Sentinel® LDK and Sentinel HASP® Run-time Environment Command-line Installer for Windows: Readme

Version 7.101

October 2019

This document provides information regarding the Run-time Environment Command-line Installer for Sentinel HASP and Sentinel LDK, including enhancements and limitations. Using this installer, you can use HASP4, Hardlock, Sentinel HL, Sentinel HASP, and Sentinel LDK under any of the supported operating systems. ("Sentinel LDK" is the next generation of Sentinel HASP.)

The following topics are discussed:

Operating Systems Supported

> Windows 7 SP1

> Windows 8.1 SP1

> Windows Server 2008 R2 SP1

> Windows Server 2012 R2

> Windows Server 2016

> Windows Server 2019

> Windows 10 IoT Enterprise 2019 LTSC

> Windows 10 Version 1903

> **NOTE**
>
> > Windows 10 Insider Preview builds are not supported.

The operating system versions listed in this section were tested by Gemalto and verified to be fully compatible with Sentinel LDK. For reasons of compatibility and security, Gemalto recommends that you always keep your operating system up to date with the latest fixes and service packs.

The installer detects the version of the operating system at run-time, before installing the relevant drivers.

Virtual Environments Supported

For a list of the virtual environments supported, see "Supported Platforms for End Users" in the *Sentinel LDK Release Notes*.

The latest Release Notes can be seen at: http://sentinelldk.gemalto.com/LDKdocs/RN

Back to Topics

Upgrading the Run-time Environment

- When upgrading the Run-time Environment, ensure that it is not currently being accessed (for example: by closing all applications). Although the installation program can terminate applications that are accessing the Run-time, it cannot terminate running services.
- If you are running HASP License Manager (HASP4 and HASP HL legacy License Manager) as a service, you must stop the License Manager before proceeding with the installation.

Back to Topics

Installing the Run-time Environment

- Type **`haspdinst.exe -?`** for command-line help.
- To use Sentinel EMS to produce Run-time Environment installers in the Sentinel LDK v.6.0 or later environment, overwrite the `haspdinst.exe` file in **%ProgramFiles(x86)%\Gemalto Sentinel\Sentinel EMS\EMSServer\webapps\ems\haspTools** with the file in this package. (For Win32 systems, go to **%ProgramFiles%\...**)
- A log file of the installation process is written to **aksdrvsetup.log** in the Windows directory.

> **NOTE** By default, Windows displays a **User Account Control** message during driver installation. Users must click **Continue** to continue the installation. Alternatively, users can change the default setting from the Control Panel of their operating system.

For additional information, see the topic “Upgrading Sentinel LDK Run-Time Environment (RTE) Installer” in the *Sentinel EMS Configuration Guide*.

Back to Topics

Allowing Incoming Connections From Public Networks Using Port 1947

The Run-time Environment Installer adds a firewall rule named “Sentinel License Manager” that allowed incoming connections from private networks using port 1947.

You can manually allow access from public networks using this port, but Gemalto highly recommends against this.

If you do plan to allow incoming connections from public networks using port 1947, create a rule with a different name in order to prevent future RTE upgrades from removing this access.

Back to Topics

Issues Related to Device Guard and Code Integrity Policies

The traditional method until now to protect against malicious application under Windows has been to trust the applications unless they were blocked by an antivirus or other security solution. Device Guard, available in Windows 10 Enterprise, implements a mode of operation in which the operating system trusts only applications that are authorized by your enterprise. You designate these trusted applications by creating *code integrity policies*.

You can maintain a whitelist of software that is allowed to run (a configurable code integrity policy), rather than trying to stay ahead of attackers by maintaining a constantly-updated list of "signatures" of software that should be blocked. This approach uses the trust-nothing model well known in mobile device operating systems.

Only code that is verified by Code Integrity, usually through the digital signature that you have identified as being from a trusted signer, is allowed to run. This allows full control over allowed code in both kernel and user mode.

Code integrity contains two primary components:

- kernel mode code integrity (KMCI)
- user mode code integrity (UMCI)

This section describes issues that arise and the workarounds when machines at the end user site are enabled with Device Guard, and the code integrity policy set to “enforce” mode.

> **NOTE** The procedures described in this document should be performed by an IT professional who is familiar with Device Guard and code integrity policies.

Issue 1: Windows blocks the installation of the Run-time Environment

During installation of the Run-time Environment on your computer, Windows displays a message similar to this: "Your organization used Device Guard to block this app. Contact your support person for more info."

Solution:

To install the Run-time Environment on a machine where Device Guard is enabled in enforce mode (which make use of PcaCertificate level code signing check), ensure that DigiCert is listed/added in the Signers list of the policy file.

Import the DigiCert Intermediate certificate to the trusted list of Intermediate Certification Authorities(ICA) store on the golden computer before creating code integrity policy.

A Digicert Intermediate certificate is available from https://aboutssl.org/digicert-trusted-root-authority-certificates/#intermediates. Under **Intermediate Certificates**, locate and download the DigiCert EV Code Signing CA (SHA2) certificate. You can also fetch this intermediate certificate from your trusted source.

To add the DigiCert Root Certificate

1. Download the certificate on the golden computer and double-click the certificate file. The Certificate dialog box is displayed.
2. Click **Install Certificate**. Follow the Certificate Install Wizard to complete the import of this certificate to the ICA store.
3. Run the steps to create a new or updated policy. This will allow Sentinel software to be installed without any issues on a machine where Device Guard is enabled.

Repeat the installation of the Run-time Environment.

Issue 2: Protected application does not operate at the customer site

(LDK-17267)) When you distribute applications that are protected with SL keys, the customized vendor library (haspvlib_*vendorID*.*) that are required for these applications are not signed. As a result, Device Guard does not allow the software to operate at the customer site.

Workaround A:

This workaround must be performed at the customer site.

Do the following to add an exception for the customized vendor library file in the code integrity policy:

1. Use Windows PowerShell in elevated mode to create a policy for the exception.
2. Use the Group Policy editor to deploy the policy file.

Each of these procedures is described below. For additional details, go to: https://technet.microsoft.com/en-us/itpro/windows/keep-secure/deploy-code-integrity-policies-steps?f=255&MSPPError=-2147217396

To create the policy for the exception

1. Open PowerShell in elevated mode.
2. Run the command to create a policy (referred to below as P1) in audit mode.
3. Deploy this policy.
4. Operate the protected application as you would normally.
5. Create another policy (referred to as P2) that captures audit information from the events log.

> **NOTE** Before proceeding with the next step, review policy P2 carefully. This policy contains information about all the binaries that were used in your system while you operated the protected application. Any unwanted application that was executed during this time is logged in the policy. If not removed, any such application will be treated as a trusted binary.

6. Merge policies P1 and P2.
7. Disable audit mode.

8. Deploy the merged policy.

To deploy the policy file

1. Open the Group Policy editor by running **GPEdit.msc**.
2. Navigate to: **Computer Configuration\Administrative Templates\System\Device Guard**
3. Select **Deploy Code Integrity Policy**. Enable this setting by using the path to the relevant policy file created above.

Workaround B (not recommended):

This workaround must be performed at the customer site.

Before deploying the code integrity policy, disable UMCI (user mode code integrity) mode.

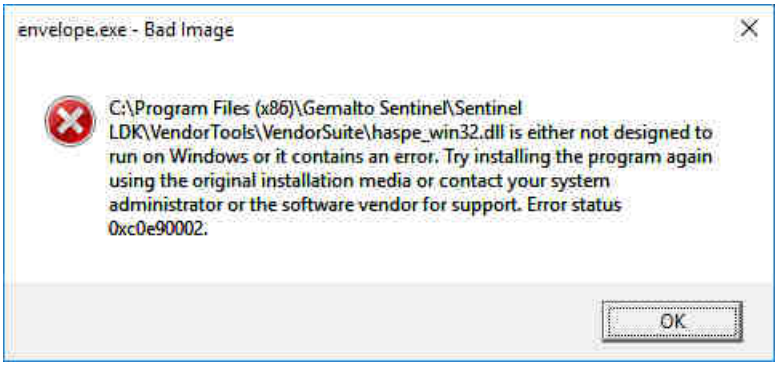
To accomplish this, run the following command in Windows PowerShell in elevated mode:

```
Set-RuleOption -FilePath <Policy path> -Option 0 -Delete
```

Issue 3: Vendor Tools fail to load

(SM-907) Sentinel LDK Vendor Tools fail to load. An error message is displayed, stating that a DLL, LIB, COM, or EXE file is not designed to run on Windows or that the DLL contains an error.

For example:



Workaround A:

Do the following to add a policy for the Sentinel LDK Vendor Tools in the code integrity policy file:

1. Use Windows PowerShell in elevated mode to create a policy for the Vendor Tools.
2. Use the Group Policy editor to deploy the policy file.

Each of these procedures is described below. For additional details, go to: https://technet.microsoft.com/en-us/itpro/windows/keep-secure/deploy-code-integrity-policies-steps?f=255&MSPPError=-2147217396

To create the policy for the Vendor Tools

1. Open PowerShell in elevated mode.
2. Run the command to create a policy (referred to below as P1) in audit mode.
3. Deploy this policy.
4. Execute all of the Vendor Tools that you will require at your site and perform all of the functions in these tools that you will require. If you miss any required Vendor Tools or Vendor Tool functions, the required entries will not be added in the new code integrity policy, and these tools or functions will generate an error message when they are eventually used.
5. Create another policy (referred to as P2) that captures audit information from the events log.

   > **NOTE** Before proceeding with the next step, review policy P2 carefully. This policy contains information about all the binaries that were used in your system while you operated the Vendor Tools. Any unwanted application that was executed during this time is logged in the policy. If not removed, any such application will be treated as a trusted binary.

6. Merge policies P1 and P2.

7. Disable audit mode.
8. Deploy the merged policy.

To deploy the policy file

1. Open the Group Policy editor by running **GPEdit.msc**.
2. Navigate to: **Computer Configuration\Administrative Templates\System\Device Guard**
3. Select **Deploy Code Integrity Policy**. Enable this setting by using the path to the relevant policy file created above.

Workaround B (not recommended):

Perform this workaround at your development site.

Before deploying the code integrity policy, disable UMCI (user mode code integrity) mode.

To accomplish this, run the following command in Windows PowerShell in elevated mode:

```
Set-RuleOption -FilePath <Policy path> -Option 0 -Delete
```

No further actions are required.

(SM-18780) When a user creates a Run-time Environment Installer using Sentinel EMS, the digital signature is removed from the Installer. As a result, Device Guard blocks the RTE Installer from executing.

Enhancements in Version 7.101

Reference	
SM-61960	configuration check box in Admin Control Center. When selected, the License Manager generates ID files. When cleared, the License Manager stops generating any new ID files. However, the existing ID files are retained.

Issues Resolved in Version 7.101

Reference	
SM-62902	https://sentinel.gemalto.com/technical-support/security-updates-sm/

Back to Topics