Managing Secure Boot Keys in VMware ESXi 9.1

Planning / Implementation

Secure Boot is a UEFI firmware security feature developed by the UEFI Consortium designed to ensure that only immutable and digitally signed software is loaded during the boot process. By leveraging digital signatures, Secure Boot validates the authenticity and origin of the loaded code, thereby guaranteeing its integrity. The VMware ESXi bootloader comprises several UEFI applications and drivers; when Secure Boot is enabled within the firmware, the system will reject any UEFI application or driver that is not cryptographically signed by a trusted key. VMware ESXi supports Secure Boot by ensuring all bootloader components are properly signed and by initiating a verification chain to confirm that the remainder of the ESXi system is signed with a VMware key. These validation measures are implemented to preclude the execution of malicious code and to mitigate threats such as rootkits and bootkits.

Secure Boot for ESXi hosts

Figure 1 illustrates the structure of signatures and keys utilized by Secure Boot. The platform is secured via a Platform Key (PK), which is embedded in the firmware by the Original Equipment Manufacturer (OEM) during the manufacturing process. Key Exchange Keys (KEK) are employed by Secure Boot to regulate access to the databases that manage the execution permissions of firmware. The Authorized Signature Database (db) contains the public keys and certificates of trusted firmware components and operating system loaders. Conversely, the Forbidden Signature Database (dbx) stores hashes of malicious or vulnerable components, as well as compromised keys and certificates, to effectively block the execution of unauthorized code.

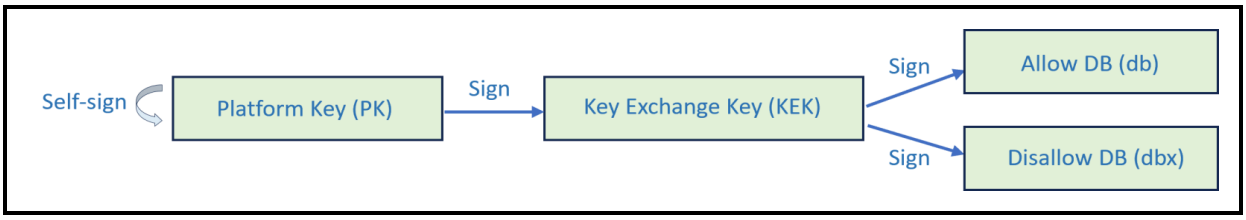

Figure 1. Secure Boot Key Structure

Figure 2 illustrates the VMware ESXi secure boot sequence, which indicates the following steps:

1. One or more public key certificates enrolled in the UEFI firmware's "db" variable serve as the initial root of trust.
2. The UEFI firmware utilizes these keys to validate the cryptographic signature of the ESXi bootloader
3. The ESXi bootloader, which incorporates a VMware public key, employs this key to verify the signatures of the kernel and a critical subset of the system, including the Secure Boot VIB verifier.
4. The VIB verifier subsequently validates every VIB package installed on the system to ensure integrity.

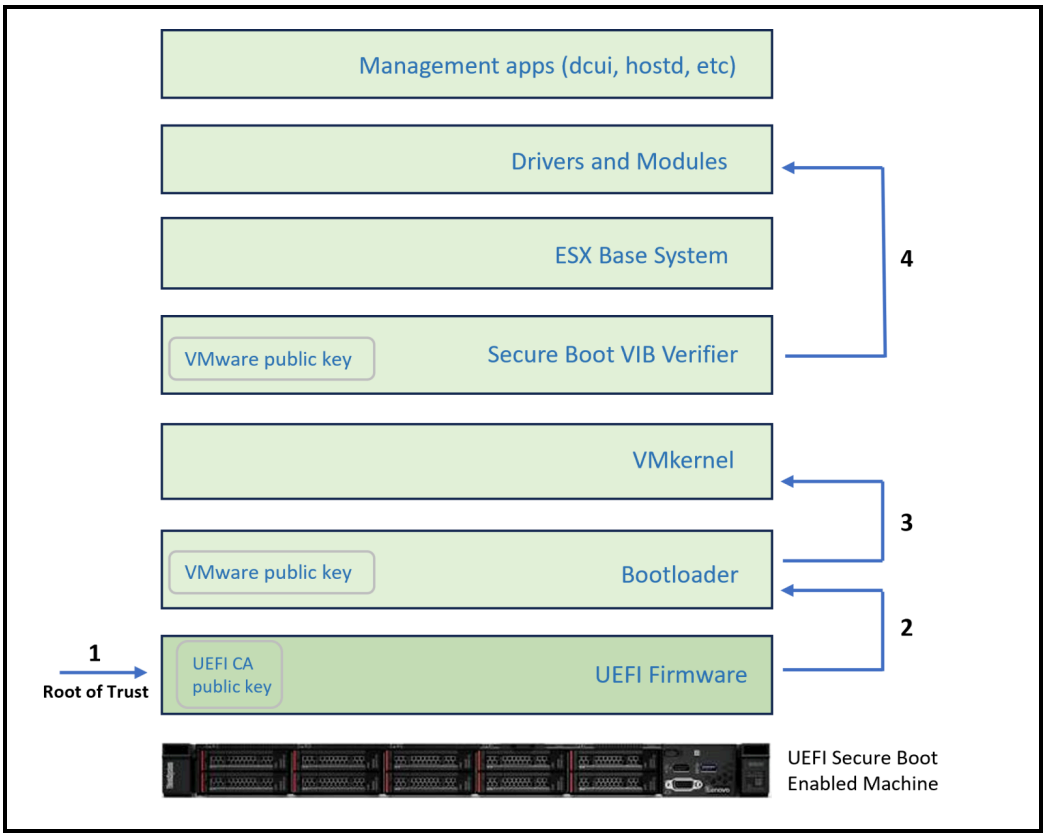

Figure 2. Secure Boot sequence on the ESXi host

Within the ESXi 9.1 environment, the parseSB utility can be utilized to inspect the Secure Boot keys enrolled in the host firmware, as illustrated in Figures 3 and 4.

To display the KEK keys, execute the following command:

```
# /usr/lib/vmware/uefi/bin/parseSB -u KEK | less
```

```
[root@localhost:~] /usr/lib/vmware/uefi/bin/parseSB -u KEK | less
SignatureOwner: GUID 77fa9abd-0359-4d32-bd60-28f4e78f784b Microsoft Corporation
1462 bytes of data
SignatureType: x509 certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            33:00:00:00:13:14:16:b8:61:6d:82:82:4b:00:00:00:00:00:13
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = US, O = Microsoft Corporation, CN = Microsoft RSA Devices Root CA 2021
        Validity
            Not Before: Mar  2 20:21:35 2023 GMT
            Not After : Mar  2 20:31:35 2038 GMT
        Subject: C = US, O = Microsoft Corporation, CN = Microsoft Corporation KEK 2K CA 2023
        X509v3 extensions:
            X509v3 Key Usage: critical
                Digital Signature, Certificate Sign, CRL Sign
            1.3.6.1.4.1.311.21.1:
                ...
            X509v3 Subject Key Identifier:
                E0:AB:72:BC:96:3E:FF:B8:66:9B:7D:10:5A:43:3E:5C:42:54:87:5F
            1.3.6.1.4.1.311.20.2:
                .
.S.u.b.C.A
            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Authority Key Identifier:
                84:44:86:06:00:98:3F:2C:AA:B3:C5:89:F3:AC:2E:C9:E6:9D:09:03
            X509v3 CRL Distribution Points:
                Full Name:
                  URI:http://www.microsoft.com/pkiops/crl/Microsoft%20RSA%20Devices%20Root%20CA%202021.crl
```

Figure 3. Display KEK certificate keys

To display the db keys, execute the following command:

```
# /usr/lib/vmware/uefi/bin/parseSB -u db | less
```

```
[root@localhost:~] /usr/lib/vmware/uefi/bin/parseSB -u db | less
SignatureOwner: GUID a3d5e95b-0a8f-4753-8735-445afb708f62 VMware by Broadcom
975 bytes of data
SignatureType: x509 certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            e9:d9:ff:69:e6:f3:e7:e4
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = US, ST = California, L = Palo Alto, O = "VMware, Inc."
        Validity
            Not Before: Oct 16 17:16:05 2008 GMT
            Not After : Dec 31 17:16:05 2019 GMT
        Subject: C = US, ST = California, L = Palo Alto, O = "VMware, Inc."
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                4A:D8:BA:04:72:07:3D:28:12:77:06:DD:C6:CC:B9:05:04:41:BB:C7
            X509v3 Authority Key Identifier:
                keyid:4A:D8:BA:04:72:07:3D:28:12:77:06:DD:C6:CC:B9:05:04:41:BB:C7
                DirName:/C=US/ST=California/L=Palo Alto/O=VMware, Inc.
                serial:E9:D9:FF:69:E6:F3:E7:E4
            X509v3 Basic Constraints:
                CA:TRUE



SignatureOwner: GUID a3d5e95b-0a8f-4753-8735-445afb708f62 VMware by Broadcom
955 bytes of data
SignatureType: x509 certificate
Certificate:
```

Figure 4. Display db certificate keys

The Microsoft Corporation KEK CA 2011 and Microsoft Corporation UEFI CA 2011 certificates utilized

within the DB expired in June 2026. Additionally, the Microsoft Windows Production PCA 2011 certificate is set to expire in October 2026. While the expiration of these 2011 certificates precludes Microsoft from generating new signatures, it does not invalidate existing signatures created prior to the expiration date. To address this, Microsoft has issued the Microsoft UEFI CA 2023 certificates (specifically, the Microsoft UEFI CA 2023 and Microsoft Option ROM UEFI CA 2023) to replace the expiring 2011 UEFI CA certificate.

Lenovo systems proactively incorporate updated UEFI firmware that includes the Microsoft Secure Boot CA 2023 certificates, enabling a seamless transition for customers without the need to disable Secure Boot or perform manual key enrollment.For detailed information regarding the minimum Lenovo UEFI build that supports the Microsoft Secure Boot CA 2023 certificates for each server model, refer to the following Lenovo Press paper, Secure Boot CA 2023 Guide for Azure Local and Windows Server.

VMware has published KB article 434297, which outlines the management of this transition at the physical server layer, and KB article 423893, which provides guidance for the virtual machine layer.

Secure Boot for ESXi virtual machines

With VMware ESXi VMs, Secure Boot is implemented via the virtual UEFI (vUEFI) firmware, which validates each boot component using Secure Boot databases stored in virtual NVRAM. During the boot sequence, the firmware verifies the bootloader (e.g., Windows Boot Manager), which subsequently validates all following components to prevent the execution of untrusted code. Secure Boot can be enabled for a VM only when all prerequisites are satisfied, including the use of EFI firmware, virtual hardware version 14 or later, and a guest operating system that supports UEFI Secure Boot. If these requirements are not met, the Secure Boot option will not be visible within the ESXi Client. Secure Boot certificates are initialized by the host during the initial power-on sequence and remain static throughout the VM's lifecycle unless explicitly updated.

Table 1 details the default certificates present in ESXi 9.1 at the time of VM creation, along with their respective expiration dates.

Table 1. Default certificates on ESXi 9.1 VM

vUEFI database	Certificate	Expiration date
PK	Windows OEM Devices PK	Sep 18 2038
KEK	Microsoft Corporation KEK 2K CA 2023	Mar 2 2038
	Microsoft Corporation KEK CA 2021	Jun 24 2026
DB	VMware Certificate	Dec 31 2019
	VMware Secure Boot Signing	Oct 19 2037
	Microsoft Corporation UEFI CA 2011	Jun 27 2026
	Microsoft Windows Production PCA 2011	Oct 19 2026
	Windows UEFI CA 2023	Jun 13 2035
	Microsoft UEFI CA 2023	Jun 13 2038

Secure Boot key management for virtual machines

In this section, we outline the procedures for configuring and updating Secure Boot certificate keys for virtual machines on ESXi 9.1, utilizing either the vUEFI interface or VMX configuration.

The test configuration for the ThinkSystem SR665 V3 is summarized in Table 2.

Table 2. ThinkSystem SR665 V3 Server Configuration

Component	Configuration
Server	ThinkSystem SR665 V3 Server
CPU	2x AMD EPYC 9634 84-Core Processor
Memory	2x DDR5 4800MHz 64GB DIMMs
HDD	1.0 TB SATA HDD
Host OS	ESXi 9.1 Custom Image for Lenovo ThinkSystem
Guest VM	RHEL9 U8 with vTPM disabled

It is strongly recommended to follow the OS vendor's official guidance when updating secure boot keys. While VMware's update methods—via the vUEFI interface or VMX configuration—are available as fallback options, updating these keys from outside the guest OS may compromise the functionality of security applications (such as Windows BitLocker or Linux LUKS) on vTPM-enabled VMs. This is because such updates modify Secure Boot variables and vTPM measurements without the OS's awareness, potentially triggering security failures.

The certificate keys employed for this demonstration are listed in Table 3.

Table 3. Certificates keys

Key Name	Variable	File Source
Windows OEM devices pk	PK	https://go.microsoft.com/fwlink/?linkid=2255361
Microsoft Corporation KEK 2K CA 2023	KEK	https://go.microsoft.com/fwlink/?linkid=2239775
Windows UEFI CA 2023	db	https://go.microsoft.com/fwlink/?linkid=2239776
Microsoft UEFI CA 2023	db	https://go.microsoft.com/fwlink/?linkid=2239872
kernel-signing-ca.cer secureboot-uki-x86_64.cer	db	From OS RHEL 9.8 under directory /usr/share/doc/kernel-keys/5.14.0-687.5.3.el9_8.x86_64/

Management via the vUEFI interface

The following steps describe the procedure for managing Secure Boot keys via the vUEFI interface on an existing RHEL 9.8 VM:

1. Download the certificate keys specified in Table 3 and copy the files to a FAT32-formatted USB storage device.
2. Attach the USB storage device to the physical host and configure the USB passthrough settings to map the device to the virtual machine.
3. Log in to the VMware Host Client, power off the VM, and navigate to Edit Settings. Under the Advanced Parameters tab, add a new configuration parameter by entering uefi.allowAuthBypass as the attribute and TRUE as the value. Click the Add button to save the parameter, as illustrated in Figure 5.

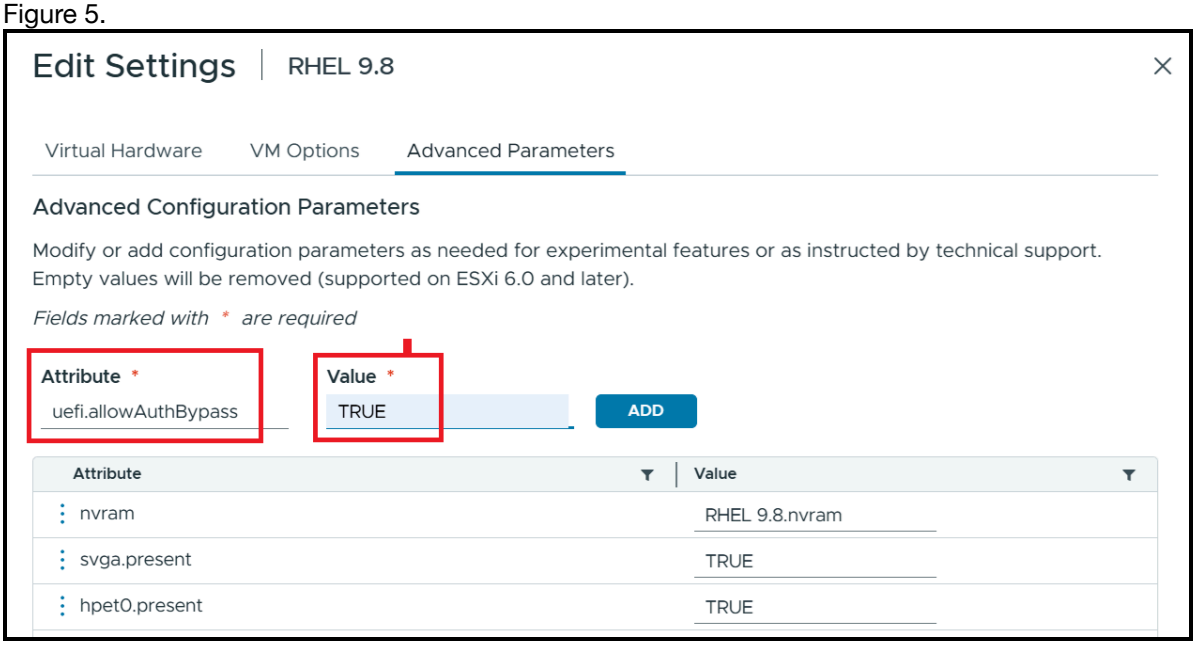

Figure 5. Add “uefi.allowAuthBypass” parameter

4. Navigate to the VM Options tab and select Boot Options. Enable the "Force EFI setup" option, as illustrated in Figure 6.

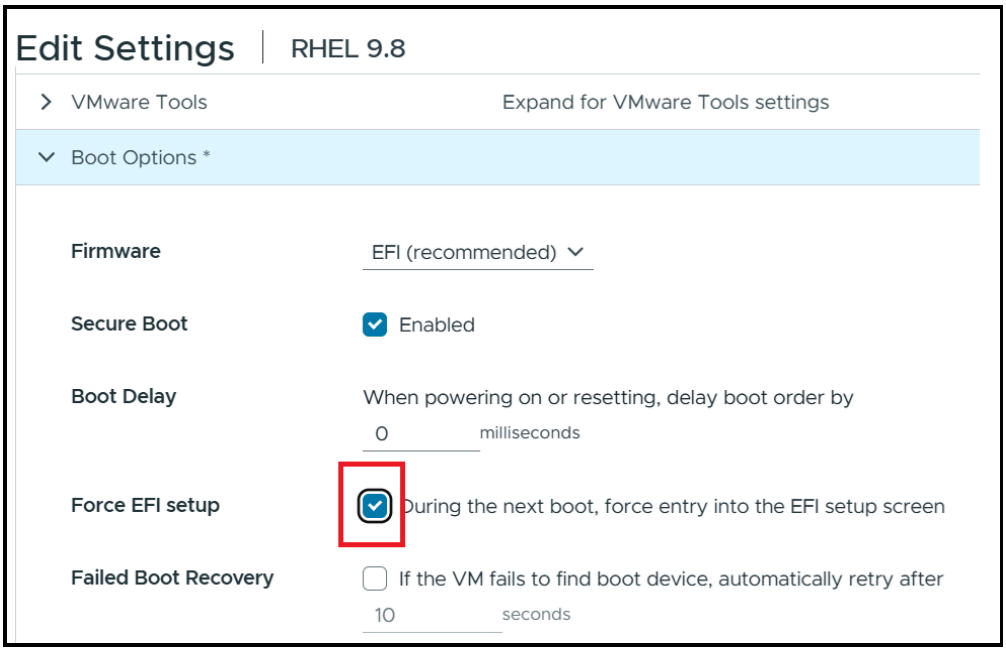

Figure 6. Force EFI setup option

5. Click OK to save the changes and power on the virtual machine to enter the vUEFI interface.
6. Within the vUEFI setup menu, navigate to Enter setup -> Secure Boot Configuration. This menu provides four distinct options for Secure Boot key management, as illustrated in Figure 7.

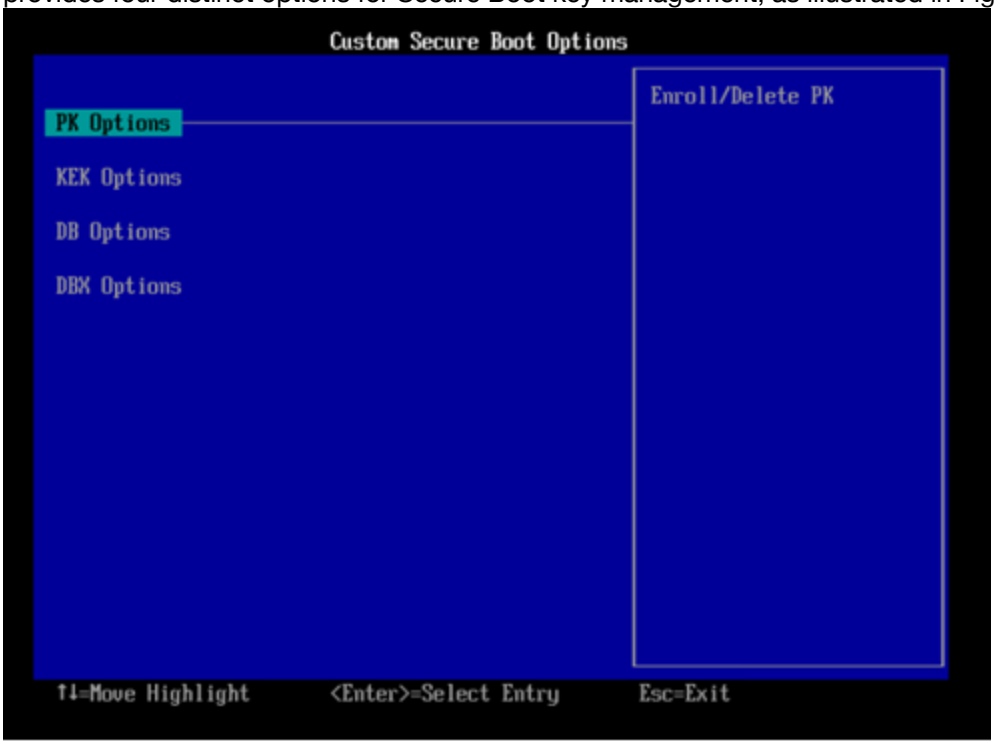

Figure 7. Secure Boot options

7. To enroll a new Key Exchange Key (KEK), navigate to KEK Options -> Enroll KEK $-> Enroll KEK Using File. Select the appropriate certificate file from the attached USB storage, as illustrated in

Figure 8.

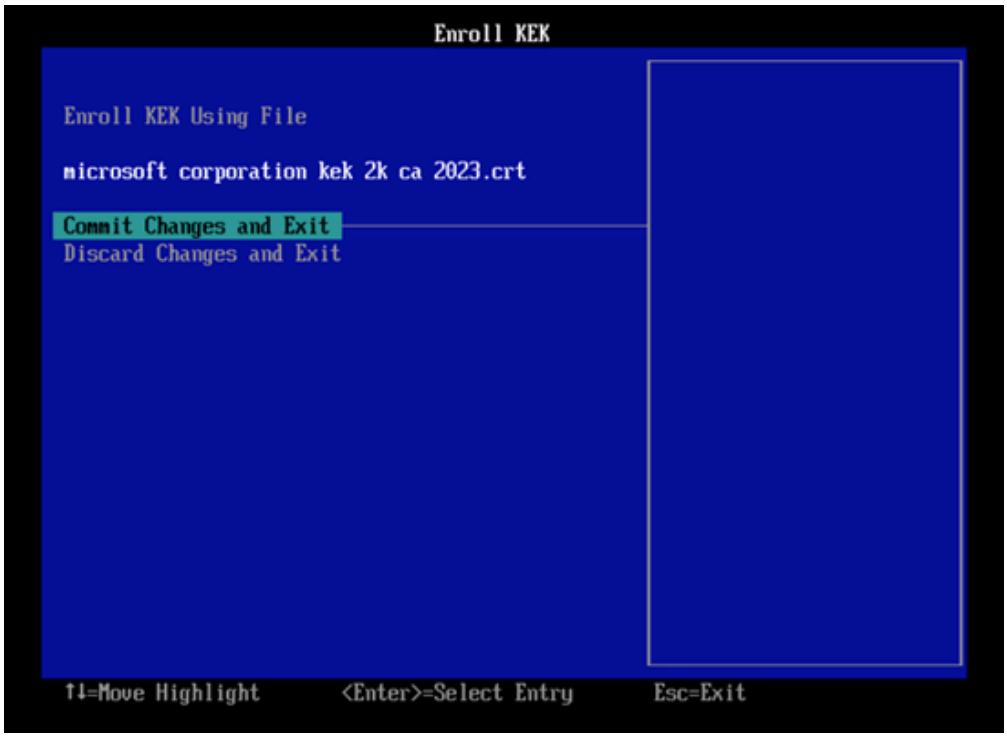

Figure 8. Enroll a KEK key file

8. The "Commit Changes and Exit" option was selected to persist the modifications to the firmware.
9. Steps 6 through 8 were repeated iteratively for each additional certificate key requiring enrollment.
10. The VM was shut down, and the uefi.allowAuthBypass parameter was removed from the advanced configuration to ensure the system returned to a secure state.
11. Upon powering on the VM, the mokutil utility was employed within the guest operating system to validate the successful application of the updates, as depicted in Figure 9.

```
root@localhost:~# mokutil --kek | grep -A13 "\[key"
[key 1]
Owner: 3d5f6e85-6791-4e26-a883-3f85a05169e4
SHA1 Fingerprint: 45:9a:b6:fb:5e:28:4d:27:2d:5e:3e:6a:bc:8e:d6:63:82:9d:63:2b
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            33:00:00:00:13:14:16:b8:61:6d:82:82:4b:00:00:00:00:00:13
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, O=Microsoft Corporation, CN=Microsoft RSA Devices Root CA 2021
        Validity
            Not Before: Mar  2 20:21:35 2023 GMT
            Not After : Mar  2 20:31:35 2038 GMT
        Subject: C=US, O=Microsoft Corporation, CN=Microsoft Corporation KEK 2K CA 2023
root@localhost:~#
```

Figure 9. Windows KEK certificate key

Management via VMX configuration

Beginning with ESXi 9.0, users can manage the certificate keys stored in Secure Boot variables (specifically PKDefault, KEKDefault, dbDefault, and dbxDefault) for both newly deployed and existing virtual machines to apply default key sets. For the purpose of this demonstration, dbDefault is used as the primary example.

Table 4 lists the related VMX configuration options and their descriptions.

Table 4. VMX configuration options

VMX options	Description
uefi.secureBoot.dbDefault.file0	Specify the certificate filename if the file is located within the VM's directory; otherwise, provide the absolute file path for certificates stored in other directories. To enroll multiple certificate keys, repeat this process for each subsequent file (e.g., file1, file2, etc.).
uefi.secureBoot.dbDefault.value0	Enter the hexadecimal representation of the certificate file's Authenticode hash into the db or dbx database.
uefi.secureBoot.dbDefault.append	If the value is set to TRUE, the certificate will be appended to the default db database. If the value is set to FALSE, the existing db database will be replaced.
uefi.secureBoot.db.resetOnce	When applying default certificate keys to an existing VM, the corresponding parameter will be automatically removed from the VMX configuration once the Secure Boot variable has been successfully reset.

The following steps describe the procedure for managing Secure Boot keys via VMX configuration parameters on an existing RHEL 9.8 VM:

1. Copy the Secure Boot certificate files to the virtual machine's directory on the ESXi datastore (e.g., /vmfs/volumes/datastore1/RHEL 9.8/).
2. Log in to the VMware Host Client, select the target VM, and power it off. Navigate to **Edit Settings > Advanced Parameters**, and enter the parameters as illustrated in Figure 10. These configurations will append the two RHEL certificate keys (kernel-signing-ca.cer and secureboot-uki-x86_64.cer) to the dbDefault Secure Boot variable.

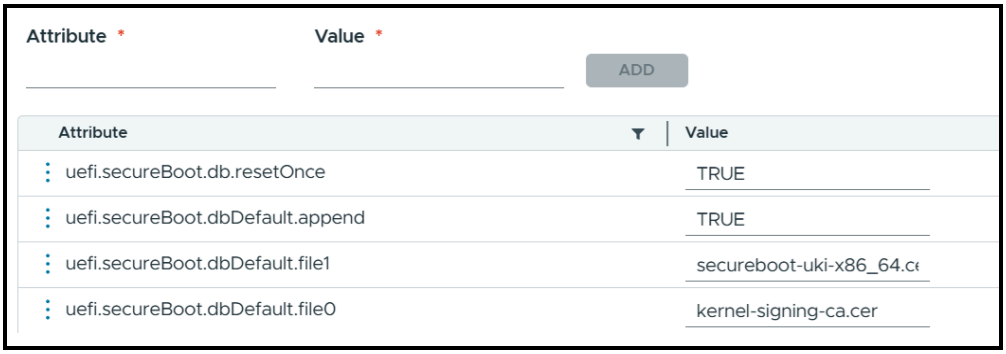

Figure 10. VMX parameters to manage secure boot keys

3. Power on the virtual machine and execute the mokutil utility within the guest operating system to verify that the changes have taken effect, as illustrated in Figure 11.

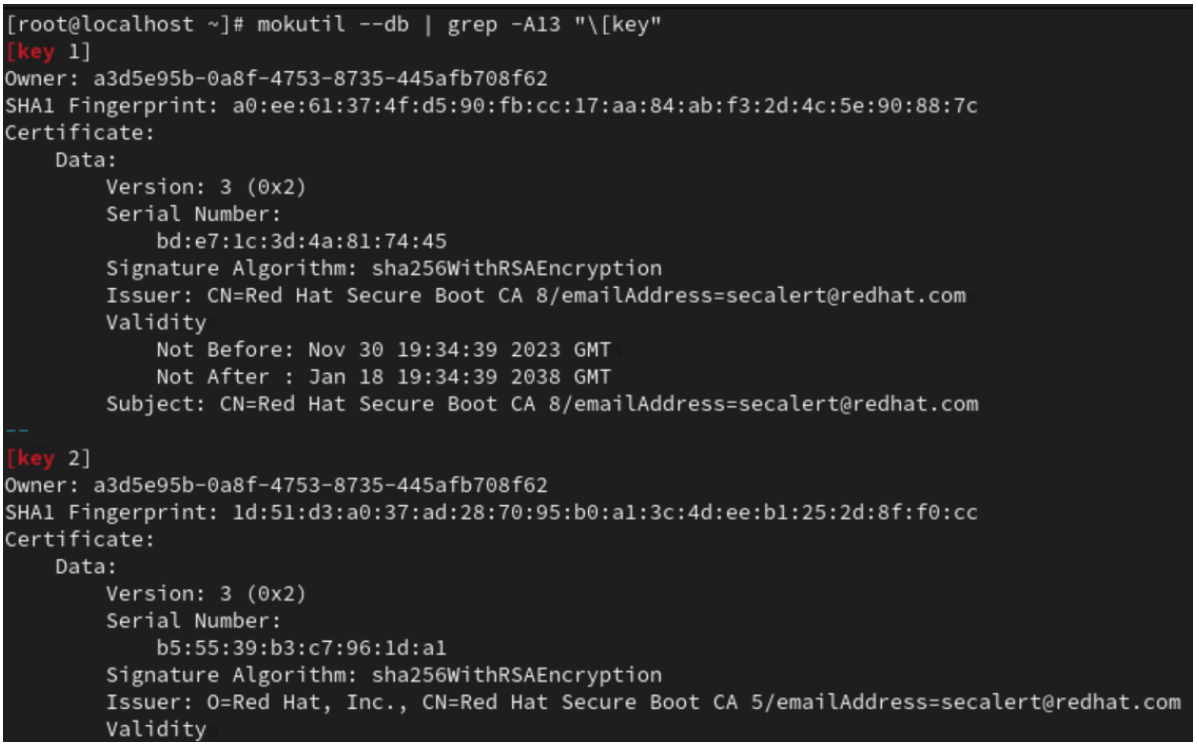

Figure 11. RHEL certificate keys

References

For more information, see these resources:

- Secure Boot Certificate Expirations: Guidance for ESXi Host Secure Boot
  https://knowledge.broadcom.com/external/article/434297
- Secure Boot Certificate Expirations and Update Failures in VMware Virtual Machines
  https://knowledge.broadcom.com/external/article/423893
- Secure Boot Custom Certificates
  https://knowledge.broadcom.com/external/article/377306
- Secure Boot CA 2023 Transition Guide for Windows Server-based Lenovo products.
  https://lenovopress.lenovo.com/lp2441
- Updating Windows Boot Manager and WinPE with the Windows UEFI CA 2023 Certificate
  https://lenovopress.lenovo.com/lp2353

Author

Alpus Chen is an OS Engineer at the Lenovo Infrastructure Solutions Group in Taipei, Taiwan. As a specialist in Linux and VMware technical support for several years, he is interested in operating system operation and recently focuses on VMware OS.

Related product families

Product families related to this document are the following:

- VMware Alliance
- VMware vSphere

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, LP2532, was created or updated on September 24, 2026.

Send us your comments in one of the following ways:

- Use the online Contact us review form found at:
  https://lenovopress.lenovo.com/LP2532
- Send your comments in an e-mail to:
  comments@lenovopress.com

This document is available online at https://lenovopress.lenovo.com/LP2532.

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

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

Microsoft®, Authenticode®, Azure®, BitLocker®, Windows Server®, and Windows® are trademarks of Microsoft Corporation in the United States, other countries, or both.

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