**Figure 18: Create Virtual Machine - Settings**



| Parameter | Description |
|---|---|
| Storage | |
| Used Managed Disks | Specify whether you want Azure to automatically manage the availability of disks to provide data redundancy and fault tolerance without you creating and managing a storage account. Click **No**. |
| Storage Account | If you need to change the storage account for the vSRX VM, click the right arrow to access the Choose Storage Account blade. Select an existing storage account for the vSRX VM, or click **Create new (+)** to create a new one. See "Create a Storage Account" on page 36 for details about creating a new storage account. |
| Network | |
| Virtual Network | If you need to change the virtual network for the vSRX VM, click the right arrow to access the Choose Virtual Network blade. Select an existing virtual network for the vSRX VM, or click **Create new (+)** to create a new one. See "Create a Virtual Network" on page 42 for details about creating a new virtual network. |

*(Continued)*

| Parameter | Description |
|---|---|
| Subnet | Enter a subnet, which is a range of IP addresses in your virtual network to isolate VMs. Public subnets have access to the Internet gateway, but private subnets do not.<br><br>A vSRX VM requires two public subnets and one or more private subnets for each individual instance group. The public subnets consist of one for the management interface (fxp0) and another for the two revenue (data) interfaces. The private subnets, connected to other vSRX interfaces, ensure that all traffic between applications on the private subnets and the Internet must pass through the vSRX instance.<br><br>To modify the subset for the virtual network, click the right arrow to access the Create Subnet blade.<br><br>Configure the following parameters:<br><br>• Subnet name—A unique name for the subnet in the Azure virtual network.<br><br>• Subnet address range—The subnet's address range in CIDR notation. It must be contained by the address space of the virtual network. Subnet address ranges cannot overlap one another. By default, the address range is 10.0.0.0/24.<br><br>**NOTE**: The address range of a subnet that is already in use cannot be edited. |

*(Continued)*

| Parameter | Description |
| --- | --- |
| Public IP address | Specify the public IP address that allows communication to the vSRX VM from outside the Azure virtual network. To modify the public IP address for the vSRX VM, click the right arrow to access the Choose Public IP Address blade. Select a public IP address in your Azure subscription and location, or click **Create new (+)** to create a new one.<br><br>Configure the following parameters:<br><br>• Name—A unique name for the public IP address.<br><br>• Assignment—There are two methods in which an IP address is allocated to a public IP resource: dynamic or static. By default, public IP addresses are dynamic, where an IP address is not allocated at the time of its creation. Instead, the public IP address is allocated when you start (or create) the resource. The IP address associated to them may change when the vSRX VM is deleted.<br><br>To guarantee that the vSRX VM always uses the same public IP address, we recommend you assign a static public IP address. |

*(Continued)*

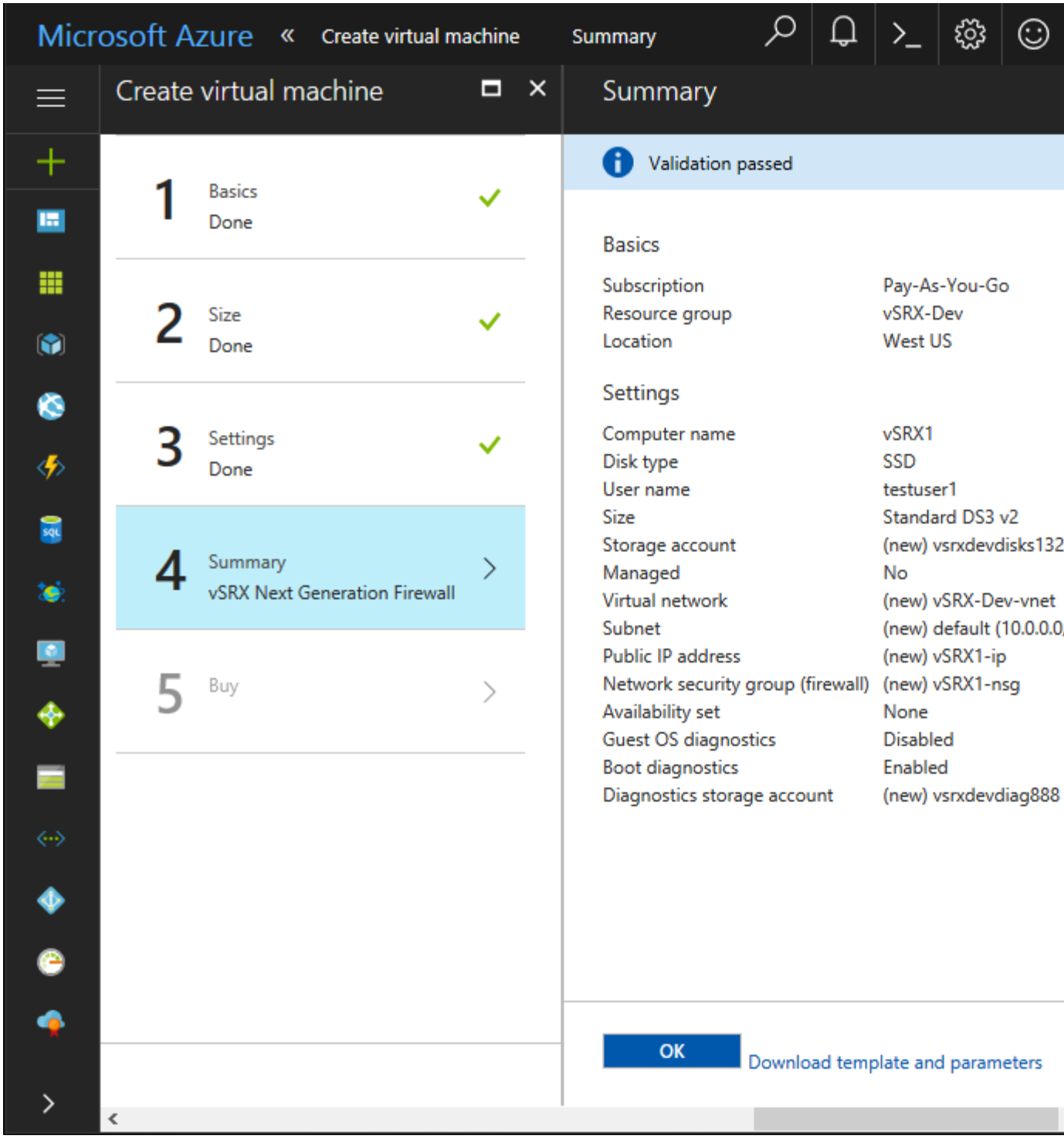
| Parameter | Description |
|---|---|
| Network security group | Specify a network security group, which is a set of firewall rules that control traffic to and from the vSRX VM. Each network security group can contain multiple inbound and outbound security rules that enable you to filter traffic by source and destination IP address, port, and protocol. You can apply a network security group to each NIC in the VM.<br><br>To modify the network security group for the vSRX VM to filter traffic, click the right arrow to access the Choose Network Security blade. Select a network security group in your Azure subscription and location, or click **Create new (+)** to create a new one.<br><br>Configure the following parameters:<br><br>• Name—A unique name for the network security group.<br><br>• Inbound rules—You can add one or more inbound security rules to allow or deny traffic to the vSRX VM.<br><br>• Outbound rules—You can add one or more outbound security rules to allow or deny traffic originating from the vSRX VM. |
| Extensions | |
| Extensions | No extensions are used for the vSRX VM. |
| High Availability | |
| Availability Set | Configure two or more VMs in an availability set to provide redundancy to an application.<br><br>**NOTE**: Availability Set should be set to **None** for the vSRX VM. Availablilty Set is not used for the vSRX VM in Azure because chassis clustering is not supported by the vSRX at this time. |

*(Continued)*

| Parameter | Description |
|---|---|
| Monitoring | |
| Boot Diagnostics | Enables or disables the capturing of serial console output and screenshots of the VM running on the host to help diagnose start-up issues. The default is Enabled. |
| Guest OS Diagnostics | Enables or disables the ability to obtain metrics every minute for the VM. Choices are: **Disabled** or **Enabled**. The default is Disabled. |
| Diagnostics Storage Account | Click the right arrow to view the details of the diagnostics storage account. Automatically fills in with the name of the diagnostics storage account from which you can analyze a set of metrics with your own tools. |

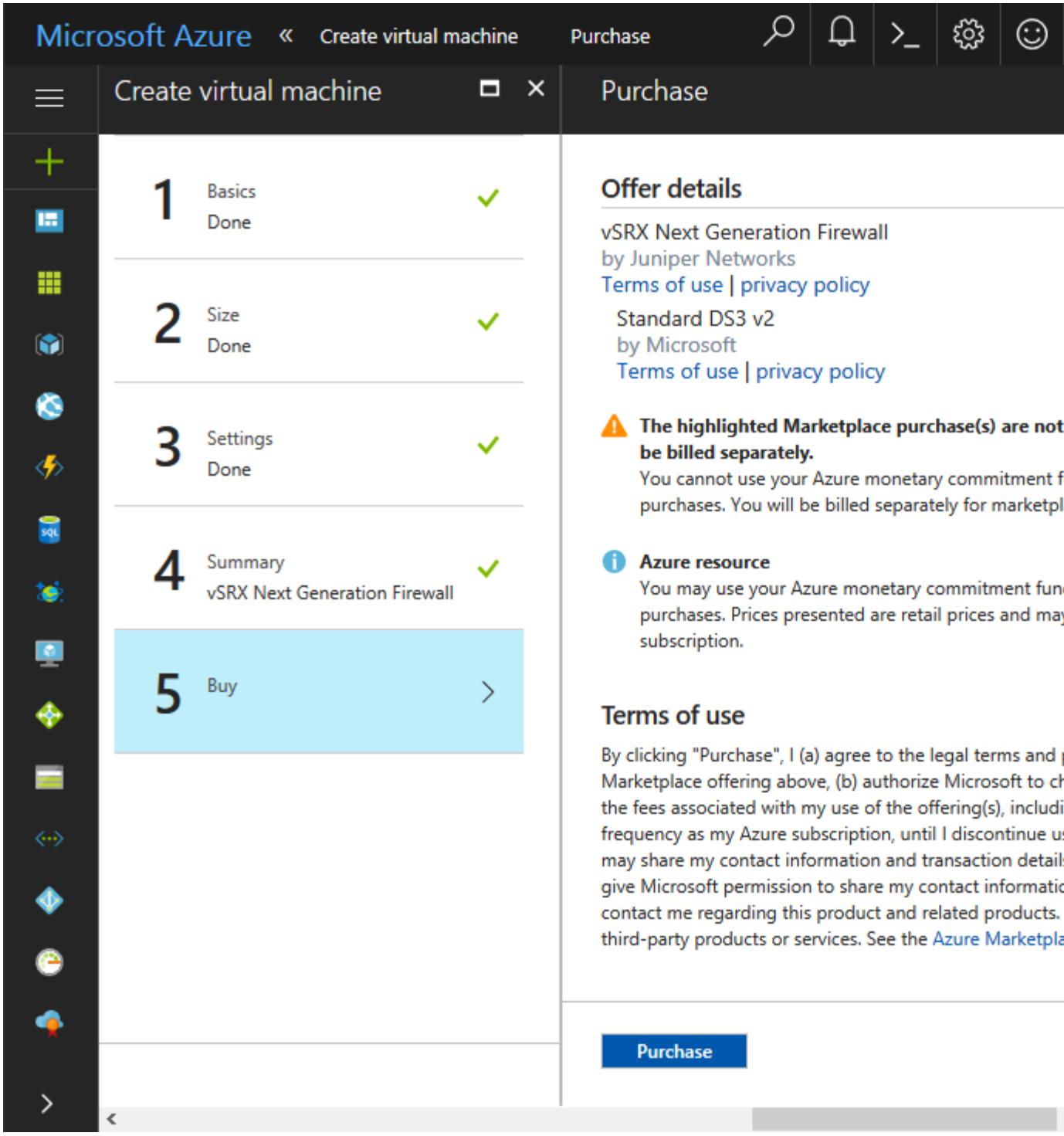
7. From the Create Virtual Machine blade, **4 Summary** , review the configuration settings (see Figure 19 on page 62). If you are satisfied with the configuration settings, click **OK**.

**Figure 19: Create Virtual Machine - Summary**



8. From the Create Virtual Machine blade, **5 Buy** review the offer details and the terms of use (see Figure 20 on page 63). If you are satisfied with the offer details and terms of use, click **Purchase**.

**Figure 20: Create Virtual Machine - Purchase**



You return to the Azure portal dashboard, and the dashboard displays the deployment status of the vSRX VM.

## Verify Deployment of vSRX to Microsoft Azure

After the vSRX VM is created, the Azure portal dashboard lists the new vSRX VM under Resource Groups. The corresponding cloud service and storage account also are created and listed. Both the vSRX VM and the cloud service are started automatically and their status is listed as **Running**

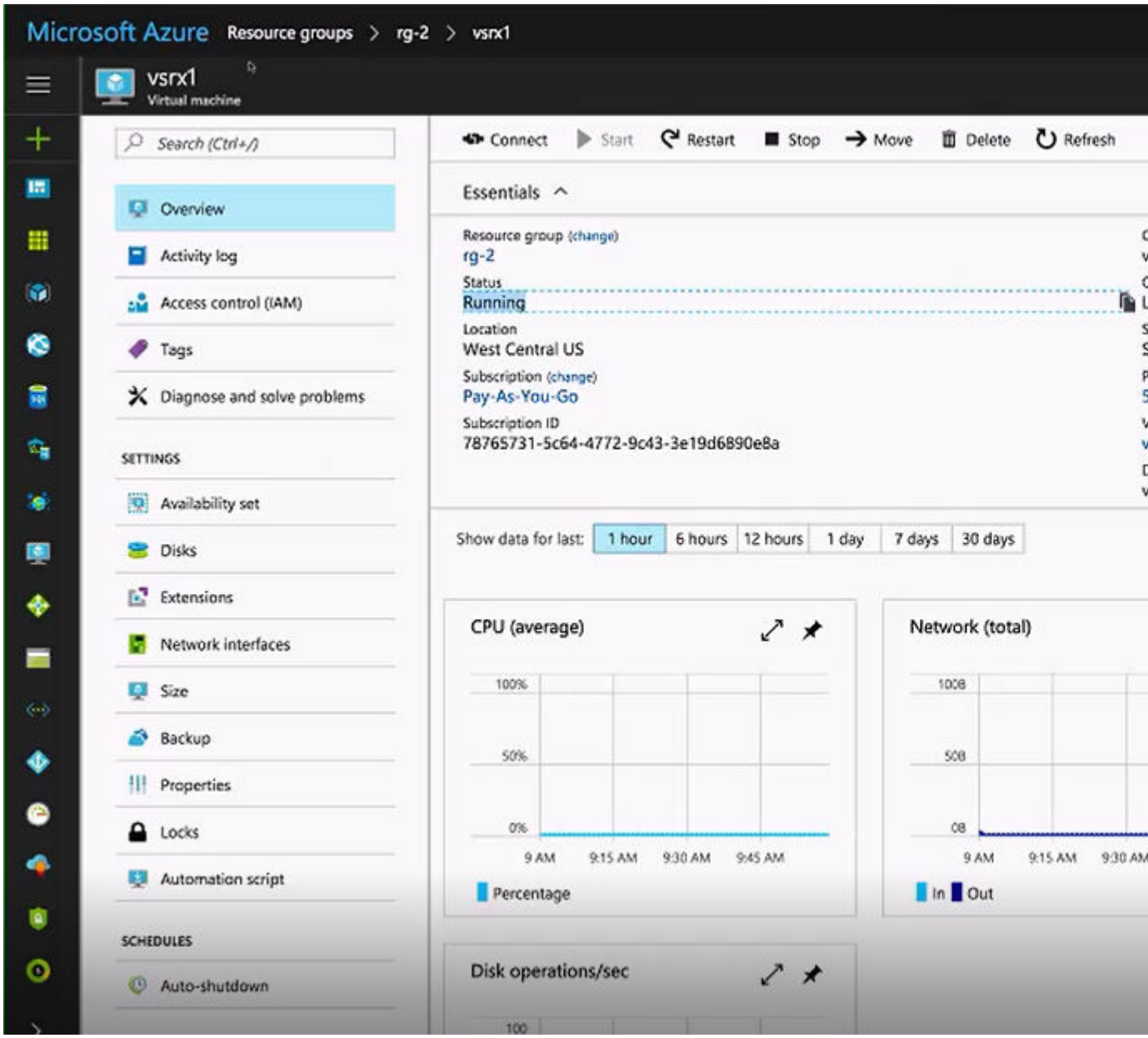
To verify the deployment of the vSRX instance to Microsoft Azure:

1. To view the vSRX resource group and its resources after deployment is completed, from the right-hand menu, click **Resource groups** to access the Resource Groups page.
2. To view details of the vSRX VM associated with the resource group, click the name of the vSRX VM. Observe that the status is **Running**.

> **NOTE**: You can stop, start, restart, and delete a vSRX VM from the Virtual Machine page in the Microsoft Azure portal.

Figure 21 on page 65 shows an example of a Resource groups vSRX VM in the Microsoft Azure portal.

**Figure 21: Microsoft Azure Resource Groups VM Example**



## Log In to a vSRX VM

After vSRX deployment is completed, the vSRX VM is automatically powered on and launched. At this point you can use an SSH client to log in to the vSRX VM.

> **NOTE**: In Microsoft Azure, individuals and enterprises can host servers and services on the cloud as a pay-as-you-go (PAYG) or bring-your-own-license (BYOL) service. For the vSRX on Microsoft Azure deployment, only the BYOL model is supported.

To log in to the vSRX VM:

1. From the Azure portal, click **Resource groups** from the menu of services on the dashboard, and then select the vSRX VM. Locate the public IP address of the vSRX VM from the Settings blade.
2. Use an SSH client to log in to a vSRX VM.
3. At the prompt, enter the following login credentials:

   > **NOTE**: The vSRX instance is automatically configured for username and password authentication. To log in, use the login credentials that were defined during the vSRX VM configuration (see "Deploy the vSRX Image" on page 48). After initially logging in to the vSRX, you can configure SSH public and private key authentication.

   **# ssh <username@vsrx_vm_ipaddress>**

   ```
   The authenticity of host 'x.x.x.x (x.x.x.x)' ...
   ECDSA key fingerprint is SHA256:XXXXXXXXXXXXXXXXXXXXXXX.
   Are you sure you want to continue connecting (yes/no)? yes
   Warning: Permanently added 'x.x.x.x' (ECDSA) to the list of known hosts.
   Password: xxxxxxxx
   username@vsrx_vm_ipaddress>
   ```

4. Configure the basic settings for the vSRX VM (see "Configure vSRX Using the CLI" on page 92).

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D91 | Starting in Junos OS Release 15.1X49-D91 for vSRX, you can deploy the vSRX virtual security appliance in your Azure virtual network by selecting the vSRX image from Azure Marketplace and customizing the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud. |

## RELATED DOCUMENTATION

- How to Deploy in Microsoft Azure using Azure Portal and Template
- Microsoft Azure portal overview
- Virtual networks and Windows virtual machines in Azure
- Create, change, or delete network interfaces
- Create a VM (Classic) with multiple NICs

CHAPTER 3

# Deploying vSRX from the Azure CLI

# Before You Deploy vSRX Using the Azure CLI

Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX from the Azure CLI and customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud.

To help automate and simplify the deployment of the vSRX in the Microsoft Azure virtual network, Juniper Networks provides a series of scripts, Azure Resource Manager (ARM) templates and parameter files, and configuration files in the GitHub repository https://github.com/Juniper/vSRX-Azure. The ARM template includes resource parameters that enable you to customize your vSRX VM deployment, such as login credentials, network interfaces, and storage container name. The template consists of JavaScript Object Notation (JSON) expressions for your vSRX deployment.

The vSRX deployment files in the GitHub repository include:

- The **deploy-azure-vsrx.sh** shell script to automate the deployment and configuration of the vSRX virtual machine (VM).
- The **vsrx.json** template file to define the components of the Azure resource group and virtual hardware settings (VM size, interface number and network) of the vSRX VM.
- The **vsrx.parameters.json** parameter file to identify the network interface parameters used to deploy the vSRX VMin Azure.

Before you deploy the vSRX virtual security appliance from the Azure CLI:

- Review the requirements for deploying a vSRX VM in Microsoft Azure Cloud in "Requirements for vSRX on Microsoft Azure" on page 9.
- Obtain an account for and a subscription to Microsoft Azure (see Microsoft Azure).
- From the Azure portal, you must first manually deploy the vSRX image (only once) by using either the **vSRX Next Generation Firewall (BYOL)** or the **vSRX Next Generation Firewall (PAYG)** SKU to accept the EULA terms. This is a requirement before you can deploy the vSRX image from the Azure CLI. By default, the Azure portal deployment tool uses **vSRX Next Generation Firewall (BYOL)** SKU as the source image. Use your Microsoft account username and password to log into the Microsoft Azure portal.

  **NOTE**: You will encounter a **MarketplacePurchaseEligibilityFailed** error if do not first accept the EULA terms for the vSRX image in the Azure portal before attempting to deploy the vSRX image from the Azure CLI.

- Install Azure command line interface (Azure CLI) 1.0 and enable Azure Resource Management (ARM) mode (see Install the Azure CLI).

  **NOTE**: The vSRX for Azure deployment shell script **deploy-azure-vsrx.sh** is written in shell and Azure CLI version 1.0 commands and does not support Azure CLI version 2.0.

- Purchase a vSRX license or request an evaluation license. Licenses can be procured from the Juniper Networks License Management System (LMS).

**NOTE**: Deployment of vSRX to Microsoft Azure does not support the use of the Azure CLI from Microsoft Windows. This is because the deploy-azure-vsrx.sh shell script that is used as part of the deployment procedure can be run only from the Linux or Mac OS CLI.

When you deploy a vSRX VM in an Azure virtual network, note the following specifics of the deployment configuration:

- Use your Microsoft account username and password to log into the Microsoft Azure portal.
- Ensure that your Azure subscription includes the following for your vSRX VM:
  - Resource group, as described in "Create a Resource Group" on page 32.
  - Storage account, as described in "Create a Storage Account" on page 36.
  - Virtual network, as described in "Create a Virtual Network" on page 42.

vSRX deployment from the Azure CLI is described in detail in "Deploy vSRX from the Azure CLI" on page 71.

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX from the Azure CLI and customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud. |

## RELATED DOCUMENTATION

Azure Resource Manager overview

Deploy resources with Resource Manager templates and Azure CLI

# Deploy vSRX from the Azure CLI

IN THIS SECTION



Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX from the Azure CLI and customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud.

Use the following procedure to deploy and configure vSRX as a virtual security appliance in a Microsoft Azure virtual network from the Azure CLI. In this procedure, you use the Azure CLI running in Azure Resource Manager (ARM) mode.

**NOTE**: Be sure you have an account for and a subscription to Microsoft Azure before deploying the vSRX to Azure (see Microsoft Azure).

If you do not have an Azure subscription, then you can create a free account before you begin. See the Microsoft Azure website for more details.

**NOTE**: From the Azure portal, you must first manually deploy the vSRX image (only once) by using either the **vSRX Next Generation Firewall (BYOL)** or the **vSRX Next Generation Firewall (PAYG)** SKU to accept the EULA terms. This is a requirement before you can deploy the vSRX image from the Azure CLI. By default, the Azure portal deployment tool uses **vSRX Next Generation Firewall (BYOL)** SKU as the source image. Use your Microsoft account username and password to log into the Microsoft Azure portal.

You will encounter a **MarketplacePurchaseEligibilityFailed** error if do not first accept the EULA terms for the vSRX image in the Azure portal before attempting to deploy the vSRX image from the Azure CLI.

## Install the Microsoft Azure CLI

To install and log in to the Microsoft Azure CLI:

1. Install the Microsoft Azure CLI 1.0 as outlined in Install the Azure CLI. You have several options to install the Azure CLI package for either the Linux or Mac OS; be sure to select the correct installation package.

   **NOTE**: The vSRX for Azure deployment shell script **deploy-azure-vsrx.sh** is written in shell and Azure CLI version 1.0 commands and does not support Azure CLI version 2.0.

   **NOTE**: Deployment of vSRX to Microsoft Azure does not support the use of the Azure CLI from Microsoft Windows. This is because the **deploy-azure-vsrx.sh** shell script that is used as part of the deployment procedure can be run only from the Linux or Mac OS CLI.

2. Log into the Azure CLI.

   **> azure login**

3. At the prompt. copy the code that appears in the command output.

```
Executing command login
To sign in, use a web browser to open the page http://aka.ms/devicelogin.
Enter the codeXXXXXXXXX to authenticate
```

4. Open a Web browser to http://aka.ms/devicelogin, enter the code, and then click **Continue**. Enter your Microsoft Azure username and password credentials. When the process completes, the command shell completes the login process.

```
Added subscription Microsoft Azure Enterprise
To sign in, use a web browser to open the page http://aka.ms/deviceloginlogin
command OK
```

> **NOTE**: If you have multiple Azure subscriptions, connecting to Azure grants access to all subscriptions associated with your credentials. One subscription is selected as the default, and used by the Azure CLI when performing operations. You can view the subscriptions, including the current default subscription, using the **azure account list** command.

5. Ensure that the Azure CLI is in Azure Resource Manager (ARM) mode.

   **> azure config mode arm**

> **NOTE**: When the Azure CLI is initially installed, the CLI is in ARM mode.

## Download the vSRX Deployment Tools

Juniper Networks provides a set of scripts, templates, parameter files, and configuration files in Juniper's GitHub repository. These tools are intended to help simplify the deployment of the vSRX to Azure when using the Azure CLI.

> **NOTE**: For background information on the scripts, templates, parameter files, and configuration files, see "Before You Deploy vSRX Using the Azure CLI" on page 69.

To download the vSRX deployment tools:

1. Access GitHub by using the following link: https://github.com/Juniper/vSRX-Azure.

2. Click **Clone or download** to download to you computer the **vSRX-Azure-master.zip** file from Github containing all files and directories from **vSRX-Azure**. The **vSRX-Azure-master** directory includes the following directories and files:

```
vSRX-Azure-master
├── README.md
├── LICENSE
├── sample-templates
│   ├── arm-templates-tool
│       ├── README.md
│       ├── deploy-azure-vsrx.sh
│       ├── templates
│       │   ├── app-vm
│       │   │   ├── vm.json
│       │   │   └── vm.parameters.json
│       │   └── vsrx-gateway
│       │       ├── vsrx.json
│       │       └── vsrx.parameters.json
│       └── utils
│           ├── decode_param_file.py
│           ├── gen_param_file.py
│           └── gen_template_file.py
│   ├── simple-vsrx-demo
│       ├── README.md
│       ├── vsrx.json
│       ├── vsrx.parameters.json
└── marketplace-solution-templates
    └── vpn-gateway
        ├── createUiDefinition.json
        ├── mainTemplate.json
        ├── vSRX-password.json
        └── vSRX-sshPublicKey.json
```

3. Extract the compressed **vSRX-Azure-master.zip** file to a location on your computer.

## Change Parameter Values in the vsrx.parameter.json File

In the **vsrx.parameters.json** file, you need to modify parameter values specific to your vSRX deployment in Microsoft Azure. These parameters are used as part of the automatic deployment performed by the **deploy-azure-vsrx.sh** script.

Keep in mind that by default vSRX uses fxp0 as the egress interface to the Internet. For features requiring Internet connections that use a revenue port (such as VPN, UTM, and so on), routing instances are required to isolate the traffic between the management network and the revenue network.

To change parameter values in the **vsrx.parameters.json** file:

1. Open the **vsrx.parameters.json** file with a text editor.
2. Modify the values in the **vsrx.parameters.json** file based on the specifics of your vSRX deployment. As an example, the following table outlines the parameters in the **vsrx.parameters.json** file found in **sample-templates\arm-templates-tool\templates\vsrx-gateway** that might require modification.

> **CAUTION**: It is critical that you change the vsrx-username and vsrx-password login credentials listed in the **vsrx.parameters.json** file before you launch the vSRX instance and login for the first time. Note that you cannot reset login credentials for the vSRX using the Microsoft Azure portal or the Azure CLI.

| Parameter | Default Value | Comment |
|---|---|---|
| *storageAccountName* | juniperstore01 | Must be unique for each deployment. |
| *storageContainerName* | vhds | Name of the Microsoft Azure storage container (VHDs). |
| *vsrx-name* | vsrx-gw | Specifies the vSRX hostname. |
| *vsrx-addr-ge-0-0-0* | 192.168.10.20 | IP address of vSRX interface ge-0/0/0.0. |
| *vsrx-addr-ge-0-0-1* | 192.168.20.20 | IP address of vSRX interface ge-0/0/1.0. |

*(Continued)*

| Parameter | Default Value | Comment |
|---|---|---|
| *vsrx-username* | demo | Change to an appropriate username for the login credentials used to access the vSRX. |
| *vsrx-password* | Demo123456 | Change to an appropriate password for the login credentials used to access the vSRX. |
| *vsrx-sshkey* | ssh-rsa placeholder | Specifies the root authentication password for the vSRX VM by entering an SSH public key string ( RSA or DSA). By default, the **deploy-azure-vsrx.sh** deployment script selects the password authentication method, unless **–p**, followed by the SSH RSA public key file (id_rsa.pub by default), is specified.<br><br>**NOTE**: Starting in Junos OS Release 15.1X49-D100 for vSRX, both password and SSH public key authentication are supported, and password authentication is chosen by default. |

*(Continued)*

| Parameter | Default Value | Comment |
|---|---|---|
| *vsrx-disk* | placeholder | The source image to create the vSRX instance. By default, the **deploy-azure-vsrx.sh** script uses the **vSRX Next Generation Firewall (BYOL)** SKU in the Azure Marketplace as the source image to deploy vSRX instance, unless **–i** is used to explicitly specify the vSRX instance image location. |
| *vnet-prefix* | 192.168.0.0/16 | IP address prefix of the virtual network. |
| *vnet-mgt-subnet-basename* | mgt-subnet | Name of management network connected to fxp0. |
| *vnet-mgt-subnet-prefix* | 192.168.0.0/24 | IP address prefix of management network connected to fxp0. |
| *vnet-trust-subnet-basename* | trust-subnet | Name of network connected to trust security zone: ge-0/0/1.0 on the vSRX. |
| *vnet-trust-subnet-prefix* | 192.168.20.0/24 | IP address prefix of network connected to trust security zone: ge-0/0/1.0 on the vSRX. |
| *vnet-untrust-subnet-basename* | untrust-subnet | Name of network connected to untrust security zone: ge-0/0/0.0 on the vSRX. |

*(Continued)*

| Parameter | Default Value | Comment |
|---|---|---|
| *vnet-untrust-subnet-prefix* | 192.168.10.0/24 | IP address prefix of network connected to untrust security zone: ge-0/0/0.0 on the vSRX. |

3. Save your changes to the **vsrx.parameters.json** file.

## Deploy the vSRX Using the Shell Script

The **deploy-azure-vsrx.sh** shell script deploys the vSRX virtual machine in a resource group that is based on your Azure Cloud geographic location. The script uses the storage account and network values defined in the **vsrx.parameters.json** file.

To deploy vSRX to the Azure virtual network:

1. At the bash prompt in the Azure CLI, run the **deploy-azure-vsrx.sh** script. By default, the script deploys the vSRX VM using the **vSRX Next Generation Firewall (BYOL)** SKU as the source image from the Azure Marketplace. The following information is read from the vsrx.json file as part of the deployment:
   - VM Size: Standard_D3_v2
   - Publisher: Juniper Networks
   - SKU: vsrx-byol-azure-image
   - Offering: vsrx-next-generation-firewall

   The following is an example of the command syntax. In this example, the script uses the vSRX image to deploy the vSRX VM in resource group “example_rg” at the Azure location “westus.” The storage account and network values are defined in the **vsrx.parameters.json** file.

   **> ./deploy-azure-vsrx.sh -g example_rg -l westus -f vSRX-Azure/sample-templates/arm-templates-tool/templates/vsrx-gateway/vsrx.json -e vSRX-Azure/sample-templates/arm-templates-tool/templates/vsrx-gateway/vsrx.parameters.json**

**NOTE**: When you specify the vSRX source image URL with the option **-i**, the script copies the vSRX source image to create the virtual hardware disk file and to set the **vsrx-disk** parameter in **vsrx.parameters.json** to this value.

The default parameter values in the command syntax include:

- *example_rg* is the resource group name (**-g**).
- *westus* is the Azure location (**-l**).
- *vsrx.json* in the folder **vSRX-Azure/sample-templates/arm-templates-tool/templates/vsrx-gateway** is the default Azure template file (**-f**).
- *vsrx.parameters.json* in the folder **vSRX-Azure/sample-templates/arm-templates-tool/templates/vsrx-gateway** is the default parameter file (**-e**).

2. Monitor the stages of deployment of vSRX to Microsoft Azure as they occur on screen. Deployment encompasses operations such as creating a resource group, storage account, template group (including configuration parameters).

**NOTE**: Creation of the storage account can take approximately 3 to 5 minutes on average. However, in some cases, it might take as long as 15 to 20 minutes.

```
➜  arm-templates-tool ./deploy-azure-vsrx.sh
Use default resource group name 'vsrx'
info:    Executing command config mode
info:    New mode is arm
info:    config mode command OK
info:    Executing command group create
+ Getting resource group vsrx
+ Creating resource group vsrx
info:    Created resource group vsrx
data:    Id:                  /subscriptions/1c3367ba-71fc-48df-898a-
d9eab4f1d673/resourceGroups/vsrx
data:    Name:                vsrx
data:    Location:            westus
data:    Provisioning State:  Succeeded
data:    Tags: null
data:
info:    group create command OK
```

```
info:    Executing command storage account create
…
data:    DeploymentName     : deployvsrx
data:    ResourceGroupName  : vsrx
data:    ProvisioningState  : Succeeded
data:    Timestamp          : Thu Jul 20 2017 12:31:45 GMT+0800 (CST)
data:    Mode               : Incremental
data:    CorrelationId      : a99b89f8-5919-4dbc-b8a5-6d76b30fcb67
data:    DeploymentParameters :
data:    Name                           Type          Value
data:    -----------------------------  ------------  -------------------
data:    storageAccountName             String        jnprsa01
data:    storageContainerName           String        vhds
data:    vsrx-name                      String        vsrx-test01
data:    vsrx-addr-ge-0-0-0             String        192.168.10.20
data:    vsrx-addr-ge-0-0-1             String        192.168.20.20
data:    vsrx-username                  String        demo
data:    vsrx-password                  SecureString  undefined
data:    vsrx-sshkey                    String        ssh-rsa placeholder
data:    vsrx-disk                      String        placeholder
data:    vnet-prefix                    String        192.168.0.0/16
data:    vnet-mgt-subnet-basename       String        mgt-subnet
data:    vnet-mgt-subnet-prefix         String        192.168.0.0/24
data:    vnet-trust-subnet-basename     String        trust-subnet
data:    vnet-trust-subnet-prefix       String        192.168.20.0/24
data:    vnet-untrust-subnet-basename   String        untrust-subnet
data:    vnet-untrust-subnet-prefix     String        192.168.10.0/24
info:    group deployment create command OK
```

When the deployment process completes, you will see the message **“info: group deployment create command Ok**.

## Verify Deployment of vSRX to Microsoft Azure

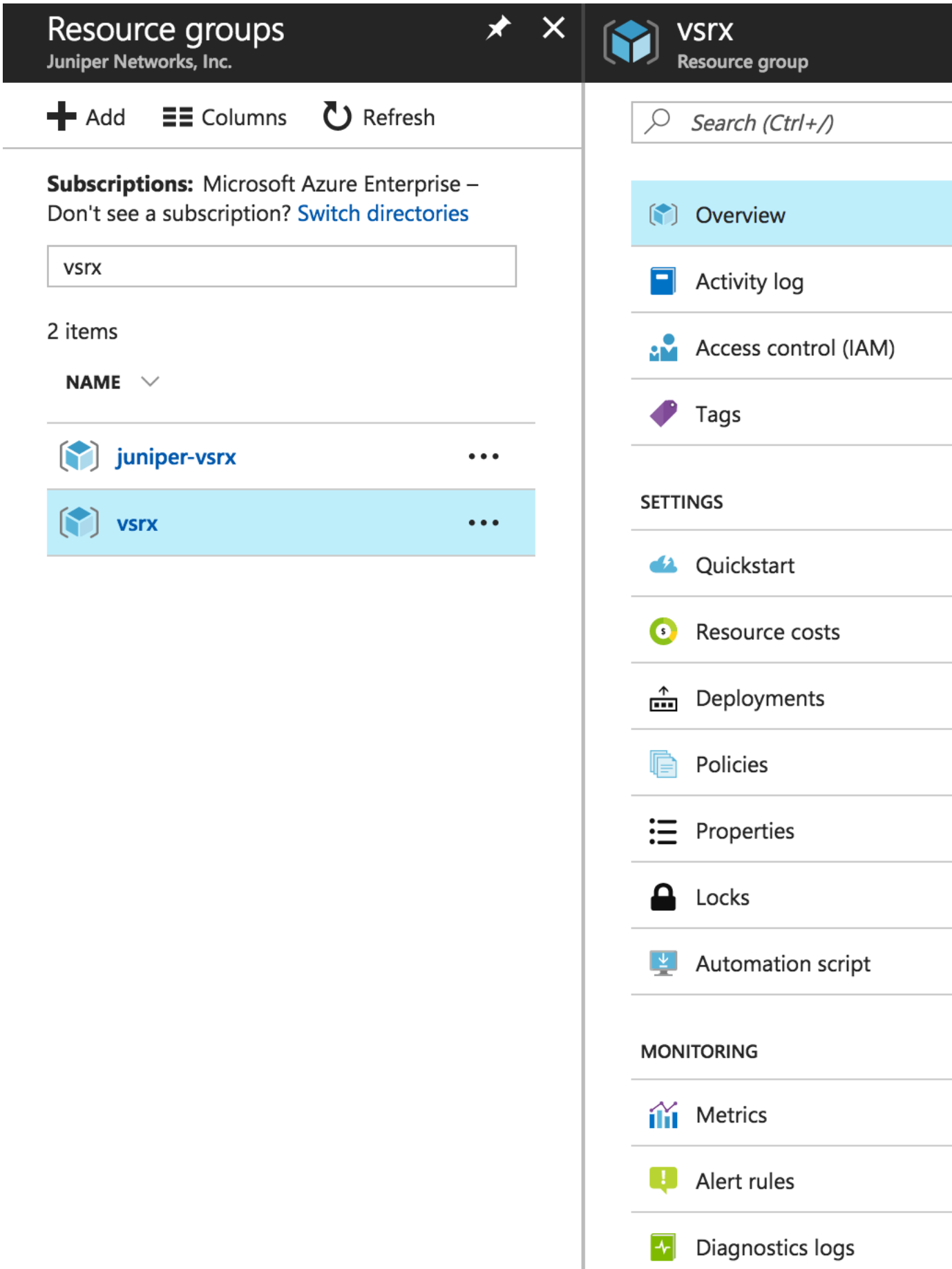
To verify the deployment of the vSRX instance to Microsoft Azure:

1. Open a Web browser to https://portal.azure.com/ and login to the Microsoft Azure portal using your login credentials. The Dashboard view appears in the Azure portal . You will see a unified dashboard for all your assets in Azure. Verify that the Dashboard includes all subscriptions to which you currently have access, and all resource groups and associated resources.

2. To view the vSRX resource group and its resources after deployment is completed, from the right-hand menu, click **Resource groups** to access the Resource Groups page.

Figure 22 on page 83 shows an example of the Resources group page in the Microsoft Azure portal.

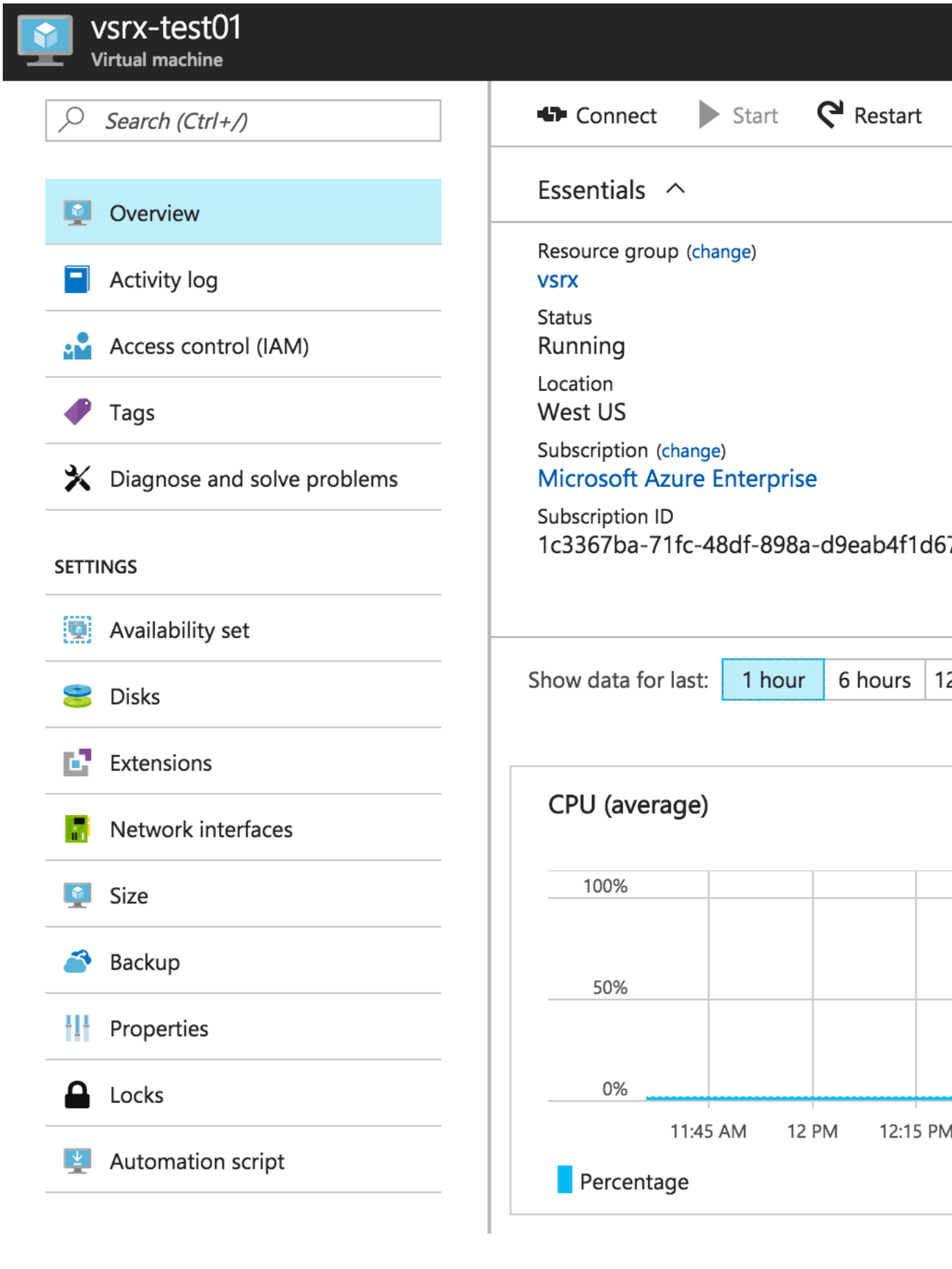
**Figure 22: Microsoft Azure Resource Groups Page Example**



3. To view details of the vSRX VM associated with the resource group, click the name of the vSRX.

Figure 23 on page 86 shows an example of the Resource groups VM in the Microsoft Azure portal.

**Figure 23: Microsoft Azure Resource Groups VM Example**

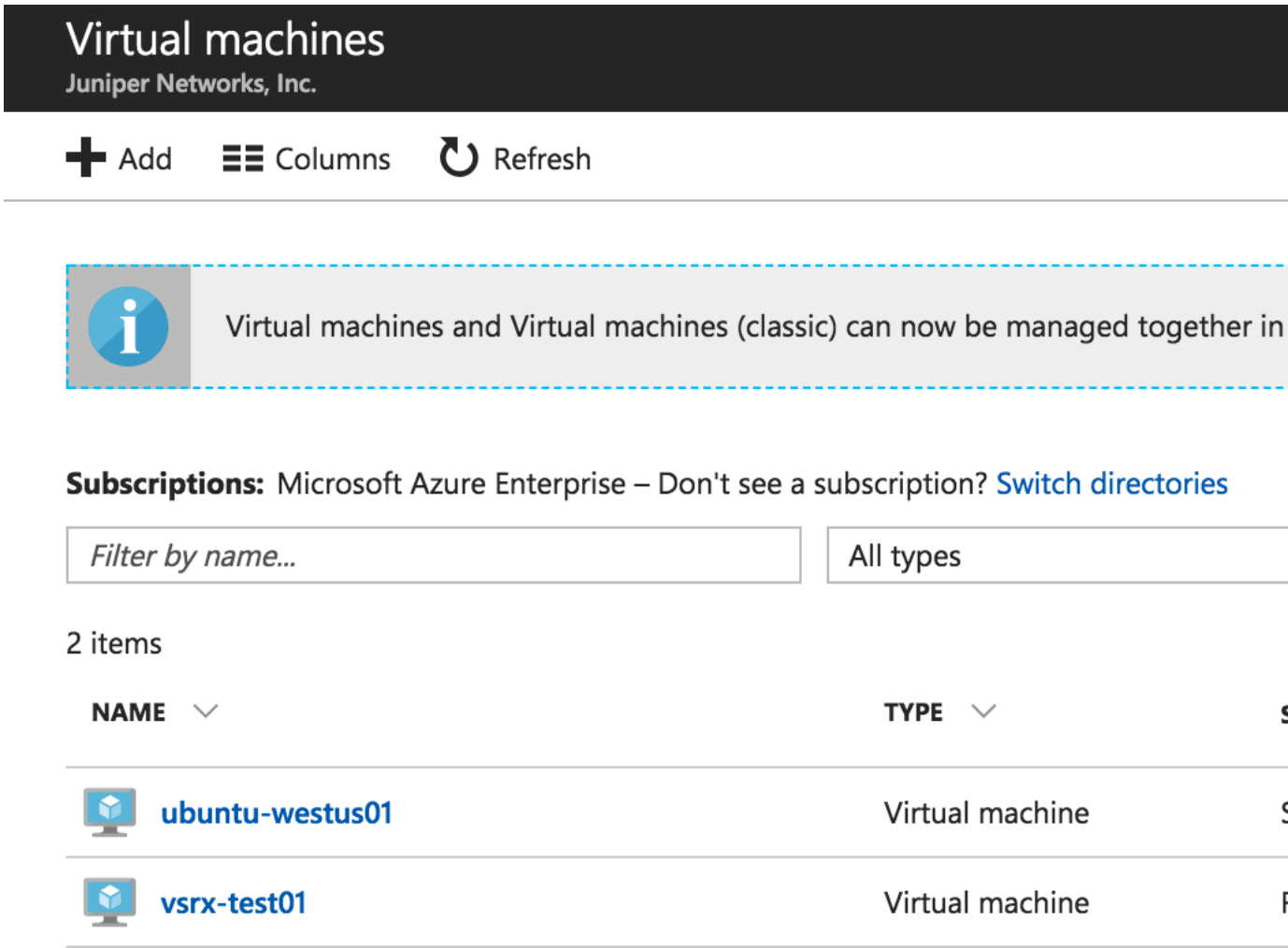


4. To see a summary view of the VMs in your subscription, including the newly deployed vSRX, click the Virtual Machines icon in the left pane. On the Virtual machines page, check the vSRX VM status after deployment is completed. Observe that the status is **Running**.

> **NOTE**: You can stop, start, restart, and delete a VM from the Virtual machines page in the Microsoft Azure portal.

Figure 24 on page 87 shows an example of the Microsoft Azure Virtual machines page.

**Figure 24: Microsoft Azure Virtual Machines Page Example**



## Log In to a vSRX Instance

After vSRX deployment is completed, the vSRX instance is automatically powered on and launched. At this point you can use an SSH client to log in to the vSRX instance.

> **NOTE**: In Microsoft Azure, individuals and enterprises can host servers and services on the cloud as a pay-as-you-go (PAYG) or bring-your-own-license (BYOL) service. For the vSRX on Microsoft Azure deployment, only the BYOL model is supported.

To log in to the vSRX VM:

1. From the Azure portal, click **Resource groups** from the menu of services on the dashboard, and then select the vSRX VM. Locate the public IP address of the vSRX VM from the Settings blade.
2. Use an SSH client to log in to a vSRX instance.
3. At the prompt, enter the following login credentials:

   > **NOTE**: Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, only password authentication is supported. Starting in Junos OS Release 15.1X49-D100 for vSRX, both password and SSH public key authentication are supported, and password authentication is chosen by default.
   >
   > The vSRX instance is automatically configured for username and password authentication. To log in, use the login credentials that were defined in the **vsrx.parameters.json** file (see "Change Parameter Values in the vsrx.parameter.json File" on page 75). After initially logging to the vSRX, you can configure SSH public and private key authentication.

   **# ssh <username@vsrx_vm_ipaddress>**

   ```
   The authenticity of host 'x.x.x.x (x.x.x.x)' ...
   ECDSA key fingerprint is SHA256:XXXXXXXXXXXXXXXXXXXXXXX.
   Are you sure you want to continue connecting (yes/no)? yes
   Warning: Permanently added 'x.x.x.x' (ECDSA) to the list of known hosts.
   Password: xxxxxxxx
   username@vsrx_vm_ipaddress>
   ```

4. Configure the basic settings for the vSRX VM (see "Configure vSRX Using the CLI" on page 92).

**Release History Table**

| Release | Description |
|---|---|
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, you can deploy the vSRX from the Azure CLI and customize the vSRX VM deployment settings and dependencies based on your network requirements in Microsoft Azure Cloud. |
| 15.1X49-D80 | Starting in Junos OS Release 15.1X49-D80 and Junos OS Release 17.3R1, only password authentication is supported. |
| 15.1X49-D100 | Starting in Junos OS Release 15.1X49-D100 for vSRX, both password and SSH public key authentication are supported, and password authentication is chosen by default. |
| 15.1X49-D100 | Starting in Junos OS Release 15.1X49-D100 for vSRX, both password and SSH public key authentication are supported, and password authentication is chosen by default. |

## RELATED DOCUMENTATION

- Connect from Microsoft Azure CLI

4

CHAPTER

# Configuring and Managing vSRX

# vSRX Configuration and Management Tools

**SUMMARY**

This topic provides an overview of the various tools available to configure and manage a vSRX VM once it has been successfully deployed.

**IN THIS SECTION**



## Understanding the Junos OS CLI and Junos Scripts

Junos OS CLI is a Juniper Networks specific command shell that runs on top of a UNIX-based operating system kernel.

Built into Junos OS, Junos script automation is an onboard toolset available on all Junos OS platforms, including routers, switches, and security devices running Junos OS (such as a vSRX instance).

You can use the Junos OS CLI and the Junos OS scripts to configure, manage, administer, and troubleshoot vSRX.

## Understanding the J-Web Interface

The *J-Web* interface allows you to monitor, configure, troubleshoot, and manage vSRX instances by means of a Web browser. J-Web provides access to all the configuration statements supported by the vSRX instance.

## Understanding Junos Space Security Director

As one of the Junos Space Network Management Platform applications, Junos Space Security Director helps organizations improve the reach, ease, and accuracy of security policy administration with a scalable, GUI-based management tool. Security Director automates security provisioning of a vSRX

instance through one centralized Web-based interface to help administrators manage all phases of the security policy life cycle more quickly and intuitively, from policy creation to remediation.

RELATED DOCUMENTATION

- CLI User Interface Overview
- J-Web Overview
- Security Director
- Mastering Junos Automation Programming
- Spotlight Secure Threat Intelligence

# Configure vSRX Using the CLI

To configure the vSRX instance using the CLI:

1. Verify that the instance is powered on.
2. Log in using the username and password credentials for your vSRX VM deployment.
3. Start the CLI.

```
root#cli
root@>
```

4. Enter configuration mode.

```
configure
[edit]
root@#
```

5. Set the root authentication password by entering a *cleartext* password, an encrypted password, or an SSH public key string (*DSA* or *RSA*).

```
[edit]
root@# set system root-authentication plain-text-password
New password: password
Retype new password: password
```

6. Configure the traffic interfaces.

```
[edit]
root@# set interfaces ge-0/0/0 unit 0 family inet address assigned_ip/netmask
root@# set interfaces ge-0/0/1 unit 0 family inet address assigned_ip/netmask
```

> **NOTE**: Configuration of the management interface fxp0 for the vSRX is not necessary, because it is configured during vSRX VM deployment. Do not change the configuration for interface fxp0 and the default routing table or you will lose connectivity.

7. Configure routing interfaces to isolate management network and traffic network.

```
[edit]
root@# set routing-instances vsrx-vr1 instance-type virtual-router
root@# set routing-instances vsrx-vr1 interface ge-0/0/0.0
root@# set routing-instances vsrx-vr1 interface ge-0/0/1.0
```

8. Verify the configuration changes.

```
[edit]
root@# commit check
configuration check succeeds
```

9. Commit the current configuration to make it permanent and to avoid the possibility of losing connectivity to the vSRX instance.

```
[edit]
root@# commit confirmed
commit confirmed will be automatically rolled back in 10 minutes unless
confirmed
commit complete
# commit confirmed will be rolled back in 10 minutes
```

10. Commit the configuration to activate it on the instance.

```
[edit]
root@# commit
commit complete
```

11. Optionally, use the **show** command to display the configuration to verify that it is correct.

**NOTE**: Certain Junos OS software features require a license to activate the feature. To enable a licensed feature, you need to purchase, install, manage, and verify a license key that corresponds to each licensed feature. To conform to software feature licensing requirements, you must purchase one license per feature per instance. The presence of the appropriate software unlocking key on your virtual instance allows you to configure and use the licensed feature. See Managing Licenses for vSRX for details.

RELATED DOCUMENTATION

- Junos OS for SRX Series
- CLI User Guide

# Configure vSRX Using the J-Web Interface

IN THIS SECTION

## Access the J-Web Interface and Configuring vSRX

Use the Junos OS CLI to configure, at a minimum, the following parameters before you can access a vSRX VM using J-Web:

> **CAUTION**: Do not change the configuration for interface fxp0 and default routing table or you will lose connectivity to the vSRX instance.

To configure vSRX using the *J-Web* Interface:

1. Launch a Web browser from the management instance.
2. Enter the vSRX fxp0 interface IP address in the Address box.
3. Specify the username and password.
4. Click **Log In**, and select the **Configuration Wizards** tab from the left navigation panel. The J-Web Setup wizard page opens.
5. Click **Setup**.

   You can use the Setup wizard to configure the vSRX VM or edit an existing configuration.

   - Select **Edit Existing Configuration** if you have already configured the wizard using the factory mode.
   - Select **Create New Configuration** to configure the vSRX VM using the wizard.

     The following configuration options are available in the guided setup:

     - Basic

       Select **basic** to configure the vSRX VM name and user account information as shown in Table 9 on page 95.

       - Instance name and user account options

         **Table 9: Instance Name and User Account Information**

| Field | Description |
|---|---|
| Instance name | Type the name of the instance. For example: **vSRX**. |

**Table 9: Instance Name and User Account Information *(Continued)***

| Field | Description |
|---|---|
| Root password | Create a default root user password. |
| Verify password | Verify the default root user password. |
| Operator | Add an optional administrative account in addition to the root account.<br><br>User role options include:<br><br>• **Super User**: This user has full system administration rights and can add, modify, and delete settings and users.<br><br>• **Operator**: This user can perform system operations such as a system reset but cannot change the configuration or add or modify users.<br><br>• **Read only**: This user can only access the system and view the configuration.<br><br>• **Disabled**: This user cannot access the system. |

- Select either **Time Server** or **Manual**. Table 10 on page 96 lists the system time options.

**Table 10: System Time Options**

| Field | Description |
|---|---|
| **Time Server** | |
| Host Name | Type the hostname of the time server. For example: **ntp.example.com**. |
| IP | Type the IP address of the time server in the IP address entry field. For example: **192.0.2.254**. |
| **NOTE**: You can enter either the hostname or the IP address. | |

**Table 10: System Time Options *(Continued)***

| Field | Description |
|---|---|
| **Manual** | |
| Date | Click the current date in the calendar. |
| Time | Set the hour, minute, and seconds. Choose **AM** or **PM**. |
| **Time Zone (mandatory)** | |
| Time Zone | Select the time zone from the list. For example: GMT Greenwich Mean Time GMT. |

- Expert

  a. Select **Expert** to configure the basic options as well as the following advanced options:

  - Four or more internal zones
  - Internal zone services
  - Application of security policies between internal zones

  b. Click the **Need Help** icon for detailed configuration information.

  You see a success message after the basic configuration is complete.

## Apply the Configuration

To apply the configuration settings for vSRX:

1. Review and ensure that the configuration settings are correct, and click **Next**. The Commit Configuration page appears.
2. Click **Apply Settings** to apply the configuration changes to vSRX.
3. Check the connectivity to the vSRX instance because you might lose connectivity if you have changed the management zone IP. Click the URL for reconnection instructions on how to reconnect to the instance.
4. Click **Done** to complete the setup.

After successful completion of the setup, you are redirected to the J-Web interface.

> **CAUTION**: After you complete the initial setup, you can relaunch the J-Web Setup wizard by clicking **Configuration>Setup**. You can either edit an existing configuration or create a new configuration. If you create a new configuration, the current configuration in vSRX will be deleted.

## Add vSRX Feature Licenses

Certain Junos OS software features require a license to activate the feature. To enable a licensed feature, you need to purchase, install, manage, and verify a license key that corresponds to each licensed feature. To conform to software feature licensing requirements, you must purchase one license per feature per instance. The presence of the appropriate software unlocking key on your virtual instance allows you to configure and use the licensed feature.

To understand more about vSRX Licenses, see, Licenses for vSRX. Please refer to the Licensing Guide for general information about License Management. Please refer to the product Data Sheets for further details, or contact your Juniper Account Team or Juniper Partner.

# Managing Security Policies for Virtual Machines Using Junos Space Security Director

**SUMMARY**

This topic provides you an overview of how you can manage security policies for VMs using security director.

Security Director is a Junos Space management application designed to enable quick, consistent, and accurate creation, maintenance, and application of network security policies for your security devices, including vSRX instances. With Security Director, you can configure security-related policy management including IPsec VPNs, firewall policies, NAT policies, IPS policies, and UTM policies. and push the configurations to your security devices. These configurations use objects such as addresses, services,

NAT pools, application signatures, policy profiles, VPN profiles, template definitions, and templates. These objects can be shared across multiple security configurations; shared objects can be created and used across many security policies and devices. You can create these objects prior to creating security configurations.

When you finish creating and verifying your security configurations from Security Director, you can publish these configurations and keep them ready to be pushed to all security devices, including vSRX instances, from a single interface.

The Configure tab is the workspace where all of the security configuration happens. You can configure firewall, IPS, NAT, and UTM policies; assign policies to devices; create and apply policy schedules; create and manage VPNs; and create and manage all the shared objects needed for managing your network security.

RELATED DOCUMENTATION

- Security Director

# Remove a vSRX Instance from Microsoft Azure

To remove a vSRX instance from Microsoft Azure:

1. Log in to the Azure Portal.
2. In the left pane of the Azure Portal, click the Virtual Machines icon.
3. To remove the vSRX instance, in the right pane, select the vSRX instance you want to remove, then click Delete.

   **NOTE**: You can delete a VM when the VM is running. If desired, you can stop the vSRX instance before deleting.

4. To delete the disks attached to the deleted vSRX virtual machine, click Delete and then select Delete the Associated VHD.
5. To delete the related cloud service for the deleted vSRX virtual machine, access the Cloud Service tab and click Delete to remove the related cloud services.

# Upgrade Junos OS Software on a vSRX Instance

IN THIS SECTION



This section outlines how to upgrade Junos OS software on your vSRX instance to a newer release. Depending upon your preference, you can replace the vSRX software in one of two ways:

## Upgrade the Junos OS for vSRX Software Release

You can directly upgrade the Junos OS for vSRX software using the CLI. Upgrading or downgrading Junos OS can take several hours, depending on the size and configuration of the network. You download the desired Junos OS Release for vSRX .tgz file from the Juniper Networks website.

You also can upgrade using J-Web (see J-Web) or the Junos Space Network Management Platform (see Junos Space).

For the procedure on upgrading a specific Junos OS for vSRX software release, see the *Migration, Upgrade, and Downgrade Instructions* topic in the release-specific *vSRX Release Notes* available on the vSRX TechLibrary.

## Replace the vSRX Instance on Azure

To replace a vSRX instance on Azure with a different software release:

1. Log in to the vSRX instance using SSH and start the CLI.

```
root@% cli
root@>
```

2. Enter configuration mode.

```
root@> configure
root@#
```

3. Copy the existing Junos OS configuration from the vSRX. The contents of the current level of the statement hierarchy (and below) are saved, along with the statement hierarchy containing it.

**NOTE**: By default, the configuration is saved to a file in your home directory.

- See Saving a Configuration File for additional background information on saving a Junos OS configuration file.
- See file copy for information on how to copy files from one location to another location on the local device or to a location on a remote device that is reachable by the local device.

```
root@#save <filename>
[edit]
root@#
```

4. Remove the vSRX instance on Azure as described in "Remove a vSRX Instance from Microsoft Azure" on page 99.
5. Once the vSRX instance on Azure has been successfully removed, define the specifics of a vSRX instance prior to launching it.
6. Launch the vSRX image using the desired software version available from Azure Marketplace.
7. Load the previously copied Junos OS configuration file onto your new (upgraded) vSRX instance as described in Loading a Configuration File.

# Software Receive Side Scaling

IN THIS SECTION

## Overview

Contemporary NICs support multiple receive and transmit descriptor queues (multi-queue). On reception, a NIC can send different packets to different queues to distribute processing among CPUs. The NIC distributes packets by applying a filter to each packet that assigns it to one of a small number of logical flows. Packets for each flow are steered to a separate receive queue, which in turn can be processed by separate CPUs. This mechanism is generally known as Receive-side Scaling (RSS). The goal of RSS technique is to increase performance uniformly. RSS is enabled when latency is a concern or whenever receive interrupt processing forms a bottleneck. Spreading load between CPUs decreases queue length. For low latency networking, the optimal setting is to allocate as many queues as there are CPUs in the system (or the NIC maximum, if lower). The most efficient high-rate configuration is likely the one with the smallest number of receive queues where no receive queue overflows due to a saturated CPU. You can improve bridging throughput with Receive Side Scaling.

As per flow thread affinity architecture each flow thread (FLT) polls for packet from dedicated receiving queue of NIC and process the packets until run to completion. Therefore, flow threads are bound to NIC receiving (RX) and transmitting (TX) queues for packet processing to avoid any disagreement. Hence, NIC must have same number of RX and TX queues as number of vSRX data plane CPU to support multi core vSRX flavors. Software RSS (SWRSS) removes this limitation of NIC HW queues to run vSRX multi-core flavors by implementing software-based packet spraying across various FLT thread.

Software RSS offloads the handling of individual flows to one of the multiple kernel, so the flow thread that takes the packets from the NIC can process more packets. Similar to RSS, network throughput improvement when using SWRSS has a linear correlation with CPU utilization.

In SWRSS, each NIC port is initialized with equal or lesser number of hardware RX/TX queues as that of I/O threads. I/O threads are determined based on total data-path CPU and minimum of NIC queues among all the NIC interface in vSRX. For example, if I/O thread is computed as 4, then number of HW queue per NIC port can be less or equal to 4 queues.

If NICs do not have sufficient number of queues as FLT threads in vSRX instances supported, then Software RSS (SWRSS) is enabled by flowd data-path. SWRSS implements software model of packet distribution across FLTs after obtaining the packets from NIC receiving queues. By removing NIC HW queue limitation, SWRSS helps to scale vCPUs by supporting various vSRX instance types.

During the I/O operation the packets are fetched from receiving queues of NIC ports and packet classification is performed. Followed by distribution of packets to FLT threads virtual queues. These virtual queues are implemented over DPDK ring queue. In the transmission path, SWRSS fetches the packets from virtual transmitting queues of FLT threads and pushes these packets to NIC transmitting queues for transmit.

Number of SWRSS I/O threads are selected based on total CPU and number of NIC queues found in vSRX instances. Mix mode of operation with HWRSS and and SWRSS is not supported.

## Understanding Software Receive Side Scaling Configuration

This topic provide you details on types of Software Receive Side Scaling (SWRSS) and its configuration.

SWRSS supports two modes of operation and it gets enabled based on number of data-path CPU needed. These modes are Shared IO mode and dedicated IO mode. These modes are enabled based on number of data-path CPUs needed. vSRX and vSRX3.0 supports dedicated I/O mode only.

In dedicated I/O mode flowd process creates dedicated I/O threads for I/O operation. Based on number of required I/O threads for vSRX, I/O thread is associated to a dedicated NIC port. NIC ports receiving and transmitting queue is then bonded to each I/O thread in round robin method for uniform distribution and to avoid I/O thread locks. Each dedicated I/O thread pulls the packets in burst mode from NIC receiving queue and distributes to FLT threads and vice versa for TX path for packet transmit.

SWRSS is enabled based on the number of vCPUs. If NIC does not have sufficient number of queues as flow thread (FLT) in vSRX with different vCPUs, then Software RSS (SWRSS) is enabled by flowd process.

SWRSS is not enabled in the following scenarios:

- When the NIC has sufficient number of hardware RX or TX queues for required PFE data-path CPU.
- When the vSRX (based on number of vCPUs) and NIC result the smaller number of FLT CPUs as that obtained in nearest hardware RSS (HWRSS) mode. In such scenario, vSRX will be enabled with HWRSS mode which results more FLT CPU than SWRSS mode, providing better packet processing throughput.
- SWRSS is not recommended for vSRX with certain type of NIC that supports lesser number of NIC queues than needed to run dedicated IO thread. In such cases, SWRSS is enabled but extra CPUs are attached to FLT CPU, until I/O CPUs are completely utilized.

If SWRSS is not enabled use the **set security forwarding-options receive-side-scaling software-rss mode enable** command to enable SWRSS. When you run this command SWRSS will be enabled by force regardless of the NIC RSS or the number of vCPUs. If you do not enable SWRSS using the CLI then enabling of SWRSS automatically is decided based on the default ratio of FLT: IO ( 4:1).

To configure the number of required IO threads, use the **set security forwarding-options receive-side-scaling software-rss io-thread-number *<1-8>*** command. To view the actual number of vCPUs assigned to IO flow threads use the **show security forwarding-options resource-manager** command.

You can decide enabling of SWRSS automatically or by force based on the architecture and conception of IO thread and worker thread. Enabling SWRSS impacts the performance, so we recommend that the number of IO thread should be changed only if required and until the performance impact bottleneck point is reached.

# GTP Traffic with TEID Distribution and SWRSS

IN THIS SECTION



## Overview GTP Traffic Distribution with TEID Distribution and SWRSS

IN THIS SECTION



The topic provides an overview of asymmetric fat tunnel solution for GTP traffic with TEID distribution and SWRSS.

With TEID-based hash distributions feature, the GTP packets would be distributed to the flow thread according to the hash value calculated by TEID. The algorithm of hash calculation is same as GTP distribution in flow module, which ensures the GTP packets would not be reinjected again in the flow process.

There is a 4-byte field inside GTP payload called tunnel endpoint identifier (TEID), which is used to identify different connections in the same GTP tunnel.

A fat GTP tunnel carries data from different users. IPsec tunnels on the security gateway could be a fat tunnel due to the fat GTP tunnel. vSRX can create one GTP session with a high-bandwidth of GTP traffic. However, the throughput is limited to one core processor's performance.

If you use TEID-based hash distribution for creating GTP-U sessions, then you can:

- Enable vSRX and vSRX 3.0 instances to process asymmetric fat tunnels for parallel encryption on multiple cores for one tunnel.
- You can split a fat GTP session to multiple sessions and distribute them to different cores. This helps to increase the bandwidth for fat GTP tunnel.

The TEID based hash distribution creates GTP-U sessions to multiple cores. The clear text traffic acts as a fat GTP tunnel. This helps a fat GTP session to split into multiple slim GTP sessions and handle them on multiple cores simultaneously.

## GTP Traffic Performance with TEID Distribution and SWRSS

vSRX instances support Software Receive Side Scaling (SWRSS) feature. SWRSS is a technique in the networking stack to increase parallelism and improve performance for multi-processor systems. If NICs do not have sufficient number of queues as flow thread (FLT), based on vSRX type, then Software RSS (SWRSS) is enabled by flowd process.

With Software Receive Side Scaling (SWRSS) support on vSRX and vSRX 3.0, you can assign more vCPUs to the vSRX regardless of the limitation of RSS queue of underlying interfaces.

Based on SWRSS you can improve the GTP traffic performance using Tunnel endpoint identifier (TEID) distribution and asymmetric fat tunnel solution by:

- Assigning specific number of vCPUs for input output flow usage—With SWRSS enabled, you can assign more vCPUs for input/output (IO) threads when the IO threads are less. Or you can assign less vCPUs for IO threads if the flow process is consuming more vCPU. Use the **set security forwarding-options receive-side-scaling software-rss io-thread-number <io-thread-number>**.

- Distributing the packets to flow threads according to the TEID inside the packet, which would avoid reinjecting the packets in flow process—This feature is enabled when both SWRSS is enabled and when you configure the **set security forwarding-process application-services enable-gtpu-distribution** command.

  With this feature, the GTP packets would be distributed to the flow thread according to the hash value calculated by TEID. The algorithm of hash calculation is same as GTP distribution in flow module, which ensures the GTP packets would not be reinjected again in flow process.

- Utilizing fragment matching and forwarding mechanism in input/output thread when GTPU distribution is enabled—This mechanism ensures that all the fragments of the same packet would be distributed to one flow thread according to the TEID.

  SWRSS uses IP pair hash to distribute packets to flow threads. For GTP traffic with GTPU distribution enabled, TEID distribution is used to distribute packets to the flow threads. For fragmented packets, TEID cannot be retrieved from non-first fragments. This will require fragment matching and forwarding logic to ensure all fragments are forwarded to the flow thread based on TEID.

## Enabling GTP-U TEID Distribution with SWRSS for Asymmetric Fat Tunnels

The following configuration helps you enable PMI and GTP-U traffic distribution with SWRSS enabled.

Before you begin, understand:

- SWRSS concepts and configurations.
- How to establish PMI and GTP-U

With Software Recieve Side Scaling (SWRSS) enabled, you can assign more vCPUs for input/output (IO) threads when the IO threads are less. Or you can assign less vCPUs for IO threads if the flow process is consuming more vCPU. You can configure the number of IO threads required. With SWRSS is enabled and IO threads configured, reboot the vSRX for configuration to take effect. After IO threads are configured, distribute the GTP traffic to the configured IO threads according to TEID-based hash distribution for splitting a fat GTP session to multiple slim GTP sessions and process them on multiple cores in parallel.

**NOTE**: When PMI mode is enabled with TEID distribution and SWRSS support, performance of PMI is improved. If you want to enable PMI mode then run the **set securtiy flow power-mode-ipsec** command.

The following steps provide you details on how to enable SWRSS, configure IO threads, enable PMI mode for GTP sessions with TEID distribution for obtaining asymmetric fat tunnels:

1. SWRSS is enabled by default when NICs do not have sufficient number of queues as flow thread (FLT) based on vSRX type, then Software RSS (SWRSS) is enabled by flowd process. But, when SWRSS is not enabled use the following CLIs to enable. When you run this command SWRSS will be be enabled by force regardless of the NIC RSS or number of vCPUs.

   Enable SWRSS.

```
[edit]
user@host# set security forwarding-options receive-side-scaling software-rss mode enable
```

2. Configure the number of IO threads required. In this configuration we are configuring eight IO threads. The assigned number of vCPUs would be assigned for IO threads, and the rest vCPUs would be assigned for flow thread.

```
[edit]
user@host# set security forwarding-options receive-side-scaling software-rss io-thread-number 8
```

3. 

```
[edit security]
user@host# set flow power-mode-ipsec
```

4. Configure GTP-U session distribution.

```
[edit security]
user@host# set forwarding-process application-services enable-gtpu-distribution
```

5. From the configuration mode, confirm your configuration by entering the **show** command.

```
[edit security]
user@host# show
forwarding-options {
    receive-side-scaling {
        software-rss {
             mode enable;
        io-thread-number 8;
flow {
    power-mode-ipsec;
}
forwarding-process {
    application-services {
        enable-gtpu-distribution;
    }
}
```

From the operational mode run the following command to view the actual number of vCPUs assigned to IO/flow threads.

```
show security forward-options resource-manager settings
```

```
------------------------------------------------------------------------
Owner          Type                      Current settings      Next settings
SWRSS-IO       CPU core number           2                     2
SWRSS          SWRSS mode                Enable                Enable
```

6. Commit the configuration.

```
[edit security]
user@host# commit
```

7. Reboot the vSRX for the configuration to take effect. After rebooting the whole device, PFE would check the IO-thread value according to the NIC RSS queue and its memory.

# Deployment of Microsoft Azure Hardware Security Module on vSRX 3.0

**IN THIS SECTION**

## Microsoft Azure Key Vault Hardware Security Module Integration Overview

Microsoft Azure Key Vault hardware security module (HSM) is a cloud service that works as a secure secrets store. You can securely store keys, passwords, certificates, and other secrets. This service from cloud vendors helps us to securely generate, store and manage Crypto keys. vSRX applications use these Crypto keys to protect data at rest, such as private keys, passwords and other sensitive data. Azure Key Vault HSM can also be used as a Key Management solution. Azure Key Vault makes it easy to create and control the encryption keys used to encrypt your data. When you provide the master encryption password then that password is used to encrypt the sensitive data and save encrypted data (AES256) on disk. The master encryption password is also protected using RSA key-pair generated and stored in HSM.

vSRX (mgd process) generates hash of configuration. This hash (and other sensitive data) is protected using master encryption password as key for AES-GCM 256 encryption.

The master password is used to protect secrets such as the RADIUS password, IKE preshared keys, and other shared secrets in the Junos OS management process (mgd) configuration. The master password is protected using the master encryption password. The master password itself is not saved as part of the configuration. The password quality is evaluated for strength, and the device gives feedback if weak passwords are used.

Sensitive data such as PKI private keys and configuration that are stored in plain text on vSRX 3.0 instances can now be protected using HSM service.

When you enable Microsoft Azure Key Vault HSM on vSRX, vSRX creates, an RSA key pair of 2048 size and uses it to encrypt, a PKI private key file located in **/var/db/certs/common/key-pairs**, configuration hash and a master password, which is saved in: **/config/unrd-master-password.txt**.

**NOTE**: Existing keypairs prior to enabling HSM will not be encrypted and are deleted.

By enabling the HSM, the software layer leverages the use of the underlying HSM service that protects sensitive information such as private keys, system master passwords, and so on, by storing the information using 256-bit AES encryption (instead of storing in cleartext format). The device also generates a new SHA256 hash of the configuration each time the administrator commits the configuration. This hash is verified each time the system boots up. If the configuration has been

tampered with, the verification fails and the device will not continue to boot. Both the encrypted data and the hash of the configuration are protected by the HSM module using the master encryption password.

Hash validation is performed during any commit operation by performing a validation check of the configuration file against the saved hash from previous commits. In a chassis cluster system, hash is independently generated on the backup system as part of the commit process.

Hash is saved only for the current configuration and not for any rollback configurations. Hash is not generated during reboot or shutdown of the device.

vSRX uses HSM to encrypt the following secrets:

- SHA256 hash of the configuration
- Device master password
- All key pairs on the device

Keys created by each vSRX 3.0 instance will be tagged and/or named using the UUID of each VM. You can log in to the cloud portal, access the keys, and verify their properties or the operations requested.

## Configure Microsoft Azure Key Vault HSM on vSRX 3.0

Key vault on Azure stack provides cloud HSM service for all Azure applications. All applications need to be registered in Azure active directory to use services such as Key Vault.

vSRX3.0 is integrated with Microsoft Azure Cloud HSM when running on Azure. You can login to cloud portal, access the keys, and verify their properties or operations requested for.

For each public cloud vendor, there are unique steps to be performed to integrate vSRX with cloud HSM. This section provides the steps needed to integrate vSRX 3.0 with Microsoft Azure Key Vault HSM.

You will need the following listed items to integrate vSRX with Microsoft Azure Key Vault HSM:

- vSRX 3.0 instance
- Microsoft Azure Key vault
- Setup key vault authentication for vSRX
- Microsoft Azure-specific configurations for integrating HSM

Microsoft Azure Key Vault is a cloud-hosted management service that allows users to encrypt keys and small secrets by using keys that are protected by hardware security modules (HSMs).

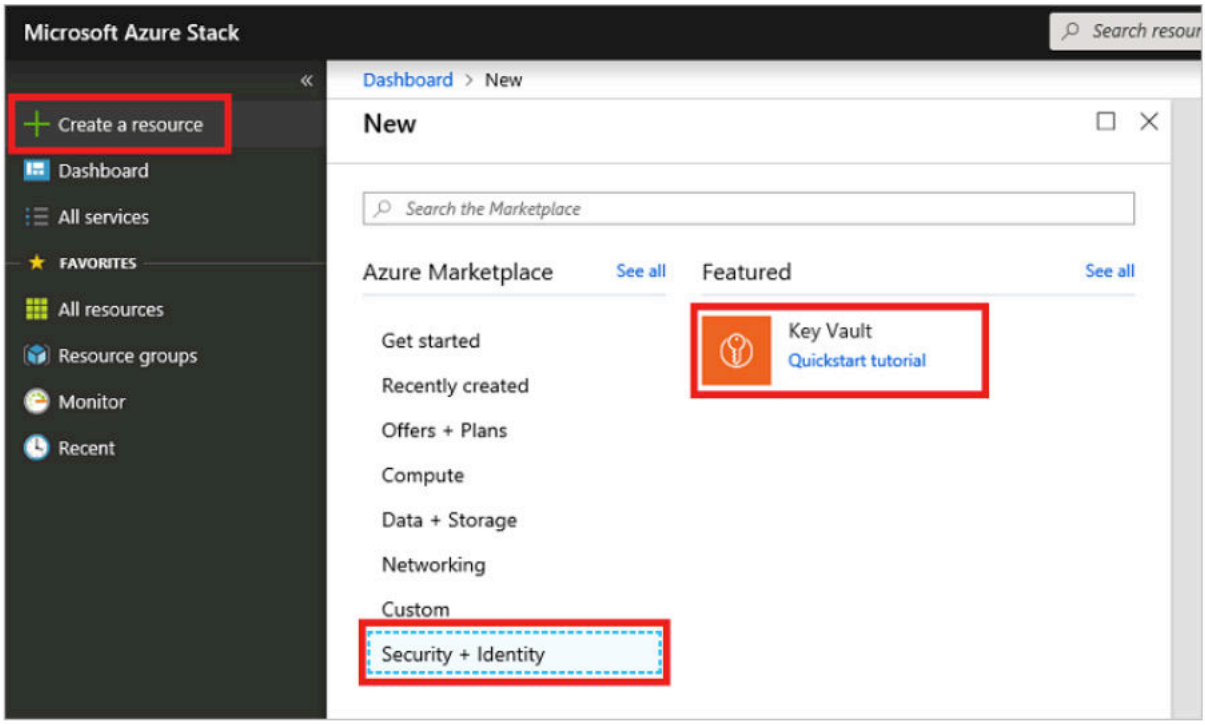
This procedure provides the general steps to integrate Microsoft Azure Key Vault HSM with vSRX 3.0.

1. Launch vSRX 3.0 instance in Microsoft Azure environment.

   For launching vSRX 3.0 instances see, vSRX Deployment Guide for Microsoft Azure Cloud.

2. Create Key vault. From the dashboard, select **+ Create a resource**, **Security + Identity**, and then **Key Vault** as shown in Figure 25 on page 111.

**Figure 25: Create Key Vault**




You need to create “premium” key vault to access cryptographic key features needed by vSRX 3.0. After you create a key vault, for more information on how to create and manage keys and secrets within the vault, see Manage Key Vault in Azure Stack using the portal.

3. Enable managed identity for vSRX 3.0.

   System assigned managed identity helps vSRX authenticate to other services (example Key vault) without saving credentials in the code by registering your application to Azure Active directory. Enabling this identity will generate unique object ID, which can be used to refer it across other vSRX instances.

   To enable managed identity for vSRX on Microsoft Azure, you need to configure managed identities for Microsoft Azure resources on a VM using the Azure portal as shown in Figure 26 on page 112 and Figure 27 on page 113.

For more information, see Configure managed identities for Azure resources on a VM using the Azure portal

**Figure 26: Enable System Assigned Managed Identity During Creation of a VM**

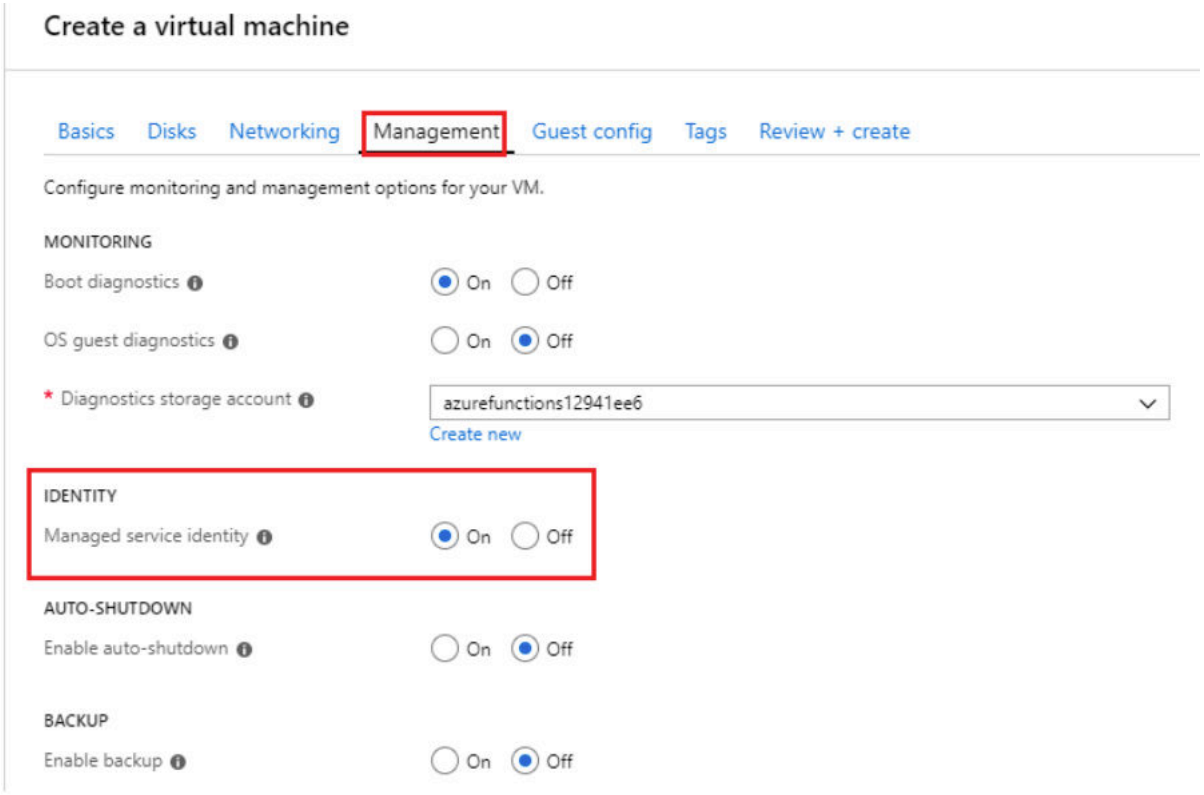
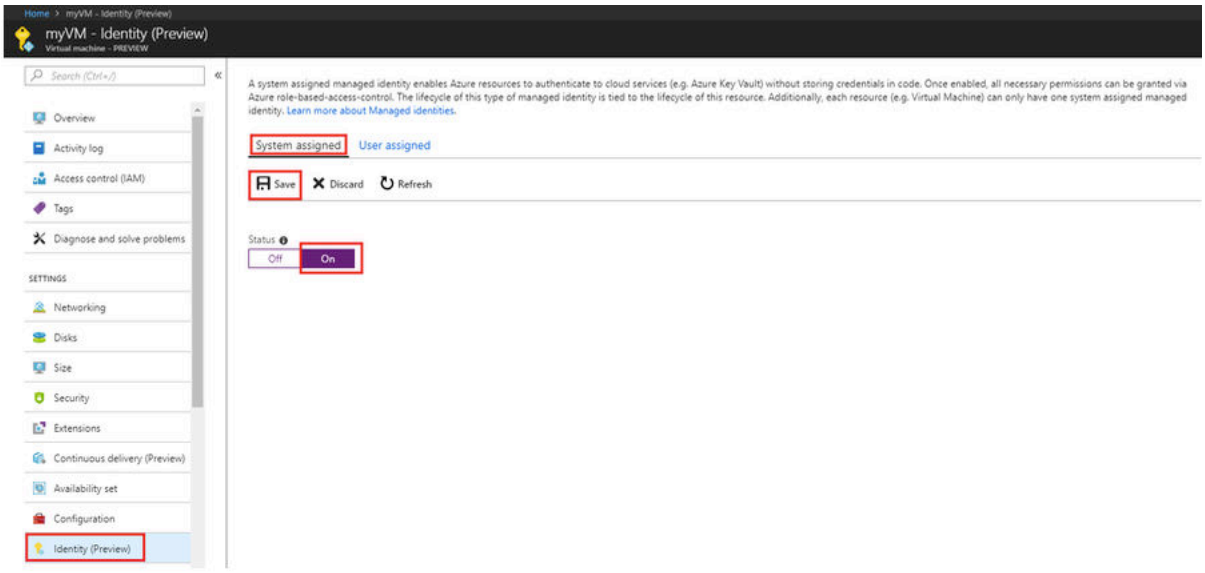


**Figure 27: Enable System Assigned Managed Identity on an Existing VM**




4. Add access policy in Microsoft Azure Key Vault.

   For applications such as vSRX 3.0 VM to access Microsoft Azure Key Vault, access policies have to be enabled. For more information on how to add new policy, see Secure access to a key vault refer this link to add new policy.

   Steps to add access policy in Microsoft Azure Key Vault are:

   a. Go to **Key Vault Resource** page on Microsoft Azure portal.

   b. Click **Access Policies** tab on the left side of the page.

   c. Click on **Add New** tab and then click **Select Principal**, where you search for your vSRX user name assigned when it was created.

   d. Select all the **key permissions** and click **Save**.

   > **NOTE**: Do not select any **Authorized application**.

5. Check fxp0 (management) interface status

   vSRX3.0 uses fxp0 for communication with the Microsoft Azure Key Vault. Use the **show interface terse fxp0** command and ensure to check if fxp0 is configured and is able to ping external servers.

**NOTE**: vSRX 3.0 connects to cloud HSM using management interface. If management interface is not configured or does not get connected, then cloud HSM features cannot be used.

6. Enable and start communicating with key vault.
   - To enable key vault, run the **request security hsm set key-vault <name-of-key-vault>** command.

     **NOTE**: URL used to access Microsoft Azure Key Vault is generally in the format as: https://<name-of-key-vault>.vault.azure.net/keys.

   - To establish communication with key vault, create RSA key pair in HSM, generate and encrypt configuration hash, and encrypt master password and PKI key pair files run the **request security hsm master-encryption-password set plain-text-password**.
   - You will be prompted to enter the master encryption password twice, to make sure that these passwords match. The master encryption password is validated for required password strength. After the master encryption password is set, the system encrypts the sensitive data with the master encryption password, that is encrypted by the MEK that is owned and protected by HSM.
   - To configure the master password run the **set system master-password plain-text-password** command. Otherwise, certain sensitive data will not be protected by the HSM. If HSM is not enabled, master password will be saved in plain text format in the **/config/unrd-master-password.txt** file

     **NOTE**: To ensure master password is not saved as plain text on vSRX 3.0, an error will be displayed on console indicating that, it is insecure to set master password without enabling HSM and command operation will be terminated.

## Change the Master Encryption Password

If you want to change the master encryption password then you can run the **request security hsm master-encryption-password set plain-text-password** command from operational mode:

**NOTE**: It is recommended that no configuration changes are made while you are changing the master encryption password.

The system checks if the master encryption password is already configured. If master encryption password is configured, then you are prompted to enter the current master encryption password.

The entered master encryption password is validated against the current master encryption password to make sure these master encryption passwords match. If the validation succeeds, you will be prompted to enter the new master encryption password as plain text. You will be asked to enter the key twice to validate the password.

The system then proceeds to re-encrypt the sensitive data with the new master encryption password. You must wait for this process of re-encryption to complete before attempting to change the master encryption password again.

If the encrypted master encryption password file is lost or corrupted, the system will not be able to decrypt the sensitive data. The system can only be recovered by re-importing the sensitive data in clear text, and re-encrypting them.

## Verify the Status of the HSM

**IN THIS SECTION**



### Purpose

To check connectivity with HSM.

### Action

You can use the **show security hsm status** command to verify the status of the HSM. The following information is displayed:

- If HSM is enabled and reachable or disabled

- Is Master Binding Key (RSA Key pair) created in HSM
- Is Master Encryption Key configured - master encryption password status (set or not set)
- Cloud vendor Information

## request security hsm master-encryption-password set

IN THIS SECTION



### Syntax

```
request security hsm master-encryption-password set plain-text-password
```

### Release Information

Command introduced in Junos OS Release 19.4R1.

### Description

Use this command to set or replace the password (in plain text).

### Options

**plain-text-password** Set or replace the password (in plain text).

### Required Privilege Level

maintenance

### Output Fields

When you enter this command, you are provided feedback on the status of your request.

### Sample Output

#### request security hsm master-encryption-password set plain-text-password

```
user@host>                    request security hsm master-encryption-password set plain-text-password
```

```
Enter new master encryption password:
Repeat new master encryption password:
Binding password with HSM
Master encryption password is bound to HSM
Encoding master password ..
Successfully encoded master password
Deleting all previous local certificates, keypairs and certificate requests
```

## show security hsm status

IN THIS SECTION

## Syntax

```
show security HSM status
```

## Release Information

Command introduced in Junos OS Release 19.4R1.

## Description

Display the current status of the Hardware Security Module (HSM). You can use this **show security hsm status** command to check the status of HSM, master binding key, master encryption password, and cloud vendor details.

## Options

This command has no options.

## Required Privilege Level

security

## Output Fields

Table 11 on page 118 lists the output fields for the **show security hsm status** command.

**Table 11: show security hsm status Output Fields**

| Field Name | Field Description |
|---|---|
| **Enabled** | Specifies whether HSM is enabled or disabled. |

**Table 11: show security hsm status Output Fields *(Continued)***

| Field Name | Field Description |
|---|---|
| **Master Binding Key** | Displays the HSM's Master Binding Key status whether it is created or not created in HSM. HSM generates cryptographic keys and encrypts them so that those can only be decrypted by the HSM. This process is know as binding. Each HSM has a master binding key, which is also know as storage root key. |
| **Master Encryption Key** | Displays Master Encryption configuration status whether it is set or not set. The encrypted data and the hash of the configuration is protected by vSRX using Microsoft Key Vault (HSM) service. |
| **Cloud vendor Details** | Displays the details specific to the cloud vendor. |

## Sample Output

### show security hsm status (HSM status command output when vSRX initially boots up but this feature is not enabled)

```
user@host>                    show security hsm status
```

```
HSM Status:
  Accessible: no
  Master Binding Key: not-created
  Master Encryption Key: not-configured
  Azure Key Vault: unknown
```

### Sample Output

**show security hsm status (HSM status command output after successful integration with key vault)**

```
user@host>                    show security hsm status
```

```
HSM Status:
  Accessible: yes
  Master Binding Key: created
  Master Encryption Key: configured
  Azure Key Vault: vsrx3-hsm-kv
```

**SEE ALSO**

request security hsm master-encryption-password set | 0

Deployment of Microsoft Azure Hardware Security Module on vSRX 3.0 | 108

## Understanding VPN Functionality with Microsoft Azure Key Vault HSM Service

**IN THIS SECTION**

- Deployment Scenario | 121

With the integration of Microsoft Azure Key Vault HSM Service on vSRX3.0, you can now use the HSM service to create, store, and perform the required VPN keypair operations. Keypair creation is now enabled in HSM service.A PKI based VPN tunnel can now be established using the keypairs generated using the HSM. Once the master encryption key is configured, you can configure the VPN functionality using HSM service. You can generate only RSA keypairs of length 2048 and 4096 bits. Operations such as private key signing during CSR creation in PKID, private key signing during verification of the

certificate received from the CA server in PKID, and private key signing during IKE negotiations at IKED is off-loaded from vSRX and is now performed by the HSM service.

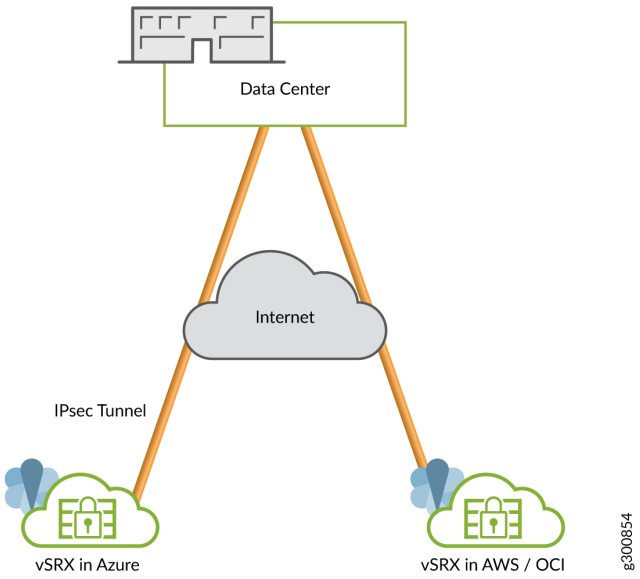
**NOTE**: Keypair generation using HSM service is only for pkid and iked processes. Also, existing keypairs in the filesystem before HSM service is enabled are not encrypted and those keypairs are deleted.

## Deployment Scenario

This section provides a deployment scenario where vSRX 3.0 instance is launched as a gateway in a virtual network connecting to a data center using a pure IPsec connection.

Figure 28 on page 121 shows the deployment scenario.

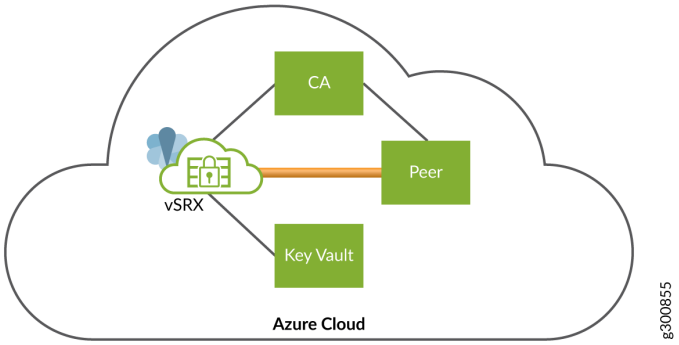
**Figure 28: Deployment Scenario of vSRX using an IPsec Connection**




You can generate key pairs using Microsoft Azure cloud HSM service for pkid process and use these keypairs for getting a local certificate from the CA server. Use the keypair present in the cloud HSM service for private key signing during IKE negotiations.

The VPN functionality performed within the Microsoft Azure cloud using HSM service is as shown in Figure 29 on page 122.

**Figure 29: Components for VPN with HSM in Microsoft Azure Cloud**




The components involved here are:

- vSRX 3.0 launched in the Microsoft Azure cloud.
- Peer—Second vSRX 3.0 instance launched in the Azure cloud. A tunnel is established between the frist vSRX 3.0 and and the Peer.
- Key Vault—The HSM service launched in the Azure cloud. You can interact between the vSRX 3.0 and the HSM. and the peer can create and store keypairs locally.
- Certificate Authority Server—Any CA server that can be accessed by the vSRX instances. The CA server is launched on the Azure Cloud.

This procedure provides steps on how to allow access from vSRX to the HSM by authenticating the vSRX with the cloud HSM service.

1. Initialize a session with the HSM service—Each process that needs to interact with the HSM has to initialize a separate session of its own. For the VPN functionality you must establish 2 sessions with the HSM service for each device involved. One session is established with pkid process and another session with iked process. These sessions with the HSM service are established only once during the init process of the daemon. If a daemon is restarted, a new session is established with the HSM service. When a session is successfully established with the HSM service, a valid session context is returned. Sessions will be established with the HSM service only if Master Encryption Key (MEK) is enabled. Each session will be a secure TLS connection between the vSRX and the cloud HSM.
2. Handling Keypairs at the HSM—To create and store keypairs at the HSM use the **request security pki generate-key-pair certificate-id certificate-id-name <size> <type>** command.

**NOTE**: The term certificate-id is just an identifier associated with the keypair that has been generated. There is no connection to a certificate creation yet. If no type and size are mentioned, then the default values of type as RSA and and size of 2048 is considered.

3. Redirection to the HSM—With HSM enabled, the same CLI command will be redirected to the HSM. A new keypair with the given parameters is created at the HSM. Keys created by each vSRX will be tagged using the UUID of each VM. You can login to cloud portal, access the keys and verify their properties/operations that you want. The UUID of each key is of the following format:**<key-name>_<unique vm-instance id>**. You need to provide the key name at the time of key creation. The VM instance is the factor that will make the key id unique in the HSM service. Thus, it is required that the **vm-instance id** must be unique for each VM which is up and running. This is ensured by Microsoft Azure. The HSM redirection will be a timed call, wherein if no response is received within *x* seconds, then an error message **call to HSM failed** is displayed.

4. Retrieval of Public Key Information—After the creation of the keypair at the HSM, we retrieve the public key components of the keypair. The HSM returns the modulus and the exponent. These components are converted into EVP_PKEY structure using OpenSSL API's. The public key structure is then stored as a new entry in the hash of keys. In this way, the public key components can be retrieved from the hash when required. Currently, the HSM does not detect duplicate keypairs, instead when error key id is received again, the HSM will overwrite the pre-existing keypair. To avoid this overwrite of keypairs, the public key is saved in the hash at the time of key creation itself. This way, a duplicate keypair creation is stopped at the device level itself, without making a call to the HSM.

   You will receive an error **error: Failed to generate key pair at HSM. Found a key with the same name at HSM. Use a different certificate id next time. Refer to PKID logs for more details** when you try to use the same name to create a new keypair, even if you have deleted the previous keypair.

5. Deletion of Keypairs—HSM does not support an API to delete keypairs created at the HSM. The delete keypair command issued at the CLI will result in the public key component being deleted from the disk and the key hash. The keypair will not be deleted from the HSM. To delete the keypair from the HSM, you need to access the HSM and manually delete the keypair. If Azure key vault has soft delete feature enabled, you will also need to eliminate the keypair from the keypair before you can re-use the keypair name.

**NOTE**: Exporting keys from the file is not supported. When you use the **request security pki local-certificate export** and **request security pki key-pair export** commands to export keys, you will receive an error message **Export of keypairs/certificate is not supported when HSM is enabled**.

6. Private Key Signing—The private key is now present at the HSM. So, all operations requiring the private key have been offloaded to the HSM. The operations involve:

   Private key signing operation are used during:

   - Creating the Certificate Signing Request (CSR)
   - Verification of the local certificate received from the CA
   - RSA signing during IKE negotiations
   - SHA-1 Inter-operability. The Azure key vault supports private key signing for only SHA-256 digests.

## CLI Behavior With and Without HSM

| CLI | Non-HSM | HSM |
|---|---|---|
| **request security pki generate-key-pair** | Creates a keypair locally | Creates a keypair at the HSM |
| **request security pki generate-certificate-request** | Creates a CSR locally | Contacts the HSM for private key signing while creating the CSR. Digest has to be SHA-256 |
| **request security pki local-certificate enroll** | Creates a CSR locally. Sends the CSR to the CA server and receives a certificate | Contacts the HSM for private key signing while creating the CSR. Sends the CSR to the CA server and receives a certificate. Digest has to be SHA-256 |
| **request security pki local-certificate export** | Exported local certificate to other device | Not possible as key pair not present locally |
| **request security pki key-pair export** | Exported locally present key pair to other device | Not possible as key pair not present locally |

| | | |
|---|---|---|
| **request security pki local-certificate generate-self-signed** | Generates self signed certificate | Contacts HSM for signing and then generates self signed certificate |
| **show security pki local-certificate** | Shows local certificate present on device | Shows keypair is generated locally or at cloud HSM |

## request security pki local-certificate enroll scep

IN THIS SECTION



### Syntax

```
request security pki local-certificate enroll scep
    ca-profile  ca-profile name
    certificate-id certificate-id-name
    challenge-password challenge-password
    digest (sha-1 | sha-256)
    domain-name domain-name
    email email-address
     ip-address ip-address
    ipv6-address ipv6-address
    scep-digest-algorithm (md5 | sha-1)
```

```
    scep-encryption-algorithm (des | des3)
    subject subject-distinguished-name
```

## Release Information

Command introduced in Junos OS Release 9.1. Serial number (SN) option added to the subject string output field in Junos OS Release 12.1X45. **scep** keyword and **ipv6-address** option added in Junos OS Release 15.1X49-D40.

Starting in Junos OS Release 20.1R1 on vSRX 3.0, you can safeguard the private keys used by PKID and IKED using Microsoft Azure Key Vault hardware security module (HSM) service. You can establish a PKI based VPN tunnel using the keypairs generated at the HSM. The hub **certificate-id** option under certificate-id is not available for configuration after generating HSM key-pair.

Starting in Junos OS Release 20.4R1 on vSRX 3.0, you can safeguard the private keys used by PKID and IKED using AWS Key Management Service (KMS). You can establish a PKI based VPN tunnel using the keypairs generated by the KMS. The hub **certificate-id** option under certificate-id is not available for configuration after generating PKI key-pair.

## Description

Enroll and install a local digital certificate online by using Simple Certificate Enrollment Protocol (SCEP).

If you enter the **request security pki local-certificate enroll** command without specifying the **scep** or **cmpv2** keyword, SCEP is the default method for enrolling a local certificate.

## Options

| | |
|---|---|
| **ca-profile** *ca-profile-name* | CA profile name. |
| **certificate-id** *certificate-id-name* | Name of the local digital certificate and the public/private key pair. |
| **challenge-password** *password* | Password set by the administrator and normally obtained from the SCEP enrollment webpage of the CA. The password is maximum 256 characters in length. You can enforce the limit to the required characters. |
| **digest (sha-1 \| sha-256)** | Hash algorithm used for signing RSA certificates, either SHA-1 or SHA-256. SHA-1 is the default. |

| | |
|---|---|
| **domain-name** *domain-name* | Fully qualified domain name (FQDN). The FQDN provides the identity of the certificate owner for Internet Key Exchange (IKE) negotiations and provides an alternative to the subject name. |
| **email** *email-address* | E-mail address of the certificate holder. |
| **ip-address** *ip-address* | IP address of the router. |
| **ipv6-address** *ipv6-address* | IPv6 address of the router for the alternate subject. |
| **scep-digest-algorithm (md5 \| sha-1)** | Hash algorithm digest, either MD5 or SHA-1; SHA-1 is the default. |
| **scep-encryption-algorithm (des \| des3)** | Encryption algorithm, either DES or DES3; DES3 is the default. |
| **subject** *subject-distinguished-name* | Distinguished Name (DN) format that contains the domain component, common name, department, serial number, company name, state, and country in the following format: DC, CN, OU, O, SN, L, ST, C.<br>• **DC**—Domain component<br>• **CN**—Common name<br>• **OU**—Organizational unit name<br>• **O**—Organization name<br>• **SN**—Serial number of the device<br>If you define SN in the subject field without the serial number, then the serial number is read directly from the device and added to the certificate signing request (CSR).<br>• **ST**—State<br>• **C**—Country |

## Required Privilege Level

maintenance and security

## Output Fields

When you enter this command, you are provided feedback on the status of your request.

## Sample Output

### command-name

```
user@host> request security pki local-certificate enroll scep certificate-id r3-entrust-scep ca-profile
entrust domain-name router3.example.net subject "CN=router3,OU=Engineering,O=example,C=US"
challenge-password 123
```

```
Certificate enrollment has started. To view the status of your enrollment, check
the public key infrastructure log (pkid) log file at /var/log/pkid. Please save
the challenge-password for revoking this certificate in future.  Note that this
password is not stored on the router.
```

## Sample Output

### Sample output for vSRX 3.0

```
user@host> request security pki generate-key-pair certificate-id example
```

```
Generated key pair example, key size 2048 bits
```

```
user@host> request security pki local-certificate enroll certificate-id ?
```

```
Possible completions:
<certificate-id> Certificate identifier
example
```

```
user@host> request security pki generate-key-pair certificate-id Hub
```

```
error: Failed to generate key pair at HSM. Found a key with the same name at
HSM. Use a different certificate id next time. Refer to PKID logs for more
details
```

### SEE ALSO

- request security pki local-certificate enroll cmpv2
- show security pki local-certificate (View)
- clear security pki local-certificate (Device)

### RELATED DOCUMENTATION

- What is Azure Key Vault?

# vSRX 3.0 Scaling for Internal and Outbound Traffic Using Azure Load Balancer and Virtual Machine Scale Sets

IN THIS SECTION



This section provides you details on vSRX 3.0 scaling and performance improvements for internal traffic (traffic between the Microsoft Azure virtual network and the internet) and outbound traffic using Microsoft Azure Load Balancer (LB) and Microsoft Azure Virtual Machine Scale Sets (VMSS). It also provides you details on how you deploy vSRX 3.0 with Azure Load Balancer and Virtual Machine Scale Sets in various ways to scale out or scale in vSRX 3.0.

## Overview

IN THIS SECTION



vSRX instances are inline firewalls that serve as security gateways on Azure Coud to protect traffic between the west and east subnets. Sometimes a single vSRX instance cannot handle huge traffic throughput, and any throughput or connection scaling limitations on these firewalls limit the performance and scaling of the entire virtual network.

To handle such huge throughput, you can use multiple vSRX 3.0 instances for the traffic inside the virtual network and for the outbound traffic, as required. You can scale out or scale in vSRX 3.0 instances by adding in and removing vSRX 3.0 instances using Azure infrastructure.

Starting in Junos OS Release 20.3R1, vSRX 3.0 can automatically scale out or scale in for internal and outbound traffic using Azure LB and Azure Virtual Machine Scale Sets. You can use the suggested deployments with Azure Load Balancer and VMSS to achieve vSRX 3.0 scaling and better performance for your business needs.

With Azure Load Balancer, you can scale your applications and create high availability for your services. Azure Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications. Load Balancer distributes new inbound flows that arrive on the Load Balancer's front-end to back-end pool instances, according to rules and health probes.

Azure VMSS let you create and manage a group of identical, load balanced virtual machines (VMs). The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update a large number of VMs. With VMSS, you can build large-scale services for areas such as compute, big data, and container workloads.

Azure Load Balancer also checks vSRX 3.0 health by means of a health probe. If one vSRX 3.0 instance is not healthy according to the health probe, it will be moved out of load balancing.

> **NOTE**: Do not configure NAT for west-east traffic. If NAT is configured, traffic might be distributed to different vSRX 3.0 instances for each direction.

For information about Azure Load Balancer high availability port limitations, see Azure Load Balancer – HA ports.

The core architecture of the vSRX 3.0 scale-out and scale-in solution consists of the following components:

- **Azure Load Balancer**—Provides traffic distribution toward vSRX 3.0 in the back-end pool.
- **Azure Virtual Machine Scale Sets (VMSS)**—Creates and manages a group of vSRX 3.0 VMs as the back-end pool of Load Balancer. Defines automatic scale-in and scale-out rule to trigger automatic scaling.
- **Initial Junos OS configuration**—With the help of cloud-init, autoconfigures each vSRX 3.0 instance in VMSS.

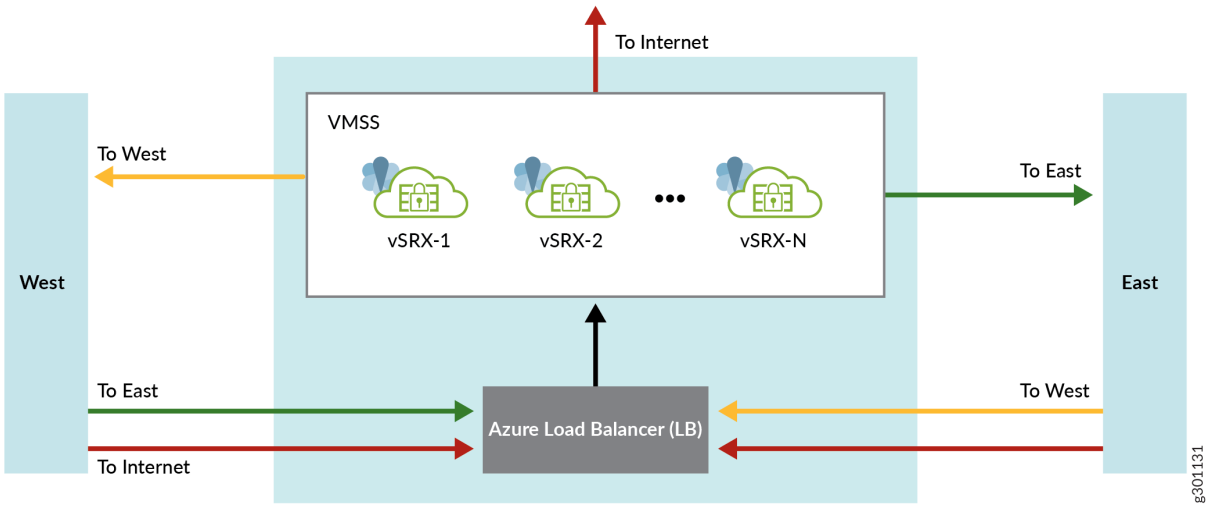
## Benefits of vSRX 3.0 Scaling Using Azure Load Balancer and Virtual Machine Scale Sets

- **Build highly reliable applications**—Improve application reliability through health checks. Azure Load Balancer probes the health of your application instances, automatically takes unhealthy instances out of rotation, and reinstates them when they become healthy again. Use Load Balancer to improve application uptime.
- **Instantly add scale to your applications**—With built-in load balancing for cloud services and VMs, you can create highly available and scalable applications in minutes. Azure Load Balancer supports TCP/UDP-based protocols such as HTTP, HTTPS, and SMTP, and protocols used for real-time voice and video messaging applications.
- **High availability and robust performance for your applications**—Azure Load Balancer automatically scales with increasing application traffic. Without you needing to reconfigure or manage the Load Balancer, your applications provide a better customer experience.
- **Load-balance Internet and private network traffic**—Use the internal Azure load balancer for traffic between VMs inside your private virtual networks, or use it to create multitiered hybrid applications.
- **Secure your networks**—Provides flexible NAT rules for better security. Control your inbound and outbound network traffic, and protect private networks using built-in Network Address Translation (NAT). Secure your network and integrate network security groups with Azure Load Balancer.

## Understanding the vSRX Scale-Out and Scale-In Solution for East-West Traffic

You can manage the east-west traffic internal traffic in Azure Cloud by deploying vSRX 3.0 as demonstrated in Figure 30 on page 133.

**Figure 30: vSRX 3.0 Scaling Deployment for East-West Traffic**



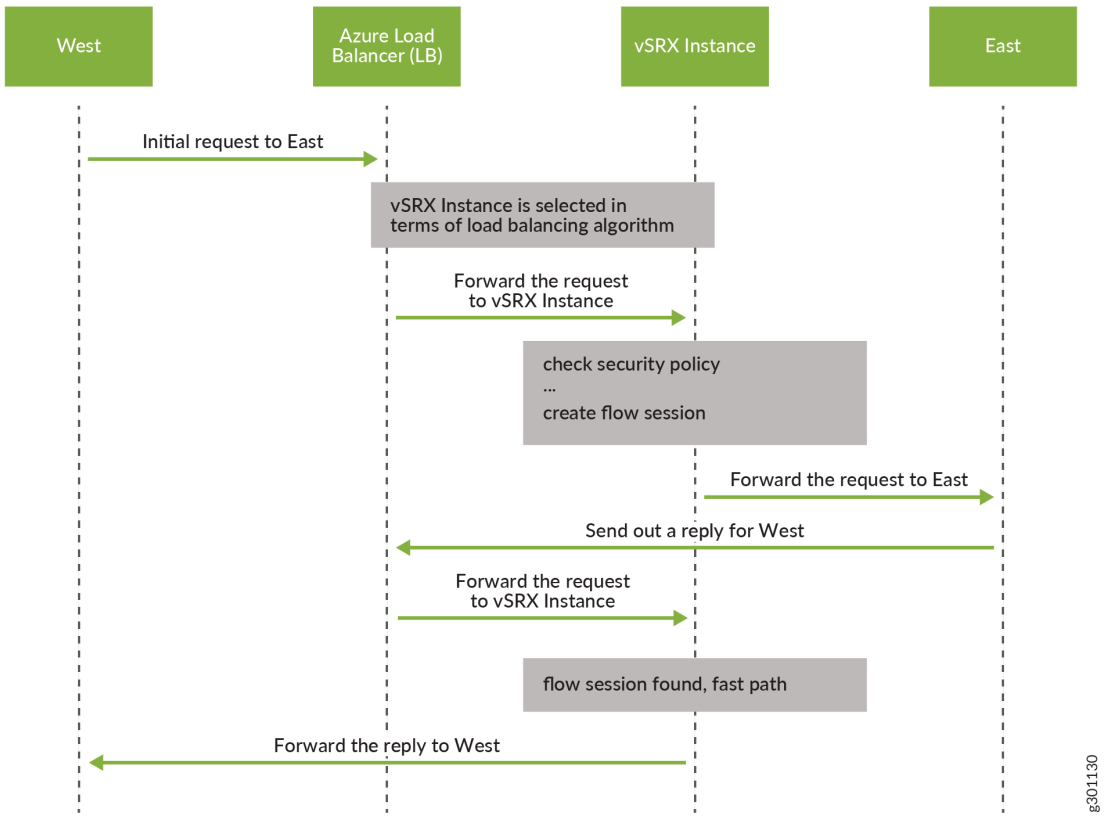

Components of this deployment are:

- West and east Azure vnet or subnet
- Internal Azure Load Balancer
- vSRX 3.0 VMSS

The west and east segments in this illustration represent two user networks, and these networks need to access each other (west-east traffic) or Internet (outbound traffic). The standard Azure internal Load Balancer that helps load-balance all traffic that comes from west and east. The high availability rule for load balancing is configured on the Azure Load Balancer's high availability ports. The high availability ports rule is set with front-end and back-end as port 0 and protocol is set as ALL. The vSRX 3.0 VMSS builds up a vSRX 3.0 group with multiple identical vSRX 3.0 VMs. It acts as a back-end pool of the internal Load Balancer. The Azure internal Load Balancer only distributes traffic to vSRX 3.0 VMSS per flow (5-tuples) according to the load-balancing algorithm. It does not make changes on packets, like destination NAT translation. Its front-end IP is only a route next hop for west or east networks.

Traffic flow and management illustrated in Figure 31 on page 134.

**Figure 31: Packet Flow Between West -East Traffic**




## Manual Deployment of vSRX Scale-In and Scale-Out Solution for East-West Traffic

To manually implement this demonstrated deployment:

1. Sign in to the Azure Portal.
2. Add an Azure network in a new resource group, with four subnets: MGMT, EAST, WEST, SOUTH.
   - Add a Frontend IP
   - Add a Backend pool
   - Add a health probe

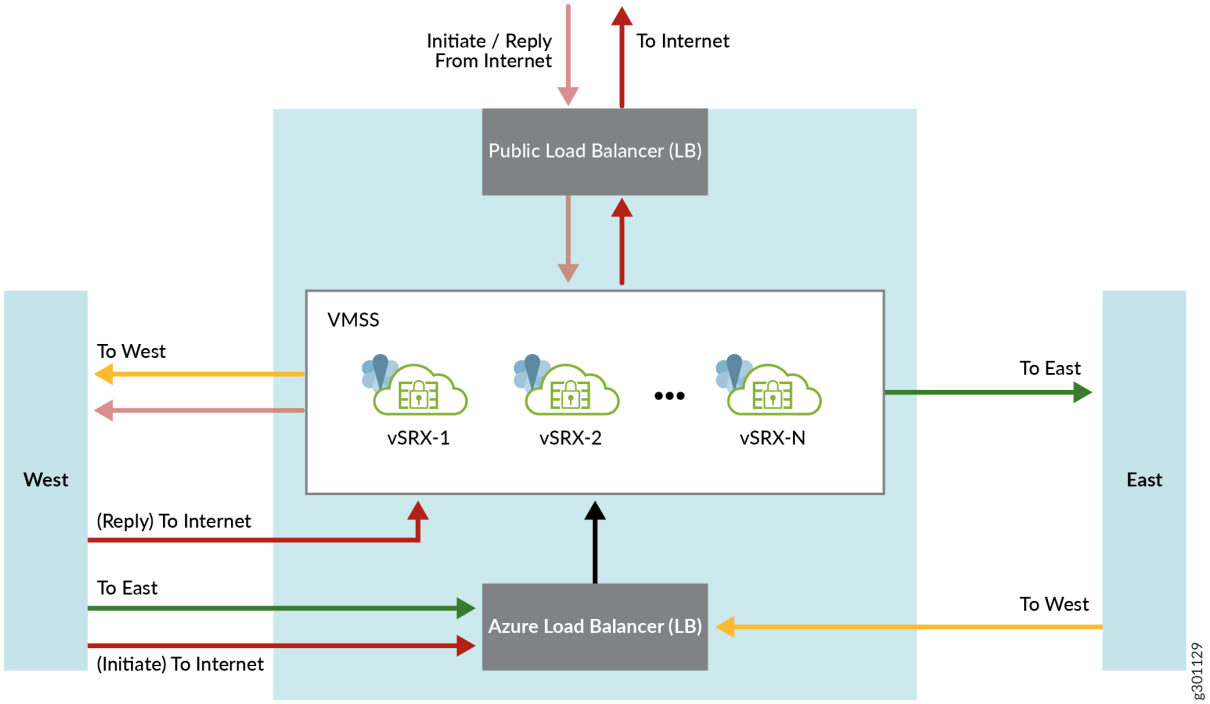
- Add a load-balancing rule (enable high availability [HA]
- Add a network security group, and add inbound security rules for enabling SSH and Wweb service.

3. Add an Azure Load Balancer with type as **Internal** and SKU as **Standard**.
4. Create a VMSS with **Image** as **vSRX** and **Size** as **Standard_DS3_v2**.
   a. In the Networking section, create two network information collectors (NICs) in subnets MGMT and South, respectively and specify the above values in the internal Azure Load Balancer.
   b. In the scaling section, specify the initial instance count as 2, and add a custom scale-out and scale-in rule based on CPU available.
5. Deploy a Linux host in the west or east subnet.
6. Define a route table for subnets west and east, add a User Defined Routing (UDR) route with the next hop as the front-end IP of internal Load Balancer.
7. Configure each vSRX 3.0 instance.

## Understanding vSRX Scale-Out and Scale-In Deployment for South-North Traffic

You can manage the south-north traffic in Azure Cloud by deploying vSRX 3.0 as demonstrated in Figure 32 on page 136.

**Figure 32: vSRX 3.0 Scaling Deployment for South-North Traffic**




Components of this deployment are:

- West and East Azure vnet or subnet
- Internal Microsoft Azure Load Balancer
- vSRX 3.0 VMSS

Traffic flow and management are illustrated in Figure 33 on page 137 and Figure 34 on page 138.

**Figure 33: Packet Flow Between Internet (as request starter) and West Network Segment**

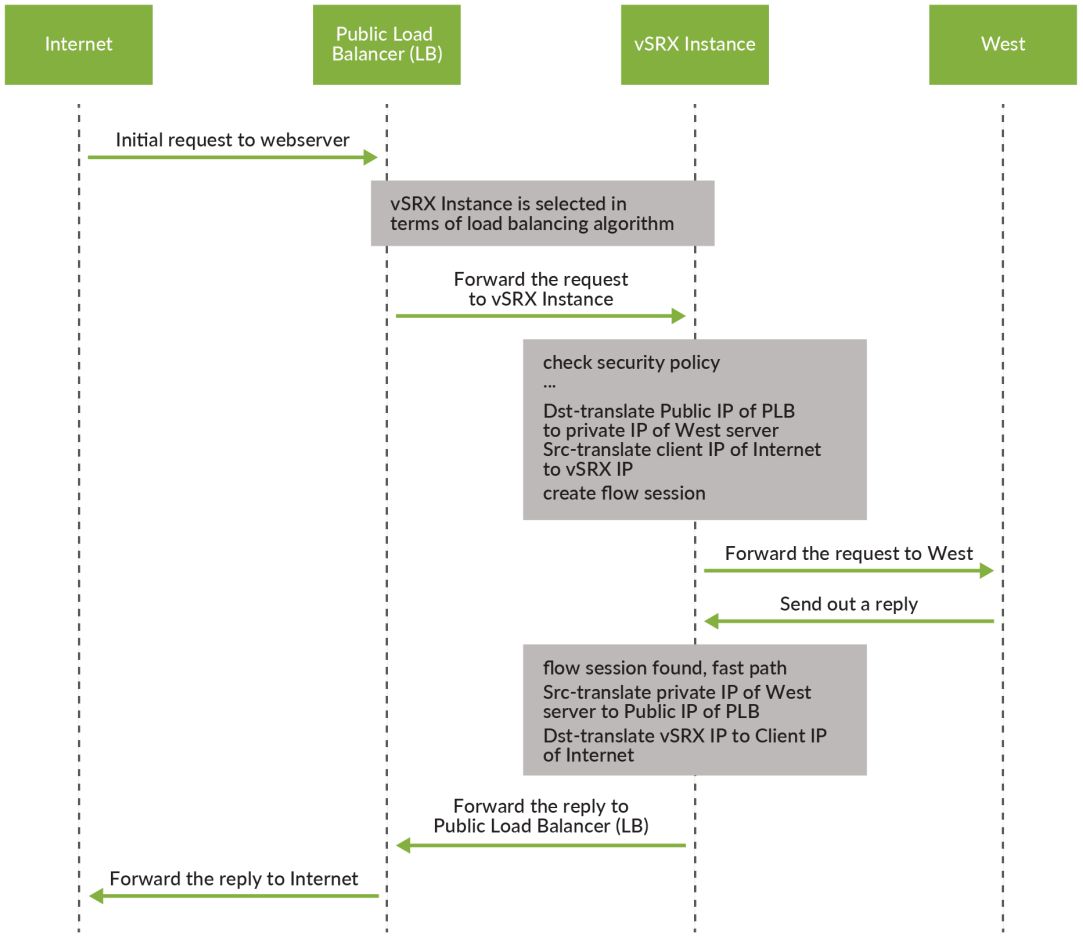
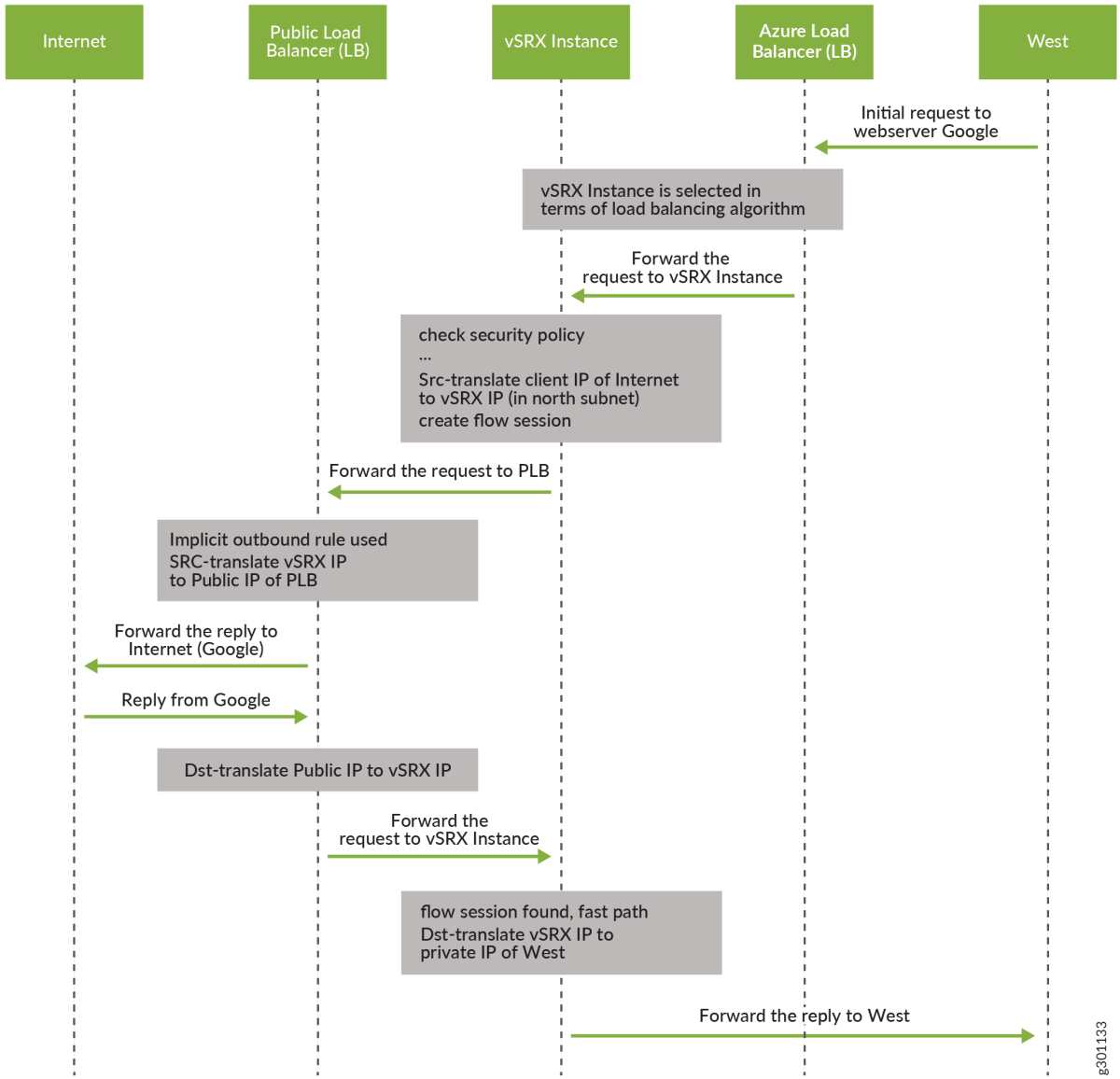


**Figure 34: Packet Flow Between Internet and West Network Segment (as Request Starter)**




## Manual Deployment of vSRX Scale-Out and Scale-In Solution for South-North Traffic

To manually implement this demonstrated deployment:

1. Sign in to the Azure Portal.
2. Add an Azure network in a new resource group, with four subnets: MGMT, EAST, WEST, SOUTH.

- Add a front-end IP
- Add a back-end pool
- Add a health probe
- Add a load-balancing rule (enable high availability [HA])
- Add a network security group, and add Inbound security rules for enabling SSH and web service.

3. Add an Azure Load Balancer with type as **Internal** and SKU as **Standard**.
4. Create a virtual machine scale set with **Image** as **vSRX** and **Size** as **Standard_DS3_v2**.
   a. In the Networking section, create two NICs in subnets MGMT and SOUTH, respectively and specify the above values in the internal Load Balancer.
   b. In the Scaling section, specify initial instance count as 2, and add custom scale-out and scale-in rule based on CPU available.
5. Deploy a Linux host in the west or east subnet.
6. Define a route table for subnets west and east, add a UDR route with a next hop as the front-end IP of internal laod balancer.
7. Add an Azure Load Balancer with type as **Public** and SKU as **Standard**, and create a new public IP.
   - Add a front-end IP by using the above public IP
   - Add a back-end pool, and associate it with vSRX 3.0 VMSS
   - Add a health probe
   - Add a load-balancing rule with port 80, back-end port 80
8. Configure a webserver in the west host.
9. Configure each vSRX 3.0 instance.

## Automatic Deployment of Solutions for vSRX Scaling

This topic provides you steps on how to automatically deploy the vSRX 3.0 scaling solutions, for east-west and south-north traffic.

1. Download vSRX-Azure tool: https://github.com/Juniper/vSRX-Azure/archive/primary.zip.
2. Change the directory using the cd vSRX-Azure/sample-templates/arm-templates-tool command.
3. Deploy east-west solution:

   **./templates/vsrx-scale-out/vsrx.scale.e-w.vsrx.conf.sh > vsrx.scale.e-w.vsrx.conf ./deploy-azure-vsrx.sh -f templates/vsrx-scale-out/vsrx.scale.e-w.json -e templates/vsrx-scale-out/**

**vsrx.scale.parameters.json -r vsrx.scale.e-w.vsrx.conf -g vsrx_scale_e_w ./templates/vsrx-scale-out/linux.deploy.sh vsrx_scale_e_w**

4. Deploy south-north solution:

   **./templates/vsrx-scale-out/vsrx.scale.s-n.vsrx.conf.sh > vsrx.scale.s-n.vsrx.conf ./deploy-azure-vsrx.sh -f templates/vsrx-scale-out/vsrx.scale.s-n.json -e templates/vsrx-scale-out/vsrx.scale.parameters.json -r vsrx.scale.s-n.vsrx.conf -g vsrx_scale_s_n ./templates/vsrx-scale-out/linux.deploy.sh --web vsrx_scale_g_n**

   After deploying south to north solution by using deploy-azure-vsrx.sh, you cannot access the web server in Azure west or east vnet with front-end public IP of the public Load Balancer. You must replace IP address 1.1.1.1 with the real front-end public IP at each vSRX instance of Azure VMSS using the **replace pattern 1.1.1.1 with x.x.x.x** command and then commit the configuration.

5

CHAPTER

# vSRX in Microsoft Azure Use Cases

# Example: Configure an IPsec VPN Between Two vSRX Instances

IN THIS SECTION



This example shows how to configure an IPsec VPN between two instances of vSRX in Microsoft Azure.

## Before You Begin

Ensure that you have installed and launched a vSRX instance in Microsoft Azure virtual network.

See SRX Site-to-Site VPN Configuration Generator and How to troubleshoot a VPN tunnel that is down or not active for additional information.

## Overview

You can use an IPsec VPN to secure traffic between two VNETs in Microsoft Azure using two vSRX instances.

# vSRX IPsec VPN Configuration

IN THIS SECTION



## vSRX1 VPN Configuration

### Step-by-Step Procedure

To configure IPsec VPN on vSRX1:

1. Log in to the vSRX1 in configuration edit mode (see "Configure vSRX Using the CLI" on page 92).
2. Set the IP addresses for vSRX1 interfaces.

```
set interfaces ge-0/0/0 unit 0 family inet address 10.0.0.10/24
set interfaces ge-0/0/1 unit 0 family inet address 10.10.10.10/24
set interfaces st0 unit 1 family inet address 10.0.250.10/24
```

3. Set up the untrust security zone.

```
set security zones security-zone untrust screen untrust-screen
set security zones security-zone untrust host-inbound-traffic system-services ike
set security zones security-zone untrust interfaces ge-0/0/0.0
set security zones security-zone untrust interfaces st0.1
```

4. Set up the trust security zone.

```
set security zone trust host-inbound-traffic system-services https
set security zone trust host-inbound-traffic system-services ssh
set security zone trust host-inbound-traffic system-services ping
set security security-zone trust interfaces ge-0/0/1.0
```

5. Configure IKE.

```
set security ike proposal ike-phase1-proposalA authentication-method pre-shared-keys
set security ike proposal ike-phase1-proposalA dh-group group2
set security ike proposal ike-phase1-proposalA authentication-algorithm sha-256
set security ike proposal ike-phase1-proposalA encryption-algorithm aes-256-cbc
set security ike proposal ike-phase1-proposalA lifetime-seconds 1800
set security ike policy ike-phase1-policyA mode aggressive
set security ike policy ike-phase1-policyA proposals ike-phase1-proposalA
set security ike policy ike-phase1-policyA pre-shared-key ascii-text <preshared-key>
set security ike gateway gw-siteB ike-policy ike-phase1-policyA
set security ike gateway gw-siteB address 198.51.100.10
set security ike gateway gw-siteB local-identity user-at-hostname "source@example.net"
set security ike gateway gw-siteB remote-identity user-at-hostname "dest@example.net"
set security ike gateway gw-siteB external-interface ge-0/0/0.0
```

**NOTE**: Be sure to replace **198.51.100.10** in this example with the correct public IP address.

6. Configure IPsec.

```
set security ipsec proposal ipsec-proposalA protocol esp
set security ipsec proposal ipsec-proposalA authentication-algorithm hmac-sha1-96
set security ipsec proposal ipsec-proposalA encryption-algorithm aes-256-cbc
set security ipsec policy ipsec-policy-siteB proposals ipsec-proposalA
set security ipsec vpn ike-vpn-siteB bind-interface st0.1
set security ipsec vpn ike-vpn-siteB ike gateway gw-siteB
set security ipsec vpn ike-vpn-siteB ike ipsec-policy ike-phase1-policyA
set security ipsec vpn ike-vpn-siteB establish-tunnels immediately
```

7. Configure routing.

```
set routing-instances siteA-vr1 instance-type virtual-router
set routing-instances siteA-vr1 interface ge-0/0/0.0
set routing-instances siteA-vr1 interface ge-0/0/1.0
set routing-instances siteA-vr1 interface st0.1
set routing-instances siteA-vr1 routing-options static route 0.0.0.0/0 next-hop 10.0.0.1
```

```
set routing-instances siteA-vr1 routing-options static route 10.20.20.0/24 next-hop st0.1
commit
```

## vSRX2 VPN Configuration

### Step-by-Step Procedure

To configure IPsec VPN on vSRX2:

1. Log in to the vSRX2 in configuration edit mode (See "Configure vSRX Using the CLI" on page 92.

2. Set the IP addresses for the vSRX2 interfaces.

```
set interfaces ge-0/0/0 unit 0 family inet address 10.1.0.10/24
set interfaces ge-0/0/1 unit 0 family inet address 10.20.20.10/24
set interfaces st0 unit 1 family inet address 10.0.250.20/24
```

3. Set up the untrust security zone.

```
set security zones security-zone untrust screen untrust-screen
set security zones security-zone untrust host-inbound-traffic system-services ike
set security zones security-zone untrust interfaces ge-0/0/0.0
set security zones security-zone untrust interfaces st0.1
```

4. Set up the trust security zone.

```
set security zones security-zone trust host-inbound-traffic system-services https
set security zones security-zone trust host-inbound-traffic system-services ssh
set security zones security-zone trust host-inbound-traffic system-services ping
set security zones security-zone trust interfaces ge-0/0/1.0
```

5. Configure IKE.

```
set security ike proposal ike-phase1-proposalA authentication-method pre-shared-keys
set security ike proposal ike-phase1-proposalA dh-group group2
set security ike proposal ike-phase1-proposalA authentication-algorithm sha-256
set security ike proposal ike-phase1-proposalA encryption-algorithm aes-256-cbc
set security ike proposal ike-phase1-proposalA lifetime-seconds 1800
```

```
set security ike policy ike-phase1-policyA mode aggressive
set security ike policy ike-phase1-policyA proposals ike-phase1-proposalA
set security ike policy ike-phase1-policyA pre-shared-key ascii-text preshared-key
set security ike gateway gw-siteB ike-policy ike-phase1-policyA
set security ike gateway gw-siteB address 203.0.113.10
set security ike gateway gw-siteB local-identity user-at-hostname "dest@example.net"
set security ike gateway gw-siteB remote-identity user-at-hostname "source@example.net"
set security ike gateway gw-siteB external-interface ge-0/0/0.0
```

**NOTE**: Be sure to replace **203.0.113.10** in this example with the correct public IP address. Also note that the SiteB local-identity and remote-identity should be in contrast with the SiteA local-identity and remote-identity.

6. Configure IPsec.

```
set security ipsec proposal ipsec-proposalA protocol esp
set security ipsec proposal ipsec-proposalA authentication-algorithm hmac-sha1-96
set security ipsec proposal ipsec-proposalA encryption-algorithm aes-256-cbc
set security ipsec policy ipsec-policy-siteB proposals ipsec-proposalA
set security ipsec vpn ike-vpn-siteB bind-interface st0.1
set security ipsec vpn ike-vpn-siteB ike gateway gw-siteB
set security ipsec vpn ike-vpn-siteB ike ipsec-policy ike-phase1-policyA
set security ipsec vpn ike-vpn-siteB establish-tunnels immediately
```

7. Configure routing.

```
set routing-instances siteA-vr1 instance-type virtual-router
set routing-instances siteA-vr1 interface ge-0/0/0.0
set routing-instances siteA-vr1 interface ge-0/0/1.0
set routing-instances siteA-vr1 interface st0.1
set routing-instances siteA-vr1 routing-options static route 0.0.0.0/0 next-hop 10.0.0.1
set routing-instances siteA-vr1 routing-options static route 10.20.20.0/24 next-hop st0.1
commit
```

# Verification

IN THIS SECTION



## Verify Active VPN Tunnels

### Purpose

Verify that the tunnel is up on both vSRX instances.

### Action

```
root@> show security ipsec security-associations
```

```
Total active tunnels: 1
ID      Algorithm                SPI      Life:sec/kb    Mon lsys Port  Gateway
<131074 ESP:aes--cbc--256/sha1 de836105 1504/ unlim -- root 4500 52.200.89.XXX
>131074 ESP:aes--cbc--256/sha1 b349bc84 1504/ unlim -- root 4500 52.200.89.XXX
```

## RELATED DOCUMENTATION

- IPsec VPN Overview
- Application Firewall Overview

# Example: Configure an IPsec VPN Between a vSRX and Virtual Network Gateway in Microsoft Azure

IN THIS SECTION



This example shows how to configure an IPsec VPN between a vSRX instance and a virtual network gateway in Microsoft Azure.

## Before You Begin

Ensure that you have installed and launched a vSRX instance in Microsoft Azure virtual network.

See SRX Site-to-Site VPN Configuration Generator and How to troubleshoot a VPN tunnel that is down or not active for additional information.

## Overview

You can use an IPsec VPN to secure traffic between two VNETs in Microsoft Azure, with one vSRX protecting one VNet and the Azure virtual network gateway protecting the other VNet.

# vSRX IPsec VPN Configuration

IN THIS SECTION



## Procedure

### Step-by-Step Procedure

To configure IPsec VPN on vSRX:

1. Log in to the vSRX in configuration edit mode (see "Configure vSRX Using the CLI" on page 92).
2. Set the IP addresses for vSRX interfaces.

```
set interfaces ge-0/0/0 unit 0 family inet address 10.0.0.10/24
set interfaces ge-0/0/1 unit 0 family inet address 10.10.10.10/24
set interfaces st0 unit 1 family inet address 10.0.250.10/24
```

3. Set up the untrust security zone.

```
set security zones security-zone untrust screen untrust-screen
set security zones security-zone untrust host-inbound-traffic system-services ike
set security zones security-zone untrust interfaces ge-0/0/0.0
set security zones security-zone untrust interfaces st0.1
```

4. Set up the trust security zone.

```
set security zone trust host-inbound-traffic system-services https
set security zone trust host-inbound-traffic system-services ssh
set security zone trust host-inbound-traffic system-services ping
set security security-zone trust interfaces ge-0/0/1.0
```

5. Configure IKE.

```
set security ike proposal ike-phase1-proposalA authentication-method pre-shared-keys
set security ike proposal ike-phase1-proposalA dh-group group2
set security ike proposal ike-phase1-proposalA authentication-algorithm sha-256
set security ike proposal ike-phase1-proposalA encryption-algorithm aes-256-cbc
set security ike policy ike-phase1-policyA mode main
set security ike policy ike-phase1-policyA proposals ike-phase1-proposalA
set security ike policy ike-phase1-policyA pre-shared-key ascii-text <preshared-key>
set security ike gateway gw-siteB ike-policy ike-phase1-policyA
set security ike gateway gw-siteB address 52.175.210.65
set security ike gateway gw-siteB version v2-only
set security ike gateway gw-siteB external-interface ge-0/0/0.0
```

**NOTE**: Be sure to replace **52.175.210.65** in this example with the correct public IP address.

6. Configure IPsec.

The following example illustrates a vSRX IPsec configuration using the CBC encryption algorithm:

```
set security ipsec proposal ipsec-proposalA protocol esp
set security ipsec proposal ipsec-proposalA authentication-algorithm hmac-sha1-96
set security ipsec proposal ipsec-proposalA encryption-algorithm aes-256-cbc
set security ipsec proposal ipsec-proposalA lifetime-seconds 7200
set security ipsec proposal ipsec-proposalA lifetime-kilobytes 102400000
set security ipsec policy ipsec-policy-siteB proposals ipsec-proposalA
set security ipsec vpn ike-vpn-siteB bind-interface st0.1
set security ipsec vpn ike-vpn-siteB ike gateway gw-siteB
set security ipsec vpn ike-vpn-siteB ike ipsec-policy ike-phase1-policyA
set security ipsec vpn ike-vpn-siteB establish-tunnels immediately
```

If required, you can use AES-GCM as the encryption algorithm in the vSRX IPsec configuration instead of CBC:

```
set security ipsec proposal ipsec-proposalA protocol esp
set security ipsec proposal ipsec-proposalA encryption-algorithm aes-256-gcm
set security ipsec proposal ipsec-proposalA lifetime-seconds 7200
set security ipsec proposal ipsec-proposalA lifetime-kilobytes 102400000
```

```
set security ipsec policy ipsec-policy-siteB proposals ipsec-proposalA
set security ipsec vpn ike-vpn-siteB bind-interface st0.1
set security ipsec vpn ike-vpn-siteB ike gateway gw-siteB
set security ipsec vpn ike-vpn-siteB ike ipsec-policy ike-phase1-policyA
set security ipsec vpn ike-vpn-siteB establish-tunnels immediately
```

7. Configure routing.

```
set routing-instances siteA-vr1 instance-type virtual-router
set routing-instances siteA-vr1 interface ge-0/0/0.0
set routing-instances siteA-vr1 interface ge-0/0/1.0
set routing-instances siteA-vr1 interface st0.1
set routing-instances siteA-vr1 routing-options static route 0.0.0.0/0 next-hop 10.0.0.1
set routing-instances siteA-vr1 routing-options static route 10.20.20.0/24 next-hop st0.1
commit
```

## Microsoft Azure Virtual Network Gateway Configuration

IN THIS SECTION



### Procedure

#### Step-by-Step Procedure

1. To configure the Microsoft Azure virtual network gateway, refer to the following Microsoft Azure procedure:

   Configure IPsec/IKE policy for S2S VPN or VNet-to-VNet connections

   Ensure the IPSec IKE parameters in Microsoft Azure virtual network gateway match the vSRX IPSec IKE parameters when the site-to-site VPN connection is formed.

2. Verify Active VPN Tunnels.

Verify that the tunnel is up between the vSRX instance and the Azure virtual network gateway.

```
root@> show security ike security-associations
```

```
Index   State  Initiator cookie  Responder cookie  Mode           Remote
Address
8290401 UP     b1adf15fc3dfe0b0  89cc2a12cb7e3cd7  IKEv2
52.175.210.65
```

```
root@> show security ipsec security-associations
```

```
Total active tunnels: 1
  ID    Algorithm       SPI      Life:sec/kb  Mon lsys Port  Gateway
  <131073 ESP:aes-gcm-256/None c0e154e2 5567/  102399997 - root 4500
52.175.210.65
  >131073 ESP:aes-gcm-256/None 383bd606 5567/  102399997 - root 4500
52.175.210.65
```

## RELATED DOCUMENTATION

- IPsec VPN Overview
- Application Firewall Overview

# Example: Configure Juniper Sky ATP for vSRX

**IN THIS SECTION**

This example shows how to configure Juniper Sky™ Advanced Threat Prevention (Juniper Sky ATP) on a vSRX instance that is deployed in a virtual private cloud (VPC).

## Before You Begin

Ensure that you have installed and launched a vSRX instance in a VPC.

## Overview

You can use Juniper Sky ATP, a cloud-based solution, along with vSRX to protect all hosts in your network against evolving security threats.

## Juniper Sky ATP Configuration

IN THIS SECTION



### Procedure

#### Step-by-Step Procedure

To configure Juniper Sky ATP on a vSRX instance:

1. Log in to the vSRX instance using SSH and start the CLI.

```
root@% cli
root@>
```

2. Enter configuration mode.

```
root@> configure
[edit]
root@#
```

3. Set up the correct data interface for the active advanced antimalware (AAMW) service instead of using the default fxp0 interface.

```
root@# set services advanced-anti-malware connection source-interface ge-0/0/0.0
```

4. Configure NAT.

```
root@# set security nat source rule-set rs1 from zone trust
root@# set security nat source rule-set rs1 to zone untrust
root@# set security nat source rule-set rs1 rule r1 match source-address 0.0.0.0/0
root@# set security nat source rule-set rs1 rule r1 match destination-address 0.0.0.0/0
root@# set security nat source rule-set rs1 rule r1 then source-nat interface
```

5. Set up virtual routing instance for the correct data interface for AAMW service.

```
root@# set routing-instances vsrx-vr1 instance-type virtual-router
root@# set routing-instances vsrx-vr1 routing-options static route 0.0.0.0/0 next-hop 10.4.1.1
root@# set routing-instances vsrx-vr1 interface ge-0/0/0.0
root@# set routing-instances vsrx-vr1 interface ge-0/0/1.0
```

6. Verify the configuration.

```
root@# commit check
configuration check succeeds
```

7. Commit the configuration to activate it on the vSRX instance.

```
root@# commit
commit complete
```

8. Optionally, you can verify the configuration by running the following show commands in the configuration mode:

   - show services advanced-anti-malware connection | display set
   - show security nat | display set
   - show routing-instances vsrx-vr1 | display set

RELATED DOCUMENTATION

Juniper Sky Advanced Threat Prevention Administration Guide

# vSRX 3.0 Scaling for Internal and Outbound Traffic Using Azure Load Balancer and Virtual Machine Scale Sets

IN THIS SECTION



This section provides you details on vSRX 3.0 scaling and performance improvements for internal traffic (traffic between the Microsoft Azure virtual network and the internet) and outbound traffic using Microsoft Azure Load Balancer (LB) and Microsoft Azure Virtual Machine Scale Sets (VMSS). It also provides you details on how you deploy vSRX 3.0 with Azure Load Balancer and Virtual Machine Scale Sets in various ways to scale out or scale in vSRX 3.0.

# Overview

IN THIS SECTION



vSRX instances are inline firewalls that serve as security gateways on Azure Coud to protect traffic between the west and east subnets. Sometimes a single vSRX instance cannot handle huge traffic throughput, and any throughput or connection scaling limitations on these firewalls limit the performance and scaling of the entire virtual network.

To handle such huge throughput, you can use multiple vSRX 3.0 instances for the traffic inside the virtual network and for the outbound traffic, as required. You can scale out or scale in vSRX 3.0 instances by adding in and removing vSRX 3.0 instances using Azure infrastructure.

Starting in Junos OS Release 20.3R1, vSRX 3.0 can automatically scale out or scale in for internal and outbound traffic using Azure LB and Azure Virtual Machine Scale Sets. You can use the suggested deployments with Azure Load Balancer and VMSS to achieve vSRX 3.0 scaling and better performance for your business needs.

With Azure Load Balancer, you can scale your applications and create high availability for your services. Azure Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications. Load Balancer distributes new inbound flows that arrive on the Load Balancer's front-end to back-end pool instances, according to rules and health probes.

Azure VMSS let you create and manage a group of identical, load balanced virtual machines (VMs). The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update a large number of VMs. With VMSS, you can build large-scale services for areas such as compute, big data, and container workloads.

Azure Load Balancer also checks vSRX 3.0 health by means of a health probe. If one vSRX 3.0 instance is not healthy according to the health probe, it will be moved out of load balancing.

**NOTE**: Do not configure NAT for west-east traffic. If NAT is configured, traffic might be distributed to different vSRX 3.0 instances for each direction.

For information about Azure Load Balancer high availability port limitations, see Azure Load Balancer – HA ports.

The core architecture of the vSRX 3.0 scale-out and scale-in solution consists of the following components:

- **Azure Load Balancer**—Provides traffic distribution toward vSRX 3.0 in the back-end pool.
- **Azure Virtual Machine Scale Sets (VMSS)**—Creates and manages a group of vSRX 3.0 VMs as the back-end pool of Load Balancer. Defines automatic scale-in and scale-out rule to trigger automatic scaling.
- **Initial Junos OS configuration**—With the help of cloud-init, autoconfigures each vSRX 3.0 instance in VMSS.

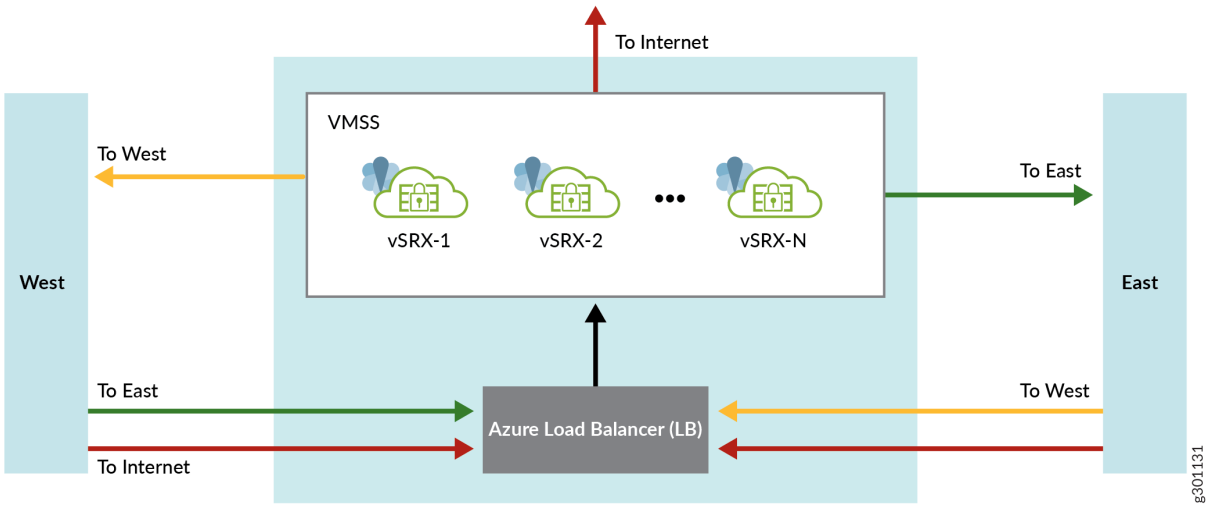
## Benefits of vSRX 3.0 Scaling Using Azure Load Balancer and Virtual Machine Scale Sets

- **Build highly reliable applications**—Improve application reliability through health checks. Azure Load Balancer probes the health of your application instances, automatically takes unhealthy instances out of rotation, and reinstates them when they become healthy again. Use Load Balancer to improve application uptime.
- **Instantly add scale to your applications**—With built-in load balancing for cloud services and VMs, you can create highly available and scalable applications in minutes. Azure Load Balancer supports TCP/UDP-based protocols such as HTTP, HTTPS, and SMTP, and protocols used for real-time voice and video messaging applications.
- **High availability and robust performance for your applications**—Azure Load Balancer automatically scales with increasing application traffic. Without you needing to reconfigure or manage the Load Balancer, your applications provide a better customer experience.
- **Load-balance Internet and private network traffic**—Use the internal Azure load balancer for traffic between VMs inside your private virtual networks, or use it to create multitiered hybrid applications.
- **Secure your networks**—Provides flexible NAT rules for better security. Control your inbound and outbound network traffic, and protect private networks using built-in Network Address Translation (NAT). Secure your network and integrate network security groups with Azure Load Balancer.

## Understanding the vSRX Scale-Out and Scale-In Solution for East-West Traffic

You can manage the east-west traffic internal traffic in Azure Cloud by deploying vSRX 3.0 as demonstrated in Figure 30 on page 133.

**Figure 35: vSRX 3.0 Scaling Deployment for East-West Traffic**



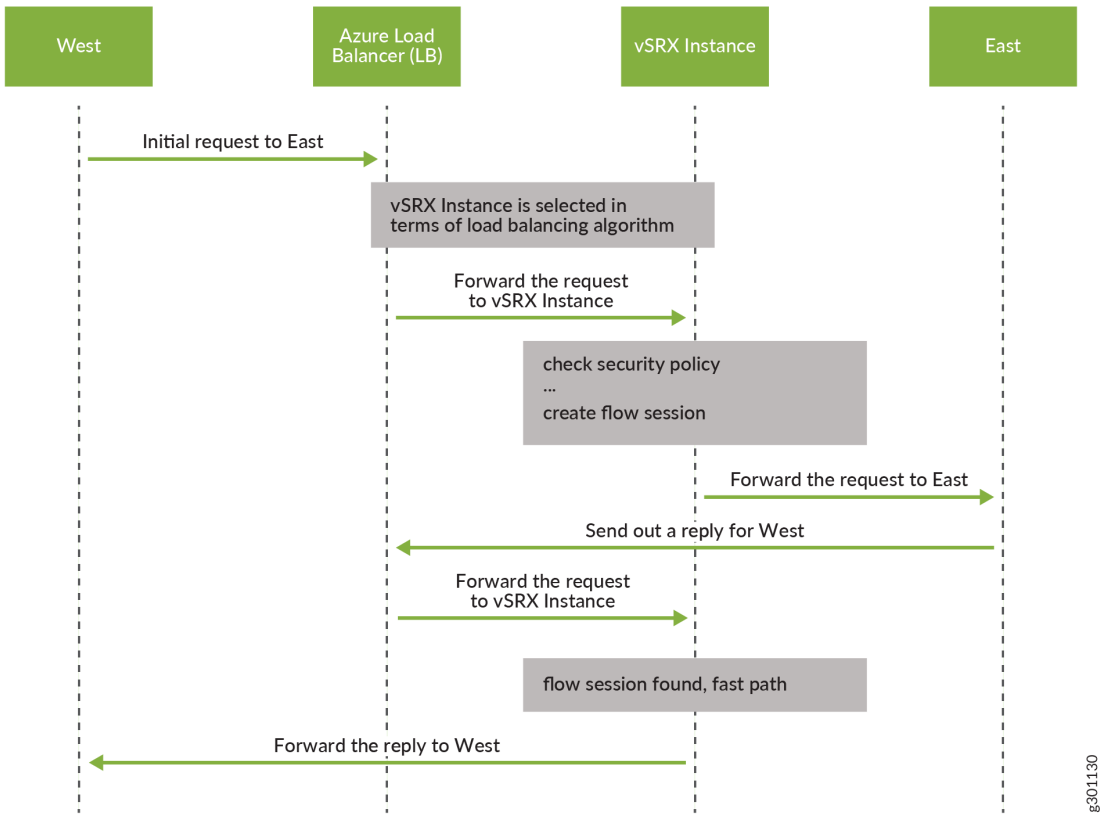

Components of this deployment are:

- West and east Azure vnet or subnet
- Internal Azure Load Balancer
- vSRX 3.0 VMSS

The west and east segments in this illustration represent two user networks, and these networks need to access each other (west-east traffic) or Internet (outbound traffic). The standard Azure internal Load Balancer that helps load-balance all traffic that comes from west and east. The high availability rule for load balancing is configured on the Azure Load Balancer's high availability ports. The high availability ports rule is set with front-end and back-end as port 0 and protocol is set as ALL. The vSRX 3.0 VMSS builds up a vSRX 3.0 group with multiple identical vSRX 3.0 VMs. It acts as a back-end pool of the internal Load Balancer. The Azure internal Load Balancer only distributes traffic to vSRX 3.0 VMSS per flow (5-tuples) according to the load-balancing algorithm. It does not make changes on packets, like destination NAT translation. Its front-end IP is only a route next hop for west or east networks.

Traffic flow and management illustrated in Figure 31 on page 134.

**Figure 36: Packet Flow Between West -East Traffic**




## Manual Deployment of vSRX Scale-In and Scale-Out Solution for East-West Traffic

To manually implement this demonstrated deployment:

1. Sign in to the Azure Portal.
2. Add an Azure network in a new resource group, with four subnets: MGMT, EAST, WEST, SOUTH.
   - Add a Frontend IP
   - Add a Backend pool
   - Add a health probe

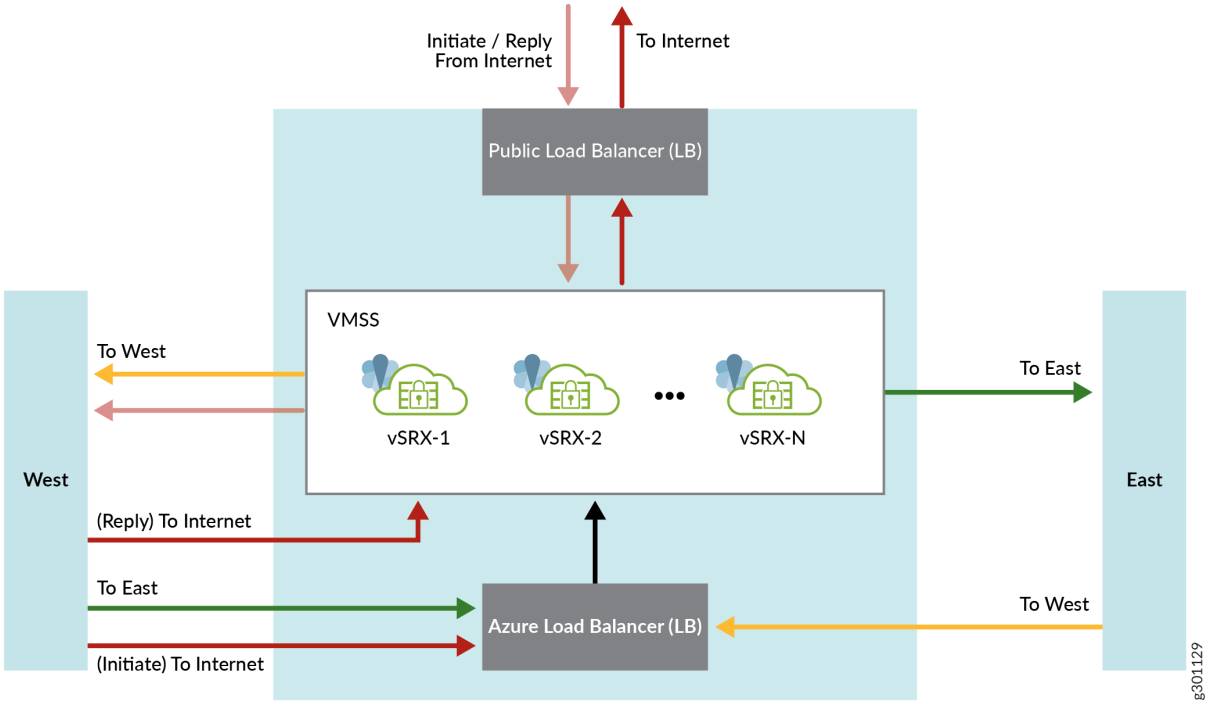
- Add a load-balancing rule (enable high availability [HA]
- Add a network security group, and add inbound security rules for enabling SSH and Wweb service.

3. Add an Azure Load Balancer with type as **Internal** and SKU as **Standard**.
4. Create a VMSS with **Image** as **vSRX** and **Size** as **Standard_DS3_v2**.
   a. In the Networking section, create two network information collectors (NICs) in subnets MGMT and South, respectively and specify the above values in the internal Azure Load Balancer.
   b. In the scaling section, specify the initial instance count as 2, and add a custom scale-out and scale-in rule based on CPU available.
5. Deploy a Linux host in the west or east subnet.
6. Define a route table for subnets west and east, add a User Defined Routing (UDR) route with the next hop as the front-end IP of internal Load Balancer.
7. Configure each vSRX 3.0 instance.

# Understanding vSRX Scale-Out and Scale-In Deployment for South-North Traffic

You can manage the south-north traffic in Azure Cloud by deploying vSRX 3.0 as demonstrated in Figure 32 on page 136.

**Figure 37: vSRX 3.0 Scaling Deployment for South-North Traffic**




Components of this deployment are:

- West and East Azure vnet or subnet
- Internal Microsoft Azure Load Balancer
- vSRX 3.0 VMSS

Traffic flow and management are illustrated in Figure 33 on page 137 and Figure 34 on page 138.

**Figure 38: Packet Flow Between Internet (as request starter) and West Network Segment**

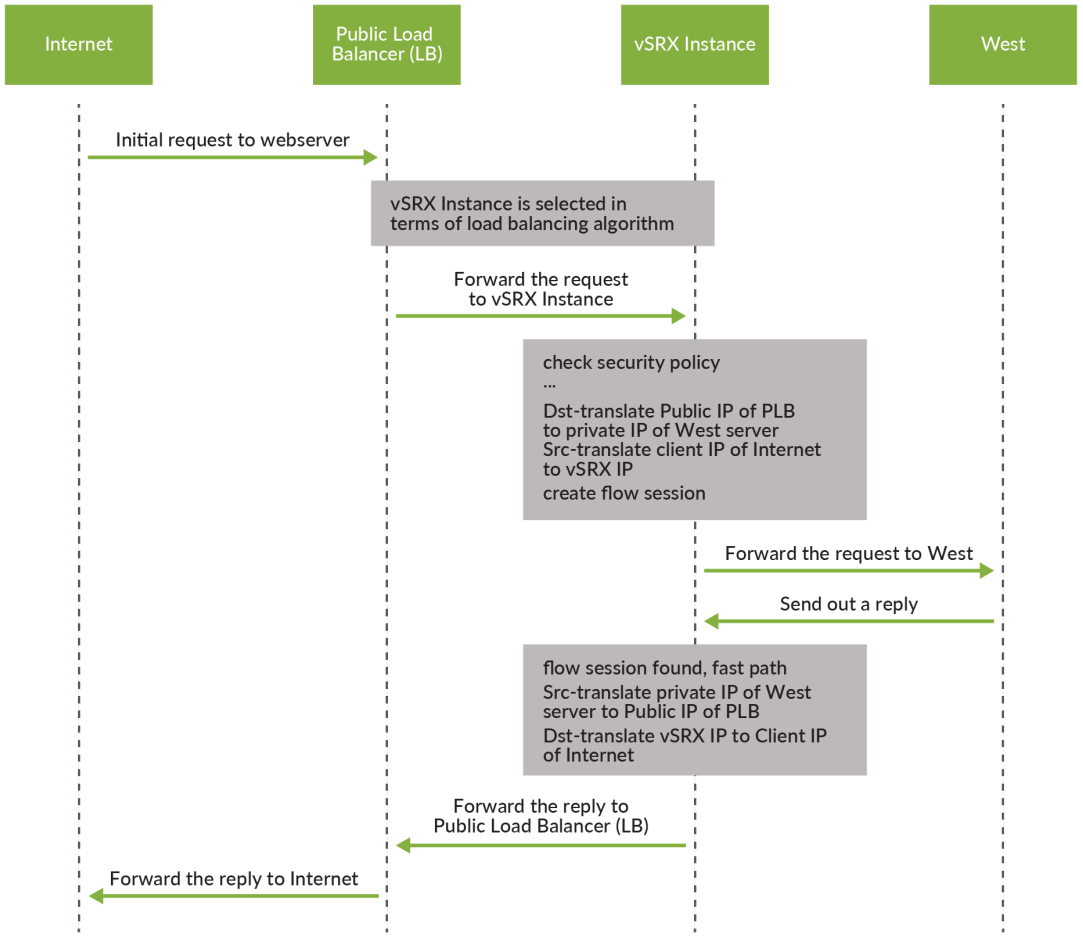
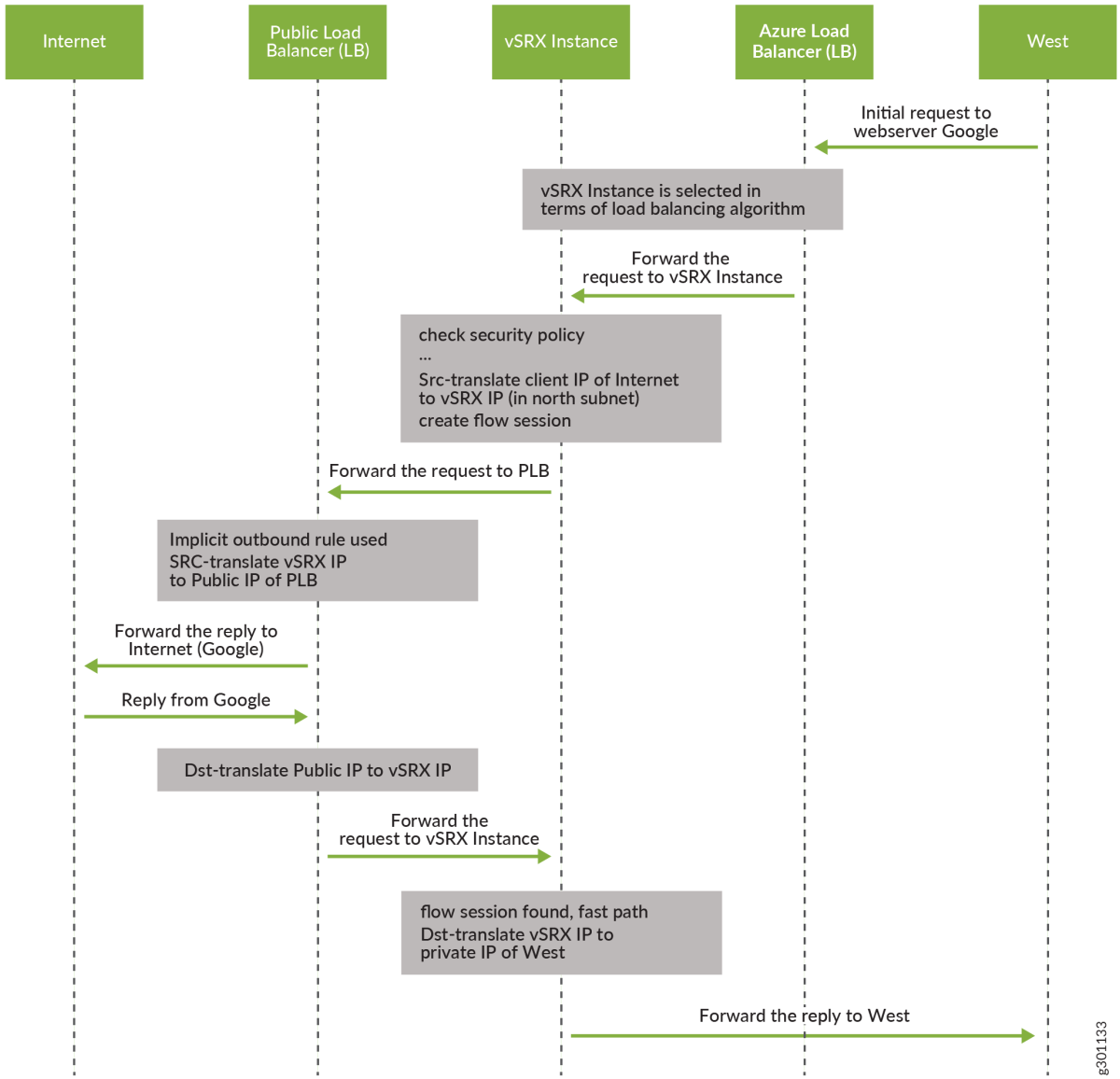


**Figure 39: Packet Flow Between Internet and West Network Segment (as Request Starter)**




## Manual Deployment of vSRX Scale-Out and Scale-In Solution for South-North Traffic

To manually implement this demonstrated deployment:

1. Sign in to the Azure Portal.
2. Add an Azure network in a new resource group, with four subnets: MGMT, EAST, WEST, SOUTH.

- Add a front-end IP
- Add a back-end pool
- Add a health probe
- Add a load-balancing rule (enable high availability [HA])
- Add a network security group, and add Inbound security rules for enabling SSH and web service.

3. Add an Azure Load Balancer with type as **Internal** and SKU as **Standard**.
4. Create a virtual machine scale set with **Image** as **vSRX** and **Size** as **Standard_DS3_v2**.
   a. In the Networking section, create two NICs in subnets MGMT and SOUTH, respectively and specify the above values in the internal Load Balancer.
   b. In the Scaling section, specify initial instance count as 2, and add custom scale-out and scale-in rule based on CPU available.
5. Deploy a Linux host in the west or east subnet.
6. Define a route table for subnets west and east, add a UDR route with a next hop as the front-end IP of internal laod balancer.
7. Add an Azure Load Balancer with type as **Public** and SKU as **Standard**, and create a new public IP.
   - Add a front-end IP by using the above public IP
   - Add a back-end pool, and associate it with vSRX 3.0 VMSS
   - Add a health probe
   - Add a load-balancing rule with port 80, back-end port 80
8. Configure a webserver in the west host.
9. Configure each vSRX 3.0 instance.

## Automatic Deployment of Solutions for vSRX Scaling

This topic provides you steps on how to automatically deploy the vSRX 3.0 scaling solutions, for east-west and south-north traffic.

1. Download vSRX-Azure tool: https://github.com/Juniper/vSRX-Azure/archive/primary.zip.
2. Change the directory using the cd vSRX-Azure/sample-templates/arm-templates-tool command.
3. Deploy east-west solution:

   **./templates/vsrx-scale-out/vsrx.scale.e-w.vsrx.conf.sh > vsrx.scale.e-w.vsrx.conf ./deploy-azure-vsrx.sh -f templates/vsrx-scale-out/vsrx.scale.e-w.json -e templates/vsrx-scale-out/**

**vsrx.scale.parameters.json -r vsrx.scale.e-w.vsrx.conf -g vsrx_scale_e_w ./templates/vsrx-scale-out/linux.deploy.sh vsrx_scale_e_w**

4. Deploy south-north solution:

   **./templates/vsrx-scale-out/vsrx.scale.s-n.vsrx.conf.sh > vsrx.scale.s-n.vsrx.conf ./deploy-azure-vsrx.sh -f templates/vsrx-scale-out/vsrx.scale.s-n.json -e templates/vsrx-scale-out/vsrx.scale.parameters.json -r vsrx.scale.s-n.vsrx.conf -g vsrx_scale_s_n ./templates/vsrx-scale-out/linux.deploy.sh --web vsrx_scale_g_n**

   After deploying south to north solution by using deploy-azure-vsrx.sh, you cannot access the web server in Azure west or east vnet with front-end public IP of the public Load Balancer. You must replace IP address 1.1.1.1 with the real front-end public IP at each vSRX instance of Azure VMSS using the **replace pattern 1.1.1.1 with x.x.x.x** command and then commit the configuration.