

Sentinel LDK 10.0

SOFTWARE PROTECTION AND LICENSING API TUTORIAL:
DEMO KIT



Disclaimer and Copyrights

All information herein is either public information or is the property of and owned solely by Thales and/or its subsidiaries and affiliates who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Thales and any of its subsidiaries and affiliates (collectively referred to herein after as "Thales") information.

This document can be used for informational, non-commercial, internal and personal use only provided that:

- The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any publicly accessible network computer or broadcast in any media and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided "AS IS" without any warranty of any kind. Unless otherwise expressly agreed in writing, Thales makes no warranty as to the value or accuracy of information contained herein. The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Thales reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Thales hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Thales be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Thales does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms' resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Thales be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Thales products. Thales disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service or loss of privacy.

Copyright © 2024 THALES. All rights reserved. Thales, the Thales logo and Sentinel are trademarks and service marks of Thales and/or its subsidiaries and affiliates and are registered in certain countries. All other trademarks and service marks, whether registered or not in specific countries, are the properties of their respective owners.

Build 2303-1 DK-API

CONTENTS

Which Tutorial is Right for Me?	5
Introduction	6
Sentinel LDK System	6
Sentinel Keys	7
Conventions	8
Lessons in This Tutorial	8
Additional Resources	9
Lesson 1: Preliminary Preparation	10
Installing the Sentinel LDK Software	10
Starting the Sentinel LDK Launcher	11
Lesson 2: Defining Features and Products	12
Defining Features	12
Defining Products	12
Our ERP System	13
Lesson 3: Using Sentinel LDK ToolBox	15
Selecting Your Programming Language	16
Working With the License Generation API	17
Starting a License Definition	17
What's Next?	18
Lesson 4: Protect Once	19
Protecting the Sample Bouncing Ball Program	19
Protecting the MyNotepad Program	21
Lesson 5: Defining an Unlocked Trialware Product	23
Defining an Unlocked Trialware Product	23
Preparing a Run-time Environment Installer	26
Lesson 6: Creating and Distributing a Sentinel SL-protected Software Order	27
Installing a Trial Product	28
Defining a Perpetual License	29
Collecting Information on the End User's Computer	29
Defining the License for the End User	30
Activating a Product With a License Update	31
Lesson 7: Creating and Distributing a Sentinel HL-protected Software Order	32

Defining a “Rental” Product Order With Sentinel HL Locking	33
Using an Application With a Sentinel HL Key	34
Creating an Update for a License in the Field	35
Updating a License at the Customer's Site	36
Lesson 8: Using Sentinel Licensing API	38
Starting With Login	38
Encrypting Data and Writing to Memory	39
Retrieving Sentinel Protection Key Information	40
Reading and Programming Sentinel Protection Key Memory	40
What's Next?	41

Which Tutorial is Right for Me?

A number of tutorials for learning about Sentinel LDK are available. We recommend that you review the information below to ensure that you choose the tutorial that is most appropriate for your requirements.

All the tutorials are included in the Sentinel LDK Demo kit and Starter kit.

Software Protection and Licensing Tutorial — Sentinel SL (Software-based) Keys

This tutorial demonstrates how to protect and license an application with Sentinel SL (software) protection keys.

Requirements: Download the free Sentinel LDK Demo Kit. *No hardware keys are required to perform this tutorial.*

NOTE The remaining tutorials described below demonstrate how to protect and license an application with either Sentinel SL (software) or Sentinel HL (hardware) protection keys.

Software Protection and Licensing Tutorial — “Demo Kit” or “Starter Kit”

Sentinel LDK-EMS (Entitlement Management System) is used as the back office system to manage definition of Features and Products and to generate and track customer entitlements.

> The **Demo Kit** tutorial uses a provided demonstration hardware key.

Requirements: Obtain the free physical Sentinel LDK Demo Kit from your Thales sales representative or from the Thales web site.

> The **Starter Kit** tutorial uses your own Sentinel Developer key and Sentinel HL key to protect and license an application.

Requirements: Purchase the Sentinel LDK Starter kit and Sentinel HL protection keys from your Thales sales representative.

Software Protection and Licensing API Tutorial — “Demo Kit” or “Starter Kit”

These tutorials demonstrate how you can use Sentinel LDK APIs to generate and update licenses. It is assumed that you would use your own back office systems to handle all ERP functionality.

A **Demo Kit** tutorial and a **Starter Kit** tutorial are available as described above.

Introduction

Thank you for choosing to evaluate Sentinel License Development Kit (Sentinel LDK) as the method to protect and license your software against unauthorized use. In this Tutorial, you use the Sentinel LDK system to protect, license, and distribute software.

The strength, uniqueness, and flexibility of Sentinel LDK are based on two primary principles:

Protect Once—Deliver Many—Evolve Often™: the concept of separating the software engineering and business processes

Cross-Locking: the technology that supports the Protect Once—Deliver Many—Evolve Often concept, enabling a protected application to work with a Sentinel HL (hardware) or a Sentinel SL (software) key

All commercial decisions, package creation, and license definitions are executed by product or marketing managers after the protection has been implemented.

This workflow model provides you with greater flexibility and freedom when defining new sales and licensing models, including feature-based and component licensing, evaluation, rental, floating, subscription, trialware (trial/grace), pay-per-use, and more, enabling you to focus on revenue growth.

Sentinel LDK System

The Sentinel LDK system provides you with a wide range of options for handling the business processes associated with producing and distributing your protected applications and Sentinel protection keys:

- > You can use the Sentinel LDK-EMS web interface and Sentinel LDK-EMS Server as a complete standalone system.
- > You can use the Sentinel LDK-EMS Web Services and Sentinel LDK-EMS Server software to connect the Sentinel LDK system with your existing ERP and CRM back office systems.
- > You can use the Sentinel License Generation API together with your existing licensing server software and ERP and CRM back office system for maximum flexibility and control over your business processes.

Regardless of the option you choose for handling your business processes, you will use Sentinel LDK Envelope or Sentinel Licensing API (or both) to protect your application and define the features that can be licensed.

This tutorial demonstrates the use of the Sentinel License Generation API to generate and maintain Sentinel protection keys. The License Generation API provides the functionality for securely updating Sentinel HL keys and SL keys, both locally and at the customer site.

In this tutorial, it is assumed that your organization will employ its own back office systems to handle all other aspects of tracking Features and Products, and of managing and tracking customer interactions. Your back office systems for Features and Products are represented in the tutorial lessons by a spreadsheet (shown below).

	A	B	C	D	E
1	Vendor Back Office ERP System				
2					
3	Features				
4		Feature Name			
5		Feature ID			
6					
7	Products				
8		Base Products			
9		Product ID			
10		Base Product			
11		Locking Type (Enforcement Type)			
12		Feature ID			
13		License Type			
14		Number of Executions			
15		Memory Offset			
16		Memory Length			
17		Memory Content			
18					
19		Unlocked Trialware Products			
20		Unlocked Trialware Product			
21		Product ID			
22		Feature			
23		License Type			
24		Number of Days			

Sentinel Keys

Your Sentinel LDK - Demo Kit includes the following Sentinel Demo keys:

- > Sentinel HL Time (Driverless configuration) key



- > Sentinel HL Max (Driverless configuration) key



The Sentinel HL Time key provides an on-board real-time clock. All Sentinel HL (Driverless configuration) keys can be used without installing a Run-time environment on the end user's computer. For a complete description of each key, see the [Sentinel HL Data Sheet](#).

You can use any of these keys to perform the lessons in this tutorial. (Use the same key for all lessons in the tutorial.)

Conventions

Each of the lessons in this tutorial begins with a focused breakdown of the lesson, similar to the following:

Objective:	A short description of the lesson objective
Roles in this lesson:	The role or roles that you assume during the lesson
Applications used:	Sentinel LDK applications that you use in the lesson

Lessons in This Tutorial

This tutorial is based on a Windows operating system and includes the following lessons:

> ["Preliminary Preparation" on page 10](#)

In this lesson, you install the Sentinel LDK system and learn how to launch Sentinel Vendor Suite.

> ["Defining Features and Products" on page 12](#)

In this lesson, you will use your back office ERP system to record the attributes of the Features that you will later license and protect, and the Products that will be based on these Features.

> ["Using Sentinel LDK ToolBox" on page 15](#)

In this lesson, you learn about the Sentinel ToolBox. This advanced user interface enables you to work with API function calls before you write even a single line of programming code.

> ["Protect Once" on page 19](#)

In this lesson, you use Sentinel LDK Envelope to apply protection to your software.

> ["Defining an Unlocked Trialware Product" on page 23](#)

In this lesson, you generate a license for an Unlocked Trialware Product, based on one of the Products you defined. This will enable you to distribute a trial version of your software later.

> ["Creating and Distributing a Sentinel SL-protected Software Order" on page 27](#)

In this lesson, you lock your protected software to a Sentinel SL key for later distribution. You will code the Sentinel SL key for a customer, and use the software as an end user. You will also learn how to activate a license, in order to convert the trial version of a program to a full-featured program.

> ["Creating and Distributing a Sentinel HL-protected Software Order" on page 32](#)

In this lesson, you lock your protected software to a Sentinel HL key for later distribution. You will code the Sentinel HL key for a customer order, and use the software as an end user. You will also learn how to create a license update for a deployed Product that has its license locked to a Sentinel HL key.

> ["Using Sentinel Licensing API" on page 38](#)

In this lesson, you learn how to use the Licensing API to program the memory on Sentinel protection keys.

We recommend that you follow the lessons sequentially to familiarize yourself with the Sentinel LDK system and to gain a clear understanding of the strength of the Protect Once—Deliver Many—Evolve Often model, and the flexibility it provides.

Additional Resources

You can use the following resources to help you learn about the Sentinel LDK system:

> [Sentinel LDK Software Protection and Licensing Guide](#)

This book describes Sentinel LDK licensing concepts and provides a detailed overview of the system.

> [Sentinel LDK Release Notes](#)

The Release Notes provide information about the current release, including:

- latest enhancements
- supported platforms
- available documentation
- known issues

LESSON 1: Preliminary Preparation

Objective:	Prepare your computer and Sentinel LDK for the tutorial
Applications used:	<ul style="list-style-type: none">> Sentinel LDK installation package> Sentinel Vendor Suite

The procedures in this lesson are required in order for you to be able to implement the processes outlined in subsequent lessons of this tutorial. They include:

- > Instructions for installing the Sentinel LDK software, including Sentinel LDK-EMS Server
- > Instructions for launching Sentinel Vendor Suite

Installing the Sentinel LDK Software

Before proceeding with this tutorial, install the Sentinel LDK software on your computer.

CAUTION! Do not connect your Sentinel Demo key before installing the Sentinel LDK software!

For instructions on how to install Sentinel LDK-EMS and Sentinel LDK Vendor Tools, refer to the section “Installing Sentinel LDK With Sentinel LDK-EMS (On-Premises)” in the [Sentinel LDK Installation Guide](#).

During the installation procedure, you can accept most defaults.

When prompted for Installation Options, select:

- > **Vendor Suite**
- > **Entitlement Management System**
 - **Sentinel LDK-EMS - install on this machine**

When prompted to choose a setup type for Sentinel LDK-EMS, click the **Express** option.

Starting the Sentinel LDK Launcher

After completing the installation, launch Sentinel Vendor Suite as follows:

1. In the Windows Services window, ensure that the Sentinel LDK-EMS Service is active
2. From the **Start** menu, open **Sentinel LDK**. The Sentinel LDK Launcher is displayed. You can access the primary Vendor Suite applications from this screen.

You are now ready to start the tutorial.

LESSON 2: Defining Features and Products

Objective:	Learn how to define Features and Products
Role in this lesson:	Product Manager
Applications used:	Spreadsheet or Your ERP system

A Feature is an identifiable functionality of a software application that can be independently protected or licensed. In Sentinel LDK, a Feature may be an entire application, a module, or a specific functionality such as Print, Save, or Draw.

A Product is a licensing entity that represents one of the vendor's marketable software products. The Product contains one or more Features. License terms are defined for each Feature in the Product.

Defining Features

Features are typically defined by the Product Manager based on business decisions, and are then implemented by the company's software engineers.

You define Features by:

- > selecting the functions in your application that you want the ability to license individually.
- > adding calls to the Sentinel Licensing API from the appropriate functions in your application.
- > protecting your application using Sentinel LDK Envelope.

Features to be licensed in a products are typically recorded in Sentinel LDK-EMS or in your ERP system. Since the scope of this tutorial does not include Sentinel LDK-EMS, we will record two Features in our ERP system. For the purposes of this tutorial, we will use a spreadsheet as our back office ERP system.

Defining Products

You define a Product by:

- > generating a license definition in the Sentinel License Generation API.
- > applying the generated license definition to a Sentinel protection key.

Like Features, Products are typically defined using Sentinel LDK-EMS. Typically, this function is performed by the Product Manager, and implemented by the software engineer. In a true production environment, a number of Products might be defined and held in readiness for inclusion in customer orders. However, you can define new Products—mixing and matching Features—at any time.

For the lessons in this tutorial, we will define a number of Products. The Products will include license terms for the two Features mentioned above. The Products that we define now will be included in the orders that you define in later lessons. As with the Features, we will record the Products in our ERP “system.”

In this lesson, we define the following Products:

- > “Bounce_Lite”—A “rental” Product, in which the license expires after three executions. Following the third execution, the license must be updated to enable further use of the program.
- > “Bounce_Full”—A “full-featured” Product, in which the license enables unrestricted access to all the Features in the program.
- > “MyNotepad_Full”— A “full-featured” Product that enables unrestricted access to all the Features in the program. In ["Defining an Unlocked Trialware Product" on page 23](#), you will define an evaluation **MyNotepad** Product. The **MyNotepad_Full** Product you define in this lesson will later be used to update the evaluation license to a full-featured license.

NOTE In ["Creating and Distributing a Sentinel SL-protected Software Order" on page 27](#), and ["Creating and Distributing a Sentinel HL-protected Software Order" on page 32](#) you will assume the role of an end user, and will install licenses and run these Products.

Our ERP System

The spreadsheet displayed below contains the information that would be stored in the ERP system.

The spreadsheet contains:

- > The two Features that we will protect in our applications. Each Feature is assigned a name and an ID.
- > The three base Products described above. For each Product, the system lists the Features contained in the Product and the attributes of the Product.

	A	B	C	D	E
1	Vendor Back Office ERP System				
2					
3	Features				
4		Feature Name	Bounce	MyNotepad	
5		Feature ID	50	10	
6					
7	Products				
8		Base Products			
9		Product ID	11	12	13
10		Base Product	Bounce_Lite	Bounce_Full	MyNotepad_Full
11		Locking Type (Enforcement Type)	HL or SL-AdmiMode	HL or SL-AdmiMode	HL or SL-AdmiMode
12		Feature ID	50	50	10
13		License Type	Execution	Perpetual	Perpetual
14		Number of Executions	3		
15		Memory Offset			0
16		Memory Length			11
17		Memory Content			Hello World
18					
19		Unlocked Trialware Products			
20		Unlocked Trialware Product			
21		Product ID			
22		Feature			
23		License Type			
24		Number of Days			

Figure 1: ERP System - Features and Products

LESSON 3: Using Sentinel LDK ToolBox

Objective:	Learn how to work with Sentinel LDK ToolBox
Role in this lesson:	Development
Applications used:	Sentinel LDK ToolBox

Before you begin the other lessons in this tutorial, you should have a basic understanding of how Sentinel LDK ToolBox works. Sentinel LDK ToolBox is an interactive GUI application that helps you familiarize yourself with various Sentinel LDK APIs. In addition, you can use Sentinel LDK ToolBox to actually execute API functions.

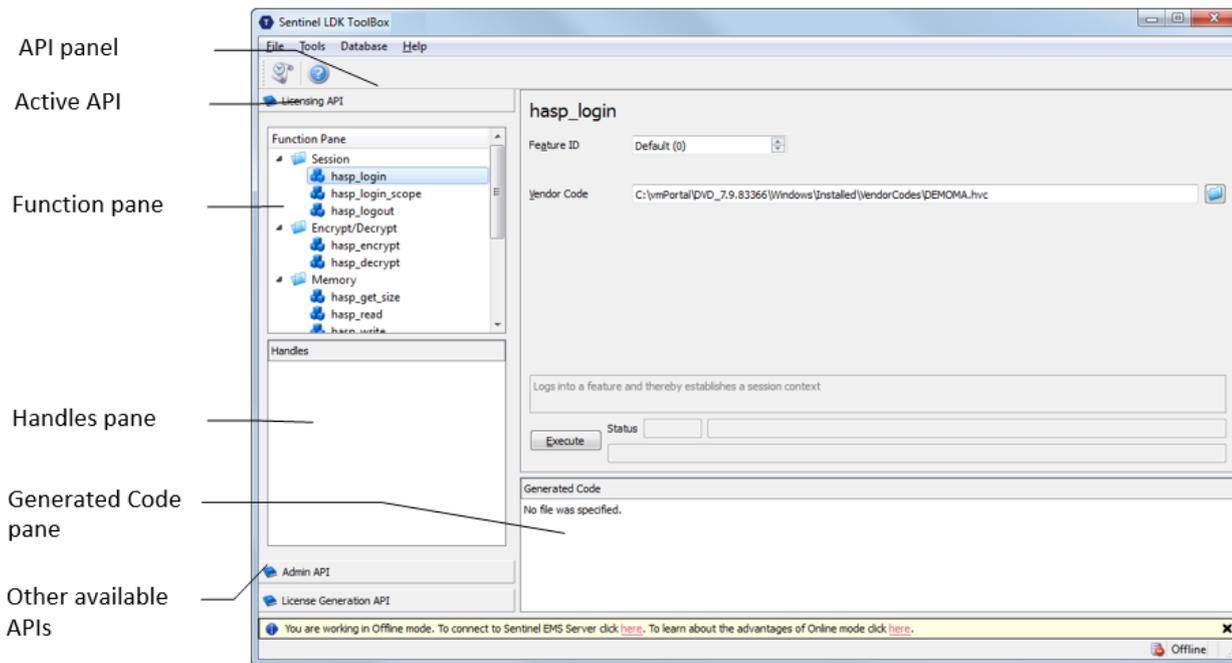
This lesson introduces you to:

- > **Sentinel LDK License Generation API.** This API is used in the software vendor production environment to generate new and updated licenses for Sentinel protection keys.
- > **Sentinel LDK Licensing API.** This API exists in the software vendor production environment and in the Runtime Environment on the end user's computer. The API is used for advanced protection techniques for the protected application and to enforce licensing parameters for the protected application.

You can also use Sentinel LDK ToolBox to generate source code that you can immediately copy and paste into your own application source code.

The Sentinel LDK ToolBox window contains these elements:

- > API panel, in which you select the API to work with
- > Function pane, in which you select the function to perform
- > Display pane, in which you can view and edit the details of the function call you selected
- > Generated Code pane, in which code is displayed. You can copy this code and paste it into your own files.
- > Handles pane, in which the session handle is displayed



To launch Sentinel LDK ToolBox

1. From the **Start** menu, open **Sentinel LDK**. The Sentinel LDK Launcher is displayed.
2. Click **ToolBox** to launch the Sentinel LDK ToolBox application. The Sentinel LDK ToolBox main window is displayed.

The API panel on the left of the ToolBox window contains all the available APIs. For the selected API, the available functions are displayed.
3. Browse the list. You can select any API, and then select an available function for that API. After you select a function, the Display pane on the right of the ToolBox window displays the parameters that you must specify in order to execute the function call.

Selecting Your Programming Language

Sentinel LDK ToolBox can generate code in several programming languages.

To select your programming language

1. From the **File** menu in Sentinel LDK ToolBox, select **Settings**. The Settings window is displayed.
2. Select the **ToolBox** tab.
3. From the **Programming language** list, select your preferred language.

NOTE For this tutorial, it is recommended that you select **C-API**. If C-API is not selected, select it and click **OK**. If C-API is already selected, click **Cancel**.

The Settings window closes with the programming language selected.

Working With the License Generation API

This section demonstrates how to get started with the License Generation API in order to generate new or modified licenses for a Sentinel protection key.

This is not a complete exercise. The purpose of this section is to demonstrate how you perform functions in Sentinel LDK ToolBox.

Initializing a Session

Each License Generation API session begins with the `sntl_ig_initialize` function call, which initializes libraries and establishes a session handle.

NOTE While you work with License Generation API in ToolBox or in your own application, you can initialize and maintain up to 512 simultaneous license generation sessions. For each session, you use the unique handle created by the `sntl_ig_initialize` function in all subsequent function calls to isolate the functions from other active sessions.

In this tutorial, we will work with only one active session at any given time.

To initialize a license generation session

1. In the API panel, click **License Generation API**.
2. In the **Function** pane, select `sntl_ig_initialize`.
3. This function requires no parameters, so we simply execute the function.

At the bottom left of the `sntl_ig_initialize` pane, click **Execute**. The result of the executed initialization is summarized to the right of the **Execute** button in three **Status** fields—number, code and message. If the function complete successfully, the session handle is displayed in the **Handles** pane.

Note that the programming code required to perform this same function in your own application is now displayed in the **Generated Code** pane. As long as ToolBox remains active, you can click  to access the API Call History dialog box. This screen contains a history of all the function calls that you executed and the generated programming code for each function call.

Starting a License Definition

Each license that you generate begins with the `sntl_ig_start` function call. This function starts the process of defining a new license or an update to an existing license. (In many cases, this is the only function that you require to define a license.) You use this function to assemble all the information required for a license. This consists of:

- > The type of license to generate

- > Your Vendor code
- > Current state of the protection key
- > XML code that contains the license definition parameters. This can be a combination of:
 - code copied from a template
 - code that you enter manually
 - code created using tools included in the ToolBox.

To start a license definition

1. In the **Function** pane, select **sntl_lg_start**. The parameters for this function appear in the Function Display.
2. For **License Type**, select **SNTL_LG_LICENSE_TYPE_PROVISIONAL**.
3. In the **Vendor Code** field, specify the location of your stored Vendor Code file.

To locate your Vendor Code file, click the browse button.

By default, the DEMOMA Vendor Code is stored in:

```
%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorCodes
```

The file is named according to the label printed on your keys. For example, if the keys are labeled with Batch Code ABCDE, by default, your Vendor Code file is named **ABCDE.hvc**.

4. Select **DEMOMA.hvc** and click **Open**. The path name is displayed in the **Vendor Code** field.
5. For a trialware license, the **Current State** field is left empty.

The **License Definition** field can be filled out using one or more of the following techniques:

 - a. You can select a value for the **License Template** field to enter a predefined license definition.
 - b. You can use the built-in function buttons (such as **New Definition** or **Edit Current Block**) to enter or modify a license definition.
 - c. You can manually enter or modify XML code in the **License Definition** field.
6. For this demonstration, select **New Provisional Product** in the **License Templates** field. ToolBox fills in the **License Definition** field with the XML code to define a sample Unlocked Trialware Product.
7. At the bottom left of the **sntl_lg_start** pane, click **Execute**. The results of the executed **start** function are displayed to the right of the **Execute** button in three **Status** fields—number, code and message.

What's Next?

You have actually started the process to define a trialware license for a Sentinel protection key. We will not continue this specific process at this point, but we will use the techniques that were demonstrated here in the lessons that follow.

LESSON 4: Protect Once

Objective:	Learn how to apply Sentinel LDK automatic protection functionality
Roles in this lesson:	Development
Applications used:	Sentinel LDK Envelope

In this lesson, you use Sentinel LDK Envelope to define protection parameters for two programs, using the two Features you defined earlier.

Sentinel LDK Envelope is an innovative, advanced solution for protecting software against illegal or unauthorized use. The solution deters illegal access and execution of protected applications.

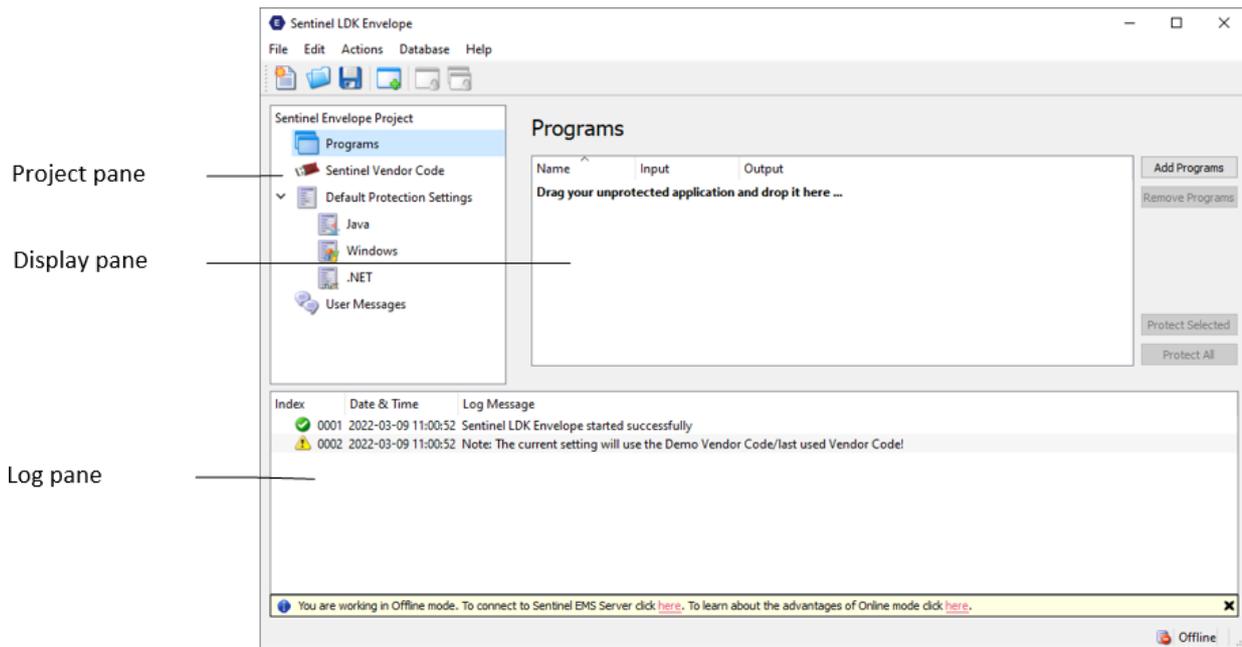
A deployed Sentinel LDK-protected program requires access to a Sentinel protection key in order to run. The protected program queries the Sentinel protection key for predefined information. If the Sentinel protection key is not present, or the information returned is incorrect, the program does not execute, or stops functioning.

Protecting the Sample Bouncing Ball Program

In this section, you launch Sentinel LDK Envelope and identify the Vendor Code with which you will protect the software. You then define an end-user error message. Finally, you protect the Bouncing Ball program.

The Sentinel LDK Envelope window includes the following:

- > Project pane, in which you select the function to perform
- > Display pane, in which you select items and perform actions. The name of the pane changes to reflect the function you selected in the Project pane—for example, in the screen below, the Display pane has the title **Programs**.
- > Log pane, in which Sentinel LDK Envelope messages are displayed



To prepare for protection

1. Connect your Demo key.
2. From the **Start** menu, open **Sentinel LDK**. The Sentinel LDK Launcher is displayed.
3. Click **Envelope** to launch Sentinel LDK Envelope. The Sentinel LDK Envelope main window is displayed.
4. Before protecting an application, Sentinel LDK Envelope must identify the correct Vendor Code. In the **Sentinel LDK Envelope Project** pane, select **Sentinel Vendor Code**.
5. In the right pane of the Sentinel Vendor Code window, click **Toggle connection with the Server**. The Log In to Sentinel LDK-EMS screen is displayed.
6. Enter the user name and password to log in to Sentinel LDK-EMS; then click **OK**. (The default user name and password are **admin – admin**.) The available Vendor Codes are displayed in the **Sentinel Vendor Codes** pane.
7. Select **Use Vendor Code from Database**, and then select the DEMOMA Vendor Code.

To protect the Bouncing Ball program

1. In the Sentinel LDK Envelope **Project** pane, select **Programs**.
2. Click **Add Programs** in the top right of the **Programs** pane.

The Add Programs window is displayed, showing the contents of the \VendorSuite\samples folder.

If a different folder is displayed, browse to:

```
%ProgramFiles(x86)%\Thales\Sentinel LDK\VendorTools\VendorSuite\samples\
```

3. Select **Win32_Bounce.exe** and click **Open**. Win32_Bounce is now listed in the **Programs** pane.

4. In the **Project** pane, under **Programs**, select **Win32_Bounce**. The **Display** pane lists the Protection Details.
5. In the **Name** field (to the right of the **Feature ID** field), select **Bounce** from the list (the Feature that you defined for the Bounce program in ["Defining Features and Products" on page 12](#)).
6. In the **Project** pane, under **Default Protection Settings**, select **Windows**. The **Display** pane now displays **Windows - Default Protection Settings**.
7. In the **Windows - Default Protection Settings** pane, decrease the number of **Periodic background checks** from the default value to **5** seconds. This means that during run-time, the protected program will check every 5 seconds to ensure that the required Sentinel HL key is connected.
8. In the **Project** pane, under **Programs**, select **Win32_Bounce**, then in the **Display** pane, click **Protect Selected**. A message informs you that the protection process is running.
9. Click **Close** when Sentinel LDK Envelope notifies you that the protection process has completed.

The protected Bouncing Ball program is saved in this folder:

```
%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\Protected
```

Protecting the MyNotepad Program

In this section, you create a copy of the Thales text viewer sample application, define protection parameters for protecting it using the MyNotepad Feature, and protect the program.

To protect the MyNotepad program

1. Locate the Thales text viewer application (textviewer.exe). This utility can be found under:


```
%ProgramFiles(x86)%\Thales\Sentinel LDK\VendorTools\VendorSuite\samples\
```
2. Create a copy of the text viewer application and rename it **MyNotepad.exe**.
3. Verify that your Demo key is connected to your computer.
4. If it is not already open, launch Sentinel LDK Envelope. The Sentinel LDK Envelope main window is displayed.
5. Select the program to protect:
 - a. In the **Sentinel LDK Envelope Project** pane, select **Programs**.
 - b. Click **Add Programs** in the top right of the **Programs** pane. The Add Programs window is displayed.
 - c. Navigate to the folder where MyNotepad.exe is located. Select the file and click **Open**. Note that a new **MyNotepad** entry is displayed in the **Project** pane under **Programs**.

6. Protect the selected program:
 - a. In the **Project** pane, under **Programs**, select **MyNotepad**. The **Display** pane lists the Protection Details.
 - b. In the **Feature ID** field, type **10** (the Feature defined for the MyNotepad program in "[Defining Features and Products](#)" on page 12).
 - c. In the **Project** pane, under **Default Protection Settings**, select **Windows**. The **Display** pane now displays **Windows - Default Protection Settings**.
 - d. In the **Windows - Default Protection Settings** pane, ensure that **Periodic background checks** is set to **5** seconds.
 - e. In the **Project** pane, under **Programs**, select **MyNotepad**, then in the **Display** pane, click **Protect**. A message informs you that the protection process is running.
 - f. Click **Close** when Sentinel LDK Envelope notifies you that the protection process has completed. The protected MyNotepad program is saved in this folder:

`%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\Protected`
7. Close Sentinel LDK Envelope. You are prompted to save the changes to the current project.
8. Click **Save**. The Save As dialog box is displayed.
9. In the **File Name** field, type **Tutorial** and click **Save**. The dialog box closes and the project is saved.

LESSON 5: Defining an Unlocked Trialware Product

Objective:	Learn how to define an Unlocked Trialware Product that can be supplied as a trial product
Roles in this lesson:	<ul style="list-style-type: none">> Product Manager> Development
Applications used:	<ul style="list-style-type: none">> Sentinel LDK-EMS> Sentinel LDK ToolBox (License Generation API)

In this lesson, still in the Production Manager role, you define the “MyNotepad_Evaluation” Product as a trial version of the “MyNotepad_Full” Product you defined in ["Defining Features and Products" on page 12](#).

An Unlocked Trialware Product is typically used for evaluation purposes, to enable the provision of a grace period, or for super-distribution. An Unlocked Trialware Product can be used on any computer for a limited time period. After the allotted period expires, the software no longer runs until it is activated with a Sentinel protection key.

Defining an Unlocked Trialware Product

In this section, in the Production Manager role, you define the “MyNotepad_Evaluation” Product—an Unlocked Trialware Product that is a trial version of the “MyNotepad_Full” Product, with a license that expires 60 days after the application is first used.

To define an Unlocked Trialware Product in your back office ERP system

- > Fill out your back office spread sheet with the desired parameters for the trialware product.

	A	B	C	D	E
1	Vendor Back Office ERP System				
2					
3	Features				
4		Feature Name	Bounce	MyNotepad	
5		Feature ID	50	10	
6					
7	Products				
8	Base Products				
9		Product ID	11	12	13
10		Base Product	Bounce_Lite	Bounce_Full	MyNotepad_Full
11		Locking Type (Enforcement Type)	HL or SL-AdmiMode	HL or SL-AdmiMode	HL or SL-AdmiMode
12		Feature ID	50	50	10
13		License Type	Execution	Perpetual	Perpetual
14		Number of Executions	3		
15		Memory Offset			0
16		Memory Length			11
17		Memory Content			Hello World
18					
19	Unlocked Trialware Products				
20		Unlocked Trialware Product			MyNotepad_Evaluation
21		Product ID			13
22		Feature			10
23		License Type			Time Period
24		Number of Days			60
25					

You prepare the V2C Unlocked Trialware license file using the License Generation API. For this tutorial, we will use Sentinel LDK ToolBox to call License Generation API functions.

To generate a V2C file for the “MyNotepad_Evaluation” Unlocked Trialware Product

1. From the **Start** menu, open **Sentinel LDK**. The Sentinel LDK Launcher is displayed.
2. In Sentinel LDK Launcher, click **ToolBox**. The Sentinel LDK ToolBox main window is displayed.
3. In the API panel, click **License Generation API**. The functions for the License Generation API are displayed.
4. In the **Function** pane, select **sntl_ig_initialize**. In the **Display** pane, click **Execute**. ToolBox calls the **sntl_ig_initialize()** function. The return code for the call is displayed in the **Status** fields. The session handle is displayed in the **Handles** pane.
5. In the **Function** pane, select **sntl_ig_start**. You will use this function to define the trialware license.
6. In the **Display** pane:
 - a. For **License type**, select **SNTL_LG_LICENSE_TYPE_PROVISIONAL**.
 - b. For **Vendor code**, click the browse button. In the VendorCodes directory, select the DEMOMA Vendor Code file.
 - c. For a trialware license, you leave **Current State** empty.

d. Click **New Definition**.

NOTE You will use the built-in interface screens to generate the license definition XML code. Therefore, there is no need to select a specific license template.

7. In the Edit Template dialog box:

a. For **Locking Type**, select **SL-AdminMode**.

b. In the **Product** pane, click **New**.

8. In the Edit Product dialog box:

a. Set the **Product ID** to 3 and the **Product Name** to MyNotepad_Evaluation.

b. In the **Feature** pane, click **New**.

9. In the Edit Feature dialog box, set the **Feature ID** to 10, **Feature Name** to MyNotepad, **License Type** to Time Period, and **Number of Days** to 60. Click **OK**.

10. In the Edit Product dialog box, click **OK**.

11. In the Edit Template dialog box, click **OK**.

ToolBox modifies the XML code in the **License Definition** field according to the information you just entered. The license definition is now configured to generate the **MyNotepad_Evaluation** Unlocked Trialware Product.

TIP You can save this Product definition for future use. Click **Save Template**. In the Save Template dialog box, select **Save as a new Template**. Type the name MyNotepad_Evaluation and then click **OK**. The Product will be available in the **License Templates** list.

12. Click **Execute**. ToolBox executes the **sntl_ig_start** function and displays the results.

Make sure that the **Status** field contains 0. This indicates that the license definition is valid and that the **sntl_ig_start** function completed successfully.

13. In the **Function** pane, select **sntl_ig_generate_license**.

14. Click **Execute**. ToolBox generates the license code.

Make sure that the **Status** field contains 0. This indicates that the license generation function completed successfully.

15. Under the **License** field, click **Save As**. In the Save As dialog box, type the file name MyNotepad_Eval and click **Save**. The trialware license for the My_Notepad_Evaluation Product is saved to:

%HomePath%\Documents\Thales\Sentinel LDK 10.0\Vendor Tools\Vendor Suite\V2C\MyNotepad_Eval.v2c

16. In the **Function** pane, select **sntl_ig_cleanup**, and then click **Execute**. This terminates the session handle and leaves ToolBox ready for the next License Generation API session.

Preparing a Run-time Environment Installer

In this section, you assume the Development role. You now prepare a Sentinel LDK Run-time Environment Installer executable that includes the V2C file created earlier. (The Installer automatically applies any V2C file located in the same directory.) The Run-time Environment may be required on the end user's computer in order for your Sentinel LDK protected and licensed program to function correctly. (This depends on the type of protection key that you will be providing. For this tutorial, you will install the Run-time Environment.)

In a true production environment, you would only need to embed the Run-time Environment installer and V2C file in your software setup to create an application that is ready-to-run and is protected and licensed by Sentinel LDK. In this scenario, you have already specified that the MyNotepad program will run in Trial mode for a maximum of 60 days.

To prepare a Sentinel LDK Run-time Environment Installer executable

1. Create a temporary directory to contain the Run-time Environment Installer.
2. A customized Run-time Environment Installer is available on the machine where you installed Sentinel LDK. Copy the file **haspdinst.exe** from the directory `%ProgramFiles(x86)%\Thales\Sentinel LDK\Redistribute\Runtime Environment\cmd Install` to the temporary directory created above.
3. Browse to the directory where you saved the V2C file in step 15 in "[Defining an Unlocked Trialware Product](#)" on page 23. Copy the the **MyNotepad_Eval.V2C** file from there to the temporary directory created above.

CAUTION! Do not execute the Run-time Environment installer at this time! You will install this bundle in a later lesson.

You are now ready to ship your Product with its license locked to either:

- > a Sentinel SL key, as described in "[Creating and Distributing a Sentinel SL-protected Software Order](#)" on page 27.

OR

- > a Sentinel HL key, as described in "[Creating and Distributing a Sentinel HL-protected Software Order](#)" on page 32.

LESSON 6: Creating and Distributing a Sentinel SL-protected Software Order

Objective:	<ul style="list-style-type: none">> Learn how to define an order that is locked to a Sentinel SL key, specify the customer for the order, and produce the order.> Simulate a variety of end-user interactions with programs protected by Sentinel protection keys.
Roles in this lesson:	<ul style="list-style-type: none">> Entitlement Manager> Production> End User
Applications used:	<ul style="list-style-type: none">> Sentinel LDK ToolBox (License Generation API)> Sentinel LDK ToolBox (Licensing API)> Sentinel Admin Control Center

In this lesson, we assume that you have received an order for the “MyNotepad” Product that you defined in ["Defining Features and Products" on page 12](#). Initially, you will ship the trial MyNotepad Product to the customer for installation on the customer’s computer. The license will be locked to a Sentinel SL key that will be activated on the end-user’s machine to convert the trial MyNotepad software to a full-featured product.

When you defined the Products, you specified the license terms for the Features included within those Products at that time. Note that instead of specifying the license terms when you defined the Product, you could have enabled the functionality for specifying the license terms at this stage as part of order processing.

NOTE Typically, a customer first installs a trial Product on their computer, and then uses the procedures described in this lesson to convert the Product to a full-featured Product.

You also have the option to bypass the trial phase and to install a full-featured Product on the customer’s computer. This procedure is not within the scope of this tutorial.

In this lesson, you will also assume the end-user role to install and use the Product. You will then use the license update received from the vendor to convert a trial version of the software to a full-featured program.

NOTE When a customer wants to purchase additional Features or to extend the current license for a Product that has its license locked to a Sentinel SL key, you can update the deployed license, as described in ["Creating an Update for a License in the Field" on page 35](#).

Installing a Trial Product

In this section, you simulate the experience of the end user when installing and running your MyNotepad “trial” software program.

Although in a true production environment your software engineer would typically embed the Run-time Environment installation which includes the trialware license (for trial use) into your software installation so that it executes as part of the installation process, in this tutorial you perform several installation and simulation steps.

To install the Sentinel Run-time Environment and trialware license on an end-user machine

1. From the **Start** menu, select **Run**.
2. Click **Browse** and navigate to the location of the RTE Installer (haspdinst.exe) that you prepared earlier on ["Preparing a Run-time Environment Installer" on page 26](#).
3. Select the **haspdinst.exe** file and click **Open**. The browse window closes, and the file path is displayed in the **Open** field.
4. In the **Open** field, type a space followed by -i at the end of the file path.

For example:

```
"C:\Users\Thomas\Downloads\haspdinst.exe" -i
```

5. Click **OK**. The Sentinel Run-time Environment is installed and a success message is displayed. The V2C file that is present in Download location together with Run-time Environment Installer is applied automatically to create the trial license.

To confirm the license status of the trial Product

1. Navigate to the following folder:
`%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\Protected`
2. Double-click **MyNotepad.exe**. A message is displayed to warn you that the program is only protected with a demo Sentinel protection key.
3. Click **OK**.
4. While the application is running, from the **Start** menu, open **Sentinel LDK**. In Sentinel LDK Launcher, click **Additional Tools > Admin Control Center**.

(The Sentinel Admin Control Center is part of the Run-time Environment.)

- In the **Options** pane, select **Sentinel Keys**. All the Sentinel protection keys that are accessible on the Sentinel License Manager are displayed.

#	Location	Vendor	Key ID	Key Type	Version	Sessions	Actions
0	Local	DEMOMA	1143840225975854213	SL Admin-mode	2.00	-	Products Features Sessions Certificates

- Locate the entry for your Demo key and click the **Features** button in the **Actions** column. The Features that are licensed in your key are displayed.

Note that the **Logins** field shows that there is currently one login to MyNotepad_Evaluation and that the **Restrictions** field for Feature ID 10 indicates that the 60-day trial has begun.

- Close the MyNotepad program.

Defining a Perpetual License

In this section, we assume that the end user is satisfied with the MyNotepad_Evaluation product and has placed an order for a perpetual license.

You would include a small routine in the Unlocked Trialware Product installer that uses Sentinel Licensing API to collect information from the end user's computer and generate a C2V file. The end user would activate this routine and then send you the generated C2V file by email.

Next, you will create a license for the "MyNotepad_Full" Product that you defined in ["Defining Features and Products" on page 12](#), locked to the user's Sentinel SL key. This license will then be used to convert the trial MyNotepad_Evaluation Product on the user's computer (that you defined in ["Defining an Unlocked Trialware Product" on page 23](#)) to a full-featured product.

Collecting Information on the End User's Computer

In the role of the end user, you will collect information about your computer and send it to the vendor. This information is required in order to prepare the perpetual license that you ordered. (This process can be partially or completely automated in the protected application.)

To simulate collection of information on the end user's computer

- Start Sentinel LDK ToolBox.
- In the API panel, click **Licensing API**; then select the **hasp_get_info** function.
- In the **Display** pane:
 - In the **Scope Template** field, select **Local Only**.
 - In the **Format Template** field, select **Get Fingerprint**.
 - In the **Vendor code** field, click the browse button. In the VendorCodes directory, select the DEMOMA Vendor Code file.
 - Click **Execute**. The information for the local computer is displayed on the **Output** tabbed page.

- e. On the **Output** tabbed page, click **Save As**. Save the C2V file. This file must be transmitted to the vendor.

Defining the License for the End User

To define a perpetual license

1. Start Sentinel LDK ToolBox (if it is not already active) and click **Work Offline**. The Sentinel LDK ToolBox main window is displayed.
2. At the bottom of the **Function** pane, click **License Generation API**. The functions for the License Generation API are displayed.
3. In the **Function** pane, select **sntl_ig_initialize**. In the **Display** pane, click **Execute**.
4. In the **Function** pane, select **sntl_ig_start**.
5. In the **Display** pane:
 - a. For **License type**, select **SNTL_LG_LICENSE_TYPE_UPDATE**.
 - b. For **Vendor code**, click the browse button. In the VendorCodes directory, select the DEMOMA Vendor Code file.
 - c. For **Current State**, click the browse button. Navigate to the C2V file you received from the end user.
 - d. Click **New Definition**.
6. In the Edit Template dialog box:
 - a. For **Locking Type**, select **SL-AdminMode**.
 - b. In the **Product** pane, click **New**.
7. In the Edit Product dialog box:
 - a. Set the **Product ID** to 3 and the **Product Name** to MyNotepad_Full.
 - b. In the **Feature** pane, click **New**.
8. In the Edit Feature dialog box, set the **Feature ID** to 10, **Feature Name** to MyNotepad, and **License Type** to Perpetual. Click **OK**.
9. In the Edit Product dialog box, click **OK**.
10. In the Edit Template dialog box:
 - In the **Memory Segment** pane, click **New**.
11. In the Edit Memory Segment dialog box:
 - a. Set **Memory Type** to **HASP_FILEID_RW**, **Offset** to 0 and **Length** to 16.
 - b. Erase the sample text on the right side of the pane. Type in the text: Hello World
 - c. Click **OK**.

12. In the Edit Template dialog box, click **OK**.

ToolBox modifies the XML code in the **License Definition** field according to the information you just entered. The license definition is now configured to generate the MyNotepad_Full Product.

13. Click **Execute**. ToolBox executes the `sntl_ig_start` function and displays the results.

Make sure that the **Status** field contains 0. This indicates that the license definition is valid and that the `sntl_ig_start` function completed successfully.

14. In the **Function** pane, select `sntl_ig_generate_license`.

15. Click **Execute**. ToolBox generates the license code.

Make sure that the **Status** field contains 0. This indicates that the license generation function completed successfully.

16. Under the **License** field, click **Save As**. In the Save As dialog box, type the file name MyNotepadFull and click **Save**. The perpetual license for the MyNotepad_Full Product is saved as **MyNotepadFull.v2c**.

17. In the **Function** pane, select `sntl_ig_cleanup`, and then click **Execute**.

Activating a Product With a License Update

Several methods are available to deliver the license update to the end user and activate the Product. For this tutorial, we will use Sentinel Admin Control Center to activate the Product.

In this section, in the role of the end user, you assume that you have purchased the MyNotepad_Full program and received a License Update (V2C) file from the vendor. You use this file to convert the trial version of the software to a full-featured program that is locked to your machine.

To simulate activation of the program

1. From the **Start** menu, open **Sentinel LDK**. The Sentinel LDK Launcher is displayed.
2. From the navigation pane, click **Additional Tools > Admin Control Center**. Sentinel Admin Control Center opens.
3. In the **Options** pane, select **Update/Attach**.
4. In the Update/Attach screen, click **Browse** and navigate to the **MyNotepadFull.v2c** file provided by the vendor. Select the file and click **Open**.
5. In the Update/Attach screen, click **Apply File**. A confirmation message is displayed.
6. In the **Options** pane, select **Features**. You can see that you now have a **Perpetual** license for Feature ID 10.

LESSON 7: Creating and Distributing a Sentinel HL-protected Software Order

Objective:	<ul style="list-style-type: none">> Learn how to define Products to be locked to Sentinel HL keys, and to burn the keys> Simulate a variety of end-user interactions with programs protected by Sentinel protection keys> Learn how to update a license in the field
Roles in this lesson:	<ul style="list-style-type: none">> Entitlement Manager> Production> End User
Applications used:	<ul style="list-style-type: none">> Sentinel LDK ToolBox (License Generation API)> Sentinel LDK ToolBox (Licensing API)> Sentinel Admin Control Center

In this lesson, we assume that you have received two different orders, which you will produce so that they are ready for shipment. You also assume the end-user role to test different types of Sentinel protection keys and licenses.

When a customer wants to purchase additional Features, or to extend the current license for your protected program, you can update the deployed license without having to recall or redeploy keys.

In this lesson, you will also learn how to update the deployed license.

The orders that you will process are based on the Products that you defined in ["Defining Features and Products" on page 12](#), and include:

- > An order for the "Bounce_Lite" Product.
- > An order to upgrade the "Bounce_Lite" Product to the "Bounce_Full" Product. This order will be used to update the license on the Sentinel HL key provided to the end user with the "Bounce_Lite" Product. You will learn how to convert its functionality from Bounce_Lite to Bounce_Full in the field.

When you defined the Products, you specified the license terms for the Features included within those Products at that time. Note that instead of specifying the license terms when you defined the Product, you could have enabled the functionality for specifying the license terms at this stage as part of order processing.

Defining a “Rental” Product Order With Sentinel HL Locking

In this section, you define a license for the “Bounce_Lite” Product, and lock the license to a Sentinel HL key.

To prepare the Bounce_Lite license

1. Connect to your computer the Demo Sentinel HL key to which you want to burn the order.
2. Start Sentinel LDK ToolBox.
3. In the API panel, click **Licensing API**; then select the **hasp_get_info** function.
4. In the **Display** pane:
 - a. In the **Scope Template** field, select **Local HASP HL Only**.
 - b. In the **Format Template** field, select **Create C2V**.
 - c. In the **Vendor code** field, click the browse button. In the VendorCodes directory, select the DEMOMA Vendor Code file.
 - d. Click **Execute**. The information for the connected HL key is displayed on the **Output** tabbed page.
 - e. Note that one of the XML tags near the top of the page is this:


```
<hasp id="[protection_key_id]">
```

This number is the unique identifier for the Sentinel HL key that is currently attached to the computer. Save this number in a temporary location - we will use it later in this procedure.
 - f. On the **Output** tabbed page, click **Save As**. Save the C2V file to the default location.
5. In the API panel, click **License Generation API**. The functions for the License Generation API are displayed.
6. In the **Function** pane, select **sntl_ig_initialize**. In the **Display** pane, click **Execute**. The session handle is displayed in the **Handles** pane.
7. In the **Function** pane, select **sntl_ig_start**. You will use this function to define the license.
8. In the **Display** pane:
 - a. For **License type**, select **SNTL_LG_LICENSE_TYPE_UPDATE**.
 - b. For **Vendor code**, your Vendor Code file should appear by default.
 - c. For **Current State**, click the browse button. Select the C2V file that you saved in step 4f above.
 - d. Click **New Definition**. Use the information from our ERP system ("[ERP System - Features and Products](#)" on page 14) to define the **Bounce_Lite** Product.
9. Click **Execute**. ToolBox executes the **sntl_ig_start** function and displays the results. Make sure that the **Status** field contains 0.
10. In the **Function** pane, select **sntl_ig_generate_license**.
11. Click **Execute**. ToolBox generates the license code. Make sure that the **Status** field contains 0.

12. Under the **License** field, click **Save As**. In the Save As dialog box, save the license as `Bounce_Lite.v2c` on the desktop.
13. Under the **Resultant State** field, click **Save As**. In the Save As dialog box, create a folder called `Key_States` on your desktop. Save the state file as `keyState_[key_id].xml` in this folder. `key_id` is the identifier of the protection key ID that you recorded in step 4e above. (For this tutorial, you can use just the last 2 digits of the key ID.)

NOTE In a production environment, you should record the protection key ID and save the state file in your ERP system as part of the information for the customer.

The state file contains the state of the end user's protection key that will exist after the V2C file is applied to the protection key. You need this file for future updates to the end user's protection key. If you do not save this file now, then in the future, you must obtain the state of the end user's protection key before you can send any protection key updates to the customer.

14. In the **Function** pane, select `sntl_ig_cleanup`, and then click **Execute**.

To burn the `Bounce_Lite` license to the connected Sentinel HL key

1. In the ToolBox API panel, click **Licensing API**; then select the `hasp_update` function.
2. In the **Update Data** field, click the browse button. Navigate to the `Bounce_Lite.v2c` file that you saved in the previous procedure. Select the file and click **Open**. The full path of the file appears in the **Update Data** field.
3. Click **Execute**. The license is burned to the HL key. (Make sure that the **Status** field contains 0.)

Using an Application With a Sentinel HL Key

When a customer receives an application you protected with a Sentinel HL key, the actual key must be accessible by the application in order to run the program. In the following procedure, you assume the end-user role to test Bounce, with a `Bounce_Lite` license locked to a Sentinel HL key.

In this part of the lesson, you use Sentinel Admin Control Center to examine your licensed software. In a true production environment, a system administrator in your customer's organization uses this Web-based application to manage network licenses for your software.

To operate a Product protected with a Sentinel HL key

1. Connect the Demo key on which you burned the order to your computer. The required licensing information for `Bounce_Lite` is contained, in encrypted format, in the key. As long as the license is valid, the protected application can run.
2. (Optional) Examine the license on the HL key as follows:
 - a. From the **Start** menu, open **Sentinel LDK**. In the Sentinel LDK Launcher, click **Additional Tools > Admin Control Center**. Sentinel Admin Control Center is displayed. This interface enables customers to

track the status and usage of their Sentinel licenses.

- b.** In the **Options** pane, select **Sentinel Keys**. All the Sentinel protection keys that are accessible on the Sentinel License Manager are displayed.

#	Location	Vendor	Key ID	Key Type	Version	Sessions	Actions
0	Local	DEMOMA	1133728459601576022	SL Admin-mode	2.00	-	Products Features Sessions Certificates
1	Local	DEMOMA	1527760096	HL NetTime 10	3.25	-	Products Features Sessions Blink on

- c.** Locate the entry for your Demo key and click the **Features** button in the **Actions** column. The Features that are licensed in your key are displayed.

Note that the **Restrictions** field for Feature ID 50 displays **Executions 3 left**, indicating that the remaining number of logins permitted by the license is **3**.

- 3.** Navigate to the following folder:

```
%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\Protected
```

- 4.** Double-click **Win32_Bounce.exe**. (A message is displayed, warning you that the program is only protected with a demo Sentinel key. Click **OK**.) The Bouncing Ball program launches.
- 5.** Close the Bounce software.
- 6.** Repeat step **4** and step **5** two more times, to consume all the permitted executions.
- 7.** Repeat step **4**. Bounce fails to open and a message is displayed indicating that the Feature has expired.
- 8.** In Sentinel Admin Control Center, in the **Options** pane, select **Sentinel Keys**.
- 9.** Locate the entry for your Demo key and click the **Features** button in the **Actions** column.

TIP If you are not sure which entry relates to your specific key, click the **Blink on** button and identify the key on which the LED is blinking.

Note that the **Restrictions** field for Feature ID 50 displays **Expired**, meaning that the number of allowed logins defined in the license has been exhausted.

Creating an Update for a License in the Field

Your customer requested a license upgrade—from Bounce_Lite to Bounce_Full. In this part of the lesson, you will generate an update to the HL key that is located on the end user's computer. You will then assume the role of the end user to activate the license update.

To produce an update license for the customer's HL key

1. Start Sentinel LDK ToolBox.
2. In the API panel, click **License Generation API**. The functions for the License Generation API are displayed.

3. In the **Function** pane, select **sntl_ig_initialize**. In the **Display** pane, click **Execute**. The session handle is displayed in the **Handles** pane.
4. In the **Function** pane, select **sntl_ig_start**. You will use this function to define the update license.
5. In the **Display** pane:
 - a. For **License type**, select **SNTL_LG_LICENSE_TYPE_UPDATE**.
 - b. For **Vendor code**, your Vendor Code file should appear by default.
 - c. For **Current State**, click the browse button. Navigate to the file **keyState_[key_id].xml** that you saved in step 13 in "[Defining a "Rental" Product Order With Sentinel HL Locking](#)" on page 33. This file contains the current state of the customer's HL key.
 - d. Click **New Definition**. Use the information from our ERP system ("[ERP System - Features and Products](#)" on page 14) to define the **Bounce_Full** Product.
6. Click **Execute**. ToolBox executes the **sntl_ig_start** function and displays the results. Make sure that the **Status** field contains 0.
7. In the **Function** pane, select **sntl_ig_generate_license**.
8. Click **Execute**. ToolBox generates the license code. Make sure that the **Status** field contains 0.
9. Under the **License** field, click **Save As**. In the Save As dialog box, save the license as **Bounce_Full.v2c** in the path:

%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\v2c

NOTE In a true production environment, you would also save the content of the **Resultant State** field as **keyState_[key_id].xml** in your ERP system in place of the key state file you saved earlier. You need this file for future updates to the protection key. If you do not save this file now, then in the future, you must obtain the state of the end user's protection key before you can send any protection key updates to the customer.

10. In the **Function** pane, select **sntl_ig_cleanup**, and then click **Execute**.

Updating a License at the Customer's Site

In this section, you retain the role of the end user. You have purchased the full-featured version (**Bounce_Full**) of the program to replace the execution-based **Bounce_Lite** version. You update your license using the Sentinel Admin Control Center and the **Bounce_Full.v2c** file that you receive from the vendor.

To apply a license update

1. Ensure that the Demo key is connected to your computer.
2. From the **Start** menu, open **Sentinel LDK**. In the Sentinel LDK Launcher, click **Additional Tools > Admin Control Center**. Sentinel Admin Control Center is displayed.

3. In the **Options** pane, select **Update/Attach**. The Update/Attach License window is displayed.
4. Click **Browse** and navigate to the following folder:
`%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\v2c`
5. When producing the order for the license update in your Entitlement Manager role, select the Bounce_Full.v2c file that you created in step 9 in "[Creating an Update for a License in the Field](#)" on page 35.
6. Click **Open**. The file path is displayed in the **Select a V2C... file** field.
7. Click **Apply File**. A message is displayed confirming that your update was applied successfully.
8. In the **Options** pane, select **Sentinel Keys**.
9. Locate the entry for your Demo key and click the **Features** button. The Features that are licensed in your key are displayed.

Note that an additional row was created for Feature ID 50. The **Restrictions** field in this row is **Perpetual**, indicating that the remaining number of logins permitted by the license is unlimited.

10. Navigate to the following folder:

`%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorTools\VendorSuite\Protected`

11. Double-click **Win32_Bounce.exe**. A message is displayed to warn you that the program is only protected with a demo Sentinel key. Click **OK**.

The Bouncing Ball program launches.

12. Close the Bouncing Ball program and disconnect the Demo key.

NOTE This lesson demonstrates using Sentinel Admin Control Center to examine licenses and to update a license at the customer's site.

Sentinel Admin Control Center is part of the Sentinel LDK Run-time Environment. The Run-time Environment was placed on your computer when you installed Sentinel LDK Vendor Tools. However, the Run-time Environment is not required for Sentinel HL Driverless keys.

If you have not included the Run-time Environment with your protected application, you can update licenses at the customer's site using other methods. For more information, see the description of the Sentinel Remote Update System in the [Sentinel LDK Software Protection and Licensing Guide](#).

LESSON 8: Using Sentinel Licensing API

Objective	Learn how to implement Sentinel Licensing API functions using Sentinel LDK ToolBox
Role in this lesson	Development
Applications used	Sentinel LDK ToolBox

Now that you have completed the other lessons in this tutorial, you should have a good understanding of how Sentinel LDK works and how to operate Sentinel LDK ToolBox.

This lesson builds on your accumulated knowledge and demonstrates more advanced techniques to work with Sentinel LDK Licensing API through Sentinel LDK ToolBox.

To perform this lesson, launch Sentinel LDK ToolBox and proceed through the tasks that follow.

NOTE Before you perform this lesson, make sure that you have completed "[Using Sentinel LDK ToolBox](#)" on page 15.

Starting With Login

Each Sentinel Licensing API session typically begins with the `hasp_login` API call, which establishes a connection with a Sentinel protection key and enables access to its resources. For this lesson, you will connect the Demo key.

NOTE `hasp_get_info()` and `hasp_update()` are standalone functions. You do not have to log in before you call these functions.

To log in to the Demo key

1. Connect the Demo key.
2. In the **Function** pane, select **hasp_login**. The **Display** pane displays all the parameters with the default values that are appropriate for the login operation.
3. Specify the location of your stored Vendor Code file.

To locate your Vendor Code file, click the browse button.

By default, the DEMOMA Vendor Code is stored in:

```
%HomePath%\Documents\Thales\Sentinel LDK 10.0\VendorCodes
```

4. Select **DEMOMA.hvc** and click **Open**. The code is displayed in the **Vendor Code** field.
5. In the **Feature ID** field, type `50`.
You must specify a Feature ID that exists in the protection key. If you did not define Feature ID 50 earlier in the tutorial, select **0** in the **Feature ID** field.
6. At the bottom left of the **hasp_login** pane, click **Execute**. The result of the executed login is summarized to the right of the **Execute** button in three **Status** fields—number, code and message.

Encrypting Data and Writing to Memory

Now that you have established a session with the connected Sentinel protection key, you can access its resources through other API calls.

One of the most important and powerful resources available in a Sentinel protection key is its internal encryption engine. You can optimize protection by using this engine and the encryption facilities in Sentinel LDK as an additional method for verifying that the correct key is accessible when the protected program is running. To do this, proceed as follows:

- > Encrypt a string using the Sentinel protection key
- > Incorporate the encrypted string into the application source code
- > Define your source code to send the encrypted string to the key for decryption and then to verify the result

In its encrypted form, the string prevents the program from running correctly. The protected program will run only if it can decrypt the string, and this can only be achieved if the Sentinel protection key is accessible. This section demonstrates how the key encrypts and decrypts a string.

To encrypt a string

1. In the **Function** pane, select **hasp_encrypt**. The **Display** pane displays parameters for the **hasp_encrypt** API call.
2. Specify the **Size** as **100**.
3. Click the first dot to the right of the memory editor, on the far right of the window, and type: `Sentinel LDK`.
4. Click **Execute**. Note that the string has been encrypted by the Sentinel protection key and is now illegible.

NOTE Sentinel LDK uses a unique encryption key for each encryption process, based on your unique Sentinel Vendor keys and the current Feature. Therefore, the string you type is encrypted differently from any other software vendor (and differently for each Feature).

5. In the **Function** pane, select **hasp_decrypt**. The editor still displays the encrypted string.

6. Click **Execute**. The Sentinel protection key decrypts the string. **Sentinel LDK** is now displayed in the editor.

Retrieving Sentinel Protection Key Information

Sentinel LDK ToolBox enables you to retrieve key-related information. In this section, you use the **hasp_get_sessioninfo** function to retrieve information relating to the identity and memory size of the Sentinel protection key.

To retrieve Sentinel protection key information

1. In the **Function** pane, under **Management**, select **hasp_get_sessioninfo**. The **Display** pane displays parameters for the **hasp_get_sessioninfo** API call.
2. In the **Display** pane, from the **Format Template** list, select **HASP_KEYINFO**. Note that a description of the Format definition is displayed in the gray box below the **Information** field.
3. Click **Execute**. The **Information** pane displays XML tags with information on the key, including:
 - The unique ID number of the connected key
 - The key Read/Write memory size
 - The key Read-only memory size

Reading and Programming Sentinel Protection Key Memory

In this section, you log in to Feature ID 10 on the Sentinel SL key that you defined in ["Defining an Unlocked Trialware Product" on page 23](#). You write to and then read from the memory of the protection key.

To log in to the Sentinel protection key

- > Follow the Demo key login procedure on ["To log in to the Demo key" on page 38](#), but this time, in step 5, type 10.

To program the Sentinel protection key memory

1. In the **Function** pane, under **Memory**, select **hasp_write**. The **Display** pane displays parameters for the **hasp_write** API call and an editor for the Sentinel protection key memory.
2. In the memory editor, set the **Offset** field to **16** and the **Length** field to **24**.
3. On the right side of the pane, click the first "." and type:


```
The World says Hello
```
4. Click **Execute**. The string is stored in the protection key memory.

To read the Sentinel protection key memory

1. In the **Function** pane, under **Memory**, select . The **Display** pane displays parameters for the **hasp_read** API call and an editor for the Sentinel protection key memory.
2. In the memory editor, set the **Offset** field to **0** and the **Length** field to **40**.
3. Click **Execute**. The memory editor now displays:
 - the string **Hello World**, which you stored in the key in "[Creating and Distributing a Sentinel SL-protected Software Order](#)" on page 27.
 - the string that you typed in the procedure above, in step 3.

What's Next?

This concludes the Sentinel LDK API tutorial. You are now ready to use Sentinel LDK applications to license your software and protect it against unauthorized use.

For additional information about the Sentinel LDK system, refer to the [Sentinel LDK Software Protection and Licensing Guide](#) and to the Help documentation available in the Sentinel LDK applications.