os-traits Documentation

Release 3.2.1.dev5

OpenStack Foundation

Jan 28, 2025

CONTENTS

os-traits is a library containing standardized trait strings.

Traits are strings that represent a feature of some resource provider. This library contains the catalog of constants that have been standardized in the OpenStack community to refer to a particular hardware, virtualization, storage, network, or device trait.

CHAPTER

ONE

INSTALLATION GUIDE

1.1 Installation

At the command line:

```
$ pip install os-traits
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv os-traits
$ pip install os-traits
```

CHAPTER

TWO

USAGE GUIDE

2.1 Usage

os-traits is primarily composed of a set of constants that may be referenced by simply importing the `os_traits` module and referencing one of the modules traits constants:

```
$ python3
Python 3.6.8 (default, Oct  7 2019, 12:59:55)
[GCC 8.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import os_traits as ot
>>> print(ot.HW_CPU_X86_SSE42)
HW_CPU_X86_SSE42
```

You can get a list of the `os_traits` symbols by simply doing a `dir(os_traits)`.

Want to see the trait strings for a subset of traits? Theres a method for that too:

```
>>> import pprint
>>> pprint.pprint(ot.get_traits(prefix='HW_CPU_X86_'))
['HW_CPU_X86_FMA3',
'HW_CPU_X86_AVX',
'HW_CPU_X86_MMX',
'HW_CPU_X86_MPX',
'HW_CPU_X86_CLMUL',
'HW_CPU_X86_AVX512VL',
'HW_CPU_X86_AVX512CD',
'HW_CPU_X86_BMI',
'HW_CPU_X86_AVX512DQ',
'HW_CPU_X86_SSE3',
'HW_CPU_X86_ABM',
'HW_CPU_X86_SSE4A',
'HW_CPU_X86_AESNI',
'HW_CPU_X86_F16C',
'HW_CPU_X86_VMX',
'HW_CPU_X86_SVM',
'HW_CPU_X86_TSX',
'HW_CPU_X86_AVX512PF',
'HW_CPU_X86_SSE41',
'HW_CPU_X86_ASF',
```

(continues on next page)

(continued from previous page)

```
'HW_CPU_X86_SGX',
'HW_CPU_X86_SSE',
'HW_CPU_X86_SSSE3',
'HW_CPU_X86_SHA',
'HW_CPU_X86_TBM',
'HW_CPU_X86_SSE42',
'HW_CPU_X86_3DNOW',
'HW_CPU_X86_BMI2',
'HW_CPU_X86_AVX512BW',
'HW_CPU_X86_XOP',
'HW_CPU_X86_AVX2',
'HW_CPU_X86_AVX512F',
'HW_CPU_X86_SSE2',
'HW_CPU_X86_FMA4',
'HW_CPU_X86_AVX512ER']
```

CHAPTER

THREE

CONTRIBUTOR GUIDE

3.1 Contributing

3.1.1 Trait lifecycle policy

It is the policy of this project that once registered, traits should never be removed, even those which will never be used by code (e.g. as a result of pivots in design or changes to the namespaces).

The general principle behind this policy is simply that an extensible-only enumeration is easier to manage than one than can be shrunk. One particular example concerns the need for the placement service to keep its database in sync with the strings in os-traits. Whenever a placement service sees a new version of os-traits it syncs up its database with the strings that are in the package, creating a row in the traits table, with an id that becomes a foreign key in other tables. If traits could be removed, then extra clean-up code might be needed in several places to handle this, and this would be particularly error-prone when execution of that code would need to be correctly orchestrated across multiple projects.

3.1.2 Generic instructions for contributing

If you would like to contribute to the development of OpenStack, you must follow the steps in this page:

> https://docs.openstack.org/infra/manual/developers.html

If you already have a good understanding of how the system works and your OpenStack accounts are set up, you can skip to the development workflow section of this documentation to learn how changes to OpenStack should be submitted for review via the Gerrit tool:

> https://docs.openstack.org/infra/manual/developers.html#development-workflow

Pull requests submitted through GitHub will be ignored.

Bugs should be filed on StoryBoard, not GitHub:

> https://storyboard.openstack.org/#!/project/openstack/os-traits

The code is hosted at:

> https://opendev.org/openstack/os-traits

CHAPTER

FOUR

REFERENCE

4.1 Reference

4.1.1 CUDA

Applications that need to perform massively parallel operations, like processing large arrays, may use the CUDA framework to accelerate their processing on graphics processing units (GPUs). The CUDA framework has two complementary pieces to it.

There are a set of GPU instruction set extensions that are implemented by various graphics cards. These instruction set extensions are known as the CUDA Compute Capability.

The second part of the framework is an SDK that allows developers to take advantage of the hardwares instruction set extensions of a particular version (a specific CUDA Compute Capability version, that is).

An application will link with a version of the CUDA SDK, and the version of the CUDA SDK controls which CUDA Compute Capability versions the application will be able to work with.

The `os_traits.hw.gpu.cuda` module contains traits for both the CUDA compute capability version as well as the CUDA SDK version. For example, `os_traits.hw.gpu.cuda.COMPUTE_CAPABILITY_V3_2` and `os_traits.hw.gpu.cuda.SDK_V6_5`.

The `os_traits.hw.gpu.cuda` module contains a utility function called `compute_capabilities_supported()` that accepts a trait indicating the CUDA SDK version and returns a `set()` containing the matching CUDA compute capability traits that that version of the CUDA SDK knows how to utilize.

Here is an example of listing the CUDA compute capability version traits that the CUDA SDK 8.0 is capable of working with:

```
>>> from os_traits.hw.gpu import cuda
>>> import pprint
>>>
>>> sdk8_caps = cuda.compute_capabilities_supported(cuda.SDK_V8_0)
>>> pprint.pprint(sdk8_caps)
set(['HW_GPU_CUDA_COMPUTE_CAPABILITY_V2_0',
     'HW_GPU_CUDA_COMPUTE_CAPABILITY_V2_1',
```

(continues on next page)

(continued from previous page)

```
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_0',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_2',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_5',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_7',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_0',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_2',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_3',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_0',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_1',
    'HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_2'])
```

For more information on CUDA, see the Wikipedia article.

4.1.2 AMD SEV

While data is typically encrypted today when stored on disk, it is stored in DRAM in the clear. This can leave the data vulnerable to snooping by unauthorized administrators or software, or by hardware probing. New non-volatile memory technology (NVDIMM) exacerbates this problem since an NVDIMM chip can be physically removed from a system with the data intact, similar to a hard drive. Without encryption any stored information such as sensitive data, passwords, or secret keys can be easily compromised.

AMDs SEV (Secure Encrypted Virtualization) is a VM protection technology which transparently encrypts the memory of each VM with a unique key. It can also calculate a signature of the memory contents, which can be sent to the VMs owner as an attestation that the memory was encrypted correctly by the firmware. SEV is particularly applicable to cloud computing since it can reduce the amount of trust VMs need to place in the hypervisor and administrator of their host system.

The `os_traits.hw.cpu.amd.SEV` trait is reserved in order to indicate that a compute host contains support for SEV not only on-CPU, but also in all other layers of the hypervisor stack required in order to take advantage of this feature.

4.2 Available Traits

Below is a list of all traits currently available.

- COMPUTE_DEVICE_TAGGING
- COMPUTE_NODE
- COMPUTE_TRUSTED_CERTS
- COMPUTE_SAME_HOST_COLD_MIGRATE
- COMPUTE_RESCUE_BFV
- COMPUTE_ACCELERATORS
- COMPUTE_SOCKET_PCI_NUMA_AFFINITY
- COMPUTE_REMOTE_MANAGED_PORTS
- COMPUTE_MEM_BACKING_FILE
- COMPUTE_MANAGED_PCI_DEVICE
- COMPUTE_ADDRESS_SPACE_PASSTHROUGH

- COMPUTE_ADDRESS_SPACE_EMULATED
- COMPUTE_SHARE_LOCAL_FS
- COMPUTE_ARCH_AARCH64
- COMPUTE_ARCH_PPC64LE
- COMPUTE_ARCH_MIPSEL
- COMPUTE_ARCH_S390X
- COMPUTE_ARCH_RISCV64
- COMPUTE_ARCH_X86_64
- COMPUTE_CONFIG_DRIVE_REGENERATION
- COMPUTE_EPHEMERAL_ENCRYPTION
- COMPUTE_EPHEMERAL_ENCRYPTION_PLAIN
- COMPUTE_EPHEMERAL_ENCRYPTION_LUKS
- COMPUTE_EPHEMERAL_ENCRYPTION_LUKSV2
- COMPUTE_FIRMWARE_BIOS
- COMPUTE_FIRMWARE_UEFI
- COMPUTE_GRAPHICS_MODEL_BOCHS
- COMPUTE_GRAPHICS_MODEL_CIRRUS
- COMPUTE_GRAPHICS_MODEL_GOP
- COMPUTE_GRAPHICS_MODEL_NONE
- COMPUTE_GRAPHICS_MODEL_QXL
- COMPUTE_GRAPHICS_MODEL_VGA
- COMPUTE_GRAPHICS_MODEL_VIRTIO
- COMPUTE_GRAPHICS_MODEL_VMVGA
- COMPUTE_GRAPHICS_MODEL_XEN
- COMPUTE_IMAGE_TYPE_AKI
- COMPUTE_IMAGE_TYPE_AMI
- COMPUTE_IMAGE_TYPE_ARI
- COMPUTE_IMAGE_TYPE_ISO
- COMPUTE_IMAGE_TYPE_QCOW2
- COMPUTE_IMAGE_TYPE_RAW
- COMPUTE_IMAGE_TYPE_VDI
- COMPUTE_IMAGE_TYPE_VHD
- COMPUTE_IMAGE_TYPE_VHDX
- COMPUTE_IMAGE_TYPE_VMDK

- COMPUTE_IMAGE_TYPE_PLOOP
- COMPUTE_MIGRATE_AUTO_CONVERGE
- COMPUTE_MIGRATE_POST_COPY
- COMPUTE_NET_ATTACH_INTERFACE
- COMPUTE_NET_ATTACH_INTERFACE_WITH_TAG
- COMPUTE_NET_VIRTIO_PACKED
- COMPUTE_NET_VIF_MODEL_E1000
- COMPUTE_NET_VIF_MODEL_E1000E
- COMPUTE_NET_VIF_MODEL_LAN9118
- COMPUTE_NET_VIF_MODEL_NETFRONT
- COMPUTE_NET_VIF_MODEL_NE2K_PCI
- COMPUTE_NET_VIF_MODEL_PCNET
- COMPUTE_NET_VIF_MODEL_RTL8139
- COMPUTE_NET_VIF_MODEL_SPAPR_VLAN
- COMPUTE_NET_VIF_MODEL_SRIOV
- COMPUTE_NET_VIF_MODEL_VIRTIO
- COMPUTE_NET_VIF_MODEL_VMXNET
- COMPUTE_NET_VIF_MODEL_VMXNET3
- COMPUTE_NET_VIF_MODEL_IGB
- COMPUTE_SECURITY_TPM_1_2
- COMPUTE_SECURITY_TPM_2_0
- COMPUTE_SECURITY_TPM_TIS
- COMPUTE_SECURITY_TPM_CRB
- COMPUTE_SECURITY_TPM_SECRET_SECURITY_USER
- COMPUTE_SECURITY_TPM_SECRET_SECURITY_HOST
- COMPUTE_SECURITY_TPM_SECRET_SECURITY_DEPLOYMENT
- COMPUTE_SECURITY_UEFI_SECURE_BOOT
- COMPUTE_SECURITY_STATELESS_FIRMWARE
- COMPUTE_STATUS_DISABLED
- COMPUTE_STORAGE_BUS_FDC
- COMPUTE_STORAGE_BUS_IDE
- COMPUTE_STORAGE_BUS_LXC
- COMPUTE_STORAGE_BUS_SATA
- COMPUTE_STORAGE_BUS_SCSI

- COMPUTE_STORAGE_BUS_USB
- COMPUTE_STORAGE_BUS_VIRTIO
- COMPUTE_STORAGE_BUS_UML
- COMPUTE_STORAGE_BUS_XEN
- COMPUTE_STORAGE_VIRTIO_FS
- COMPUTE_VIOMMU_MODEL_INTEL
- COMPUTE_VIOMMU_MODEL_SMMUV3
- COMPUTE_VIOMMU_MODEL_VIRTIO
- COMPUTE_VIOMMU_MODEL_AUTO
- COMPUTE_VOLUME_ATTACH
- COMPUTE_VOLUME_ATTACH_WITH_TAG
- COMPUTE_VOLUME_EXTEND
- COMPUTE_VOLUME_MULTI_ATTACH
- HW_ARCH_ALPHA
- HW_ARCH_ARMV6
- HW_ARCH_ARMV7
- HW_ARCH_ARMV7B
- HW_ARCH_AARCH64
- HW_ARCH_CRIS
- HW_ARCH_I686
- HW_ARCH_IA64
- HW_ARCH_LM32
- HW_ARCH_M68K
- HW_ARCH_MICROBLAZE
- HW_ARCH_MICROBLAZEEL
- HW_ARCH_MIPS
- HW_ARCH_MIPSEL
- HW_ARCH_MIPS64
- HW_ARCH_MIPS64EL
- HW_ARCH_OPENRISC
- HW_ARCH_PARISC
- HW_ARCH_PARISC64
- HW_ARCH_PPC
- HW_ARCH_PPCLE

- HW_ARCH_PPC64
- HW_ARCH_PPC64LE
- HW_ARCH_PPCEMB
- HW_ARCH_S390
- HW_ARCH_S390X
- HW_ARCH_SH4
- HW_ARCH_SH4EB
- HW_ARCH_SPARC
- HW_ARCH_SPARC64
- HW_ARCH_UNICORE32
- HW_ARCH_X86_64
- HW_ARCH_XTENSA
- HW_ARCH_XTENSAEB
- HW_CPU_HYPERTHREADING
- HW_CPU_AARCH64_FP
- HW_CPU_AARCH64_ASIMD
- HW_CPU_AARCH64_EVTSTRM
- HW_CPU_AARCH64_AES
- HW_CPU_AARCH64_PMULL
- HW_CPU_AARCH64_SHA1
- HW_CPU_AARCH64_SHA2
- HW_CPU_AARCH64_CRC32
- HW_CPU_AARCH64_FPHP
- HW_CPU_AARCH64_ASIMDHP
- HW_CPU_AARCH64_ASIMDRDM
- HW_CPU_AARCH64_ATOMICS
- HW_CPU_AARCH64_JSCVT
- HW_CPU_AARCH64_FCMA
- HW_CPU_AARCH64_LRCPC
- HW_CPU_AARCH64_DCPOP
- HW_CPU_AARCH64_SHA3
- HW_CPU_AARCH64_SM3
- HW_CPU_AARCH64_SM4
- HW_CPU_AARCH64_ASIMDDP

- HW_CPU_AARCH64_SHA512
- HW_CPU_AARCH64_SVE
- HW_CPU_AARCH64_CPUID
- HW_CPU_AMD_SEV
- HW_CPU_PPC64LE_POWER8
- HW_CPU_PPC64LE_POWER9
- HW_CPU_X86_AVX
- HW_CPU_X86_AVX2
- HW_CPU_X86_CLMUL
- HW_CPU_X86_FMA3
- HW_CPU_X86_FMA4
- HW_CPU_X86_F16C
- HW_CPU_X86_MMX
- HW_CPU_X86_SSE
- HW_CPU_X86_SSE2
- HW_CPU_X86_SSE3
- HW_CPU_X86_SSSE3
- HW_CPU_X86_SSE41
- HW_CPU_X86_SSE42
- HW_CPU_X86_SSE4A
- HW_CPU_X86_XOP
- HW_CPU_X86_3DNOW
- HW_CPU_X86_AVX512F
- HW_CPU_X86_AVX512CD
- HW_CPU_X86_AVX512PF
- HW_CPU_X86_AVX512ER
- HW_CPU_X86_AVX512VL
- HW_CPU_X86_AVX512BW
- HW_CPU_X86_AVX512DQ
- HW_CPU_X86_AVX512VNNI
- HW_CPU_X86_AVX512VBMI
- HW_CPU_X86_AVX512IFMA
- HW_CPU_X86_AVX512VBMI2
- HW_CPU_X86_AVX512BITALG

- HW_CPU_X86_AVX512VAES
- HW_CPU_X86_AVX512GFNI
- HW_CPU_X86_AVX512VPCLMULQDQ
- HW_CPU_X86_AVX512VPOPCNTDQ
- HW_CPU_X86_ABM
- HW_CPU_X86_BMI
- HW_CPU_X86_BMI2
- HW_CPU_X86_TBM
- HW_CPU_X86_AESNI
- HW_CPU_X86_SHA
- HW_CPU_X86_MPX
- HW_CPU_X86_SGX
- HW_CPU_X86_TSX
- HW_CPU_X86_ASF
- HW_CPU_X86_VMX
- HW_CPU_X86_SVM
- HW_CPU_X86_PDPE1GB
- HW_CPU_X86_STIBP
- HW_CPU_X86_AMD_SEV
- HW_CPU_X86_AMD_SEV_ES
- HW_CPU_X86_AMD_SVM
- HW_CPU_X86_AMD_IBPB
- HW_CPU_X86_AMD_NO_SSB
- HW_CPU_X86_AMD_SSBD
- HW_CPU_X86_AMD_VIRT_SSBD
- HW_CPU_X86_INTEL_MD_CLEAR
- HW_CPU_X86_INTEL_PCID
- HW_CPU_X86_INTEL_SPEC_CTRL
- HW_CPU_X86_INTEL_SSBD
- HW_CPU_X86_INTEL_VMX
- HW_GPU_API_DIRECTX_V10
- HW_GPU_API_DIRECTX_V11
- HW_GPU_API_DIRECTX_V12
- HW_GPU_API_DIRECT2D

- HW_GPU_API_DIRECT3D_V6_0
- HW_GPU_API_DIRECT3D_V7_0
- HW_GPU_API_DIRECT3D_V8_0
- HW_GPU_API_DIRECT3D_V8_1
- HW_GPU_API_DIRECT3D_V9_0
- HW_GPU_API_DIRECT3D_V9_0B
- HW_GPU_API_DIRECT3D_V9_0C
- HW_GPU_API_DIRECT3D_V9_0L
- HW_GPU_API_DIRECT3D_V10_0
- HW_GPU_API_DIRECT3D_V10_1
- HW_GPU_API_DIRECT3D_V11_0
- HW_GPU_API_DIRECT3D_V11_1
- HW_GPU_API_DIRECT3D_V11_2
- HW_GPU_API_DIRECT3D_V11_3
- HW_GPU_API_DIRECT3D_V12_0
- HW_GPU_API_VULKAN
- HW_GPU_API_DXVA
- HW_GPU_API_OPENCL_V1_0
- HW_GPU_API_OPENCL_V1_1
- HW_GPU_API_OPENCL_V1_2
- HW_GPU_API_OPENCL_V2_0
- HW_GPU_API_OPENCL_V2_1
- HW_GPU_API_OPENCL_V2_2
- HW_GPU_API_OPENGL_V1_1
- HW_GPU_API_OPENGL_V1_2
- HW_GPU_API_OPENGL_V1_3
- HW_GPU_API_OPENGL_V1_4
- HW_GPU_API_OPENGL_V1_5
- HW_GPU_API_OPENGL_V2_0
- HW_GPU_API_OPENGL_V2_1
- HW_GPU_API_OPENGL_V3_0
- HW_GPU_API_OPENGL_V3_1
- HW_GPU_API_OPENGL_V3_2
- HW_GPU_API_OPENGL_V3_3

- HW_GPU_API_OPENGL_V4_0
- HW_GPU_API_OPENGL_V4_1
- HW_GPU_API_OPENGL_V4_2
- HW_GPU_API_OPENGL_V4_3
- HW_GPU_API_OPENGL_V4_4
- HW_GPU_API_OPENGL_V4_5
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V1_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V1_1
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V1_2
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V1_3
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V2_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V2_1
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_2
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_5
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V3_7
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_2
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V5_3
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_1
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V6_2
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V7_0
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V7_1
- HW_GPU_CUDA_COMPUTE_CAPABILITY_V7_2
- HW_GPU_CUDA_SDK_V6_5
- HW_GPU_CUDA_SDK_V7_5
- HW_GPU_CUDA_SDK_V8_0
- HW_GPU_CUDA_SDK_V9_0
- HW_GPU_CUDA_SDK_V9_1
- HW_GPU_CUDA_SDK_V9_2
- HW_GPU_CUDA_SDK_V10_0
- HW_GPU_MAX_DISPLAY_HEADS_1
- HW_GPU_MAX_DISPLAY_HEADS_2

- HW_GPU_MAX_DISPLAY_HEADS_4
- HW_GPU_MAX_DISPLAY_HEADS_6
- HW_GPU_MAX_DISPLAY_HEADS_8
- HW_GPU_RESOLUTION_W320H240
- HW_GPU_RESOLUTION_W640H480
- HW_GPU_RESOLUTION_W800H600
- HW_GPU_RESOLUTION_W1024H600
- HW_GPU_RESOLUTION_W1024H768
- HW_GPU_RESOLUTION_W1152H864
- HW_GPU_RESOLUTION_W1280H720
- HW_GPU_RESOLUTION_W1280H768
- HW_GPU_RESOLUTION_W1280H800
- HW_GPU_RESOLUTION_W1280H1024
- HW_GPU_RESOLUTION_W1360H768
- HW_GPU_RESOLUTION_W1366H768
- HW_GPU_RESOLUTION_W1440H900
- HW_GPU_RESOLUTION_W1600H900
- HW_GPU_RESOLUTION_W1600H1200
- HW_GPU_RESOLUTION_W1680H1050
- HW_GPU_RESOLUTION_W1920H1080
- HW_GPU_RESOLUTION_W1920H1200
- HW_GPU_RESOLUTION_W2560H1440
- HW_GPU_RESOLUTION_W2560H1600
- HW_GPU_RESOLUTION_W3840H2160
- HW_GPU_RESOLUTION_W7680H4320
- HW_NIC_SRIOV
- HW_NIC_MULTIQUEUE
- HW_NIC_VMDQ
- HW_NIC_PROGRAMMABLE_PIPELINE
- HW_NIC_ACCEL_SSL
- HW_NIC_ACCEL_IPSEC
- HW_NIC_ACCEL_TLS
- HW_NIC_ACCEL_DIFFIEH
- HW_NIC_ACCEL_RSA

- HW_NIC_ACCEL_ECC
- HW_NIC_ACCEL_LZS
- HW_NIC_ACCEL_DEFLATE
- HW_NIC_DCB_PFC
- HW_NIC_DCB_ETS
- HW_NIC_DCB_QCN
- HW_NIC_OFFLOAD_TSO
- HW_NIC_OFFLOAD_GRO
- HW_NIC_OFFLOAD_GSO
- HW_NIC_OFFLOAD_UFO
- HW_NIC_OFFLOAD_LRO
- HW_NIC_OFFLOAD_LSO
- HW_NIC_OFFLOAD_TCS
- HW_NIC_OFFLOAD_UCS
- HW_NIC_OFFLOAD_SCS
- HW_NIC_OFFLOAD_L2CRC
- HW_NIC_OFFLOAD_FDF
- HW_NIC_OFFLOAD_RXVLAN
- HW_NIC_OFFLOAD_TXVLAN
- HW_NIC_OFFLOAD_VXLAN
- HW_NIC_OFFLOAD_GRE
- HW_NIC_OFFLOAD_GENEVE
- HW_NIC_OFFLOAD_TXUDP
- HW_NIC_OFFLOAD_QINQ
- HW_NIC_OFFLOAD_RDMA
- HW_NIC_OFFLOAD_RXHASH
- HW_NIC_OFFLOAD_RX
- HW_NIC_OFFLOAD_TX
- HW_NIC_OFFLOAD_SG
- HW_NIC_OFFLOAD_SWITCHDEV
- HW_NIC_SRIOV_QOS_TX
- HW_NIC_SRIOV_QOS_RX
- HW_NIC_SRIOV_MULTIQUEUE
- HW_NIC_SRIOV_TRUSTED

- HW_NUMA_ROOT
- HW_PCI_LIVE_MIGRATABLE
- MISC_SHARES_VIA_AGGREGATE
- OWNER_CYBORG
- OWNER_NOVA
- STORAGE_DISK_HDD
- STORAGE_DISK_SSD