



Enabling product operation

Contents

Enabling product operation 1

Product enablement checklist	1
Software-based license	1
USB hardware device	2
Software-based licensing	3
Software-based licensing terminology	3
Overview of the activation process	5
The license manager	8
Activating a license manager	11
Activating a license manager with Rational Tokens	13
Returning a software-based license key less than 31 days before expiration	14
Returning a software-based license key more than 30 days before expiration	15
Activating and configuring a license manager client	17
The USB hardware device	18
USB hardware device licensing terminology	19

State of the hardware device.	21
Capacity of the hardware device	21
Serial number of the USB hardware device	22
Expiration dates	22
Capability configuration of the USB hardware device	25
License keys and the Application Developer Controlled Distribution (ADCD)	25
Parallel Sysplex Activations	25
Rational Token Activations and the USB hardware device.	26
Planning for USB hardware devices	27
Obtaining an update file	27
Activating the USB hardware device	33
Quick setup instructions for using and migrating the product license server	37
Next steps.	40

Index 43

Enabling product operation

You need an activated license manager or activated USB hardware device to operate the product and authorize the licensee to use z Systems™ Development and Test Environment or any of its components.

The method of authorization you use depends on the version of the product that you have.

- For Rational® Development and Test Environment for z Systems v9.5 and earlier, you must use an activated USB hardware device.
- For z Systems Development and Test Environment v10 and later, the method depends on the type of license entitlement:
 - For z Systems Development and Test Environment Personal Edition v10 (Authorized User Single Install), you must use an activated USB hardware device.
 - For Resource Value Unit (RVU) and Rational Token software-based license entitlements, you must use an activated license manager.

Product enablement checklist

Use these checklists to enable product operation with a license manager or a USB hardware device.

Software-based license

About this task

If you purchased z Systems Development and Test Environment with Resource Value Unit (RVU) or Rational Token entitlement, you must use software-based licensing. Use this checklist to activate the product by using software-based licensing. These steps link to other topics for more information.

Procedure

1. Install the z Systems Development and Test Environment license manager feature on the computer that is to act as the license manager. For step-by-step instructions, see “Installing and starting the license manager” on page 8.
2. Optional: Install the z Systems Development and Test Environment license manager feature on other computers for failover. For more information, see “License manager failover” on page 10.
3. Install z Systems Development and Test Environment on computers that meet the System Requirements and use software-based licensing. For detailed installation and configuration instructions, see Installing z System Development and Test Environment.
4. Obtain a license key file. For software-based licenses, the license key file is a license manager update file. Use one of these procedures for your license metric to activate your license manager:
 - “Activating a license manager” on page 11
 - “Activating a license manager with Rational Tokens” on page 13

5. If you purchased entitlement that uses Rational Tokens, you must set up the Rational License Key Server and obtain the corresponding license from the Rational License Key Center. See Setting up Rational Tokens.
6. If the license manager is not on the same subnet, configure each computer that is hosting the offering to use the license manager that you set up in step 1. See “Activating and configuring a license manager client” on page 17.
7. On the computers that host the mainframe applications that run with the emulator, create the device maps used to define your emulated systems:
 - If you purchased entitlement to z Systems Development and Test Environment with Parallel Sysplex[®], include a line to enable the coupling facility.
 - If you purchased entitlement that uses Rational Tokens, include a line to identify the Rational License Key Server to be used. As an alternative, you can set the RDTSERVER Linux environment variable.
8. To install z/OS[®] on any host computers, follow the instructions in Installing z/OS volume images.

USB hardware device

About this task

If you purchased z Systems Development and Test Environment Authorized User Single Install, or any version before z Systems Development and Test Environment v10.0, use this checklist to activate the product by using one or more USB hardware devices. The process for activating the product is similar for all versions of z Systems Development and Test Environment before v10.0.

Procedure

1. Purchase the z Systems Development and Test Environment offering and license metric that best suits your needs.
2. Obtain one or more USB hardware devices by ordering the appropriate media pack through Passport Advantage[®].

Note: USB hardware devices are not tied to a particular release of the offering. If you have a USB hardware device from a different release, renew the activation to obtain the update file that is needed to install the z/OS[®] 2.1 or later Application Developers Controlled Distribution for z Systems Development and Test Environment.

3. Obtain an update file to activate the USB hardware device from the Rational License Key Center. If you purchased entitlement that uses Rational Tokens, you must set up the Rational License Key Server and obtain the corresponding license from the Rational License Key Center. See “Obtaining an update file” on page 27.
4. Optional: Set up a product license server:
 - a. Install z Systems Development and Test Environment on the license server.
 - b. Connect the USB hardware device to the license server.
 - c. Apply the update file to the USB hardware device attached to the license server.
 - d. Start the product license server.

For more information about setting up a product license server, see “Setting up the product license server” on page 37.

5. Install z Systems Development and Test Environment on computers that meet the System Requirements. For detailed installation and configuration instructions, see Installing z System Development and Test Environment.
6. If you are using a product license server, configure each computer that is hosting the offering to use the product license server you set up in step 4. See “Setting up computers to access the product license server” on page 38.
7. If you are not using a product license server, apply the corresponding update file to the locally attached USB hardware device of each computer that is hosting the offering. See “Activating the USB hardware device” on page 33.
8. On the computers that host the mainframe applications that run with the emulator, create the device maps used to define your emulated systems:
 - If you purchased entitlement to z Systems Development and Test Environment with Parallel Sysplex, include a line to enable the coupling facility.
 - If you purchased entitlement that uses Rational Tokens, include a line to identify the Rational License Key Server to be used. As an alternative, you can set the RDTSERVER Linux environment variable.

For more information about creating device maps, see Defining the device map.

9. To install z/OS on any host computers, follow the instructions in Installing z/OS volume images. You can now use the emulator.

Software-based licensing

When you use software-based licensing, a license manager provides a server for centralized management of license keys for one or more instances of z Systems Development and Test Environment. With this method, a USB hardware device is not required.

The z Systems Development and Test Environment license manager is installed on a machine, and the authenticated license key files are installed on the license manager. The licensee is not authorized to use z Systems Development and Test Environment or any of its components except when that license manager is activated with a license key file and is accessible by the program, as it provides proof of license entitlement.

The license manager and an instance that runs z Systems Development and Test Environment cannot run on the same platform, but it does need network connectivity to the license manager.

Note: License manager needs to be a static resource in any infrastructure configuration. In a virtualized or cloud infrastructure, it does not tolerate physical moving. If the license manager is manually or automatically moved inside the infrastructure, it will stop working. If you would like to move the License Manager, you need to return the license before moving, acquire the license after movement, and configure the License manager again.

Software-based licensing terminology

Understand the terms that are used to describe the activation process with software-based licensing.

The words *license* and *token* have more than one meaning, which depends on the context in which they are used.

You can encounter at least two different associated meanings for the word *license* as you work with z Systems Development and Test Environment:

- The word *license* can mean the product entitlement that you purchased and the corresponding terms and conditions under which use is allowed.
- The Rational License Key Center uses the word *license* and *licenses* for a quantity of product entitlement parts purchased.

To clarify the intended use of the term, the z Systems Development and Test Environment documentation uses these phrases:

- The phrase *product entitlement* refers to the terms and conditions under which the product can be run by the licensee.
- The phrase *license entitlement* refers to the Rational License Key Center interface and the quantity of entitlement parts purchased.
- The phrase *license key file* refers to a file that represents the user's license entitlement. License key files must be obtained, installed on the license manager, and available to operate z Systems Development and Test Environment.
- The phrase *license manager update file* refers to the specific license key file that is generated in the Rational License Key Center and installed on a license manager by using an **update_license** command to activate it. This file is used for software-based licensing.
- The phrase *license manager activation* refers to the condition of the license manager and its readiness for operation by clients that run the emulator.

You can encounter at least two different associated meanings for the phrase *license server* as you work with z Systems Development and Test Environment:

- The phrase *license server*, *software-based license server*, or *license manager* can mean the License Development Kit–Software License (LDK–SL) server that distributes software-based license keys without the need for a USB hardware device. This server is also referred to as a license manager.
- The same phrase *license server* can mean the Rational License Key Server that is used to share license entitlements among several IBM® products.

To clarify these terms, the z Systems Development and Test Environment documentation uses these phrases:

- The phrase *license manager* refers to the remote z Systems Development and Test Environment that distributes authentication by using software-based license keys.
- The phrase *Rational License Key Server* refers to the license server used exclusively to share Rational Token license entitlements among a set of IBM products.

The word *token* is used to refer to a type of IBM Rational product entitlement that allows great flexibility in the deployment and use of associated, purchased products. The z Systems Development and Test Environment documentation uses the phrase *Rational Token* to refer to IBM Rational Token licensing.

The word *instance* refers to a single installation of z Systems Development and Test Environment on a physical or virtual machine.

z Systems Development and Test Environment can be purchased with several different product entitlements. The various types of entitlements authorize use of the product in specific ways.

For details on the product entitlement you purchased, see the documentation that is provided with your copy of the offering or refer to the Product License available at the IBM Software License Agreement website.

The Product License can be found by searching the site for the product name.

Overview of the activation process

When you use software-based licensing, z Systems Development and Test Environment requires access to a license manager that is activated with a software-based license key file to control the licensee's access to all or portions of the program. The license manager, sometimes referred to as a software-based license server, manages all software-based license key files. The licensee is not authorized to use z Systems Development and Test Environment or any of its components except when that license manager is activated with a license key file and is accessible by the program, as it provides proof of license entitlement. License key files are obtained from the Rational License Key Center, and are specific to a customer account.

Each license key file is generated with a number of entitled emulated central processors, referred to as *Central Processors* or *CPs*. A single instance of z Systems Development and Test Environment requires at least 1 CP, and can run with a maximum of eight CPs. The number of CPs needed depends on the number of users and the types and amount of processing required. For recommendations on how many CPs to use with an instance, see section 2.3.4 "Performance" and section 13.12, "CPs, processors, threads, and tokens" in the zPDT® Guide and Reference.

Note: z Systems Development and Test Environment Authorized User Single Install, and all releases before v10 that do not include the Parallel Sysplex offering can run with a maximum of 3 CPs.

The software-based license key file that is generated by the Rational License Key Center is called a license manager update file. It is created by the Rational License Key Center for a specific license manager, and only that license manager can be activated with that license manager update file. A license manager is activated with only one license manager update file. To change the number of entitled emulated CPs on a license manager, or to renew a software-based license key that is currently installed on a license manager, you must first return the license key and then generate a new one.

A single activated license manager can authenticate many instances of z Systems Development and Test Environment. The number of instances that can be authenticated varies based on the number of emulated Central Processors or CPs used in each instance.

When an instance of the offering starts, it seeks an activation from the license manager for the number of CPs specified in the emulator's device mapping file. When an instance of the emulator is stopped, the CPs authenticated by the license manager become available for another instance. The activated license manager must remain available to the instance while the instance is running.

Software-based license key files are generated and license managers are activated with that license key file by using a multi-step process:

1. Create a license manager request file on the license manager on which the license key is to be installed.

2. Obtain from the Rational License Key Center a license manager update file unique to that server to provide proof of license entitlement.
3. Activate the license manager by installing the license manager update file.
4. If the license manager is not on the same subnet as all z Systems Development and Test Environment instances that use that manager for their license entitlement, configure network access to the license manager from any of those instances not on the same subnet. Any instance that uses a license manager is known as a license manager client.

If you purchase entitlement to z Systems Development and Test Environment through Rational tokens and software-based licensing, the offering also communicates with a Rational License Key Server to share license entitlements among several IBM products. For more information about shared license entitlements, see “Rational Token activation” on page 7 and “Activating a license manager with Rational Tokens” on page 13.

Note: License manager needs to be a static resource in any infrastructure configuration. In a virtualized or cloud infrastructure, it does not tolerate physical moving. If the license manager is manually or automatically moved inside the infrastructure, it will stop working. If you would like to move the License Manager, you need to return the license before moving, acquire the license after movement, and configure the License manager again.

License key expiration

All license key files expire after one year from the date they were generated, or after less than a year if entitlement is from a term license that expires in less than a year. If your purchase entitles you to use the product for more than one year, you need to renew the license key each year. Renewal activates the license key for a subsequent year or until the end of the fixed term if less than a year.

When z Systems Development and Test Environment is running, you can check the expiration date of a license key file by using the **token** command. You can run the token command only while the emulator is running, and you must run the command from the same user ID that started the emulator.

License keys and the Application Developer Controlled Distribution (ADCD)

z/OS 1.13 ADCD and earlier distributions can be installed without access to an activated license manager. All volumes are supplied as files in compressed (.gzip) format, and the **gunzip** command makes them usable by z Systems Development and Test Environment.

For software-based entitlement, beginning with the z/OS 2.1 ADCD for Rational Development and Test Environment for z Systems, an activated license manager is required to install the system resident volumes of the z/OS ADCD. The **Z1091_ADCD_install** command is used to install these system resident volumes. This command requires access to an activated license manager with connectivity to the instance that does the installation. The same activated license manager that entitles the user to run z Systems Development and Test Environment also entitles the user to install a z/OS ADCD.

For instructions on installing the z/OS ADCD, see Installing May 2017 Edition z/OS 2.2 ADCD for z Systems Development and Test Environment

Rational Token activation

You can purchase entitlement to z Systems Development and Test Environment through Rational Tokens. When you use Rational Tokens, entitlement can be shared among several IBM products, with license entitlements tracked by temporary use of a number of Rational Tokens. For z Systems Development and Test Environment, each emulated central processor within an instance of a z Systems virtual machine requires a particular number of Rational Tokens, based on the token value for z Systems Development and Test Environment. When you use Rational Tokens, each instance of the emulator coordinates with a Rational License Key Server instance and starts or continues to run only if sufficient Rational Tokens are available for the number of configured CPs defined for that instance.

For software-based entitlement, use of Rational Tokens does not replace the requirement for an activated license manager to be available to z Systems Development and Test Environment. Any instance that uses Rational Tokens must have access to an activated license manager, and must have access to the Rational License Key Server. That license key file must indicate that use of Rational Tokens is in effect. When z Systems Development and Test Environment is started, the emulator verifies that an activated license manager is found and enough Rational Tokens are available from the Rational License Key Server to run the emulator. Rational Tokens control product operation while enough CPs are available from the license manager.

For information on setting up and using the Rational License Key Server, see the associated documentation available on the offering's installation e-image.

z Systems Development and Test Environment with Parallel Sysplex

z Systems Development and Test Environment with Parallel Sysplex, in addition to providing the base environment that supports z/OS also includes IBM z™ Personal Development Tool support for virtual Coupling Facility and the z/VM® software distribution to enable virtual coupling of z/OS guests under z/VM. Entitlements with this offering that enable the coupling facility are referred to in the product documentation as Sysplex activations. The Parallel Sysplex offering is available only with Resource Value Unit (RVU) or Rational Token license entitlements.

A single instance of z Systems Development and Test Environment with Parallel Sysplex enabled runs z/VM, the coupling facility, and one or more instances of z/OS. The available CPs defined to the instance are shared among all systems that are running, and are defined in the *processor* setting of the *devmap*. Typically, more CPs are required to run a Parallel Sysplex offering. While in principle you can use one CP on a base PC with one processor, or *core*, to run a z/VM system with several Parallel Sysplex z/OS guests, this configuration is not practical and might result in various z/OS timeouts. Configure at least one CP for each z/OS guest on the z/VM system. A single instance of z Systems Development and Test Environment with Parallel Sysplex can be configured to use a maximum of 8 CPs. Chapter 21, “Sysplex” in the zPDT Guide and Reference and zPDT 2016 Sysplex Extensions give more information about Parallel Sysplex systems.

Although a coupling facility is not related to an individual processor in real z Systems hardware, every CP in a z Systems Development and Test Environment with Parallel Sysplex instance requires entitlement to a Parallel Sysplex CP activation, or RVU, to enable the coupling facility for the instance.

The license manager

Learn how to install, configure, and administer license managers.

Installing and starting the license manager

To use software-based licensing, you must install at least one instance of the z Systems Development and Test Environment license manager that manages all software-based license keys.

About this task

For the hardware requirements, software requirements, and steps to install a license manager, see *Installing z System Development and Test Environment*.

The installation process installs both the license manager function and the UIM server function, and starts the license manager if the appropriate glibc library is installed. The license manager is started as a service named aksusbd.

Procedure

- When you use software-based licensing, the UIM server must run on the license manager, and unlike the license manager daemon is not automatically started at installation. Start the UIM server from a non-root user ID by using this command:

```
uimserverstart
```

- **Firewall considerations:** The license manager listens for license key requests on port 1947. You must manage any firewalls that are involved with this server. When you initially install in a test environment, it is easier to diagnose errors if you are able to disable firewalls until you are satisfied with your software-based license operations. This action separates network-related problems from license manager issues. If you operate through firewalls, you must ensure that the relevant port numbers can pass through the firewalls. You can use many techniques to manage firewalls, depending on what product is being used. Many Linux systems respond to **iptables** commands, such as:

```
# iptables -I INPUT -p tcp --dport 1947 -j ACCEPT
```

Capability configuration and capacity of the license manager

Specific combinations of Parallel Sysplex support and use of Rational Tokens define the capability configuration of the license key and of the license manager. The Rational License Key Center does not combine different types of software-based license orders when you generate a software-based license key file. Any orders that are combined to create one license key file must be of the same license type. However, the number of CPs that you can generate for a software-based license key file is limited only by your number of entitlements. You can generate a license key file with any or all of your entitlements for the same capability configuration.

You can have only one installed license key file on a license manager. That license key file allows operating within only one configuration:

- Parallel Sysplex support is not enabled and you are not required to use Rational Tokens.
- Parallel Sysplex support is enabled and you are not required to use Rational Tokens.
- Parallel Sysplex support is not enabled and Rational Tokens are required.
- Parallel Sysplex support is enabled and Rational Tokens are required.

For other steps required to run if you are using Rational Tokens for your entitlement, see Setting up Rational Tokens.

To ensure that only one installed license key file is generated for a license manager, a license key file must be returned to the Rational License Key Center before it can be renewed or another file can be generated for the same server. For more information, see “Returning a software-based license key less than 31 days before expiration” on page 14.

Host ID and host name of the license manager

Each license key file is uniquely identified in the Rational License Key Center with the host name and host ID of the license manager for which the license was generated. For software-based licenses, the host name is the host name of the server, which can be displayed with the Linux **hostname** command. If the host name has periods, such as if it is an IPv4 address, the Rational License Key Center replaces the periods in the host name with underscore characters.

The host ID, which is required to be unique across all license key files in the RLKC, is a generated unique identifier for each license manager. The host ID is generated when the license key file is generated. If you are trying to find the license key file in the RLKC, by using either **Return Keys**, **View Keys by Host**, or **View Keys by User**, the host ID of the server is the field that uniquely correlates a license key file to the license manager for which it was generated.

Administering the license manager

The license manager for z Systems Development and Test Environment is controlled by line commands from a Linux command line on the server. The commands that are needed to administer the server are described in the topics on obtaining, returning, and renewing software-based license key files.

You can display information about the license key that is currently installed on the license manager from the root userid by entering this command:

```
/opt/IBM/LDK/query_license
```

Stopping and uninstalling the license manager

If you need to uninstall the license manager, either to move license managers or to discontinue use of a license manager, you must first return the activated license key.

Before you begin

For instructions for returning an active license key, see “Returning a software-based license key less than 31 days before expiration” on page 14

Procedure

1. To stop the license manager, enter **systemctl stop aksusbd.service**.
2. From a non-root user ID, enter **uimserverstop**.
3. Go to the directory where installation program is present and switch to root user id to run below program.
4. Execute the command **./zDT_Install.x86_64**.
5. Type **2** to select IBM z Systems Development and Test Environment License Manager.
6. Type **3** to uninstall the selected product.
7. Wait for the process to get finish.

8. To uninstall silently, execute the command `./zDT_Install.x86_64 --uninstall --zdtlicense`.

License manager failover

You can activate multiple license managers of the same capability configuration, and have one or more available for failover. Each instance of z Systems Development and Test Environment that uses software-based licensing can configure a list of managers. When an instance configures access to multiple managers, if both managers are active, entitlement is requested evenly between the managers. If either manager does not respond or has insufficient entitlement, either during initial activation or during product operation, the client instance attempts to authenticate entitlement through the other active managers in the list.

About this task

For more information about the steps to access a license manager, see “Activating and configuring a license manager client” on page 17.

Important: Since license key files are unique to each license manager, you cannot use one license key file on multiple license managers. Each license manager, including the ones that are used for failover, must have its own entitlements. For more information about configuring license managers, see “Planning for license managers.”

Changing license managers

If you need to move your license manager to a different platform, you must first return your license keys from the current license manager, and then generate and install license key files for each new license manager. If the original platform that hosts the license manager will not be used as a license manager in the future, you can optionally uninstall the z Systems Development and Test Environment license manager.

Do these steps:

1. Follow the steps in “Stopping and uninstalling the license manager” on page 9. Uninstalling the license manager is optional. If your current server is unusable, contact the Rational License Key Center to return your license key file.
2. For each new license manager, install and start the license manager by using the steps that are described in “Installing and starting the license manager” on page 8.
3. Generate a new license key file for each new license manager by using the steps that are described in “Activating a license manager” on page 11 or “Activating a license manager with Rational Tokens” on page 13.

Planning for license managers

You can spread your license entitlements across one or more license managers. Also, you can choose to have a license manager always active for failover if a failure on the primary license manager occurs. How many license managers are needed, and how those entitlements are activated on each license manager, is dependent upon many factors, such as

- How many instances you run,
- The access of the different users to the license managers,
- The types of license entitlements you use, and
- Having license managers for failover or not.

Consider the following rules when you are planning how to spread your license entitlements across your license managers, and how to later generate your license manager update files:

- Each license manager can have only one installed license key file at a time. That license key file can be of only one capability configuration. Therefore, you must have at least as many license managers as capability configurations you need. For example, if you have entitlement to z Systems Development and Test Environment and z Systems Development and Test Environment with Parallel Sysplex, you need at least two license managers installed.
- The number of CPs on a software-based license key is only limited by your entitlement. Therefore, if all license manager clients have access to a license manager for the required capability configuration, one license manager and one license manager for failover is sufficient.
- The license key file for all active license managers must have at least as many CPs as the combined number of processors that are needed for all concurrent instances that use the license manager. This number can be calculated by adding the number of processors that are defined in the devmap for each instance that can run concurrently.
- If you require a license manager to be available if failover occurs, have one for each capability configuration that is allowed by your entitlements.
- How many CPs to entitle each license manager when multiple license managers are activated to allow for failover can vary. If each license manager has sufficient CPs for the maximum number of concurrent users, operations will continue in the event of failure. You can spread your required entitlements evenly across license managers if you can limit users during the time one license manager fails. If a failure is not temporary, you can also generate a new license key file with more entitled CPs for one or more of the active license managers.
- You can combine orders (license entitlements) in one update file, but only if they are of the same license type

Note: The hostname of the server that is hosting an active license manager must not change while an active license key is installed. In order to change the hostname of the server, the license key must first be returned to the Rational License Key Center, then the server can be renamed, and, if you want to, you can request a new license key for the modified server.

Activating a license manager

To obtain a software-based license key file, you must create a license manager request file from the license manager where the license key file is to be installed. Then, you must go to the Rational License Key Center and install license key file on the license manager to activate it.

If you already activated a license manager with a license key file, and need to generate a new license key file either because the license key is expiring or you need to change the license entitlements that are applied to the license key, you must first return the existing license entitlement in the Rational License Key Center. For more information about returning a license key, see “Returning a software-based license key less than 31 days before expiration” on page 14 or “Returning a software-based license key more than 30 days before expiration” on page 15. After you return the license key, the procedures to generate a new one are the same as obtaining an initial license key file.

Procedure

After you install and start the license manager, do these steps to activate the license manager:

1. From the root user ID on the license manager, run this command:
opt/IBM/LDK/request_license. This command creates a request compressed file in the root home directory with a name similar to Hostname_xxxxxxxx.zip. This file is unique to the license manager, and is used to generate the software-based license key file.
2. If needed, transmit the Hostname_xxxxxxxx.zip file to the system you use to log in to the Rational License Key Center.
3. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
4. On the left side of the screen, select **Get Keys**.
5. Select the product line for z Systems Development and Test Environment. The interface displays a list of license key names that show the z Systems Development and Test Environment licenses that are available.
6. Select the z Systems Development and Test Environment license type that you want to apply to the key: IBM z Systems Development and Test Environment Resource Value Unit or IBM z Systems Development and Test Environment with Parallel Sysplex Resource Value Unit.
7. Select the check box next to all of the orders from which you would like to use licenses and click **Next**. The interface displays a table for you to complete.
8. Provide the license manager request file for the license manager for which the license key file is to be generated and the number of Emulated Central Processors you intend to enable on the manager. The total number of emulated CPs must not exceed your total number of Resource Value Unit (RVU) entitlements. For help with the table, click a column heading.
9. Select **Generate** at the bottom of the page. A window opens, prompting you to download the license manager update file.
10. To save the file, click **Download**. You can download the generated file either during this process or later. To download any files later, use the View Keys by Host link on the left side of the Rational License Key Center page.
11. Transmit the license manager update file to the license manager.
12. From the root user ID on the license manager, run this command:
/opt/IBM/LDK/update_license Hostname_xxxxxxxx_update.zip In this command, you must specify the absolute path to the Hostname_xxxx_update.zip file. This command installs the license key file on the server.
13. In the license manager, enter the command **./serverconfig** from path **/opt/IBM/LDK**. Then set up the following options:
 - a. Set **Server Access** to 'Y'. If this field is set to 'N', client or instance can not get the license from license manager.
 - b. Set **Enable Access Log** to 'Y'. If you would like to create log, enter the command **./display_gen2_acclog** in the path **/opt/IBM/LDK** once zOS is ipled. Then the log is displayed.

Alternatively, you can use serverconfig command via command line interface by entering command **/opt/IBM/LDK/./serverconfig_cli <parameter>**.

To check the list of possible parameter values, enter command **/serverconfig_cli** without parameter.

14. Optional: If you would like to deny or allow access to some instances or client machines, edit the file `/opt/IBM/LDK/rules.ini`, and execute the command `./serverconfig_cli -u` to load rules. To check all rules currently in use, enter command `./serverconfig_cli -r`.

Note: You cannot generate separate license key files for the same license manager at the same time. Before you generate a license key file you want to change for a particular manager, you must return the license key file that was previously created for that server. Returning license entitlements that are assigned to a manager also makes the corresponding active entitlements available for assignment to another license manager.

Activating a license manager with Rational Tokens

Since software-based license key files are unique to a license manager, to obtain a software-based license key file you must create a license manager request file from the license manager where the license key file is to be installed. Then, you must go to the Rational® License Key Center, use the license manager update file to generate a license key file for the license manager, and separately generate a Rational Token license file. Then, you must install the license key file on the license manager to activate it.

Note: You cannot generate separate license key files for the same license manager at the same time. Before you generate a license key file you want to change for a particular manager, you must return the license key that was previously created for that server. Returning license entitlements that are assigned to a manager also makes the corresponding active entitlements available for assignment to another license manager.

After you install and start the license manager, do these steps to activate it:

1. From the root user ID on the license manager, run this command:
opt/IBM/LDK/request_license. This command creates a request compressed file in the root home directory with a name similar to `Hostname_XXXXXXXXX.zip`. This file is unique to the license manager, and is used to generate the software-based license key file.
2. If needed, transmit the `Hostname_XXXXXXXXX.zip` file to the system you use to log in to the Rational License Key Center.
3. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
4. On the left side of the screen, select **Get Keys**.
5. Select the product line for IBM Rational Tokens.
6. The Select License Keys page displays one or more boxes for orders. Find the order that includes z Systems Development and Test Environment. Make a note of the part number for the z Systems Development and Test Environment product you want to activate. Select the order that contains the parts you noted and click **Next**.
7. Enter the required information to create the Rational Token license file for your host. Additional information for each field is available by clicking the descriptive link next to the input field. Fields with a red star are required fields.
8. Click **Generate** to generate the Rational Token license file. You are presented with a screen with the generated license information.
9. Click **Download License Keys** to download the Rational Token license file to your computer. This token license file is the file that is applied to your

Rational License Key Server. It is named `license.dat` by default. Additional information on setting up the Rational License Key Server and applying this file to the server can be found in the Rational License Key Server documentation that is provided in the Rational License Key Server media.

10. On the Select License Keys page, click **Generate Token License Keys for Software-based License Servers** to generate and download the license manager update file for your license manager.
11. A Required Information page opens and displays a table that you must complete. Provide the license manager request file for the license manager for which the license key file is to be generated. Use the **Catalog Item** list to confirm the part that you want to use to generate an update file. This confirmation is important if you have entitlement to both z Systems Development and Test Environment and z Systems Development and Test Environment with Parallel Sysplex. You cannot combine Token and Sysplex Token activations on a single software-based license key file. From the list, select the entitlement part number that corresponds to the activation you want on the specified license manager update file. For help with the table, click a column heading.
12. Select **Generate** at the bottom of the page. A window opens, prompting you to download the license manager update file.
13. To save the file, click **Download** for each file that was generated. You can download the generated file either during this process or later. To download any files later, use the **View Keys by Host** link on the left side of the Rational License Key Center page.
14. Transmit the license manager update file to the license manager.
15. From the root user ID on the license manager, run this command:
`/opt/IBM/LDK/update_license Hostname_xxxxxxxxxx_update.zip`. In this command, you must specify the absolute path to the `Hostname_xxxx_update.zip` file. This command installs the license key file on the server.
16. Restart the license server daemon to make the license key file active by entering `systemctl restart aksusbd.service`.

Related information:

Setting up Rational Tokens

Rational Token licensing is an entitlement that you can purchase and use to run z Systems Development and Test Environment. With Rational Tokens, z Systems Development and Test Environment maintains a connection to a Rational License Key Server and starts and continues to run only when sufficient Rational Tokens are available.

Returning a software-based license key less than 31 days before expiration

To generate a new license key file for a license manager, you must first return the existing license entitlement in the Rational License Key Center, and then generate a new one. Returning the license entitlements is a process of telling the Rational License Key Center that you are no longer using the license entitlements that you previously assigned to the license manager. You do not need to return the physical license manager update file to the Rational License Key Center.

You can return a license entitlement in the Rational License Key Center in several ways. The easiest method is to use the **View Keys by Host** link. That process is described here. You can also use the **Return Keys** link.

Do these steps to return a software-based license key that has less than 31 days until expiration.

1. From the root user ID on the license manager, run this command:
opt/IBM/LDK/request_license. This command creates a request compressed file in the root home directory with a name similar to `Hostname_XXXXXXXXX.zip`. This file is unique to the license manager, and is used to generate the software-based license key file.
2. If needed, transmit the `Hostname_XXXXXXXXX.zip` file to the system you use to log in to the Rational License Key Center.
3. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
4. On the left side of the screen, select **View Keys by Host**.
5. Select the **Hostname** corresponding to the license key you want to return.
6. A table is displayed with data for the **Hostname** selected. At the far right of the table, click the **Change** link.
7. The interface displays a list of the license keys for this license entitlement. Locate the license key of the license manager you are returning. Click **Browse**, and browse to and select the `Hostname_XXXXXXXXX.zip` file that you just created.
8. Click **Initiate Return**.
9. A message is displayed to confirm that the license entitlements were successfully returned.

Returning a software-based license key more than 30 days before expiration

To generate a new license key file for a license manager, you must first return the existing license entitlement in the Rational License Key Center, and then generate a new one. Returning the license entitlements is a process of telling the Rational License Key Center that you are no longer using the license entitlements that you previously assigned to the license manager. You do not need to return the physical license manager update file to the Rational License Key Center.

To return a software-based license key that has more than 30 days until expiration requires that you invalidate the license on the license manager as part of the return process. This process consists of these general steps:

- Create a license manager request file from the license manager where the license key file is to be installed.
- Obtain a license key file that invalidates the current license key from the Rational® License Key Center.
- Install the license key file that invalidates the current license key on the license manager.
- Use the update file to complete the return process in the Rational License Key Center.

You can return a license entitlement in the Rational License Key Center in several ways. The easiest method is to use the **View Keys by Host** link. That process is described here. You can also use the **Return Keys** link.

Do these steps to return a software-based license key that has more than 30 days until expiration.

1. Create a license manager request file from the license manager where the license key file is to be installed:

- a. From the root user ID on the license manager, run this command:
opt/IBM/LDK/request_license. This command creates a request compressed file in the root home directory with a name similar to Hostname_XXXXXXXXX.zip. This file is unique to the license manager, and is used to generate the software-based license key file.
 - b. If needed, transmit the Hostname_XXXXXXXXX.zip file to the system you use to log in to the Rational License Key Center.
2. Obtain a license key file that invalidates the current license key from the Rational® License Key Center.
 - a. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
 - b. On the left side of the screen, select **View Keys by Host**.
 - c. Select the **Hostname** corresponding to the license key you want to return.
 - d. A table is displayed with data for the **Hostname** selected. At the far right of the table, click the **Change** link.
 - e. The interface displays a list of license keys for this license entitlement. Locate the license key of the license manager you are returning. Click **Browse**, and browse to and select the Hostname_XXXXXXXXX.zip file that you just created.
 - f. Click **Initiate Return**.
 - g. A window opens, prompting you to download the license manager update file with the file to invalidate the license on the license manager.
 - h. To save the file, click **Download**. You can download the generated file either during this process or later. To download any files later, use the **View Keys by Host** link on the left side of the Rational License Key Center page.
 - i. Transmit the license manager update file that invalidates the license key to the license manager.
3. Install the license key file that invalidates the current license key on the license manager.
 - a. From the root user ID on the license manager, unzip the Hostname_XXXXXXXXX_update.zip file, run this command:
/opt/IBM/LDK/update_license Hostname_XXXXXXXXX_update.v2c. This command installs the invalidating license key file on the server.
 - b. Restart the license server daemon to make the license key file active by entering `systemctl restart aksusbd.service`.

 The license manager is no longer active, and the licenses cannot be used anymore. However, the license entitlements in the Rational License Key Center are still reserved until the return process is completed.
 - c. From the root user ID on the license manager, run this command:
opt/IBM/LDK/request_license. This command creates a request compressed file in the root home directory with a name similar to Hostname_XXXXXXXXX.zip. This file verifies that the license key is invalidated and is used to complete the return of the software-based license key file.
 - d. If needed, transmit the Hostname_XXXXXXXXX.zip file to the system you use to log in to the Rational License Key Center.
4. Use the verification file to complete the return process in the Rational License Key Center.
 - a. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
 - b. On the left side of the screen, select **View Keys by Host**.

- c. Select the **Hostname** corresponding to the host name of the license manager from which you want to return the license key.
- d. A table is displayed with data for the **Hostname** selected. At the far right of the table, click the **Change** link.
- e. The interface displays a list of the license keys for this license entitlement. Locate the license key of the license manager you are returning. Click **Browse**, and browse to and select the `Hostname_xxxxxxxx.zip` file that you just created.
- f. Click **Complete Return**.
- g. A message is displayed to confirm that the license entitlements were successfully returned.

Activating and configuring a license manager client

Each instance that uses software-based licensing must activate the license manager client and configure the client to have network access to the license manager. You must have network connectivity to the server.

About this task

If the license manager client is activated on an instance of z Systems Development and Test Environment, the zPDT emulator always first seeks access and entitlement from a license manager. For more information, see zPDT license servers.

Procedure

To configure and activate a license manager client:

1. Working as root, enter this command: `# /usr/z1090/bin/gen2_init`. This setup command needs to be done only once. It performs several actions:
 - a. The license manager client requires a 32-bit version of the Linux glibc library. This command first accesses the configured Linux repositories to obtain the latest version of this library. If your base Linux already has glibc-32bit installed, then you can ignore failures in attempting to fetch these libraries. If glibc-32bit is not already installed on your base Linux system and if you cannot connect to these repositories because of firewalls, for example, then you must obtain and install glibc-32bit in some other way before you start the license manager client.
 - b. The license manager client is started. The license manager client is started automatically any time the client Linux system is restarted.
2. If the client instance is running on a platform that previously authenticated with either a USB hardware device or a different license manager, issue this command from root:


```
uimreset -l
```
3. Enter this command from root:


```
cd /usr/z1090/bin/
./clientconfig
```

A window opens with several fields, then complete these fields:

- a. Set Gen2 Contact Server field to the hostname or IP address of license manager.
- b. Set Gen2 BackupServer field if you want to set up license manager for failover. Otherwise, leave the field empty. This field is optional.

- c. Leave Gen1 Contact Server empty. This field is used for product license server.
 - d. Leave Gen1 BackupServer empty. This field is used for failover product license server.
 - e. Set UIM ContactServer if you want to use different UIM server than License manager. Otherwise, leave the field empty. This field is optional.
 - f. Leave UIM Local Serial Random empty. This field is optional.
 - g. Press Enter twice to save these values.
4. Optionally, you can use command line interface to complete client configuration. To invoke the client, enter the following command:
- ```
cd /usr/z1090/bin/
./clientconfig_cli <parameter>
```
- You can write the client configuration file `sntlconfig.xml` for z1090/z1091. `clientconfig_cli` is usually run by the z1090/z1091 administrator as a superuser root.
- The following parameters are accepted.
- [g1s1 | g1s2 | g2s1 | g2s2] servername**  
Specify Gen1 or Gen2 server. One contact server (s1) and one backup server (s2) can be specified.
  - ucs server**  
Specify UIM contact server. The setup is the same as license server by default.
  - usm y/n**  
Specify UIM serial method.
  - l** List servers currently configured.
  - h** Help, and print this message.

## Results

The instance can be started. It attempts to authenticate with an activated license manager.

---

## The USB hardware device

For v9.5 or earlier, z Systems Development and Test Environment requires a USB hardware device that contains a license key that controls the licensee's access to all or portions of the program. The USB hardware device is ordered through Passport Advantage in a media pack that is separate from the electronic media that contains the offering software. It can be connected directly to the USB port on a computer that is hosting the offering or it can be plugged into the USB port of a separate computer that hosts the offering for distributing authentication to remote instances of the product through a TCP/IP network. This computer is known as a product license server.

The licensee is not authorized to use z Systems Development and Test Environment or any of its components except when that key is made available to the program. To make the key available to the program, you must obtain a license key file that provides proof of license entitlement from the Rational License Key Center. That file, called an *update file*, must then be applied to the USB hardware device to activate it and allow the licensee to use z Systems Development and Test Environment. The update file is specific to a customer account. The update file works with a single hardware device, which is identified by its unique serial

number. The USB hardware device is installed either on the computer that hosts the offering or on a remote product license server that communicates over the network to one or more instances of the offering.

A single high-capacity USB hardware device can authenticate many instances of z Systems Development and Test Environment. The number of instances that can be authenticated varies based on the number of emulated Central Processors or CPs used in each instance.

When an instance of the offering starts, it seeks an activation for the number of CPs specified in the emulator's device mapping file. When an instance of the emulator is stopped, the CPs authenticated by the USB hardware device become available for another instance.

If you purchase entitlement to z Systems Development and Test Environment by using Rational Tokens, the offering can also communicate with a Rational License Key Server to share license entitlements among several IBM products. The license entitlements are tracked by temporary use of a number of Rational Tokens. For information on setting up and using the Rational License Key Server, see the associated documentation available on the offering's installation e-image.

**Note:** Both the USB hardware device and the Rational License Key Server are required for the product to operate with Rational Tokens. When Rational Tokens are used for authentication with USB hardware devices, implement a product license server. For more information about shared license entitlements, see Rational Token activation and Setting up Rational Tokens.

## USB hardware device licensing terminology

Understand the terms that are used to describe the activation process with a USB hardware device.

The words *license* and *token* have more than one meaning, which depends on the context in which they are used.

You can encounter at least four different associated meanings for the word *license* as you work with z Systems Development and Test Environment:

- The word *license* can mean the product entitlement that you purchased and the corresponding terms and conditions under which use is allowed.
- The same word *license* can also refer to a technical indicator on the USB hardware device that is used to activate the hardware device and allow an emulated z Systems instance to operate. The zPDT Guide and Reference and the z Systems emulator messages use the words *license* and *licenses* to refer to these indicators stored on the hardware device. You can use a *remote license server* with the offering that derives its name from this particular usage of the word.
- The Rational License Key Center uses the word *license* and *licenses* for a quantity of product entitlement parts purchased.
- The file that is generated in the Rational License Key Center and that is applied to the USB hardware device to enable z Systems Development and Test Environment operation is called a generated *license* file in the website user interface.

To clarify the intended use of the term, the z Systems Development and Test Environment documentation uses these phrases:

- The phrase *product entitlement* refers to the terms and conditions under which the product can be run by the licensee.
- The phrase *license entitlement* refers to the Rational License Key Center interface and the quantity of entitlement parts purchased.
- The phrase *license key file* refers to a file that represents the user's license entitlement. License key files must be obtained, installed, and available to operate z Systems Development and Test Environment.
- The phrase *USB hardware device activation* or *activation* refers to the condition of the hardware device and its readiness for operation with the emulator.
- The phrase *update file* refers to the specific license key file that is generated in the Rational License Key Center and applied to the USB hardware device to activate it.

You can encounter at least two different associated meanings for the phrase *license server* as you work with z Systems Development and Test Environment:

- The phrase *license server* or *product license server* can mean the remote server that is set up to distribute authentication from a single high-capacity USB hardware device to remote instances of the product as an alternative to attaching discrete USB hardware devices to each computer that hosts one or more instances of the product.
- The same phrase *license server* can mean the Rational License Key Server that is used to share license entitlements among several IBM products.

To clarify these terms, the z Systems Development and Test Environment documentation uses these phrases:

- The phrase *product license server* refers to the remote z Systems Development and Test Environment license server that distributes authentication from a Sentinel Hardware Key USB hardware device.
- The phrase *Rational License Key Server* refers to the license server used exclusively to share Rational Token license entitlements among a set of IBM products.

You can encounter at least two different associated meanings for the word *token* as you work with z Systems Development and Test Environment:

- The word *token* can mean the USB hardware device that is required by z Systems Development and Test Environment. The zPDT Guide and Reference and z System emulator messages use the words *token* and *tokens* to refer to these hardware devices. In such places, you can find further reference to the 1091 token that is intended for use with the z Systems Development and Test Environment offering.
- The same word *token* can mean a type of IBM Rational product entitlement that allows great flexibility in the deployment and use of associated, purchased products.

To clarify these terms, the z Systems Development and Test Environment documentation uses these phrases

- The phrase *USB hardware device* refers to the hardware part.
- The phrase *Rational Token* refers to IBM Rational Token licensing.

The word *instance* refers to a single installation of z Systems Development and Test Environment on a physical or virtual machine.



z Systems Development and Test Environment can be purchased with several different product entitlements. The various types of entitlements authorize use of the product in specific ways.

For details on the product entitlement you purchased, see the documentation that is provided with your copy of the offering or refer to the Product License available at the IBM Software License Agreement website.

The Product License can be found by searching the site for the product name.

## State of the hardware device

The USB hardware device is not activated by default. When activated, the device allows z Systems Development and Test Environment to run one or more emulated central processors, referred to as Central Processors or CPs.

Activations are placed on the USB hardware device by using a command to apply an update file to the hardware device. For instructions on obtaining update files and applying the files to your USB hardware devices, see sections “Obtaining an update file” on page 27 and “Activating the USB hardware device” on page 33.

The number of CPs needed on the update file is dependent upon the number of z Systems Development and Test Environment instances that use the USB hardware device, and how many CPs each instance uses. A single instance of z Systems Development and Test Environment without Parallel Sysplex enabled requires 1 CP, and can run with a maximum of 8 CPs. The configuration file for each instance, called a device map or *devmap*, has a processor setting that defines the number of CPs that are used by that instance. The appropriate number that is needed for one instance depends on the number of cores in the hardware on which the instance is run, and the z/OS configuration that is running on that instance. See “zPDT instances” in the most current edition of the zPDT Guide and Reference for guidance on the number of CPs to define per instance. For the CP requirements for a z Systems Development and Test Environment instance with Parallel Sysplex enabled, see “Parallel Sysplex Activations” on page 25.

## Capacity of the hardware device

While every release of the offering included a USB hardware device, the maximum number of CPs that can be activated by a single device varied over time. The two variations that are issued are high capacity devices and low capacity devices.

### High capacity devices

High capacity USB hardware devices can activate up to 100 CPs. These parts have a green label. Since an instance is usually 3 CPs, a high capacity device can typically enable up to 33 product instances of the offering. High capacity devices became available for order with Rational Development Unit Test v8.0.3 and subsequent releases. As of Rational Development and Test Environment for z Systems v8.5, only high capacity devices are packaged with the offering.

### Low capacity devices

Low Capacity USB hardware devices can activate up to three CPs. These devices have a blue label. As of IBM z Systems Development and Test Environment v8.5, low capacity devices are no longer available for order.

If you are using a low capacity USB hardware device, you can successfully apply an update file that activates more than 3 CPs. However, you can use only three CPs due to the device limit. These hardware devices are intended for direct installation on the computer that hosts the offering. Only one instance of the offering is expected to run on the hosting machine.

## Serial number of the USB hardware device

Each USB hardware device has a unique serial number that is required during the activation process. To locate the serial number of a key, turn the key to the side opposite the colored label. You see three rows of numbers. The lowest or bottom row of numbers is the serial number. The serial number is always of the form 03-xxxxx or 02-xxxxx where xxxxx is five hexadecimal digits.

## Expiration dates

USB hardware device activations expire after one year from the date they were generated or less if entitlement is from a term license that expires in less than a year. If your purchase entitles you to use the product for more than one year, you need to renew the USB hardware device activation each year. Renewal activates the USB hardware device for a subsequent year or until the end of the fixed term if less than a year as applicable.

You can check the expiration date of an activation by using the token command. You can run the token command only while the emulator is running, and you must run the command from the same user ID that started the emulator.

### Expiration notification example

USB hardware device activations last for up to one year. If you are entitled to use z Systems Development and Test Environment beyond that period, the USB hardware device must be reactivated by obtaining and applying a new update file.

The examples in these topics show how you can set up a system to automatically send you an email each day when the expiration date for your current activation is approaching. The example is a bash shell script that is run on a Linux computer that hosts an instance of z Systems Development and Test Environment. Do not run this script on your product license server because it requires the product to run on the same system. You cannot operate a product license server and the product simultaneously on the same system.

This command works only with a single USB hardware device. If your product license server has multiple USB hardware devices that are attached, the results of this example are unpredictable because the example uses only the device activation that is used by the product. It does not search for or examine all attached USB hardware devices.

This example uses the token command to determine the expiration date of the USB hardware device activation. If the expiration date is within a specified number of days, then it formats an email and sends that email to the specified user.

### Related information:

“Expiration dates”

USB hardware device activations expire after one year from the date they were generated or less if entitlement is from a term license that expires in less than a year. If your purchase entitles you to use the product for more than one year, you need to renew the USB hardware device activation each year. Renewal activates the USB hardware device for a subsequent year or until the end of the fixed term if less than a year as applicable.

## Sample Expiration Notification Script Setup

1. This example assumes that the mail command on your Linux system already works with your network. Before you try this example, verify that mail can be sent from your Linux system with the commands:

```
echo Testing>/tmp/mymail mail s "a test subject line"
userid@your.email.address
```

2. Copy the "Sample Expiration Notification Script" on page 24 to your computer and make it available to the user ID that runs it. For example, if ibmsys1 is running the emulator, you can copy this script to /home/ibmsys1/expiredTokenCheck.sh

3. Set the executable flag

```
chmod 755 /home/ibmsys1/expiredTokenCheck.sh
```

4. Set the EMAILADDRESS and THRESHOLD variables in the script to contain the email address to be notified and the number of days before expiration that notification is to begin.
5. Ensure PATH and LD\_LIBRARY\_PATH variable exports are set correctly.

**Note:** Changing the variable exports is probably not necessary.

6. Ensure that awsstart is running continuously on the computer on which the script runs. The script might be modified to start a small instance temporarily if awsstart is not running, but that would require that a license be available.
7. Create a cron job to run this command each day. For example, under the user ID that is running the emulator and this script, enter the command

```
crontab e
```

Next, add a line to your cron file that points to your script. For example, if this script is named expiredTokenCheck.sh and is to be run by user ibmsys1, then your crontab entry would include:

```
@daily /home/ibmsys1/expiredTokenCheck.sh
```

## Sample Expiration Notification Script Output

This example, as shown, would send a note to USERID@YOUR.EMAIL.ADDRESS with a subject line similar to

```
RDT 0xDF0 expiration warning: 27 days (3/20/2014)
```

The body of the note is similar to the information shown in Table 1 on page 24.

Table 1. Output of the sample expiration notification script

```

Hostname : rdtesys1
DNS Name : rdtesys1.example.dnsname

Today's Date: : 02/20/2015 (mm/dd/yyyy)
Expiration date : 3/20/2015
Days remaining : 27 Days

**** USB Serial 0xDF0 will expire within 30 days. (27 remaining)
**** A new license file is required by

CPU 0, zPDTA(1091) available and working. Serial=4380(0x111C)
 Lic=3568(0xDF0) EXP=3/20/2015
CPU 1, zPDTA(1091) available and working. Serial=4380(0x111C)
 Lic=3568(0xDF0) EXP=3/20/2015
CPU 2, zPDTA(1091) available and working. Serial=4380(0x111C)
 Lic=3568(0xDF0) EXP=3/20/2015

End of zPDTA Status display

```

## Sample Expiration Notification Script

```

#!/bin/bash
#
Send an e-mail when a RDT USB hardware key activation is about to expire.
This works with Rational Development and Test Environment only.
For previous releases (RDzUT), the EXPDATE and SERIAL regular expressions
would need to be modified because the format of the "token" command changed.
#
This script should not be run on the product license server.
This script runs on the machine running the emulator
#
This is a sample only and may need modification on your system.
#
If you would prefer notification other than email, you might want to
- FTP a file to your z/OS system through the tunnel address.
- Run a z/OS console command through "oprmsg".
- Take other actions appropriate for your installation.

EMAILADDRESS=USERID@YOUR.EMAIL.ADDRESS # email address to receive emails
THRESHOLD=30 # number of days before expiration to start notification

export PATH=/usr/z1090/bin:$PATH # path to token command
export LD_LIBRARY_PATH=/usr/z1090/bin:$LD_LIBRARY_PATH # path to libman.so

SERIAL=`token |grep 'EXP='|head -n 1 | sed -e "s/.*Lic=.*(//\" -e "s/).*/\"`

if [! -z $SERIAL]; then # if serial number was found

 EXPDT=$(token | grep "EXP="|head -n 1 | sed -e 's/.*EXP=\\([^\]*)\\).*/\\1/g')
 TOKENDATE=`date -d $EXPDT +%s` # get expire date
 TODAY=`date +%s` # get today date in seconds
 DAYSLEFT=$((($TOKENDATE-$TODAY)/(24*3600))) # get days left

 NOTE=`mktemp`
 echo Hostname : `hostname` >> $NOTE
 echo DNS Name : $(nslookup `hostname` | \
grep -i "Name:"|sed -e 's/Name:\\s*/') >> $NOTE
 echo >> $NOTE
 echo Today's Date: : $(date +"%m/%d/%Y") \\(mm/dd/yyyy\\) >> $NOTE
 echo Expiration date : $EXPDT >> $NOTE
 echo Days remaining : $DAYSLEFT Days >> $NOTE

 if [$THRESHOLD -ge $DAYSLEFT]; then # if days left <= threshold
 # send mail or other actions.
 # This is where you would send an email or take other actions.
 echo >> $NOTE
 echo "**** USB Serial $SERIAL will expire within $THRESHOLD \" \
" days. ($DAYSLEFT remaining)" >> $NOTE
 fi
fi

```

```

echo "**** A new license file is required by $EXPDATE" >> $NOTE
echo >> $NOTE ; token >> $NOTE ; echo >> $NOTE
echo Sending $NOTE to $EMAILADDRESS for serial $SERIAL...
mail -s "RDT $SERIAL expiration warning: $DAYSLEFT days ($EXPDT)" \
$EMAILADDRESS < $NOTE
fi
cat $NOTE
rm $NOTE
fi

```

## Capability configuration of the USB hardware device

Based on details in the update file that is applied to the USB hardware device, the device is configured to make z Systems Development and Test Environment capable of operating within only one of the following parameters:

1. Parallel Sysplex support is not enabled and you are not required to use Rational Tokens.
2. Parallel Sysplex support is enabled and you are not required to use Rational Tokens.
3. Parallel Sysplex support is not enabled and Rational Tokens are required.
4. Parallel Sysplex support is enabled and Rational Tokens are required.

Although the hardware device itself permits the application of more than one update file, it is best to avoid mixing Sysplex and Non-Sysplex update files on the same USB hardware device. Mixing the update files can make it difficult to manage and track the deployment of license entitlements and potentially separate expiration dates for each update file that is applied to the device. Because of these concerns, the Rational License Key Center does not generate multiple update files for the same USB hardware device.

## License keys and the Application Developer Controlled Distribution (ADCD)

z/OS 1.13 ADCD and earlier distributions can be installed without access to an activated license key file. All volumes are supplied as files in compressed (.gzip) format, and the **gunzip** command makes them usable by z Systems Development and Test Environment.

Beginning with the z/OS 2.1 ADCD for Rational Development and Test Environment for z Systems, an activated license key file is required to install the system resident volumes of the z/OS ADCD. The **Z1091\_ADCD\_install** command is used to install these system resident volumes. This command requires access to an activated license key file, either on a locally attached USB hardware device, or on a product license server with connectivity to the instance that does the installation. The same activated license key file that entitles the user to run z Systems Development and Test Environment also entitles the user to install a z/OS ADCD.

For instructions on installing the z/OS ADCD, see Installing May 2017 Edition z/OS 2.2 ADCD for z Systems Development and Test Environment

## Parallel Sysplex Activations

Some entitlements of z Systems Development and Test Environment enable emulation of the Coupling Facility under z/VM while others do not. Entitlements that enable the coupling facility are referred to in z Systems Development and Test Environment documentation as Sysplex activations.

Although a coupling facility is not related to an individual processor in real z Systems hardware, every CP in a z Systems Development and Test Environment instance requires a corresponding Sysplex CP activation to enable the coupling facility for the instance.

Similarly, instances that do not enable the coupling facility require non-Sysplex CP activations for every CP in the instance. The update file that you receive to activate your USB hardware device is based on the type of license you purchased.

A single instance of z Systems Development and Test Environment with Parallel Sysplex enabled runs z/VM, the coupling facility, and one or more instances of z/OS. The available CPs defined to the instance are shared among all systems that are running, and are defined in the processor setting of the *devmap*. Configure at least one CP for each z/OS guest. A single instance of z Systems Development and Test Environment with Parallel Sysplex can be configured to use a maximum of 8 CPs.

## Rational Token Activations and the USB hardware device

USB activations that are specific to Rational Tokens must be used when Rational Token licensing is implemented. The emulator uses the USB hardware device activation that is stored on the USB hardware device to determine whether Rational Tokens are required, then communicates with the Rational License Key Server to determine whether enough tokens are available to activate the number of CPs requested for this instance.

**Attention:** When you are using multiple USB hardware devices, especially in an SHK license server, you must not mix activations that require Rational Tokens with activations that are not intended for Rational Tokens.

If the emulator receives inconsistent USB activations, your environment might be out of compliance with your licensing terms or you might be unable to access some of the CPs to which you are entitled.

z Systems Development and Test Environment usage can be governed by Rational Tokens. When you use Rational Tokens, the emulator coordinates with a Rational License Key Server instance and starts or continues to run only if sufficient Rational Tokens are available.

Use of Rational Tokens does not replace the USB hardware device. A USB hardware device that is activated with a valid update file is still required.

When z Systems Development and Test Environment is started, the emulator verifies that a valid USB hardware device activation is found and enough Rational Tokens are available from the Rational License Key Server to run the emulator. Rational Tokens control product operation while enough CPs are available from the USB hardware device or devices used. Each high capacity USB hardware device that is activated for use with Rational Tokens has a limit of 100 CPs. If you need to serve more than 100 CPs across all product instances, you can obtain extra high capacity USB hardware devices and activate them for use with Rational Tokens. In the Rational License Key Center, you can generate as many extra update files as needed to activate additional USB hardware devices. Multiple USB hardware devices can be attached to the product license server.

Both the activated USB hardware device and the Rational Tokens must be available to the emulator while the emulator is running. The emulator suspends operation if either becomes unavailable.

**Related information:**

Setting up Rational Tokens

Rational Token licensing is an entitlement that you can purchase and use to run z Systems Development and Test Environment. With Rational Tokens, z Systems Development and Test Environment maintains a connection to a Rational License Key Server and starts and continues to run only when sufficient Rational Tokens are available.

## Planning for USB hardware devices

You can spread your license entitlements across one or more USB hardware devices.

How many USB hardware devices are needed, and how those entitlements are activated on each device, is dependent upon many factors, such as:

- How many instances you run
- The capabilities of the underlying platforms
- The access of the different users to the different platforms
- The types of license entitlements you use

Consider the following rules when you are planning how to spread your license entitlements across your USB hardware devices, and how to later generate your update files:

- Activate a USB hardware device with only one update file.
- You can generate each update file with only one license type. That update file defines the capability configuration for the device.
- You can combine orders (license entitlements) in one update file, but only if they are of the same license type
- You can use multiple USB hardware devices in a single Linux environment, such as on a product license server. However, the following restrictions apply.
  - Do not have multiple USB hardware devices where one device is activated for Rational Tokens and one is not.
  - It is unpredictable which USB hardware device is used in obtaining entitlement to run any particular z Systems Development and Test Environment instance.

**Note:** Using multiple USB hardware devices in a single Linux environment can particularly affect product license servers in environments that use both Sysplex and non-Sysplex activations. For more information, see USB hardware device considerations when you use the coupling facility.

## Obtaining an update file

To activate a USB hardware device and enable the z Systems Development and Test Environment to run, you must first obtain a unique *update file* for each USB hardware device you are using.

You must then apply the update file to the USB hardware device.

Updates to the processes are published on the IBM Development and Test Environment Hub on developerWorks®.

**Important:** The method for obtaining update files for the Rational Development and Test Environment for z Systems USB hardware device is different from the method that is described in the zPDT Guide and Reference.

Most update files are now distributed as part of an archive with a file extension of .zip. These files are obtained through the Rational License Key Center. For version 9.1 and later, the \*.zip file is applied to the USB hardware device without being extracted. The \*.upw file that used to be delivered alone to update the USB hardware device for all entitlements is now in the compressed file (.zip file) along with other files necessary for version 9.1.0 and later releases of the offering. If you are using earlier releases of the offering and receive a compressed file, you can extract the compressed file and continue to use the \*.upw file alone with your USB hardware device just as you have in the past. Offering releases before version 9.1.0 do not require any other content in the compressed file.

To obtain an update file the first time, log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login>, select your account and use the **Get Keys** screen to select the order for which you want to generate and download an update file.

If you want to change the license entitlements for a USB hardware device you are already using, you must first return the update file that was previously created for that device, and then generate a new update file. The Rational License Key Center does not generate a new update file for a USB hardware device unless any previous update file for that same device is returned. To renew an update file, log in to the Rational License Key Center, select your account and use the **Return Keys** link to return the old update file. Then, use the **Get Keys** link to generate a new one. The return function does not require that you provide the old update file. The return function signals the Rational License Key Center that you can regenerate a new update file.

If you have a previously activated USB hardware device, you can use it without change to install IBM z Systems Development and Test Environment v9.1 or later and the z/OS 1.13 ADCD. However, if you want to install IBM z Systems Development and Test Environment v9.1 or later and the z/OS 2.1 ADCD or later with a previously activated USB hardware, the USB hardware device must be activated with an update file in the \*.zip format. If the update file is in the \*.upw format, return that update file and obtain a replacement in the new .zip file format.

**Before you sign in to the Rational License Key Center, locate the serial number of your USB hardware device.** You must know this number to generate an update file. To locate the serial number of a key, turn the key to the side opposite the colored label. Three rows of numbers are etched on the device. The lowest or bottom row of numbers is the serial number. The serial number is always of the form 03-xxxxx or 02-xxxxx where xxxxx is five hexadecimal digits.





Figure 1. USB hardware device

In Figure 1, the entire serial number is **02-00222**.

If you are unable to read the etched serial number from the USB hardware device contact IBM support. They ask for a “request file” created by the SecureUpdateUtility or the Z1091\_token\_update command with the -r switch. From this request file, IBM can determine the serial number of the USB hardware device.

**Note:** The Rational License Key Center now requires the entire serial number of the USB Hardware Device, not just the last (5) digits. For example, 03-00123.

If you are going to apply an update file that enables the coupling facility, see Enabling the coupling facility.

### Getting the update file from the Rational License Key Center

Learn how to use the Rational License Key Center to obtain update files.

If your installation is using Rational Tokens to govern entitlement to z Systems Development and Test Environment, see “Obtaining update files for Rational Tokens” on page 31.

#### Getting the initial file:

Obtain an update file from the Rational License Key Center.

1. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
2. On the left side of the screen, select **Get Keys**
3. Select the product line for z Systems Development and Test Environment. You see a list of license key names that show the z Systems Development and Test Environment licenses that are available.
4. Select the z Systems Development and Test Environment license type that you want to apply to the key.
5. You see one or more boxes that show the separate orders that you can apply to one or more USB hardware devices.

**Important:** For some license entitlements, you can combine multiple orders on a single USB hardware device or spread the licenses across multiple USB hardware devices. Regardless of how many z Systems Development and Test Environment instances you deploy for your licensed users, you must make sure that you maintain proper entitlement for the number of users who access each instance of the product. License entitlements based on Resource Value Units (RVUs) do not limit the number for users who access the offering. All other types of License entitlements require at least one product entitlement for each user who accesses the product.

6. Check the box next to all of the orders from which you would like to use licenses and click **Next**.
7. A screen is displayed showing a table that you must complete with the USB hardware device serial numbers you want to activate. Depending on the license entitlement that is selected, you might also be asked for the number of instances, the number of licenses to associate with each USB hardware device, and the number of Emulated Central Processors you intend to use for the specific USB Hardware Device activation.

You can enter information for more than one USB hardware device. A separate update file is generated for each key specified. You can download the generated update files either during this process or from the **View keys by host** screen.

More help is available by clicking the column headings in the table, but briefly stated:

- a. The **Serial Number of USB Hardware Key** field is where you specify the full eight character serial number that is etched on the last line of the USB hardware device. It has a format of 02-xxxxx or 03-xxxxx where xxxxx is five hexadecimal digits. The two-digit prefix and the dash must be included when you specify the serial number.
  - b. The **Number of Server Instances** field refers to the number of separate emulated systems you plan to enable with each USB hardware device. The number of central processors (CPs) that is available is automatically calculated for you. For some older entitlements, this value is forced to one.
  - c. The **Number of Licenses** field is applicable to some offering entitlements. Where applicable, each user who accesses a separate instance of a virtual z Systems machine requires a separate license. Individual users are licensed to a specific instance of a virtualized z Systems. The total number of licenses for all serial numbers must not exceed the number of licenses to which you are entitled. For example, if you are entitled to five Authorized User licenses, you might enable 2 instances, and 5 licenses so that one workstation can be used by two people and another workstation can be used by three people. Designating the number of licenses that are associated with a USB hardware device ensures that you are tracking the number of entitlements used.
  - d. The **Emulated Central Processor** field is applicable only to entitlements metered as Resource Value Units (RVUs). For this entitlement, each emulated central processor within an instance of a virtual z Systems machine requires a separate RVU entitlement. For each serial number, you must enter the total number of emulated central processors that you intend to enable with that specific USB hardware device. The total number of emulated CPs for all serial numbers must not exceed your total number of RVU entitlements.
8. After you indicate how you want to distribute license entitlements across USB hardware devices, select **Generate** at the bottom of the page and you are presented with a screen where you can download the update files.
  9. Click **Download** for each update file that is generated to save the update files. If you need to download the update files in the future, use the **View Keys by Host** link on the left side of the Rational License Key Center web page.

**Note:** You cannot generate separate activations for the same USB hardware device at the same time. Before you generate update files for a particular USB hardware device that you want to reuse, be sure to return the update file that was previously created for that device. Returning license entitlements that are assigned to the USB

hardware device also makes the corresponding active entitlements available for assignment to a USB hardware device again.

### Getting the replacement file:

For perpetual license entitlements, USB hardware device activations are set to expire one year from the date an update file is generated. For this type of entitlement, you can return previously generated update files at any time and generate a new update file.

For term license entitlements, if term expiration occurs in more than one year, USB hardware device activations are set to expire one year from the date an update file is generated. Update files can be returned and generated again before term expiration. If term expiration occurs in less than one year, USB hardware device activations are set to expire at term end. After term end, update files cannot be generated in the Rational License Key Center.

To generate a new update file, you must first return the existing license entitlement in the Rational License Key Center, and then generate a new update file. Returning the license entitlements is a process of telling the Rational License Key Center that you are no longer using the license entitlements that you previously assigned to your USB hardware device. You do *not* need to return the physical update file to the Rational License Key Center.

You can return a license entitlement in the Rational License Key Center in several ways. The easiest method is to use the **View Keys by Host** link. You can also use the **Return Keys** link.

For z Systems Development and Test Environment, the term **host** in the Rational License Key Center refers to the USB hardware device that is uniquely identified by its serial number.

1. Log in to Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
2. On the left side of the screen, select **View Keys by Host**.
3. Select the serial number of the USB hardware device you want to work with. This serial number is in the **Host** column.
4. A table is displayed with data for the USB hardware device selected. At the far right of the table, click the **Change** link.
5. You see a list of devices with license entitlements that are assigned to them from the same Order Line. Locate the serial number of the USB hardware device you are working with and click **Return**. A message is displayed to confirm that the license entitlements were successfully returned.
6. Use the **Get Keys** link on the left side of the page to generate a new license. For detailed instruction on generating new licenses, see "Getting the initial file" on page 29

### Obtaining update files for Rational Tokens

Learn how to use the Rational License Key Center to obtain update files if your installation is using Rational Tokens to govern entitlement to z Systems Development and Test Environment.

Use of Rational Token licensing requires purchase of Rational Tokens. After the Rational Tokens are purchased, follow these instructions for obtaining Rational Token license files and update files for your USB hardware devices.

## Getting the initial update file and Rational Token license:

Learn how to get an update file for your USB hardware device and a Rational token license file from the Rational License Key Center.

1. Log in to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> and select your account.
2. On the left side of the screen, click **Get Keys**
3. Select a product line that contains your Rational Token orders with z Systems Development and Test Environment.
4. On the "Select License Keys" screen, you see one or more boxes for orders. Find the order that includes z Systems Development and Test Environment. Make a note of the part number for the z Systems Development and Test Environment product you want to enable. Select the order that contains the parts you noted and click **Next**.
5. Enter the required information to create the Rational Token license file for your host. Additional information for each field is available by clicking the descriptive link next to the input field. Fields with a red star are required fields.
6. Click **Generate** to generate the Rational Token license file. You are presented with a screen with the generated license information.
7. Click **Download License Keys** to download the Rational Token license file to your computer. This token license file is the file that is applied to your Rational License Key Server. It is named `license.dat` by default.
8. On the same screen, press **Generate Token License Keys for USB Hardware Device** to generate and download the update file for your USB hardware device.
9. A **Required Information** screen is displayed showing a table that you must complete with the USB hardware device serial numbers and associated z Systems Development and Test Environment product you want to enable with each device.

You can enter information for more than one USB hardware device. A separate update file is generated for each key and can be downloaded either during this process or at some time in the future from the **View keys by host** screen.

The serial number is the full eight character serial number that is etched on the last line of the USB hardware device. It has the format 02-xxxxx or 03-xxxxx where xxxxx is five hexadecimal digits. The two-digit prefix and the dash must be included when you specify the serial number.

Use the **Catalog Item** list to confirm the part that you want to use to generate an update file. This confirmation is important if you have entitlement to both z Systems Development and Test Environment and z Systems Development and Test Environment with Parallel Sysplex. You cannot combine Token and Sysplex Token activations on a single USB hardware device. From the list, select the entitlement part number that corresponds to the activation you want on the specified USB hardware device.

10. After you indicate how you want to distribute licenses across USB hardware devices, click **Generate** at the bottom of the screen and you will be presented with a screen where you can download the update files.
11. Click **Download** for each update file to save the update files. If you need to download the update files at some time in the future, use the **View Keys by Host** link on the left side of the page.

**Note:** You cannot generate separate activations for the same USB hardware device at the same time. Before you generate update files for a USB hardware device that you want to reuse, be sure to return the update file that was previously created for that device. Returning license entitlements that are assigned to the USB hardware device also makes the corresponding active entitlements available for assignment to a USB hardware device again.

#### **Getting a replacement update file and Rational Token license:**

To return and regenerate your update file and existing Rational Token license files, use the **Return Keys** function in the Rational License Key Center. Use this process to renew the USB hardware device activation for another year.

Do these steps:

1. Go to the Rational License Key Center at <https://licensing.subscribenet.com/control/ibmr/login> to return the existing USB hardware device update file. For more information about returning update files, see “Getting the replacement file” on page 31.
2. After you return the update file, click **Get Keys** on the left side of the screen.
3. Click the product line for IBM Rational Tokens.
4. You see a screen titled **Select license keys**. Keys that include z Systems Development and Test Environment have a link to generate update files for USB hardware devices.

Click the link titled **Generate Token License Keys for USB Hardware Device** and complete the form to generate and download the new update file.

## **Activating the USB hardware device**

Applying the update file that you obtained from the Rational License Key Center to your USB hardware device activates the device. Each time that you apply an update file, it overwrites the previous activation on the USB hardware device. To properly activate, the update file must be generated with the same serial number as the USB hardware device that it applies to. To change the activation of a USB hardware device, you must obtain and apply a new update file that activates the total number of license entitlements you intend to use on that device. Changing the activation of a USB hardware device involves returning and regenerating licenses in the Rational License Key Center.

The process for applying update files to USB hardware devices recently changed. For instructions on applying update files to USB hardware devices for z Systems Development and Test Environment version 9.1 or later, see “Activating a USB hardware device” on page 34

For instructions on applying update files to USB hardware devices for Rational Development and Test Environment for z Systems version 9.0 or earlier, see “Activating a USB hardware device for version 9.0 or earlier” on page 35

**Important:** In a product license server configuration, do not mix USB hardware devices that require Rational Tokens with USB hardware devices that do not require Rational Tokens. This approach is not supported and can result in unpredictable behavior.

## Activating a USB hardware device

Before you insert the USB hardware device into a computer's USB port, verify that the computer's clock is set correctly. If your computer is new or recently repaired, ensure that the clock is correct before you insert the USB hardware device.

**Note:** Never set the system clock to a future date or time when the USB hardware device is plugged in or you will damage the hardware device.

**Note:** Verify that the Linux system clock is correct. If the USB hardware device is attached to a computer with a clock set to a future date, the device remembers the future date, will become inoperable when the clock is restored to the current time, and will remain inoperable until the computer's clock again reaches that future date. If z Systems Development and Test Environment is unable to authenticate with the USB hardware device because the clock was set to a future time or date, you receive a message that indicates a "time cheat" state.

Set your system clock to Coordinated Universal Time (UTC) to avoid errors that are caused by semi-annual time changes (for example, Daylight Saving Time).

The update file that you obtain from the Rational License Key Center is an archive file with the extension .zip. With Rational Development and Test Environment for z Systems version 9.1 or later, do not extract the contents of the archive.

In Rational Development and Test Environment for z Systems version 9.1 or later, the Z1091\_token\_update command replaces the SecureUpdateUtility command that is used in previous releases.

To activate your USB hardware device, connect it to a computer that has Rational Development and Test Environment for z Systems version 9.1 or later installed and run the Z1091\_token\_update command. The step-by-step procedure to activate your USB hardware device is as follows:

1. Connect the USB hardware device to the computer that is running z Systems Development and Test Environment.

**Note:** Connect only the USB hardware device that is being activated to your computer during this process, or unpredictable results might occur.

2. Use a browser on your Linux system to download the update file directly from the Rational License Key Center or use whatever method is available to copy your update file to that computer (for example, FTP, SFTP, or an external hard disk).

**Note:** Be sure to use the binary transfer mode and not a text mode such as ASCII.

**Tip:** The Z1091\_token\_update cannot process a path when any directory in the path or the update file name contains spaces. This restriction remains true even if the path and file name are supplied to the Z1091\_token\_update command in single or double quotation marks. Therefore, ensure that neither the path to the update file nor the update file name itself contains spaces.

3. Log in to the computer and, if you are not running as root, enter su followed by the root password.
4. Change to the /usr/z1090/bin directory

```
cd /usr/z1090/bin
```

5. Run the `./Z1091_token_update -u update-file.zip` command where *update-file.zip* is the full path and file name of the update file. This command produces several messages that indicate that the update was successful.
6. After the update is successfully applied, unplug the USB hardware key. Wait at least 10 seconds and then reconnect the hardware key. It is now ready for routine zPDT operation.
7. To ensure that the system sees the newly applied licenses, from root, restart the local SHK license server: `/opt/safenet_sentinel/common_files/sentinel_keys_server/loadserv restart`
8. Optionally, you can verify that licenses are now available on your USB hardware device by using the command:

```
./Z1091_token_update -status
```

**Tip:** After z Systems Development and Test Environment is started, you can verify the effective z System serial number, the token number, and the expiration date of the license. Enter the zPDT command **token** with no operands.

**Tip:** The token command might return different results based on the `cpuopt zvm_couplingfacility` setting in the device map configuration file, called the *devmap*. For more information, see *Enabling the coupling facility*.

#### Stand-alone .upw update files:

Before IBM started to distribute compressed file archives containing update files, stand-alone update files with an extension of `.upw` were distributed. If you have an update file with an extension of `.upw`, you can use the `Z1090_token_update` command to apply the update file to your USB hardware device. The activated hardware device still allows the emulator to run. However, you cannot install the z/OS 2.1 ADCD for Rational Development and Test Environment for z Systems with this kind of activation.

#### USB hardware device and Sysplex activation:

When you are using the `Z1091_token_update` command to apply an update file for Parallel Sysplex operation that is provided in compressed file format, it is no longer necessary to apply the `RDT_SP.NLF` file to prepare the USB hardware device. That step is done automatically by the `Z1091_token_update` command.

For instructions on enabling the coupling facility after you have an activated USB hardware device, see *Enabling the coupling facility*.

#### Activating a USB hardware device for version 9.0 or earlier

Before you insert the USB hardware device into a computer's USB port, verify that the computer's clock is set correctly. If your computer is new or recently repaired, ensure that the clock is correct before you insert the USB hardware device.

**Note:** Never set the system clock to a future date or time when the USB hardware device is plugged in or you will damage the USB hardware device.

**Note:** If the USB hardware device is attached to a computer with a clock set to a future date, the device remembers that future date, will become inoperable when the clock is restored to the current time, and will remain inoperable until the

computer's clock reaches that future date. If z Systems Development and Test Environment is unable to authenticate with the USB hardware device because the clock was set to a previous time or date, you receive a message that indicates a "time cheat" state.

If the update file you obtain from the Rational License Key Center is an archive file with the extension .zip, you need to extract the contents of the archive to a temporary directory during the process that is described here to get the \*.upw file that is needed for z Systems Development and Test Environment version 9.0 or earlier releases.

If you have an older update file that has an extension of .upw, you can use that file with the SecureUpdateUtility as described in step 4, step 5, and step 7.

If you follow this process, your USB hardware device might not be able to install the z/OS 2.1 volumes. This restriction is true if the hardware key was never activated for z Systems Development and Test Environment v9.1.0 or later.

Read the entire set of instructions before you start activating your USB hardware device. In summary, you connect the device to a computer that has z Systems Development and Test Environment installed. Extract the contents of your update compressed file and run the SecureUpdateUtility command against the \*.upw file that was extracted from the .zip file. The entire procedure is described here:

1. Connect the USB hardware device to the computer that is running z Systems Development and Test Environment.

**Note:** Update files are written for specific USB hardware devices, so it is acceptable to have more than one USB hardware device that is attached to the computer during this process.

2. Use a browser on your Linux computer to download the update file directly from the Rational License Key Center or use whatever method is available (for example, FTP, SFTP, or an external hard disk) to copy your update file to that computer.

**Note:** Be sure to use the binary transfer mode and not a text mode such as ASCII.

3. Log in to the computer, create a new, temporary directory and extract the contents of your update file to that directory. For example:

```
mkdir /tmp/myupdatedir
cd /tmp/myupdatedir
unzip update-file.zip ls
```

You see a list of file names. Note the name of the file that ends in .upw. The upw file has a name similar to RDT-xxxxx-yyyyymmddhmm-Lcc.upw or RDT-xxxxx-yyyyymmddhmm-Lcc.-Sysplex.upw where:

**xxxxx** Is your device's serial number

**yyyyymmddhmm**

Is the expiration date of the activation

**cc** Is the number of licensed CPs in hexadecimal format

4. If you are not running as root, enter su followed by the root password.
5. Change to the /usr/z1090/bin directory

```
cd /usr/z1090/bin
```



6. If you are applying an update file for Parallel Sysplex, run the **SecureUpdateUtility** command with the **-n** switch and the full path of the **adcd.NLF** file. For example, enter this command on one line:  

```
./SecureUpdateUtility -n /tmp/myupdatedir/adcd.NLF
```

This command produces several messages but indicates that the update was successful or that the license exists, meaning the update file was already applied. You need to run this command only one time for a USB hardware device.

**Note:** Running this command each time you apply an update file does not cause problems.

7. Run the **./SecureUpdateUtility -u update-file.upw** command where *update-file.upw* is the full path and file name of the update file. For example, enter the following command on one line:  

```
./SecureUpdateUtility -u /tmp/myupdatedir/RDT-00DF0-201502152359-L03-Sysplex.upw
```

This command produces several messages but indicates that the update was successful.

8. To ensure that the system sees newly applied licenses, run the following command to restart the local product license server:

```
/opt/safenet_sentinel/common_files/sentinel_keys_server/loadserv restart
```

9. You can verify the effective z System serial number, the token number, and the expiration date of the license after z Systems Development and Test Environment is started. Issue the **zPDT** token command with no operands.

**Tip:** The token command can return different results based on the **cpuopt zvm\_couplingfacility** setting in the device map configuration file, called the *devmap*. For more information, see *Enabling the coupling facility*.

For answers to common errors related to USB hardware device activation, see *Troubleshooting license manager and USB hardware device activation*.

## Quick setup instructions for using and migrating the product license server

Learn how to set up the product license server, how to set up other computers to access the product license server, and how to migrate the product license servers and client instances that use product license servers.

### Setting up the product license server

Learn how to set up the product license server.

Configuring a product license server by using the default ports can be accomplished with the following steps:

1. Obtain the update files for the USB hardware devices you want to use with the SHK license server.
2. Install z Systems Development and Test Environment on the computer you want to function as the SHK license server. Follow the instructions for installing z Systems Development and Test Environment in *Installing z System Development and Test Environment*. You do not have to install Rational Test Control Panel.

**Note:** This computer can be a virtual machine only if it can access the USB hardware device.

3. Connect the USB hardware device to the product license server computer. If you are running the SHK license server in a virtual machine, be sure that the USB hardware device is routed to that virtual machine.
4. Activate the attached USB hardware device. This process is described in “Activating the USB hardware device” on page 33.
5. Log in to the computer as a non-root user and enter the `uimserverstart` command.
6. If you have a firewall that is enabled, verify that your firewall allows connections to ports 9450 and 9451. If your firewall is based on IP tables, commands like

```
iptables -I INPUT -p tcp --dport 9450 -j ACCEPT
```

or

```
iptables -I INPUT -p tcp --dport 9451 -j ACCEPT
```

entered from a root user ID can be used to enable your firewall to allow traffic to these ports:

**Note:** This procedure is an example. Details about managing your Linux firewall and your external routing controls are Linux beyond the scope of this document.

You might want to alter the server configuration to allow remote access to the browser-based configuration interface, to restrict access to specific hosts to enable logging, or to complete other tasks. For details on configuring product license servers and to familiarize yourself with the product license server, see zPDT license servers.

## Setting up computers to access the product license server

The computers that authenticate from a product license server instead of a locally attached USB hardware device can be set up by running the `clientconfig` program as root. This process assumes that your product license server is using all of the default ports.

1. Log in to each computer that runs the emulator with authentication from a product license server.
2. Enter the following commands:

```
su (enter root password when prompted)
cd /usr/z1090/bin
./clientconfig
```
3. A window opens with several fields, then complete these fields.
  - a. Leave Gen2 Contact Server empty. This field is used for software-based license manager.
  - b. Leave Gen2 BackupServer empty. This field is used for software-based license manager.
  - c. Set Gen1 Contact Server field to point to IP address or hostname of product license server.
  - d. Set Gen1 BackupServer if you want to set up the failover product license server. Set the field to point to IP address or hostname of failover product license server.

- e. Set UIM ContactServer if you want to use different UIM server than product license server. Otherwise, leave the field empty. This field is optional.
  - f. Leave UIM Local Serial Random empty. This field is optional.
  - g. Press Enter twice to save these values.
4. Optionally, you can use command line interface to complete client configuration. To invoke the client, enter the following command:
- ```
cd /usr/z1090/bin/
./clientconfig_cli <parameter>
```
- You can write the client configuration file `sntlconfig.xml` for z1090/z1091. `clientconfig_cli` is usually run by the z1090/z1091 administrator as a superuser root.
- The following parameters are accepted.
- [g1s1 | g1s2 | g2s1 | g2s2] servername**
Specify Gen1 or Gen2 server. One contact server (s1) and one backup server (s2) can be specified.
 - ucs server**
Specify UIM contact server. The setup is the same as license server by default.
 - usm y/n**
Specify UIM serial method.
 - l** List servers currently configured.
 - h** Help, and print this message.

For details on configuring clients to the product license server, see the “License Server” chapter in the zPDT Guide and Reference.

Migrating product license servers and client instances that use product license servers

Learn how to migrate product license servers and client instances that use product license servers.

Migrating clients that use product license servers

In most cases, you can migrate z Systems Development and Test Environment instances that use product license servers (clients) to newer versions or releases without migrating your product license server to the newer release. The USB hardware device on the product license server generally does not need to be updated, and the client is able to use the product license server as soon as the installation of the client is complete. This capability allows for installing new releases without stopping the product license server, and thus not affecting other users.

For more information about installing the client, see *Installing z System Development and Test Environment*. For more information about configuring the client for access to the product license server, see “Setting up computers to access the product license server” on page 38.

Migrating product license servers

Upgrade your product license server to at least the highest level of z Systems Development and Test Environment in use by any of its client instances.

The product license server must be upgraded only in the following case:

- Your current product license server is running z Systems Development and Test Environment Version 9 Release 0 or below, or the USB Hardware Device of the product license server was activated with a .upw file instead of a .zip file.
- You are upgrading a client to Version 9 Release 1 or above
- You are installing a z/OS 2.2 ADCD for z Systems Development and Test Environment on the client

Tip: Even in this case, you can upgrade a client to z Systems Development and Test Environment Version 9 Release 1 and above without migrating your product license server. See <http://www.ibm.com/support/docview.wss?uid=swg21959847>.

Upgrading your product license server to z Systems Development and Test Environment Version 10 can be accomplished with the following steps:

- Obtain a *.zip update file for the USB Hardware Device that is used by the SHK license server. For the step-by-step procedures, see “Obtaining an update file” on page 27.
- Install z Systems Development and Test Environment v10. The instructions for installing z Systems Development and Test Environment are in Installing z System Development and Test Environment.
- Activate the USB Hardware Device. For the step-by-step procedure, see “Activating the USB hardware device” on page 33.
- Set up the product license server. For the quick setup instructions for setting up the product license server after you finish installing z Systems Development and Test Environment, see “Setting up the product license server” on page 37.

Note: Any z Systems Development and Test Environment instances that require authentication from the product license server during the upgrade will not run until the upgrade is complete.

After the upgrade to the product license server is complete, any z Systems Development and Test Environment instance can use the product license server both to install and run any z/OS distribution, without a locally attached USB Hardware Device. Also, assuming the product license server is at the same IP address and port as the previous version, no additional actions are required to existing z Systems Development and Test Environment instances that use that product license server. Down level z Systems Development and Test Environment instances can run accessing a z Systems Development and Test Environment v9.5 product license server without change.

Next steps

You might need to issue a **UIMRESET -1** command before you use z Systems Development and Test Environment with an activated USB hardware device.

These cases are the most common reasons to issue the command:

- If you were using a single local USB hardware device, and are replacing it with another USB hardware device or want to use multiple USB hardware devices.
- If you change the disk containing the Linux partition or upgrade the Linux kernel.
- If you are using a product license server or license manager and the UIM serial number in the local database conflicts with a serial number in the license server.

For more information about UIM serial numbers, how they are assigned, and when resetting the UIM serial numbers is required, see zPDT license servers.

If you worked through the previous sections of this IBM Knowledge Center, you now have one or more activated license keys that are ready to enable operation of one or more instances of z Systems Development and Test Environment. For the next steps to follow to use the offering, see [Installing an operating system on z Systems Development and Test Environment](#).

Index

Special characters

.upw update files 35
.upw update files, Stand-alone 35

A

access, product license server 38
activate a USB hardware device for
v9.0 35
activate a USB hardware device for
v9.1 34
activating the product
license manager 1
USB hardware device 1
Activating the USB hardware device 33
activation 27
activation process 5
activation, license 27
activation, Parallel Sysplex 26
activation, Rational Token 13, 26
activation, replacement update file 31
activation, USB hardware device 25, 27
activation, USB hardware device and
Sysplex 35
administering the license manager 9

B

Backup license managers 10

C

capacity, hardware device 21
capacity, license server 8
configuration, USB hardware device 25
consideration, security 5
considerations, terminology 3, 19

D

dates, expiration 22
device activation 25, 34, 35
device, activating 33
devices, high capacity 21
devices, low capacity 21
devices, serial number 22

E

enabling product operation 1
entitlement, license 3, 19
entitlement, product 3, 19
example, expiration notification 22
expiration 22
expiration dates 22
expiration notification example 22
expiration notification sample script 24

expiration notification sample script
output 23
expiration notification sample script
setup 23

F

file, update 32, 33

G

getting the update file 29

H

hardware device activation 25, 34, 35
hardware device serial number 22
hardware device, activating 33
hardware device, capacity 21
hardware device, high capacity 21
hardware device, low capacity 21
hardware device, state 21
high capacity devices 21
host ID and host name of the license
server 9

I

initial Rational Token license 32
initial Rational Token license, Steps to
get 32
initial update file, Steps to get 32
installing and starting the license
manager 8

K

key 19
key, activating 33

L

license 3, 19
license entitlement 3, 19
license file replacement 31
license key 19
license key expiration 6
license key, activating 33
license keys
ADCD 6, 25
license manager
installing and starting 8
license server
host ID 9
host name 9
license server, capacity 8
low capacity devices 21

N

Next steps 41
notification example, expiration 22

O

obtain an update file 29
Obtain an update file 27
Obtaining update files for Rational
Tokens 31

P

Parallel Sysplex activations 7, 26
process, activation 5
product entitlement 3, 19
product license server access, set up 38
product license server, set up 37
product operation
enabling 1

R

Rational License Key Center 29
Rational Token 3, 19
Rational Token activation 7, 13, 26
Rational Token license, initial 32
Rational Token license, replacement 33
Rational Tokens, update file 31
replacement license file 31
replacement Rational Token license 33
replacement Rational Token license, Steps
to get 33
replacement update file 31
replacement update file, Steps to get 33

S

sample script output, expiration
notification 23
sample script setup, expiration
notification 23
sample script, expiration notification 24
script output, expiration notification
sample 23
script setup, expiration notification
sample 23
script, expiration notification sample 24
security consideration 5
serial number, hardware devices 22
serial number, USB device 22
set up product license server access 38
Set up the product license server 37
Stand-alone .upw update files 35
state of the hardware device 21
Steps to get the initial update file 32
Steps to get the Rational Token
license 32, 33

Steps to get the replacement update
file 33
Sysplex activation 35
Sysplex and USB hardware device
activation 35

T

terminology considerations 3, 19
token 3, 19

U

uninstalling the license manager 9
update file 3, 19, 29

update file, initial 32
update file, obtain 27
update file, Rational License Key
Center 29
update file, replacement 31, 33
update file, Rational Tokens 31
update files, Stand-alone .upw 35
USB devices, high capacity 21
USB devices, low capacity 21
USB devices, serial number 22
USB hardware device 19, 27, 33
USB hardware device activation 19, 25,
34, 35
USB hardware device and license
activation 27

USB hardware device and Sysplex
activation 35
USB hardware device configuration 25
USB hardware device licensing 19
USB hardware device serial number 22
USB hardware device, activating 33

V

V9.0 USB hardware device activation 35
V9.1 USB hardware device activation 34