Enabling AMD Secure Nested Paging (SEV-SNP) on ThinkSystem Servers

Planning / Implementation

Trusted execution environments have become increasingly common for the execution of security critical code. In their processors, AMD first introduced Secure Encrypted Virtualization (SEV) in 2016, and then introduced Encrypted State (SEV-ES) to encrypt CPU register state of virtual machines (VM) in 2017. The third generation of SEV, Secure Nested Paging (SEV-SNP), enhances memory integrity protection for the malicious attacks from hypervisor.

SEV-SNP is supported on AMD EYPC processors starting with the AMD EPYC 7003 series processors.

AMD SEV-SNP offers powerful and flexible support for the isolation of a guest virtual machine from an untrusted host operating system. It is very useful in public cloud and any untrusted host scenario. Major public cloud vendors already used it in their products, including Amazon Web Services (AWS) and Google Cloud.

This paper describes how to enable SEV-SNP on an AMD-based ThinkSystem server running Red Hat Enterprise Linux (RHEL) 9.2.

SEV-SNP overview

This section will show how to protect the guest VM via SEV-SNP function and what threats can be prevented.

In Figure 1, “AMD Hardware and Firmware” and “SEV-SNP VM” are considered trusted in the measurement process, even though the hypervisor is untrusted.

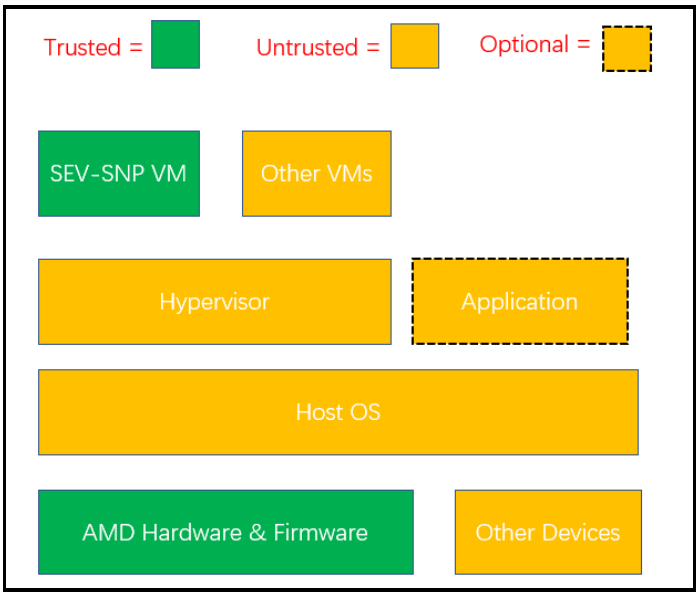

Figure 1. Threat model

Under the attestation process of SEV-SNP, only a guest owner (third-party) can decide whether the guest is trusted or not based on attestation reports.

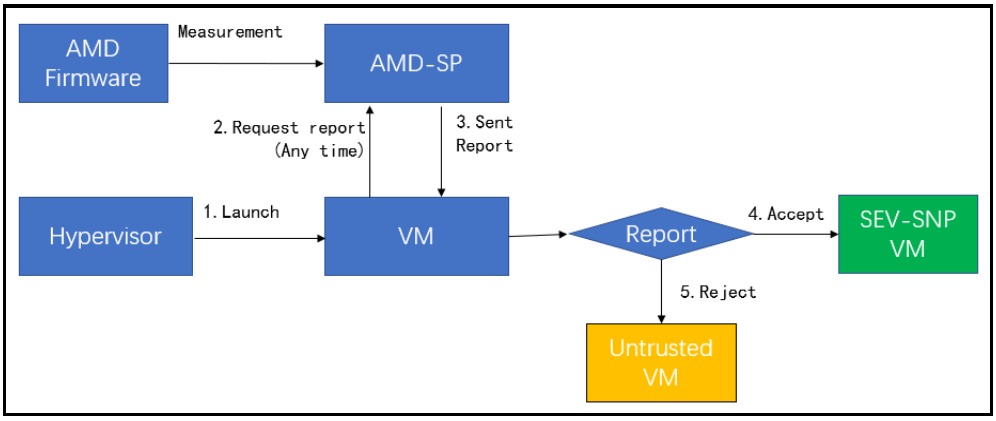

Figure 2. Attestation Process

Table 1 lists potential threads mitigated by SEV-SNP.

Table 1. Threat mitigation

Potential Threats		**Mitigated**
Confidentiality	VM Memory Example attack: Hypervisor reads private VM memory	Mitigated
	VM Register State Example attack: Read VM register state after VMEXIT	Mitigated
	DMA Protection Example attack: Device attempts to read VM memory	Mitigated
Integrity	Replay Protection Example attack: Replace VM memory with an old copy	Mitigated
	Data Corruption Example attack: Replace VM memory with junk data	Mitigated
	Memory Aliasing Example attack: Map two guest pages to same DRAM page	Mitigated
	Memory Re-Mapping Example attack: Switch DRAM page mapped to a guest page	Mitigated
Availability	Denial of Service on Hypervisor Example attack: Malicious guest refuses to yield/exit	Mitigated
Physical Access Attacks	Offline DRAM analysis Example attack: Cold boot	Mitigated
Misc.	TCB Rollback Example attack: Revert AMD-SP firmware to old version	Mitigated
	Malicious Interrupt/Exception Injection Example attack: Inject interrupt while RFLAGS.IF=0	Optional mitigated
	Indirect Branch Predictor Poisoning Example attack: Poison BTB from hypervisor	Optional mitigated
	Secure Hardware Debug Registers Example attack: Change breakpoints during debug	Optional mitigated
	Trusted CPUID Information Example attack: Hypervisors lies about platform capabilities	Optional mitigated

Preparing UEFI and the Host OS

In this section:

UEFI configuration via System Setup

The steps to activate SEV-SNP in UEFI are as follows:

1. Press F1 during boot to enter System Setup
2. In the Processors section, enable these items as shown in the figure below.
    - SVM Mode: Enable
    - SEV-SNP Support : Enable

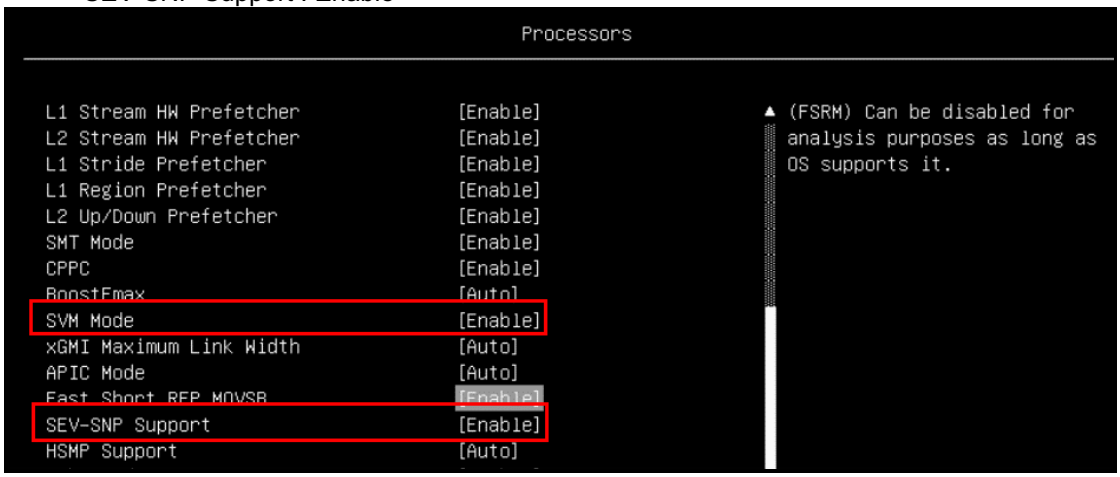

Figure 3. Processor settings in System Setup

3. In the Memory section, enable these items as shown in the figure below.
    - SMEE: Enable
    - SEV-ES ASID Count: AUTO
    - SEV-ES ASID Space Limit Control: Manual
    - SEV-ES ASID Space Limit: 10
    - SEV Control: Enable

Figure 4. Memory settings in System Setup

UEFI configuration via OneCLI

As an alternative to System Setup, you can use the OneCLI command line tool, which can be downloaded from:
https://support.lenovo.com/us/en/solutions/HT116433

1. Create a configuration file, as follows:

```
[root@sev-snp ~]# cat > snp_uefi.txt << EOF
set Processors.SEV-SNPSupport enable
set Memory.SMEE Enable
set Memory.SEVASIDCount AUTO
set Memory.SEV-ESASIDSpaceLimitControl Manual
set Memory.SEV-ESASIDSpaceLimit 10
set Memory.SEVControl Enable set Processors.SVMMode Enable
EOF
[root@sev-snp ~]#
```

2. Set up UEFI config via Onecli command:

```
[root@sev-snp ~]# chmod a+x lnvgy_utl_lxceb_onecli01l-4.3.0_linux_x86-64.bin
[root@sev-snp ~]#
[root@sev-snp ~]# ./lnvgy_utl_lxceb_onecli01l-4.3.0_linux_x86-64.bin config batch --file snp_uefi.txt -N -b XCC_USER:XCC_PASSWORD@XCC_IP
Extracting...
Executing...

Start to connect BMC at [redacted] to apply config batch
Connected to BMC at IP address [redacted] by REDFISH
Invoking BATCH command ...

Processing....

All commands executed successfully !
Changes completed successfully, but these changes will not take effect until next restart the system.
Succeed.
[root@sev-snp ~]#
```

Figure 5. Issuing the OneCLI command to run the configuration file

3. Restart the server to apply the configuration.

Operating System configuration

As RHEL 9.2 inbox kernel and QEMU hypervisor still do not fully support this feature, users need to compile it by themselves. Ensure your system has access to the Internet and source code will be downloaded automatically during compiling.

1. Register your system and enable repository “codeready-builder-for-rhel-9-x86_64-rpms" using the following commands:

```
[root@sev-snp ~]# subscription-manager register --username XXX --password
XXX
This system is already registered. Use --force to override
[root@sev-snp ~]#
[root@sev-snp ~]# subscription-manager repos --enable codeready-builder-fo
r-rhel-9-x86_64-rpms
```

2. Install the necessary packages for compiling:

```
[root@sev-snp ~]# yum install -y ninja-build.x86_64 gthread libgib* glib-d
evel.x86_64 \
> PackageKit-glib.x86_64 PackageKit-glib-devel.x86_64 pixman pixman-devel.
x86_64 \
> nasm.x86_64 uuid-devel.x86_64 glibc-static acpica-tools  perl dwarves pk
gconfig
[root@sev-snp ~]# pip install meson; ln -s /usr/lib64/libuuid.so.1.3.0 /us
r/lib64/libuuid.so; ldconfig
```

3. Build Linux kernel, QEMU and other components with the following command

```
# git clone https://github.com/AMDESE/AMDSEV.git
# cd AMDSEV; git checkout snp-latest
# ./build.sh –package
```

Enabling SEV-SNP on the Host OS

Follow these steps enable and verify SEV-SNP on a host OS:

1. Install the compiled kernel

```
# cd snp-release-<DATE>
# sudo cp kvm.conf /etc/modprobe.d/
# rpm -ivh $(find . -name "kernel*host*" | grep -v headers)
```

2. Modify the SNP kernel to the default boot entry

```
# grubby --default-kernel                 # Get current default boot entr
y
# grubby --info ALL                       # Get all the boot entry
# grubby --set-default-index=ENTRY-INDEX  # Set the SNP kernel entry index
to the default
```

3. Reboot the server

```
# reboot
```

4. Verify the feature was enabled from driver layer:

```
[root@sev-snp ~]# cat /sys/module/kvm_amd/parameters/sev_snp
Y
[root@sev-snp ~]#
```

5. Verify the dmesg log shows the SEV-SNP support information:

```
[root@sev-snp ~]# dmesg I grep SEV-SNP
[    0.569182] SEV-SNP: RMP table physical address [0x000000009b700000 - 0
x00000000a3cfffff]
[    3.905529] ccp 0000:23:00.1: SEV-SNP API:1.55 build : 14
[   15.047076] kvm_amd: SEV-ES and SEV-SNP supported: 9 ASIDs
```

Enabling SEV-SNP on a Guest OS

Follow these steps to enable and verify SEV-SNP on guest OS.

1. Create SEV-SNP VM with the following commands

```
# qemu-img create -f qcow2 /home/rh9.qcow2 40G    #Create your qcow2 file
for guest storage
# cd AMDSEV/snp-release-
# sed -i "s/CONSOLE=.*$/CONSOLE=\"virtio\"/" launch-qemu.sh
# sed -i "s/readonly/readonly=on/" launch-qemu.sh
# ./launch-qemu.sh -hda /home/rh9.qcow2 -cdrom home/RHEL-9.2.0-20230414.17
-x86_64-dvd1.iso
```

```
Launching VM ...
  /tmp/cmdline.73089
char device redirected to /dev/pts/1 (label compat_monitor0)
qemu-system-x86_64: warning: Number of hotpluggable cpus requested (255) exceeds the recommended cpus supported by KVM (64)
VNC server running on ::1:5900
```

Figure 6.

2. Finish the installation via VNC viewer based on the output about VNC server address.

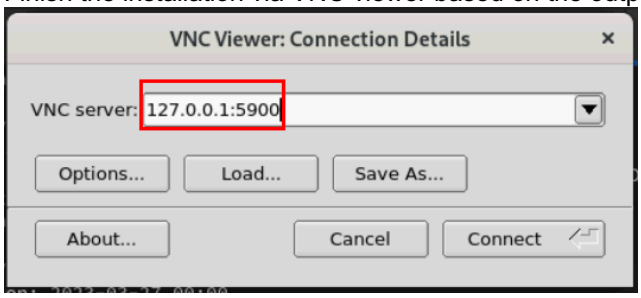

Figure 7. Launch VNC viewer

3. Launch the guest OS

```
# ./launch-qemu.sh -hda  /home/rh9.qcow2 -sev-snp   #Launch the guest
```

4. Access the guest via VNC viewer based on the output about VNC server address.

```
qemu-system-x86_64: warning: kvm_create_gmemfd: created memfd: 30, size: 20000, flags: 0
VNC server running on ::1:5900
```

Figure 8.

5. If SEV-SNP is enabled properly in a VM, the log “Memory Encryption Features active:” must include the string “SEV-SNP” in OS log (dmesg):

```
[root@snp-guest ~]# dmesg | grep -i SEV-SNP
[    0.2712931 Memory Encryption Features active: AMD SEU SEV-ES SEV-SNP
[root@snp-guest ~]#
```

For more information

For more information, see these resources:

- AMD SEV-SNP:Strengthening VM Isolation
  https://www.amd.com/content/dam/amd/en/documents/epyc-business-docs/white-papers/SEV-SNP-strengthening-vm-isolation-with-integrity-protection-and-more.pdf
- The github of SEV-SNP
  https://github.com/AMDESE/AMDSEV/tree/snp-latest
- Introduction to confidential virtual machines
  https://www.redhat.com/en/blog/introduction-confidential-virtual-machines
- AMD SEV-SNP Attestation: Establishing Trust in Guests
  https://www.amd.com/content/dam/amd/en/documents/developer/lss-snp-attestation.pdf

Author

Song Shang is a Linux Engineer in Lenovo Infrastructure Solutions Group, based in Beijing, China.

Thanks to the following people for their contributions to this project:

- David Watts, Lenovo Press
- Adrian Huang, Lenovo Linux Engineer
- Gary Cudak, Lenovo Lead Architect

Related product families

Product families related to this document are the following:

- Processors

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service. Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
8001 Development Drive
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary. Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk. Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This document, LP1893, was created or updated on February 20, 2024.

Send us your comments in one of the following ways:

- Use the online Contact us review form found at:
  https://lenovopress.lenovo.com/LP1893
- Send your comments in an e-mail to:
  comments@lenovopress.com

This document is available online at https://lenovopress.lenovo.com/LP1893.

Trademarks

Lenovo and the Lenovo logo are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. A current list of Lenovo trademarks is available on the Web at https://www.lenovo.com/us/en/legal/copytrade/.

The following terms are trademarks of Lenovo in the United States, other countries, or both:
Lenovo®
ThinkSystem®

The following terms are trademarks of other companies:

AMD and AMD EPYC™ are trademarks of Advanced Micro Devices, Inc.

Linux® is the trademark of Linus Torvalds in the U.S. and other countries.

Other company, product, or service names may be trademarks or service marks of others.