Advanced Settings Utility
Version 3.60

IBM

User's Guide

Advanced Settings Utility
Version 3.60

User's Guide

Note: Before using this information and the product it supports, read the general information in Appendix B, "Notices," on page 97.

Contents

Chapter 1. Using the Advanced Settings Utility

You can use the IBM® Advanced Settings Utility (ASU) to modify firmware settings from the command line on multiple operating-system platforms. You can perform the following tasks by using the utility:

- Modify selected basic input/output system (BIOS) CMOS settings without the need to restart the system to access F1 settings
- Modify selected baseboard management controller setup settings
- Modify selected Remote Supervisor Adapter and Remote Supervisor Adapter II setup settings
- Modify selected settings in integrated management module (IMM)-based servers for the IMM firmware and IBM System x Server Firmware. The IMM replaces the Remote Supervisor Adapter and baseboard management controller functions on IMM-based servers. IBM System x Server Firmware is the IBM implementation of Unified Extensible Firmware Interface (UEFI). The UEFI replaces the basic input/output system (BIOS) and defines a standard interface between the operating system, platform firmware, and external devices.
- Modify a limited number of VPD settings on IMM-based servers.
- Modify iSCSI boot settings. Note that to modify iSCSI settings with the ASU, you must first manually configure the values by using the server Setup utility settings on IMM-based servers. For more information, see "iSCSI configuration" on page 37.
- Remote connectivity to support setting all the listed firmware types settings on IMM-based servers. Remote connection support requires accessing the IMM external port over a LAN.

The ASU supports scripting environments because it is a command-line utility; however, it also offers easier scripting through its batch-processing mode.

For a list of the ASU commands and their descriptions, see Chapter 3, "Command reference," on page 41.

Note: Throughout this document, the term *Remote Supervisor Adapter II* refers to both the IBM Remote Supervisor Adapter II and the IBM Remote Supervisor Adapter II SlimLine, unless otherwise noted.

Note the following limitations for IMM-based servers only:

- Some settings might require that you restart the IMM before the settings become effective or for the values that are set through the ASU to be displayed in the IMM Web interface.
- For IMM-based servers, the ASU does not support the commands to generate, import, or export security certificates. This command support is still available for BIOS-based servers.

Supported firmware types

The ASU supports the following firmware types:

- BIOS firmware
- Remote Supervisor Adapter firmware
- Remote Supervisor Adapter II firmware
- Baseboard management controller firmware

- IBM System x Server Firmware
- IMM firmware

Note: The PC-DOS version of the ASU supports BIOS settings only.

To support the settings of the previously listed firmware types, the following types of settings are supported:
- Banked CMOS at 70h/71h (NS317)
- CMOS at 70h/71h and 72h/73h (NS417)
- CMOS at 72h/73h (AMD 8111)
- CMOS through baseboard management controller
- Serial EEPROM settings
- The following Remote Supervisor Adapter and Remote Supervisor Adapter II commands:
  – 8-bit values
  – 16-bit values
  – IP address values (32 bits)
  – Strings
  – Keystroke sequences
  – Certificate
  – Port
- The following baseboard management controller commands:
  – 8-bit values
  – 8-bit value within a block
  – IP address values (32 bits)
  – MAC address values (48 bits)
  – Strings
- Single and multi-node systems

The ASU retrieves and modifies user settings from the supported firmware types through its command-line interface. The ASU does not update any of the firmware code.

Operating-system support

The following operating systems (32-bit and 64-bit) support the ASU.

Note: For the latest information about IBM servers, workstations, and the operating systems that are supported, see http://www.ibm.com/servers/eserver/ serverproven/compat/us/.
- Microsoft® Windows® 2000, Windows XP, Windows Server 2003, Windows Server 2003 R2, Windows Server 2008, Windows Server 2008 R2, and Windows Vista™

  Note: For the Windows operating system, you must have administrator privileges. For more information about using the ASU in the Windows operating system, see the readme file that is extracted when you extract the ASU package.
- Microsoft Windows PE 1.6, Windows PE 2.0, and Windows PE 2.1

- Red Hat® Linux® version 4, Red Hat Linux version 5, Red Hat Linux version 7, Red Hat Linux version 8, and Red Hat Linux version 9 (see the note that follows this list)
- Red Hat Enterprise Linux AS 2.1
- Red Hat Enterprise Linux version 3 (see the note that follows this list)
- Red Hat Enterprise Linux version 4 (see the note that follows this list)
- Red Hat Enterprise Linux version 5 with Xen added (see the note that follows this list)
- SUSE Linux version 7, SUSE Linux version 8, and SUSE Linux version 9
- SUSE Linux Enterprise Server 8, SUSE Linux Enterprise Server 9, SUSE Linux Enterprise Server 10, and SUSE Linux Enterprise Server 10 with Xen added, SUSE Linux Enterprise Server 11, SUSE Linux Enterprise Server 11 SP1
- VMware ESX Server 3.x, VMware ESX Server 3.5, Vmware ESX Server 4.0, and VMware Infrastructure 3

Notes:

1. For Red Hat Enterprise Linux version 3, Red Hat Linux version 9, and other Linux distributions that do not install the compatibility libstdc++ library, the following message might be displayed:

   ```
   ./asu: error while loading shared libraries:
   libstdc++-libc6.1-1.so.2: cannot open shared object file:
   No such file or directory.
   ```

   If this message is displayed, install the compat-libstdc++*.rpm that is included on the distribution media.

2. The ASU Version 3.60 does not update the PC-DOS and Sun Solaris executable files. No technical support is available for these two operating systems.

For BIOS-based servers, run the ASU on the system that contains the settings that you want to view and change. When the ASU performs setting configuration, it interacts with the physical hardware. On operating systems that require root or administrator access to interact with physical hardware, you must run the ASU from an account that has root or administrator access. (Windows and Linux require root and administrator access.)

The Remote Supervisor Adapter device drivers and Remote Supervisor Adapter II USB daemon are used to view and change Remote Supervisor Adapter or Remote Supervisor Adapter II settings. If there is no Remote Supervisor Adapter or Remote Supervisor Adapter II device driver for an operating system (such as PC-DOS), the ASU does not support changing Remote Supervisor Adapter or Remote Supervisor Adapter II settings for that operating system.

For baseboard management controller settings, an Intelligent Platform Management Interface (IPMI) device driver and a corresponding IBM IPMI mapping layer (if required) are required. The ASU supports interfacing with different versions of the IPMI device driver (OSA, MSI, Microsoft IPMI Driver, and OpenIPMI). For Microsoft Windows-based implementations, see "IPMI device driver support for Windows" on page 7. For Linux-based implementations, see "IPMI device driver support for Linux" on page 8.

Notes:

- You cannot use the ASU to configure Remote Supervisor Adapter II settings from Red Hat Enterprise Linux AS 2.1 because there is no Remote Supervisor Adapter II device driver for the Red Hat Enterprise Linux AS 2.1 operating system.
- You cannot use the ASU to configure Remote Supervisor Adapter or Remote Supervisor Adapter II settings from PC-DOS because there is no Remote Supervisor Adapter or Remote Supervisor Adapter II device driver for PC-DOS.
- You can view or change settings on a local server only.
- You cannot use the ASU to configure baseboard management controller settings from PC-DOS because there is no baseboard management controller device driver for PC-DOS.
- The PC-DOS version of the ASU does not support a multi-node-capable server when it is configured as multi-node.
- The PC-DOS version of the ASU does not support a server in which the BIOS settings are stored in the baseboard management controller (for example, an IBM System x3950 M2 server).
- You cannot unpack the Windows ASU packages on a server or workstation that is running a 64-bit version of Windows PE. You can unpack the Windows ASU packages on a server or workstation that is running any of the other Windows operating systems that support the ASU. (For a list of operating systems that support the ASU, see "Operating-system support" on page 2.)

Windows PE support

Windows PE requires some special considerations for the ASU to function correctly. These considerations are different for Windows PE 1.6 or earlier and Windows PE 2.0 (the version that is based on the Windows Vista operating system) or later.

Windows PE 1.6 or earlier

Note: ASU is not supported on WinPE 1.6 for IMM-based servers.

Windows PE 1.6 or earlier (the Windows PE versions that are based on Windows XP and Windows Server 2003) do not have temporary file storage, which affects how you extract the device driver from the ASU executable file.

The Windows PE 1.6 or earlier versions also cannot recognize new USB devices after startup, which affects the behavior of the Remote Supervisor Adapter II reset and restart commands.

Device driver extraction for read-only media

Windows operating-system support of the ASU requires the loading of a helper device driver. This device driver is embedded in the ASU executable file and, under normal circumstances, is extracted automatically at runtime to either a temporary directory or, if no "TEMP" environment variable is defined, the directory that contains the executable file, if it does not already exist. The device driver is then automatically loaded and used by the ASU. In the case of Windows PE, where no writeable temporary directory is defined, there must be an alternative way to load the device driver.

If the ASU is unable to extract the device driver to a temporary directory, it attempts to load the device driver from the directory where the executable file is located. If

the device driver exists, it is loaded, and execution continues. If the ASU does not find the device driver and cannot extract it for loading, an error message is displayed, and execution is halted.

If you are running the ASU from nonwriteable media (such as a CD), the device driver must exist on the media. You can manually extract it from the ASU executable file (and then copy it onto the nonwriteable media) by running the following command from a writeable media that contains the ASU:

`ASU extractdriver` (32-bit Windows version)

`ASU64 extractdriver` (64-bit Windows version)

The device driver is extracted as wflash.sys into the directory that contains the executable file. Then, you can copy the device driver (together with the ASU executable file) to nonwriteable media such as a bootable Windows PE CD.

Remote Supervisor Adapter II reset and restart problem with Windows PE

Windows PE versions that are earlier than Windows PE 2.0 recognize devices that are available only at startup time and later. This causes a problem when you issue a **resetrsa** or **rebootrsa** command to the Remote Supervisor Adapter II using the ASU because it removes the Remote Supervisor Adapter II from the bus. This prevents further communication with the device before a subsequent server restart.

If the version of Windows PE is earlier than 2.0, an information message is displayed indicating that the server must be restarted before you can run another ASU command for the Remote Supervisor Adapter II.

Windows PE 2.0 or later

The standard Windows PE 2.0 (based on the Windows Vista operating system) image is missing two packages that the ASU requires for operation. For instructions for adding optional Windows PE packages to the image that you create, see the documentation that comes with the Windows Automated Installation Kit.

To run the ASU, you must add the following packages:
- WinPE-MDAC-Package
- WinPE-WMI-Package

Windows PE 2.0 has an integrated Microsoft IPMI device driver, and the ASU 2.3.0 or later has an embedded mapping layer that supports that driver. Therefore, no user-installed IPMI driver or mapping layer is required. For more information about the IPMI device-driver support in Windows PE 2.0, see "IPMI device driver support for Windows" on page 7.

Windows Server 2008 support

The Microsoft Windows Server 2008 operating system supports the ASU version 2.3.0 or later. Windows Server 2008 has an integrated Microsoft IPMI device driver, and the ASU 2.3.0 or later has an embedded mapping layer that supports that driver. Therefore, no user-installed IPMI driver or mapping layer is required. For more information about the IPMI device-driver support in Windows Server 2008, see "IPMI device driver support for Windows" on page 7.

Supported systems

The ASU supports the servers, blade servers, and computers that are listed in the following two lists. The first list shows the supported systems that do *not* require an externally applied BIOS patch. The second list shows the supported systems that require an externally applied BIOS patch.

Note: The IMM-based servers do not require any type of patch; the settings definitions are included with the IMM firmware and the IBM System x Server Firmware.

For requirements and implementation details for applying BIOS patches to the ASU, see "Obtaining the ASU and patch files" on page 9 and "Using ASU patches" on page 10.

For the latest list of supported systems, see http://www.ibm.com/systems/support/.

The ASU supports the following systems that do *not* require an externally applied BIOS patch:

- IBM xSeries 366 Type 8863
- IBM xSeries MXE 460 Type 8874
- IBM System x3200 Types 4362 and 4363
- IBM System x3200 M2 Types 4367 and 4368
- IBM System x3200 M3 Types 7327 and 7328
- IBM System x3250 M2 Types 4190, 4191, and 4194
- IBM System x3250 M3 Types 4251, 4252, and 4261
- IBM System x3350 Types 4192 and 4193
- IBM System x3400 Types 7973, 7974, 7975, and 7976
- IBM System x3400 M2 Types 3808, 3809, 3815, 3817, 7836 and 7837
- IBM System x3400 M3 Types 7378 and 7379
- IBM System x3455 Types 7940 and 7941
- IBM System x3500 Type 7977
- IBM System x3500 M2 Type 3821, 3822, and 7839
- IBM System x3500 M3 Type 7380
- IBM System x3550 Types 1913 and 7978
- IBM System x3550 M2 Type 7946
- IBM System x3550 M3 Type 4254 and 7944
- IBM System x3620 M3 Type 7376
- IBM System x3650 Types 1914 and 7979
- IBM System x3650 M2 Type 7947
- IBM System x3650 M3 Type 4255 and 7945
- IBM System x3655 Types 7943 and 7985
- IBM System x3690 X5 Type 7148
- IBM System x3755 Types 7163 and 8867
- IBM System x3800 Types 8865 and 8866
- IBM System x3850 Types 7362, 7365, 8863, and 8864
- IBM System x3850 M2 Types 7141, 7144, and 7234
- IBM System x3850 X5 Types 7145 and 7146

- IBM System x3950 Types 7963, 8872, and 8878
- IBM System x3950 E Types 7364, 7367, 8874, and 8879
- IBM System x3950 M2 Type 7143
- IBM System x3950 X5 Type 7145 and 7146
- IBM System x iDataPlex dx320 Types 6388 and 7326
- IBM System x iDataPlex dx340 Types 7832 and 7834
- IBM System x iDataPlex dx360 M2 Types 7321, 7323, and 7375
- IBM System x iDataPlex dx360 M3 Type 6391
- IBM BladeCenter HS12 Types 1916, 8014, and 8028
- IBM BladeCenter HS20 Types 1884 and 8843
- IBM BladeCenter HS21 Types 1885 and 8853
- IBM BladeCenter HS21 XM Types 1915 and 7995
- IBM BladeCenter HS22 Types 1936 and 7870
- IBM BladeCenter HS22V Types 1949 and 7871
- IBM BladeCenter HX5 Type 7872
- IBM BladeCenter LS20 Type 8850
- IBM BladeCenter LS21 Type 7971
- IBM BladeCenter LS22 Type 7901
- IBM BladeCenter LS41 Type 7972
- IBM BladeCenter LS42 Type 7902

The ASU supports the following systems that require an externally applied BIOS patch:
- IBM eServer 326m Types 7969 and 7992
- IBM xSeries 206m Types 8485 and 8490
- IBM xSeries 236 Type 8841
- IBM xSeries 260 Type 8865
- IBM xSeries 306m Types 1887, 8491, and 8849
- IBM xSeries 346 Types 1880 and 8840
- IBM System x3250 Types 4364, 4365, and 4366
- IBM System x3455 Types 7984 and 7986
- IBM BladeCenter HC10 Type 7996
- IBM BladeCenter HS20 Types 1883 and 7981
- IBM IntelliStation A Pro Types 6217 and 6224
- IBM IntelliStation M Pro Types 6218, 6225, 6228, 9229, 9230, 9236, and 9237
- IBM IntelliStation Z Pro Types 6223, 9228, and 9232

IPMI device driver support for Windows

Microsoft Windows supports the OSA and MSI IPMI device drivers. If you are using Windows Server 2003 R2, Windows Server 2008, or Windows PE 2.0 or later, you do not have to install an external (OSA or MSI) IPMI device driver. Microsoft integrates the IPMI device driver with the operating system. The ASU supports these operating systems starting with ASU 2.3.0. To support the integrated device driver, a new IBM mapping layer is required. This new mapping layer comes embedded in the ASU 2.3.0 or later.

If you want to use the Microsoft IPMI device driver and you have previously installed the IBM mapping layer, you must remove the mapping layer. This mapping layer was required to support the OSA and MSI IPMI device drivers. The ASU embedded mapping layer conflicts with the externally installed and previously required IBM mapping layer.

The Microsoft IPMI device driver is automatically installed with Windows Server 2008 and Windows PE 2.0 or later. However, it is not installed by default with Microsoft Windows Server 2003 R2. To install the Microsoft IPMI device driver, complete the following steps:

1. Click **Start → Control Panel → Add/Remove Programs**.
2. Click **Add/Remove Windows Components**.
3. From the component list, select **Management and Monitoring Tools**, and then click **Details**.
4. Select **Hardware Management**.
5. Click **Next**. The installation wizard opens and guides you through the installation. The Windows Server 2003 R2 installation disk is required.

To verify that the Microsoft IPMI device driver is installed, complete the following steps:

1. Click **Start → Control Panel → System**.
2. Click the **Hardware** tab.
3. Click **Device Manager**.
4. Click **View → Show Hidden Devices**.
5. Expand **System devices**.
6. If the Microsoft IPMI device driver is installed correctly, a device named **Microsoft Generic IPMI Compliant Device** is displayed under **System devices**. For a multi-node configuration, a device named **Microsoft Generic IPMI Compliant Device** is created for each node.

Note: For an IBM System x3950 M2 server (multi-node configuration), the Microsoft IPMI device driver is the only supported IPMI driver.

IPMI device driver support for Linux

To access baseboard management controller settings, you must use either the OSA or MSI device driver and its corresponding IBM IPMI mapping layer, or use the OpenIPMI device driver, which comes in many Linux distributions.

Notes:

1. The default version of OpenIPMI that is in Red Hat Enterprise Linux 3 Update 6 and Red Hat Enterprise Linux 3 Update 7 has a timing issue; therefore, it is not supported by the ASU. In this case, the OSA IPMI mapping layer is required.
2. For an IBM System x3950 M2 server (multi-node configuration), the OpenIPMI device driver is the only supported IPMI driver.

Obtaining the ASU and patch files

The ASU, patch files, and device drivers that you need are available from the IBM Support Web site.

Note: Patch files are not required for IMM-based servers.

You must download the following files:

- ASU package. The package contains the ASU tool and the additional files that are required for IMM-based servers. The additional files for IMM-based servers are required for configuring and activating the IMM LAN over USB interface.
- **(For BIOS-based servers only)** BIOS patch file for the server. (To determine if an external BIOS code patch is required, see "Supported systems" on page 6.)
- **(For BIOS-based servers only)** Remote Supervisor Adapter device drivers or Remote Supervisor Adapter II USB daemon (if you want to use the ASU to configure Remote Supervisor Adapter or Remote Supervisor Adapter II settings).
- **(For BIOS-based servers only)** Baseboard management controller device drivers and, if required, mapping layer (if you want to use the ASU to configure baseboard management controller settings).

Note: Changes are made periodically to the IBM Web site. Procedures for locating firmware and documentation might vary slightly from what is described in this document.

To locate and download the files that you need, complete the following steps:

1. To download the IBM Advanced Settings Utility, go to http://www-947.ibm.com/systems/support/supportsite.wss/docdisplay?lndocid=TOOL-ASU &brandind=5000016 or complete the following steps:
   a. Go to http://www.ibm.com/systems/support/.
   b. In the left pane, click **Systems Management software**.
   c. Under **Popular links**, click **Tools and utilities**.
   d. Under Configuration, click **Advanced Settings Utility**.
2. **(For BIOS-based servers only)** To download the BIOS definition file, complete the following steps:
   a. On the Support page for the server, select **Software and device drivers**.
   b. On the "Software and device drivers" page, click **BIOS**.
   c. Select **BIOS definition file for use with IBM Advanced Setting Utility**. (The BIOS definition file *must* match the BIOS level that the server is running.)
   d. Select the ASU BIOS definition .exe file.
   e. Follow the instructions that guide you through the download process.
3. **(For BIOS-based servers only)** After you download the BIOS definition file (if needed), return to the "Software and device drivers" page to install the device-driver software for the Remote Supervisor Adapter II and baseboard management controller.
4. Download the IPMI device driver only if required. The IPMI device driver is not required in Windows 2003 R2, Windows 2008, and Linux operating systems that support OpenIPMI. For more information, see "IPMI device driver support for Windows" on page 7 or "IPMI device driver support for Linux" on page 8. On the "Software and device drivers" page, click **OSA IPMI**, and then select the applicable device-driver file. Read the .txt file and follow the instructions that guide you through the download process.

5. After you have completed the download of the OSA IPMI device driver file, if it was required in step 4 on page 9, select the applicable **Mapping layer for OSA IPMI** file. Read the .txt file and follow the instructions that guide you through the download process.

6. **(For BIOS-based servers only)** On the "Software and device drivers" page, click **Remote Supervisor Adapter II**, and then select the applicable **RSA II daemon**. Read the .txt file and follow the instructions that guide you through the download process.

Using ASU patches

Note: Patch files are not required for IMM-based servers.

The ASU requires configuration information for the baseboard management controller, BIOS, and Remote Supervisor Adapter II before you can perform ASU operations that are targeted to the selected device. This configuration information is contained in a patch. There are separate patches for the baseboard management controller, BIOS, and Remote Supervisor Adapter II.

Starting with ASU 2.0, the ASU contains the baseboard management controller and Remote Supervisor Adapter II patches. Depending on the server model and BIOS level, a BIOS code patch might be required. For a list of servers that require a BIOS patch file, see "Supported systems" on page 6. For the systems that do not require a BIOS patch file, the BIOS patch is embedded in the BIOS ROM. If the BIOS patch is required, you must download the BIOS patch (definition file) from the IBM Web site and add it to the ASU (only if you are required to perform an ASU command by using the BIOS settings).

To determine whether the BIOS patch is already available for the ASU, issue the ASU **patchlist** command to display the current available patches. If the ASU determines that a BIOS patch file is embedded in the BIOS, it displays the BIOS patch files that are available (see the following example).

Example of ASU output when the **patchlist** command is issued with the BIOS patch *not* listed:

```
Patch 1: <XX[00->99] (BMC)>
Patch 2: <XX[00->99] (RSA)>
```
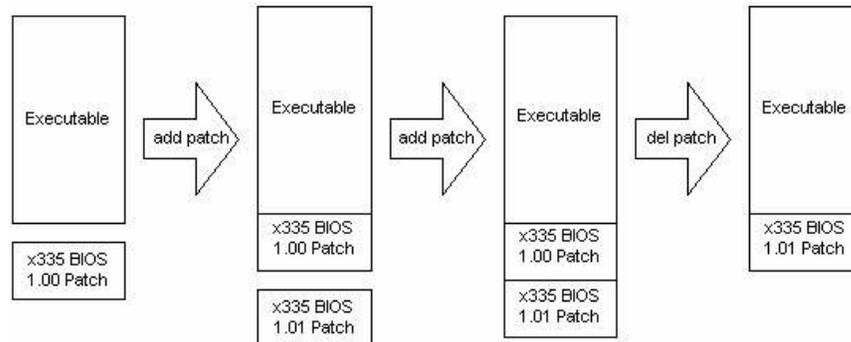
Example of ASU output when the **patchlist** command is issued with the BIOS patch listed:

```
Patch 1: <XX[00->99] (BMC)>
Patch 2: <XX[00->99] (RSA)>
Patch 3: <D0[14->14] (BIOS)>
```

If the BIOS patch is not listed, you must add the patch before you can change or view BIOS settings. You can either add or remove a patch from the ASU.

The following illustration shows how patches are added and removed in the ASU binary code. For each BIOS code and firmware type, the internal locations of the settings vary. A patch informs the ASU where the settings are located for a single BIOS code version.

If a BIOS patch is needed and is not listed when you issue the **patchlist** command, you can download the selected BIOS definition file (patch file) from the IBM Web site and perform a patch add, using the ASU.

When you run the ASU, it automatically scans the patches that are available and determines whether the applicable patch exists for the setting that you want. If an applicable patch exists, the ASU applies the setting. If the patch does not exist, the ASU returns an error.

A patch that is added remains until you run the **patchremove** command on that patch.

Notes:

1. You cannot remove the BIOS patches that are embedded in the BIOS ROM by running the **patchremove** command.
2. Only one patch is supported for any major version of BIOS. For example, if there are BIOS versions 19A and 19B, only one patch is supported for both.

Unpacking the ASU files for Windows

To unpack the ASU files if you are using a Windows operating system, complete the following steps:

1. Change to the directory that contains the downloaded ASU files.
2. Choose one of the following methods to unpack the ASU files:
   - If you are using Windows, double-click **filename.exe**, where *filename* is the file name for the Advanced Settings Utility for Windows that you downloaded.

     The files are automatically extracted to the same directory.
   - From a command prompt, type `filename.exe`, where *filename* is the file name for the Advanced Settings Utility for Windows that you downloaded.

Notes:

1. You must run the **asu** commands from the directory in which the ASU files are located.
2. The ASU requires additional files for IMM-based servers. These files are required to automatically configure and activate the LAN over USB interface. The ASU uses the LAN over USB interface as a connectivity option. For more information about connectivity options, see "Command connectivity options" on page 27.

Unpacking the ASU files for Linux

To unpack the ASU files if you are using a Linux operating system, complete the following steps:

1. Open an xterm or other terminal window.
2. Change to the directory that contains the downloaded ASU files.
3. From a shell command prompt, type one of the following commands and press Enter.

   If the .tgz file for ASU was downloaded:

   "`tar -zxvf` *filename*`.tgz`", where *filename* is the file name of the Advanced Settings Utility for Linux that you downloaded.

   The files are automatically extracted to the same directory.

   If the .rpm file for ASU was downloaded:

   "`rpm -Uvh` *filename*`.rpm`", where *filename* is the file name of the Advanced Settings Utility for Linux that you downloaded.

   The files are extracted to the `/opt/ibm/toolscenter/asu` directory.

Notes:

1. You must type the **asu** commands from the directory in which the ASU files are located.
2. The ASU requires additional files for IMM-based servers. These files are required to automatically configure and activate the LAN over USB interface. The ASU uses the LAN over USB interface as a connectivity option. For more information about connectivity options, see "Command connectivity options" on page 27.

Using the asu command

The ASU uses the **asu** or **asu64** command. Some ASU commands examples are shown in the following list:

- To see all of the **asu** command modes and options, type the following command:

  `asu` (for a 64-bit operating system, type `asu64`)

- To change a value, type the following command:

  `asu set` *setting value*

- To show the current value, type the following command:

  `asu show` *setting*

- To show all possible values, type the following command:

  `asu showvalues` *setting*

- To add a patch, type the following command:

  `asu patchadd` *filename*`.def`

  where *filename* is the file name of the definition file.

- In the commands, *setting* is the name of a setting that you want to view or change, and *value* is the value that you are placing on the setting.
- If *value* contains spaces, enclose the value string in quotation marks (").
- If you are using a Linux operating system, you must either add a period (.) to the path environment variable or type `./` before each **asu** command. For example, type `./asu`, or for a 64-bit operating system, type `./asu64`.

Using the ASU to configure Remote Supervisor Adapter or Remote Supervisor Adapter II settings

You can use the ASU to directly configure a Remote Supervisor Adapter or Remote Supervisor Adapter II.

Note: Be sure to install the Remote Supervisor Adapter device drivers or Remote Supervisor Adapter II USB daemon *before* you use the ASU. To install the device drivers, see the *IBM Remote Supervisor Adapter II Installation Instructions for Microsoft Windows Users* or *IBM Remote Supervisor Adapter II Installation Instructions for Linux Users*, which are available at http://www.ibm.com/systems/support/.

Setting up communication with the ASU

Before you can use the ASU to modify Remote Supervisor Adapter or Remote Supervisor Adapter II settings, you must configure the operating-system setting so that the ASU can communicate correctly with the Remote Supervisor Adapter or Remote Supervisor Adapter II. Use the Configuration/Setup Utility program that is part of the system BIOS code to configure the operating-system setting.

To configure the operating-system setting by using the Configuration/Setup Utility program, complete the following steps:

1. Turn on the system.
2. When the prompt `Press F1 for Configuration/Setup` is displayed, press F1.
3. On the Configuration/Setup Utility main menu, select **Advanced Setup**, and then select **RSA Settings**.
4. Select **Other OS** for a Windows operating system, or select **Linux OS** as the operating-system USB selection.
5. Select **Save the Values and Reboot RSA**, and then press Enter. Wait until the message `RSA Settings saved` is displayed.
6. Exit the Configuration/Setup Utility program and complete the startup of the operating system.

Example of configuring the Ethernet settings on a Remote Supervisor Adapter II

The Remote Supervisor Adapter II requires configuration to enable remote access to the adapter through the adapter Ethernet and serial connectors.

For detailed information about using the Remote Supervisor Adapter II Web interface for remote access, see the *IBM Remote Supervisor Adapter II User's Guide*.

You can use the ASU to configure the Remote Supervisor Adapter II Ethernet settings.

Note: If you have an accessible, active, and configured Dynamic Host Configuration Protocol (DHCP) server on your network, the host name, IP address, gateway address, and subnet mask are set automatically. You can use the Configuration/Setup Utility program that is part of the server BIOS to select DHCP server settings. For more information, see the *IBM Remote Supervisor Adapter II Installation Instructions for Microsoft Windows Users* or *IBM Remote Supervisor Adapter II Installation Instructions for Linux Users*, which is available at http://www.ibm.com/systems/support/.

You can also configure the DHCP setting by using the ASU. To use the ASU, continue with the following procedure.

If you have an enabled DHCP server and you want to configure the serial connector, go to "Example of configuring the serial connection on a Remote Supervisor Adapter II."

To configure the Ethernet settings if you do not have a DHCP server on your network, complete the following steps.

Note: If you are using a Linux operating system, be sure to type `./` before `asu`.

1. If you have not already done so, unpack the ASU files. For more information, see "Unpacking the ASU files for Windows" on page 11 or "Unpacking the ASU files for Linux" on page 12.
2. At the command prompt, change to the directory that contains the ASU files.
3. If you have not already done so, add the Remote Supervisor Adapter/Remote Supervisor Adapter II ASU patch. For more information, see "Patchadd command" on page 63 and "Patchlist command" on page 65.

    Note: This step is not required if you are using ASU version 2.0 or later.
4. To view a list of all settings and their assigned values, type the following command and press Enter:

    `asu show all`
5. From the following list, select the items that you want to set:
    - To enable the network interface on the Remote Supervisor Adapter II, type the following command and press Enter:

        `asu set RSA_Network1 Enabled`
    - To set the IP address, type the following command and press Enter:

        `asu set RSA_HostIPAddress1 192.169.70.140`
    - To set the subnet mask, type the following command and press Enter:

        `asu set RSA_HostIPSubnet1 255.255.255.0.`
    - To set the gateway IP address, type the following command and press Enter:

        `asu set RSA_GatewayIPAddress1 192.168.70.1`
    - To enable DHCP, type the following command and press Enter:

        `asu set RSA_DHCP1 Enabled`
    - To set the data-transfer rate, type the following command and press Enter:

        `asu set RSA_LANDataRate1 "10M Ethernet"`
    - To set duplex mode, type the following command and press Enter:

        `asu set RSA_Duplex1 Full`
6. Type `asu rebootrsa` and press Enter.

Example of configuring the serial connection on a Remote Supervisor Adapter II

You can use the ASU to configure the Remote Supervisor Adapter II serial connection. The serial connector connects to a modem for dial-out support only.

To configure the Remote Supervisor Adapter II serial connection for access to a modem, complete the following steps.

Note: If you are using a Linux operating system, be sure to type `./` before `asu`.

1. If you have not already done so, unpack the ASU files. For more information, see "Unpacking the ASU files for Windows" on page 11 or "Unpacking the ASU files for Linux" on page 12.
2. At the command prompt, change to the directory that contains the ASU files.
3. To view a list of all of the settings and assigned values, type the following command and press Enter:

   `asu show all`
4. From the following list, select the items that you want to set:
   - To change the modem baud rate, type the following command and press Enter:

     `asu set RSA_ModemBaudRate1 value`

     where *value* is a number from 2400 through 57600 in increments of 2400 (for example, 2400, 4800, 7200, 9600, ..., 57600). The default is 57600.

     Note: Make sure that the baud rate matches the baud rate of the device that you are connecting to the serial connector on the Remote Supervisor Adapter II.
   - To change the modem parity, type the following command and press Enter:

     `asu set RSA_ModemParity1 value`

     where *value* is None, Odd, Even, Mark, or Space. The default is None.
   - To change modem stop bits, type the following command and press Enter:

     `asu set RSA_ModemStopBits1 value`

     where *value* is either 1 or 2. The default is 1.
5. Type `asu rebootrsa` and press Enter.

Using the ASU to configure settings in IMM-based servers

The ASU 3.00 supports configuring settings on IMM-based servers. The ASU uses the same set of commands and syntax that is used by previous versions of the ASU tool. Some commands are enhanced to manage and display groups of settings. This includes new classes that are used as filters if you display the supported settings by using the **show** set of commands.

Notes:
1. To use all of the new functions in the ASU 3.01, the IMM firmware level must be version 1.05 or later.
2. For IMM-based servers: After you use the ASU to change settings, you must reset the IMM before you flash new firmware; otherwise, the changes to the settings might be lost. To reset the IMM, run the **asu rebootimm** command.

The following sections describe the functions that are available to support IMM-based servers with the ASU version 3.00 or later.

Connectivity

In IMM-based servers, all firmware settings are configured through the IMM. The ASU can connect to the IMM locally (in-band) through the Keyboard Controller Style (KCS) interface or through the LAN over USB interface. The ASU can also connect remotely over the LAN.

The IMM comes with a LAN over USB interface that can be configured and activated on the running operating system. After you install and configure the corresponding information file, the ASU can be connected to the IMM. The local

LAN over USB connection requires authentication. A new set of connectivity parameters are required when the ASU is connected over the LAN.

The local connection over the KCS interface does not require authentication and follows the online connecting model and command structure of BIOS-based servers, where no connectivity parameters are required. If you do not specify any connectivity parameters, the ASU attempts to connect to the IMM by using default LAN settings on the LAN over USB interface. If the ASU is unable to connect to the IMM over the LAN, it automatically connects over the KCS interface, provided that the correct IPMI device drivers or mapping layers are installed. For more information about the KCS interface, see the Intelligent Platform Management Interface Specification.

In the following example, the ASU attempts to connect through the LAN over USB interface. If the LAN over USB interface connection has failed, the ASU attempts to connect through the KCS interface.

```
asu show
```

You can request that the ASU connect locally, exclusively using the KCS interface, which avoids the automated connection over the LAN over USB interface (and the fallback to the KCS interface). If you use the **--kcs** option, the ASU communicates through the KCS interface only. In the following example, the ASU attempts to connect through the KCS interface only, without the need for authentication parameters.

```
asu show --kcs
```

You can also request that the ASU connect locally, exclusively using the LAN over USB interface, by specifying the **--host** connectivity option. The ASU does not attempt to fall back to use the KCS interface when this option is specified. In the following example, the ASU attempts to connect through the LAN over USB interface only by using the default user ID and password account.

```
asu show --host 169.254.95.118
```

When the ASU runs any command on an IMM-based server, it attempts to connect and automatically configure the LAN over USB interface, if it detects that this interface is not configured. The ASU provides a level of automatic and default settings. You have the option of specifying that the automatic configuration process is skipped, if you have manually configured the IMM LAN over USB interface by using different settings than those used by the ASU application default settings.

You can use the **--noimmlancfg** option to skip automatic configuration process, if you have manually configured the interface. For example, if you want to show the IMM list of settings and avoid using the automatic configuration process when you attempt to connect through the LAN over USB interface, type the following command:

```
asu show IMM --noimmlancfg
```

If the ASU is connecting remotely to the IMM over the LAN, there is no requirement for the remote operating system of the targeted IMM to be booted (online). The ASU can connect to the IMM remotely when the server is connected to power or is using standby power.

To connect remotely, the **--host** option is required. The following example indicates the minimum required parameter when the ASU connects remotely through the LAN to the IMM external port.

```
asu show --host target_IMM_external_IP_address
```

Enabling and disabling the LAN over USB interface

You can use the ASU to enable or disable the IMM LAN over USB interface by using the **IMM.LanOverUsb** setting. When you enable or disable this setting, you must use the KCS interface because the LAN over USB interface is removed during the **set** command process. This prevents the ASU **set** command from terminating correctly. You must use the **--kcs** connectivity option to make sure that the **asu** command is completed correctly and relates status.

```
asu set IMM.LanOverUsb Disabled --kcs
```

Examples:

To connect remotely to an IMM from a Windows client to display all available settings, type the following command. The IMM external IP address is 9.5.51.37.

```
asu show all --host 9.5.51.37 --user testid —password test
```

To connect locally to an IMM from a Windows operating system to display all available settings, type the following command. The ASU connects to the IMM through the LAN over USB interface.

```
asu show all --user testid —password test
```

To connect locally to an IMM from a Windows operating system to display all available settings, type the following command. The ASU attempts to connect over the LAN over USB interface by using the default IMM authentication credentials. If the default settings do not match, the ASU attempts to use the KCS interface, provided that the IPMI device drivers and mapping layers are installed.

```
asu show all
```

To connect locally forcing the ASU to use the KCS interface and avoid using the LAN over USB interface, type the following command:

```
asu show --kcs
```

For more information about connectivity parameters and usage, see "Command connectivity options" on page 27.

Settings syntax

All settings in IMM-based servers are configured through the IMM and the settings are classified into groups of settings. The term *group* is used in this document to refer to how the settings for a specific subsystem are grouped together.

The following groups of settings or firmware settings are supported:
- UEFI – UEFI (BIOS) settings.
- IMM – IMM settings. RSA/BMC functionally equivalent settings.
- BootOrder – Boot-order configuration in UEFI.
- iSCSI – iSCSI supported settings (Boot-over-iSCSI settings).
- SYSTEM_PROD_DATA – User-configurable VPD settings.

There is a new settings syntax to use for identifying the corresponding group of settings. The setting name is preceded by the corresponding group name:

group_name.setting_name

Example:

To set the external IP address in the IMM, type the following command:

`asu set IMM.HostIPAddress 9.5.51.37`

where IMM.HostIPAddress is the IMM setting that is used to configure the IMM external IP address. The setting is part of the IMM group.

There are settings that include an additional index, which is referred to in this document as an *instance*. The index is used to identify and set different instances of the same setting when these are available.

Example:

To set the first instance of a number or login ID, type the following command:

`asu set IMM.LoginId.1 testid`

where IMM.LoginId.1 is the IMM setting that is used to configure the first instance of a login account.

Instances of settings

Support for instances of settings on IMM-based servers was introduced in the ASU 3.0. The ASU 3.01 is extending the support of instances in several different ways. Commands have been created or modified to provide more information about instances and ways to create and delete them.

Instances are denoted by adding a dot, followed by the instance number to the end of the setting name. For example, if the setting name is "IMM.LoginId," instance number 1 of the setting is "IMM.LoginId.1".

Note: There is an exception to the naming convention for single instances. Single instances do not have the dot followed by an instance number. The setting instead appears like a non-instance setting. Single instance settings are denoted in the output of the **showvalues** command by having a maximum number of instances of 'single'. For example, the setting iSCSI.initiatorName is a 'single instance'. Notice that the usual dot followed by an instance number is not used. If the single instance exists, the setting iSCSI.initiatorName is displayed in the **show** command output. If it does not exist, the setting is not displayed in the **show** command output.

Instance settings are now defined to have a minimum and maximum number of allowed instances. To determine which settings can have instances and the minimum and maximum number of instances allowed, use the **showvalues** command with the new **--instances** parameter. The output of this command is detailed in "Showvalues command" on page 89.

Creating and deleting instances

Beginning with the ASU 3.01, you can create or delete instances. Use the **set** command to create an instance. If the instance does not already exist, and the instance number is between 1 and the maximum number of allowed instances, the instance is automatically created and set to the value specified in the **set** command.

Use the new **delete** command to delete an instance. This command deletes the instance, if deleting the instance does not cause the number of instances for the setting to go below the minimum number of allowed instances for the setting.

Note: There are restrictions for creating and deleting instances of settings that are part of a record. For more information about the restrictions, see "Record management."

Record management

Record management is a new feature in the ASU 3.01. Settings that have instances can be part of a record. A record is basically a group of settings that have dependencies on each other. For example, a user ID and a password are dependent on each other. A user ID must have a password and a password must have a user ID. Therefore, they are grouped in the same record.

Each record has a setting that is defined as the "record key". It represent the primary setting for the record.

To determine if a setting is part of a record, use the **showvalues** command with the **--instances** parameter. Settings that are part of a record are marked with the text "recordKey" (if the setting is the record key) or "recordKey=key_name" (if the setting is part of a record but is not the key), where key_name is the name of the setting that is the record key. Examples of the **showvalues** output for settings that are part of a record are shown in "Showvalues command" on page 89.

All settings in a record are created or deleted as a group. To create an instance of a record, you must first perform a "set" on the key setting of the record. This automatically causes an instance to be created and set to its default value for all other settings in the record. For examples of creating an instance of a setting, see the "Set command" on page 79.

To delete an instance of a record, the **delete** command is performed on the "record key" setting. This automatically deletes all other instances for the settings in the record. For examples of deleting an instance of a setting, see the "Delete command" on page 47.

Using the RDCLI to mount an image to a remote IMM-based server

The Remote Disk Command Line Interface (RDCLI) is available in ASU 3.60. RDCLI is a standalone tool that you can use to mount an ISO image file or DVD or CD drive to a remote IMM-based server. You can access the mounted ISO image file or DVD or CD drive as a local hardware device.

Notes:
1. To use the RDCLI and mount an ISO image file or DVD or CD drive to a remote IMM-based server, make sure that the server has a virtual media key installed and that the remote presence feature is enabled. For more information, see the documentation that comes with the server.
2. RDCLI does not work if there is a remote presence session started through the IMM web interface.

3. RDCLI, is supported in only rack IMM-based servers. RDCLI does not support IMM-based blade servers.

The RDCLI supports the following operating systems:
- SUSE Linux Enterprise Server 10 (32 / 64 bit)
- SUSE Linux Enterprise Server 11 (32 / 64 bit)

Limitation
This version of RDCLI supports only ISO image files and DVD or CD drives.

Package
The RDCLI package contains the following two binary files that are packaged in the rdcli32 and rdcli64 directories in the ASU Linux distribution:

rdmount
> Performs authentication and creates new file server processes that enable access to the virtual disk. It can also query the existing file server processes.

rdumount
> Binary file that is used to unmount an ISO image file or DVD or CD drive from a remote IMM-based server.

Command-line parameters
- -s *address_or_hostname*

  The address or host name of the remote IMM.
- -d *path*

  The image or local optical drive directory path.
- -l *login*

  Authorized login user to the IMM.
- -p *password*

  Password of the authorized login user.
- -w *port*

  Authentication port used to communicate with the IMM.

Chapter 2. Using the command-line interface

This chapter describes how to use the Advanced Settings Utility (ASU) command-line interface.

Command syntax

Read the following guidelines before you use the command-line interface:
- Each command has the following format:

  asu [*application*] [*command* [*command_modifier*] | [*class*]]
  [*options*] [*connect_options*]

- Each command starts with asu or asu64.
- The optional configuration *application* can be either of the following options (see "Command configuration applications"):
  – savestat
  – immcfg
- *command* is one of the commands that are listed in Chapter 3, "Command reference," on page 41.
- *command_modifier* is one or more options that apply only to a certain command. These are considered command modifiers or extensions. Each command modifier must be preceded by a double hyphen (--). (See "Command modifiers" on page 25.)
- *class* is a filter that acts on a listing of settings (settings display filters). A class is not considered a modifier or option. Classes are also used to operate a command upon a group or class of settings. A class does not require the double hyphen (--) as part of the syntax. For more information about a class, see "Classes of settings" on page 23.
- *option* is one or more general options that globally apply to the operation. Each option requires the preceding double hyphen (--) as part of its syntax. (See "Command general options" on page 28.)
- *connect_option* is one or more parameters that are related to the ASU connection to the IMM. Connect options are defined as options. Each requires the double hyphen (--) as part of its syntax. (See "Command connectivity options" on page 27.)
- Brackets ([ ]) indicate that an application, option, or class is optional. Brackets are not part of the command that you type.

Command configuration applications

The ASU integrates additional configuration applications. The following applications are included in the ASU:
- savestat
- immcfg

The following sections describe each of the applications and outline the command syntax and structure.

Savestat

The savestat application is used by the Scripting Toolkit to save and restore the state information about the system. This function is supported for both IMM-based servers and some BIOS-based servers. The use cases for these are identical, other than the following differences:

- A file called `savestat.def` is used for BIOS-based servers. It is supplied in the Scripting Toolkit run environment.
- Using savestat on IMM-based servers might require additional connectivity parameters because the data is in the IMM on the server.

Examples
savestat write data example:

```
asu savestat write datafile
```

This command uses the contents of the `datafile` file to update the persistent storage contents.

savestat read data example:

```
asu savestat read datafile
```

This command reads the contents of the persistent storage and writes the results to the `datafile` file. The file is created if it does not already exist. If the file already exists, it is overwritten.

IMM LAN over USB interface configuration

The ASU provides the ability to install and configure the LAN over USB interface that is used to communicate with the IMM as an add-on configuration application called *immcfg*. The command syntax is:

```
asu immcfg [application_commands]
```

The supported commands and the operating systems for which these are available are listed in Table 1.

The additional files that are required to perform these commands are included with the ASU package. The following required files must be in the directory from which the ASU is run:

For Windows:

```
ibm_rndis_server_os.inf
device.cat
```

For Linux:

```
cdc_interface.sh
```

Table 1. IMM LAN over USB configuration application commands

Configuration application command	Description	Operating system support	Command syntax to set the command

setip	Sets the operating system IP address for the LAN over USB interface	Windows only	`asu immcfg --setip`
			`[ [--ip ip_address]`
			`[--ipmask ipmask] ]`

detectdevice	Detects whether the IMM LAN over USB interface is activated	Windows only	`asu immcfg -—detectdevice`

Table 1. IMM LAN over USB configuration application commands (continued)

Configuration application command	Description	Operating system support	Command syntax to set the command
installdriver	Installs the IMM LAN over USB device driver	Windows only	`asu immcfg --installdriver [--inf inf_path/name]`
detectdriver	Detects whether the device driver for the IMM LAN over USB interface is installed in the operating system	Windows only	`asu immcfg -–detectdriver`
autocfg	Automatically configures the IMM LAN over USB interface using all default settings. The default settings include packaged .inf, .cat, and .sh files. The default IP address is 169.254.95.119 for the operating system IP address. **Note:** The IMM IP address is 169.254.95.118; however, this address can change if the default address is used by another device.	Windows Linux VMware	`asu immcfg --autocfg`

Classes of settings

For commands that support operating on multiple settings, classes are used to indicate groups of settings. The commands that support classes are comparedefault, help, loaddefault, show, showdefault, showlocation, and showvalues.

The ASU classes of settings are described in the following list:

all
> This class includes all settings that are listed in the ASU for the Remote Supervisor Adapter or Remote Supervisor Adapter II, baseboard management controller, and BIOS.

authentication
> This class includes all settings that are classified as authentication settings, which include passwords, user IDs, and authority-related settings.
>
> The **save** and **restore** commands do not save or restore this class of settings.
>
> You can list the user IDs and authority-related settings by using the **show** command.
>
> You cannot list the password settings by using the **show** command. To list the password settings, use the **showvalues** command.
>
> **Example:**
>
> List the settings defined as authentication, including the password settings. Password settings are normally not displayed when you use the **show** command. To display the available password, use the **showvalues** command with the password class:

```
asu showvalues authentication
```

backupctl

This class lists all settings that are not restored when you run the **restore** command. An additional flag is required for these settings to be included during a restore operation. For more information, see the "Restore command" on page 74.

Class filter for the **show, showvalues, showdefault**, and **showlocation** commands.

To list the settings that are not restored if saved, type the following command:

```
asu show backupctl
```

bios

This class includes all settings that match the installed BIOS code level.

bmc

This class includes all settings that are identified as baseboard management controller settings.

change

This class includes all settings that are not in the reboot class that can be changed safely and changed back before the system is restarted.

critical

This class includes all settings that you cannot change safely before a restart but that *must* be tested.

group

This class includes all settings that belong to the specified group. To view the supported groups, use the **showgroups** command.

Settings in BIOS-based servers and IMM-based servers are cataloged in classes or groups. If specified, the class is used as a filter for the command that is displaying or operating on the settings.

Class filter for the **show, showvalues, showdefault, showlocation, save, restore**, and **replicate** commands.

Examples:

To list the settings that are part of the IMM group (IMM-based servers):

```
asu show IMM
```

To list the settings that are part of the BIOS group (BIOS-based servers):

```
asu show bios
```

nochange

This class includes all changeable settings that are not in the reboot and change classes.

noreplicate

This class lists all settings that are not replicated when you run the **replicate** command. These settings are usually unique to each system.

Class filter for the **show, showvalues, showdefault**, and **showlocation** commands.

Example:

To list the settings that are not replicated:

```
asu show replicate
```

password

This class lists all settings that are classified as password settings.

Password setting values are not displayed during a **show** command.

Passwords setting are not allowed to be set remotely.

This class filter can be used with the following commands to list the settings classified as password: **showvalues, showdefault**, and **showlocation** (BIOS-based servers only) commands.

Example:

List the settings that are defined as password settings. Password settings are normally not displayed when you use the **show** command. To display the available password, use the **showvalues** command with the password class:

```
asu showvalues password
```

rsa

This class includes all settings that are identified as Remote Supervisor Adapter and Remote Supervisor Adapter II settings.

readonly

This class includes all settings that are read-only (for example, settings that you cannot change).

reboot

This class includes all settings that can be changed safely before a restart. If changing a setting does not preclude starting from the hard disk drive on the next startup, the setting is in this class.

writeonly

This class includes all settings that are write-only (for example, settings that you can change but that cannot be read, such as passwords).

Command modifiers

Command modifiers are optional and are used to modify the default operation of only specific commands. Table 2 describes the command modifiers and the commands to which they are applicable.

Table 2. Command modifiers

Command modifier	Description	SyntaxLAN-over-USB

--group	Optional command modifier for the **show**, **showdefault**, **showvalues**, **showlocation**,and **save** commands.
	
	When used with a supporting command, it is used to specify the name of a group section.
	
	The group_name is obtained by running the **showgroups** command.
		`asu show [--group group_name]`
		
		`asu save [--group group_name]`
		
		Obtain the *group_name* by running the following **showgroups** command:
		
		`asu showgroups`

--setlist	Optional command modifier for the **show**, **showdefault**, **showvalues**, **showlocation**, and **save** commands.
	
	When used with a supporting command, it is used to specify a list of settings on which the command will operate.
		`asu show [--setlist name1 name2 nameN]`
		
		`asu save [--setlist name1 name 2 nameN]`

Table 2. Command modifiers (continued)

Command modifier	Description	SyntaxLAN-over-USB
--excbackupctl	Optional command modifier for the **save** command. Use this command modifier during a **save** command to exclude saving the backup control settings in a file. The default in a **save** command is to include all backupctl settings. To review the backupctl settings, use the **asu show backupctl** command. This is mutually exclusive to the **incbackupctl** command modifier.	`asu save file_name [--excbackupctl]`
--incbackupctl	Optional command modifier for **restore** command. Use this command modifier during a **restore** command to include the backup control settings. The backupctl settings are not restored by default. To review the backupctl settings, use the **asu show backupctl** command. This is mutually exclusive to the **excbackupctl** command modifier.	`asu save file_name [--incbackupctl]`
--instances	Optional command modifier for the **showvalues** command. Use this command modifier during a **showvalues** command to show the names of settings that can have instances. This is mutually exclusive to the group command modifier.	asu showvalues [--instances]
--help	Optional command modifier for all commands and applications. Use this command modifier to show the help text for an ASU command or application.	asu command | application --help

Command connectivity options

The command connectivity options are a set of parameters that relate to the ASU connection to the IMM.

The connectivity options might be required when you connect the ASU to a local IMM. When you attempt to connect the ASU to a local IMM and none of the connectivity parameters are specified, the ASU attempts to connect to the IMM by using the default LAN over USB interface settings. The default settings include user ID, password, and host. The values for these settings are defined by the IMM hardware. If none of these settings are specified on a local connection and the default LAN over USB interface settings fail, the ASU attempts to connect over the KCS interface. The KCS interface does not require any of these parameters.

To connect the ASU remotely to the IMM, the **host** parameter is required. The **user** and **password** parameters are optional. If these parameters are not specified, the default settings are used.

Any local or remote LAN over USB connection requires authentication. The default authentication parameters are the default user ID and password that comes configured with each IMM. For the default authentication settings, see the documentation that comes with the server or optional devices.

Table 3. Command connectivity options

Connectivity option	Description	Syntax
host	Specifies the host name or IP address of the IMM to which the ASU should connect.	`asu cmd [command_modifier] options --host host_name|ip` `asu cmd [command_modifier] options --host=host_name	ip`

user	Specifies the user name to use when you are authenticating with the IMM.	`asu cmd [command_modifier] options --user user_id`
		
		`asu cmd [command_modifier] options --user=user_id`

password	Specifies the password to use when you are authenticating with the IMM.	`asu cmd [command_modifier] options --password password`
		
		`asu cmd [command_modifier] options --password=password`

password-file	Specifies the name of a file that contains the password to use when you are authenticating with the IMM.	`asu cmd [command_modfier] options --password-file file_name`
		
		`asu cmd [command_modfier] options --password-file=file_name`

kcs	Specify to send commands by using the KCS IPMI interface only. This option requires the IPMI device driver. For more information, see "IPMI device driver support for Windows" on page 7 or "IPMI device driver support for Linux" on page 8. This option does not require authentication.	`asu cmd [command_modifier] options --kcs`

Command general options

This section describes the command general options that are available.

Bypass command option (ASU version 2.0 or later)

Note: The bypass command option is not available for IMM-based servers.

If you try to run the **patchadd** command on a system on which a BIOS CMOS patch file is already installed in the BIOS ROM and the definition file that you use has the same BIOS code level as the system, an error message is generated and the ASU is not modified. Also, if you issue the **patchremove** command and select the BIOS CMOS patch, an error message is generated, and the ASU does not remove this patch.

To add and use a BIOS CMOS definition patch file on a system in which the BIOS ROM has a BIOS CMOS patch with the same BIOS code level, use the bypass option (`--bypass`) at the end of the **asu** command.

Examples:
```
asu  patchadd  GG16A.def --bypass
asu patchlist  --bypass
asu show all --bypass
```

When you run the **asu** command with the bypass option, the ASU ignores the BIOS CMOS patch information in BIOS ROM.

dumptofile command option (ASU version 2.4 or later)

Use the dumptofile option to redirect all output that is produced by the ASU to a log file.

Usage
The dumptofile option is inserted at the end of any ASU command.

The dumptofile option complements in function and is mutually exclusive with the silent option. When the dumptofile option is specified, the ASU runs in silent mode. All output that is produced by the ASU, whether informational or error logging, is redirected to a predefined log file. For Linux, the log file is /asulog/asuout.log. For Windows, the log file is *c*:\asulog\asuout.log, where *c* is the system drive as defined in Windows.

Output
The dumptofile option does not produce any additional output or filter any output that is normally produced by the ASU. It is a simple redirect to the predefined file.

Every time the ASU runs, the predefined file is initialized. All content from a previous ASU run is lost. There are no appends to the existing file from a previous run.

The predefined log file has no maximum file size. If, during an ASU run, the file reaches the file system available space, any additional output is lost.

Examples

The following examples show the asu command with the dumptofile option.

`asu show all dumptofile`	This ASU command shows the current value for all settings and redirects the output to a predefined log file.
`asu set CMOS_CRTRequired Disabled dumptofile`	This ASU command sets the value, and all output is redirected to the predefined log file.

-n*x* node option (ASU version 2.3 or later)

The -n*x* option supports multi-node systems. In a multi-node system, there are multiple BIOS CMOS settings, Remote Supervisor Adapter settings, and baseboard management controller settings. The ASU enables you to access any node settings by adding the -n*x* parameter to the command (where *x* is the selected node).

Before you use the -n*x* option with a multi-node system that is running a Windows operating system, see "IPMI device driver support for Windows" on page 7. If the multi-node system is running a Linux operating system, see "IPMI device driver support for Linux" on page 8.

Syntax

If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system and is a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1). The -n*x* option must be at the end of the command. If the --bypass option is also specified, the --bypass option must follow the node option.

Examples:

`asu show all -n3`
This command shows the current value for all settings for node 3.

`asu comparedefault CMOS_PrimaryBootDevice1 —n2`
This command compares the current CMOS setting in node 2 with the default value.

`asu set CMOSCRTRequired Disabled`
This command sets the CMOS setting to disabled for node 1.

`asu rebootrsa —n1`
This command restarts the Remote Supervisor Adapter node 1, which is the primary node.

`asu patchadd GG16A.def -nx 2 --bypass`
This command forces adding a patch to node 2.

When you run the **asu** command with the --bypass option, the ASU ignores the BIOS CMOS patch information in BIOS ROM.

showsptraffic command option (ASU version 2.3 or later)

Note: The showsptraffic command option is not available for IMM-based servers.

Use the showsptraffic command option (`--showsptraffic`) to show raw traffic to and from the service processor (SP). Use this option for debugging.

Usage
The showsptraffic option is inserted anywhere in any **asu** command. Any communication with the service processor is shown.

Output
When the showsptraffic option is specified on the command line, the following lines are interspersed with normal output:

```
SP Sent: <byte 1> <byte 2> ... <byte n>
SP Recv: <byte 1> <byte 2> ... <byte n>
```

Or

```
SP6 Sent: <byte 1> <byte 2> ... <byte n>
SP6 Recv: <byte 1> <byte 2> ... <byte n>
```

Examples
The showsptraffic option and corresponding output are shown in the following example:

Command line:

```
asu show RSA_SSL_Server_Enable --showsptraffic
```

Output:

```
SP Sent: 02 06 00 00 00 00 04 09 05 01 01 01
SP Recv: 04 06 01 00 00 00 04 09 05 01 01 01 00
RSA_SSL_Server_Enable=Disabled
```

Command line:

```
asu set RSA_SSH_Enable Enabled --showsptraffic
```

Output:

```
SP Sent: 00 06 01 00 00 00 04 09 05 01 04 01 01
SP Recv: 04 06 00 00 00 00 04 09 05 01 04 01
SP Sent: 02 06 00 00 00 00 04 09 05 01 04 01
SP Recv: 04 06 01 00 00 00 04 09 05 01 04 01 01
RSA_SSH_Enable=Enabled
```

silent command option (ASU version 2.4 or later)

Use the silent command option to suppress all output.

Usage
The silent option is inserted at the end of any ASU command. Any output that is produced by the ASU as either informational or error logging to screen is suppressed. To determine whether the command was successful, see the command return code.

Output
If the ASU command is successful, the return code is zero (0). If the ASU command is not successful, the return code is a positive number greater than zero (0). For information about the return codes, see "Return codes" on page 33.

Examples
In the following command example, the ASU sets the value, and no output is produced to the screen or to a file.

```
asu set CMOS_CRTRequired Disabled silent
```

--help command option (ASU version 3.01 or later)

Use the --help command option to show command-line help.

Usage

The --help option can be used to give a full description of an ASU application (for example, **savestat** or **immcfg**) or ASU command (for example, **batch**, **set**, or **loaddefault**).

Output

The output is a full description of the ASU application or command, including available options and the description of those options. Running **asu --help** displays details about the **--help** command option.

Examples

In the following command example, the ASU displays the full description of the **asu show** command.

Command line:

```
asu show --help
```

Output:

```
Description:

 If <setting> is specified, current value is only shown for setting.
 Show the current value for one or all settings.  If "--group all"
 is specified, current values are shown for all settings. If "--group
 <group>" is specified then only values are shown for settings in that
 group. If "--setlist <setting1>...<settingN>" is specified, the list
 of settings from <setting1> to <settingN> are specified.

Syntax:

 show [<setting>][<cmdmod>] [<options>] [<connect_opts>]

Where:

 <cmd_mod>

Note: Use the command "asu showgroups" to find variable classes   available.

  --setlist <name 1>...<name n> - Operate commands on list of settings
  --group <variableclass>      - Operate commands on the group of setting

 options

  --silent        - Silent execution. Use return code to retrieve status
  --dumptofile    - Silent execution and output to asuout.log file
  -n<node>        - Node number node in a multinode system
  -v              - Verbose

 <connect_opts>

  Note: These connectivity options apply to IMM-based servers only

  --host <ip>          - Address of the IMM to operate on
  --user <user>        - User name to authenticate to the IMM as
  --password <pwd>     - Password to authenticate to the IMM with
  --password-file <fn> - File containing password to authenticate with IMM
```

Return codes

When the ASU completes a request successfully, it returns a zero (0) return code. If a failure is detected, the ASU return code is a positive number greater than zero.

The ASU has grouped the failure return codes into different categories. The failure return codes are used as a signal that a failure is detected. However, these return codes are not an indicator of a specific failure. For a more detailed error isolation, see the return code explanation that is displayed on the screen with the specific error condition.

The different categories of errors that are returned by the ASU are described in the following table, including brief descriptions of the possible type of failures.

Table 4. ASU return codes

Return code value	Description	Explanation
0	Successful command	The ASU command has been completed successfully.
5	Input error	The input that was provided to the ASU has an error. This might be a user-input error or an error with the provided definition file.
10	Software error	An error occurred when you tried to use specific software.
15	Hardware error	An error occurred when the ASU tried to communicate with or find specific hardware.
20	Data error	An error occurred in the data that the ASU reads or sets.
25	Program error	An error occurred in the ASU program execution flow.
30	Invalid or missing patch detected	An ASU patch is missing or an ASU patch does not match the required level.
35	The command is not supported.	
40	Invalid value input error	The input value that is provided to the ASU has a syntax error.
45	Batch command error	One or more of the commands that were specified in the batch file has failed.
80	32-bit version running on 64-bit operating system	The ASU application is a 32-bit version, and you are attempting to run it on a 64-bit operating system.

RDCLI return code messages

The following table lists the RDCLI return code messages

Table 5. RDCLI return code messages

Situation	Return message
Success	The virtual disk mount successfully
Invalid parameters	Invalid parameter in command line
Unknown character	Unknown option character
No login name	Please indicate your login name on the IMM
Failed to read .netrc file	Error when load username and password
Cannot find ("HOME") environment	Problem read environment parameters
Failed to allocate memory to read	Could not allocate memory
Failed to open .netrc file	Could not open .netrc file
Cannot find any entry in the .netrc file	Please indicate your login or password in IMM
No IMM address provided	Please indicate a target IMM by hostname or IP
No filepath provided	Pleas indicate a target CD/DVD drive or image
Invalid process ID	Error in system call
Cannot get the program process ID (pid) or parent process ID (ppid) from the system	Malformed output from system process list
Failed to get login/drive/IMM information	Malformed output from system process list
Another mount session exists	There's already a session on the machine, unmount first.
Cannot find an existing mount session	No rdmount session found
Failed to passthrough authentication	Fail in authentication, please check your username and password or IMM setting and try later.
Failed to connect through a socket	sockConnet: Could not connect to IMM address at port PORT[1]
Failed to log in	Failed in login authentication to IMM.
Failed to map the local file to a remote disk	Failed to map the local file to the remote disk.
Unknown mounting error	Unknown error while mounting.
Invalid rdumount token is provided	Invalid token
[1]There is a default port number that the tool uses to communicate with the IMM. You can also specify a different port number that the IMM supports.	

Baseboard management controller startup sequence (boot order) settings

Note: The baseboard management controller startup sequence setting is not available for IMM-based servers.

If the startup sequence (boot order) settings for your IBM System x server are contained in the baseboard management controller and not in the BIOS CMOS memory, you must use the baseboard management controller settings when you use the **asu** commands (such as the **set** and **show** command).

To determine whether the startup sequence settings for your System x server are contained in the baseboard management controller, use the ASU **patchextract** command. The **patchextract** command syntax is

```
asu patchextract patch_number patch_ filename
```

where the *patch_number* is the patch number for the BIOS code and *patch_ filename* is the generated extracted BIOS definition file.

Example:

Command line:

```
asu patchlist          (to determine which patch is the BIOS patch)
```

Output:

```
Patch 1: <XX[00->99] <BMC>
Patch 2: <XX[00->99] <RSA>
Patch 3: <DY[14->14] <BIOS>
```

Command line:

```
asu patchextract 3 bios.def     (bios.def will contain the bios definitions)
```

Output:

```
Extracted patch 3: <DY[14->14] <BIOS>> to bios.def
```

Open the bios.def file and review the contents. If any of the settings start with `BMCSetting` (for example, `BMCSetting BMC_PrimaryBootDevice2`, `"Second Primary Boot Device"`, `critical,......`), the server startup sequence settings are contained in the baseboard management controller, and you must use the `BMCSetting` when you access the server startup sequence settings.

Note: Before you can access the baseboard management controller settings, the baseboard management controller device driver must be installed. For information about obtaining the device driver, see "Obtaining the ASU and patch files" on page 9.

Boot order settings for IMM-based servers

The boot order settings on IMM-based servers are a special group of settings. The boot order sequence is stored in the IMM and used by the server firmware during the startup process.

The following examples illustrate how to view the current settings and to set the boot order sequence. Each example shows a different connectivity option to illustrate the different method of connectivity.

To determine the current boot order sequence, start by listing the settings that belong to the special boot-order group. The special group or class is currently defined as BootOrder.

Example:

To list the boot order settings using a local connection to an IMM:

```
asu show BootOrder
```

A sample output of the command:

```
BootOrder.BootOrder=CD/DVD Rom=Floppy Disk=Hard Disk
0=Network=Hard Disk 1=Hard Disk 2=Hard Disk 3
```

The setting that contains the boot order is called **BootOrder.BootOrder**. This output represents an ordered list. The syntax of an ordered list is:

```
value1=value2=valueN
```

The values that are separated by the equal sign (=) represent each of the items in the list. Therefore, this is the list of devices in the boot order.

The listed values are those devices that are currently set in the boot order. Additional devices might be available to be included in the boot order. To learn about all the available devices, use the **showvalues** command.

Example:

To list the devices that are available to be set by using the remote connection to an IMM and the default authentication for the **BootOrder.BootOrder** setting, type the following command:

```
asu showvalues BootOrder.BootOrder --host 9.5.51.207
```

A sample output for this command:

```
BootOrder.BootOrder==CD/DVD Rom=Floppy Disk=Hard Disk
0=Network=Hard Disk 1=Hard Disk 2=Hard Disk 3=Hard Disk
4=USB Storage=Diagnostics=iSCSI=iSCSI Critical=Legacy
Only=Embedded Hypervisor
```

The list of devices is much larger than what is shown in the previous example. Note the double equal sign (==) sign nomenclature. The double equal sign represents the values in an ordered list. You can select any of these values to build a new ordered list.

After all the possible values are known, you can build a new ordered list of values by using the **set** command.

Example:

To build a new boot order sequence by using the set command on a local IMM through the LAN over USB connection:

```
asu set BootOrder.BootOrder "Network=Hard Disk 1=USB
Storage=Diagnostics=iSCSI=iSCSI Critical=Legacy
Only=Embedded Hypervisor" --user testuser --password
testpwd
```

Each of the devices in the list in the specified order constitutes the new boot order sequence.

A sample output for this command:

```
BootOrder.BootOrder=Network=Hard Disk 1=USB
Storage=Diagnostics=iSCSI=iSCSI Critical=Legacy
Only=Embedded Hypervisor --user testuser --password
testpwd
```

iSCSI configuration

The ASU supports setting iSCSI boot parameters. This section describes how to configure iSCSI settings. This section does not describe iSCSI or the actual meaning of each parameter.

The ASU is designed to configure iSCSI settings, which do not initially exist. This section describes the detailed steps for creating and configuring the iSCSI settings.

Before you configure iSCSI boot parameters, read the following general information:

- The iSCSI settings are grouped into a record. The record key for the record is the iSCSI.AttemptName setting. Each record represents an *attempt*. The term *attempt* is equivalent to the term *instance* that is defined for other settings.

- More than one attempt can be defined. In the examples in this section, all the settings that end with ".1" belong to the "first" attempt. The iSCSI.AttemptName.1 setting defines the name of the attempt group of settings.

- If more that one attempt is defined, the same setting names that define the attempt group are used, but each attempt contains a different instance number sequence.

- The iSCSI settings can now be created by using the ASU **set** command on an instance of the iSCSI.AttemptName setting that does not already exist.

- The iSCSI settings can now be deleted by using the ASU **delete** command on an instance of the iSCSIAttemptName setting. This causes all iSCSI settings for that instance to also be deleted because iSCSI instance settings are now grouped as a record.

- The only parameter that is defined as a global iSCSI setting to all attempts and that does not required an instance index is the `iSCSI.InitiatorName`. This setting does not require the attempt index that all the other settings require. The initiator name is always defined as `iSCSI.InitiatorName`.

- The iSCSI parameters are defined in the iSCSI group of settings. To list the available iSCSI settings, type the following command:

  ```
  asu show iscsi
  ```

Example:

The following example shows the list of settings if a single attempt group is defined.
All setting names with the same instance ".1" belong to the same attempt group:

```
iSCSI.InitiatorName

iSCSI.MacAddress.1
iSCSI.AttemptName.1
iSCSI.IscsiMode.1
iSCSI.ConnectRetryCount.1
iSCSI.ConnectTimeout.1
iSCSI.InitiatorInfoFromDhcp.1
iSCSI.LocalIp.1
iSCSI.SubnetMask.1
iSCSI.Gateway.1
iSCSI.TargetInfoFromDhcp.1
iSCSI.TargetName.1
iSCSI.TargetIp.1
iSCSI.TargetPort.1
iSCSI.BootLun.1
iSCSI.CHAPType.1
iSCSI.CHAPName.1
iSCSI.CHAPSecret.1
iSCSI.ReverseCHAPName.1
iSCSI.ReverseCHAPSecret.1

iSCSI.MacAddress.2
```

Example:

The following example shows the list of settings if two attempts are defined:

```
iSCSI.InitiatorName

iSCSI.MacAddress.1
iSCSI.AttemptName.1
iSCSI.IscsiMode.1
iSCSI.ConnectRetryCount.1
iSCSI.ConnectTimeout.1
iSCSI.InitiatorInfoFromDhcp.1
iSCSI.LocalIp.1
iSCSI.SubnetMask.1
iSCSI.Gateway.1
iSCSI.TargetInfoFromDhcp.1
iSCSI.TargetName.1
iSCSI.TargetIp.1
iSCSI.TargetPort.1
iSCSI.BootLun.1
iSCSI.CHAPType.1
iSCSI.CHAPName.1
iSCSI.CHAPSecret.1
iSCSI.ReverseCHAPName.1
iSCSI.ReverseCHAPSecret.1

iSCSI.MacAddress.2
iSCSI.AttemptName.2
iSCSI.IscsiMode.2
iSCSI.ConnectRetryCount.2
iSCSI.ConnectTimeout.2
iSCSI.InitiatorInfoFromDhcp.2
iSCSI.LocalIp.2
iSCSI.SubnetMask.2
iSCSI.Gateway.2
iSCSI.TargetInfoFromDhcp.2
iSCSI.TargetName.2
iSCSI.TargetIp.2
iSCSI.TargetPort.2
iSCSI.BootLun.2
iSCSI.CHAPType.2
iSCSI.CHAPName.2
iSCSI.CHAPSecret.2
iSCSI.ReverseCHAPName.2
iSCSI.ReverseCHAPSecret.2
```

RDCLI

The RDCLI is designed to mount an ISO image file or DVD or CD drive to a remote
IMM-based server. Before you start to invoke RDCLI, make sure that:

• The remote IMM is connected to the network environment.
• There is no other remote presence session opened through the RDCLI or IMM
  web interface.
• The server has a virtual media key installed and the remote presence feature is
  enabled.

Example:

Type the following commands to mount an ISO image file or DVD or CD drive to a
remote IMM-based server:

```
rdmount -s 192.168.1.12 –d /dev/cdrom -l USERID -p PASSW0RD
rdmount -s 192.168.1.12 –d /home/install.iso -l USERID -p PASSW0RD
rdmount –s 192.168.1.12 –d /dev/cdrom -l USERID –p PASSW0RD –w 90
```

If the mounting command is successful, a message stating that the mount operation is completed without error is displayed. Otherwise, a message stating that the mount operation has failed and the detailed reasons for the failure is displayed.

Type `-w 90` to switch the authentication port number to 90. The default port number is 80.

Example:

Type the following command to query existing mount sessions between the client operating system and the remote server:

```
rdmount –q
```

If the command is successful, all available token values on the client operating system are listed in the following format:

```
"Token 507: drive path /home/install.iso, mounted to SP 192.168.0.1, by user USERID"
```

Note: The token value is a unique ID of an existing successful mounting record in the client operating system.

If the command is not successful, the error message and reason for the failure are displayed.

Example:

Type the following command to unmount an ISO image file or DVD or CD drive that is already mounted to the remote system. In the command, the remote mount session has the token 507.

```
rdumount 507
```

If the command is successful, a message that the remote presence session is ended is displayed. If the command is not successful, the error message and reason for the failure are displayed.

Chapter 3. Command reference

This chapter describes the Advanced Settings Utility (ASU) commands.

Batch command

Use the **batch** command to queue ASU operations without any knowledge of the scripting capabilities of the operating system on which the ASU is running.

Syntax

The syntax of the **batch** command is

```
asu batch batch_filename [-nx] [connect_options]
```

where *batch_filename* is the name of a file that contains a list of ASU commands.

Notes:

1. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

2. Do not specify the –n*x* optional parameter in the batch file for any of the batched commands.

3. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

When you use the **batch** command on a batch file, the output that is sent to stdout and stderr is the collective output of all the commands in the batch file. The output of each command in the batch file is preceded by the command itself, surrounded by brackets ([ ]).

```
[command 1]
output of command 1
```
```
[command 2]
output of command 2
```
```
:
:
```
```
[command n]
output of command n
```

Examples

The **batch** command and corresponding output are shown in the following examples.

Example 1:

Batch file:

```
set IMM.PowerRestorePolicy "Always on"
set IMM.PowerOnAtSpecifiedTime_Hour 05
set IMM.PowerOnAtSpecifiedTime_Minute 00
set IMM.PowerOnAtSpecifiedTime_Second 00
set IMM.ShutdownAndPowerOff_WeekDay Sunday
set IMM.HTTPPort 81
set IMM.SSLPort 441
set IMM.TelnetPort 21
set SYSTEM_PROD_DATA.SysEncloseAssetTag "Server Tag"
set iSCSI.InitiatorName "iqn.2009-01.com.ibm:InitiatorName"
set uEFI.Com1BaudRate 9600
```

Output:

```
[set IMM.PowerRestorePolicy "Always on"]
IMM.PowerRestorePolicy=Always on
[set IMM.PowerOnAtSpecifiedTime_Hour 05]
IMM.PowerOnAtSpecifiedTime_Hour=05
[set IMM.PowerOnAtSpecifiedTime_Minute 00]
IMM.PowerOnAtSpecifiedTime_Minute=00
[set IMM.PowerOnAtSpecifiedTime_Second 00]
IMM.PowerOnAtSpecifiedTime_Second=00
[set IMM.ShutdownAndPowerOff_WeekDay Sunday]
IMM.ShutdownAndPowerOff_WeekDay=Sunday
[set IMM.HTTPPort 81]
IMM.HTTPPort=81
[set IMM.SSLPort 441]
IMM.SSLPort=441
[set IMM.TelnetPort 21]
IMM.TelnetPort=21
[set SYSTEM_PROD_DATA.SysEncloseAssetTag "Server Tag"]
SYSTEM_PROD_DATA.SysEncloseAssetTag=Server Tag
[set iSCSI.InitiatorName "iqn.2009-01.com.ibm:InitiatorName"]
iSCSI.InitiatorName=iqn.2009-01.com.ibm:InitiatorName
[set uEFI.Com1BaudRate 9600]
uEFI.Com1BaudRate=9600
```

Example 2:

Batch file:

```
show CMOS_PrimaryBootDevice1
show CMOS_PrimaryBootDevice2
show CMOS_PrimaryBootDevice3
show CMOS_PrimaryBootDevice4
```

Output:

```
[show CMOS_PrimaryBootDevice1]
CMOS_PrimaryBootDevice1=CD ROM
[show CMOS_PrimaryBootDevice2]
CMOS_PrimaryBootDevice2=Diskette Drive 0
[show CMOS_PrimaryBootDevice3]
CMOS_PrimaryBootDevice3=Hard Disk 0
[show CMOS_PrimaryBootDevice4]
CMOS_PrimaryBootDevice4=Network
```

Example 3:

Batch file:

```
set CMOS_PrimaryBootDevice1 "Network"
set CMOS_PrimaryBootDevice2 "Hard Disk 0"
set CMOS_PrimaryBootDevice3 "Diskette Drive 0"
set CMOS_PrimaryBootDevice4 "CD ROM"
```

Output:

```
[set CMOS_PrimaryBootDevice1 "Network"]
CMOS_PrimaryBootDevice1=Network
[set CMOS_PrimaryBootDevice2 "Hard Disk 0"]
CMOS_PrimaryBootDevice2=Hard Disk 0
[set CMOS_PrimaryBootDevice3 "Diskette Drive 0"]
CMOS_PrimaryBootDevice3=Diskette Drive 0
[set CMOS_PrimaryBootDevice4]
CMOS_PrimaryBootDevice4=CD ROM
{set CMOS_PrimaryBootDevice1]
CMOS_PrimaryBootDevice1=CD ROM
[show CMOS_PrimaryBootDevice2]
CMOS_PrimaryBootDevice2=Diskette Drive 0
[show CMOS_PrimaryBootDevice3]
CMOS_PrimaryBootDevice3=Hard Disk 0
[show CMOS_PrimaryBootDevice4]
CMOS_PrimaryBootDevice4=Network
```

Comparedefault command

Use the **comparedefault** command to compare current values to default values for one or more settings.

Syntax

The syntax of the **comparedefault** command is

```
asu comparedefault [setting | class] [-v] [-nx] [connect_options]
```

where *setting* is the name of an ASU setting and *class* is the name of an ASU class of settings.

Notes:

1. If the optional **-v** parameter is specified, the output is verbose.

2. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

3. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **comparedefault** command shows the current and default values for one or all settings.

Without the **-v** parameter:

```
<setting 1>=<current value 1><<default value1>>
<setting 2>=<current value 2><<default value2>>

.
.
.

<setting n>=<current value n><<default valuen>>
```

With the **-v** parameter:

```
<setting 1>: <setting 1 description> = <current value 1>,
<default value 1> (default)
<setting 2>: <setting 2 description> = <current value 2>,
<default value 2> (default)

.
.
.

<setting n>: <setting n description> = <current value n>,
<default value n> (default)
```

Examples

The **comparedefault** command and corresponding output are shown in the following examples.

Command line:

```
asu comparedefault uefi.com1baudrate
```

Output:

```
uEFI.Com1BaudRate=115200<115200>
```

Command line:

```
asu comparedefault CMOS_PrimaryBootDevice1
```

Output:

```
CMOS_PrimaryBootDevice1=Network<CD ROM>
```

Command line:

```
asu comparedefault CMOS_PrimaryBootDevice2 –v
```

Output:

```
CMOS_PrimaryBootDevice2: Second Startup Device = Network, CD
ROM (default)
```

Command line:

```
asu comparedefault bios
```

Output:

```
CMOS_DisketteA=1.44 MB 3.5"<1.44 MB 3.5">
CMOS_CRTRequired=Disabled<Enabled>

:
:
:
CMOS_OSUSBControl=Other OS<Other OS>
```

Createuuid command

> **Note:** This command is for IMM-based servers only.

Use the **createuuid** command to generate and set the Universally Unique Identifier (UUID).

> **Note:** When you set the UUID, the command requires the setting name. You can use the **asu show** command to identify the setting.

Syntax

The syntax of the **createuuid** command is:

```
asu createuuid UUID_setting_name [connect_options]
```

Notes:

1. The *UUID_setting_name* is usually defined as SYSTEM_PROD_DATA.SysInfoUUID.
2. To view the actual setting name, which can vary from server to server, use the **show** command to list the setting name that belongs to the SYSTEM_PROD_DATA group. To view the available groups, use the **showgroups** command.
3. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

To see the value that is set by the **createuuid** command, use the **show** command.

```
asu show SYSTEM_PROD_DATA.SysInfoUUID
```

The output of the **show** command is the new randomly generated UUID.

```
SYSTEM_PROD_DATA.SysInfoUUID=801a3b663e82b60104af001a64e50c94
```

Examples

The **createuuid** command and corresponding output are shown in the following examples.

Command line:

```
asu createuuid SYSTEM_PROD_DATA.SysInfoUUID
```

Output:

The command does not produce output. To see the generated UUID, use the **show** command.

```
asu show SYSTEM_PROD_DATA.SysInfoUUID
SYSTEM_PROD_DATA.SysInfoUUID=801a3b663e82b60104af001a64e50c94
```

Delete command

Note: This command is for IMM-based servers only.

Use the **delete** command to delete an instance of a setting.

Note: Using this command to delete the instance of a setting that is a record key results in all other settings with the same instance number to be deleted.

Syntax

The syntax of the **delete** command is

```
asu delete setting_instance [-nx] [connect_options]
```

where *setting_instance* is the name of an instance of a setting to delete. Use the command **asu show all** to show a list of available setting instances.

Use the **asu showvalues** setting command to show a list of all values that are available for the setting.

Limitations

Settings can have a minimum number of allowed instances. The **delete** command does not allow you to delete an instance if doing so causes the number of instances to drop below the minimum number of allowed instances. To determine the minimum number of instances allowed, use the command:

```
asu showvalues --instances
```

```
The output for each setting that can have instances will be displayed,
along with the minimum and maximum number of allowed instances.
```

Also, deleting instances that are part of a record is allowed for only the record key setting. To determine whether or not a setting is part of a record, use the command:

```
asu showvalues --instances
```

```
The output for each setting that can have instances will be displayed,
along with the record information.  See the asu showvalues command for
details about the record information.
```

Notes:
1. Values that contain spaces must be enclosed in quotation marks ("). If a value contains quotation marks, add a backslash before each quotation mark (\") in the value.
2. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
3. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **delete** command indicates the following information.

Deleting an instance that exists:

```
Deleting <setting_instance>
Waiting for command completion status
Command completed successfully
```

Deleting an instance that does not exist:

```
Could not find setting <setting>
```

Deleting an instance that causes the number of instances to drop below the minimum allowed number of instances:

```
The setting IMM.MacAddress.1 cannot be deleted.  Too few instances.
```

Deleting an instance that is in a record but is not the record key:

```
The setting <setting> is part of a record and cannot be deleted.
To delete the entire instance of the record you must
delete the record's key setting, <key setting name>.
```

Examples

The **delete** command and corresponding output are shown in the following examples.

Deleting an instance that exists.

Command line:

```
asu delete IMM.Community_Name.1
```

Output:

```
Deleting IMM.Community_Name.1
Waiting for command completion status
Command completed successfully
```

Deleting an instance that does not exist.

Command line:

```
asu delete IMM.Community_Name.3
```

Output:

```
Could not find setting IMM.Community_Name.3
```

Deleting an instance that causes the number of instances to drop below the minimum allowed number of instances:

Command line:

```
asu delete IMM.MacAddress.1
```

Output:

```
The setting IMM.MacAddress.1 cannot be deleted.  Too few instances.
```

Deleting an instance that is in a record but is not the record key:

Command line:

```
asu delete iSCSI.ConnectTimeout.1
```

Output:

```
The setting iSCSI.ConnectTimeout.1 is part of a record and cannot be
deleted. To delete the entire instance of the record you must
delete the record's key setting, iSCSI.AttemptName.1.
```

Dump command

Note: This command is for BIOS-based servers only.

Use the **dump** command to see the raw contents of CMOS memory.

Syntax

The syntax of the **dump** command is

```
asu dump [-nx]
```

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **dump** command is a table that contains the current raw hexadecimal contents of CMOS memory. The CMOS memory setting area is preceded by < and followed by >. CMOS memory locations that are outside the CMOS memory setting area are denoted by *. The ASU uses information from the CMOS memory map to determine how to access the second bank of CMOS memory. It also uses CMOS memory limit information from the map to determine the CMOS memory setting area.

Examples

The **dump** command and corresponding output are shown in the following example.

Command line:

```
asu dump
```

Output:

```
   0 1 2 3 4 5 6 7 8 9 A B C D E F
00: 38*00*14*00*10*00*01*07*07*03*26*02*50*80<00 00
10: 40 00 00 7e 01 80 02 ff ff 00 00 f2 00 86 c0 c8
20: 60 00 00 00 00 00 00 00 00 00 00 02 27 50 07 18
30: ff ff 20 05 0d 06 00 00 c0 00 f0 ff 00 ca 00 00
40: 00 00 00 00 00 00 00 00 00 00 00 20 52 00 00 60
50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 24
70: 10 42 08 21 00 00 81 4a 2a 00 2e 28 00 30 00 00
80: 00 00 ff 01 00>00*00*00*00*00*00*00*00*00*00*00
90: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
a0: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
b0: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
c0: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
d0: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
e0: 00*00*00*00*00*00*00*00*00*00*00*00*00*00*00*00
f0: 00*00*00*32*08*9c*00*62*90*5c*cd*ff*4f*5f*ba*9f
```

Export command

Note: This command is for BIOS-based servers only.

Use the **export** command to export a selected certificate or certificate signing request (CSR) file. This **export** command is only for the Remote Supervisor Adapter and Remote Supervisor Adapter II. This command fails if the certificate or CSR is not available in the Remote Supervisor Adapter or Remote Supervisor Adapter II. The **export** command generates a binary file that is saved in the current directory.

Syntax

The syntax of the **export** command is

```
asu export setting certificate_binary_file [-nx]
```

where *setting* is the name of a valid ASU setting and *certificate_binary_file* is the name of a file that is generated with the valid certificate information that is provided by the Remote Supervisor Adapter or Remote Supervisor Adapter II.

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **export** command is a binary file and a message that indicates that the Remote Supervisor Adapter or Remote Supervisor Adapter II has completed the command successfully.

Examples

The **export** command and corresponding output are shown in the following example.

Command line:

```
asu export RSA_SSL_Client_PrivateKey_Export asu.cert
```

Output:

```
Certificate was exported to the file successfully!
( asu.cert file in saved in the current directory)
```

Generate command

> **Note:** This command is for BIOS-based servers only.

The **generate** command is targeted to only the Remote Supervisor Adapter or Remote Supervisor Adapter II. Use the **generate** command to generate a private key and public key pair with a self-signed certificate or certificate signing request (CSR). The generation can take a few seconds to be completed, depending on the state of the Remote Supervisor Adapter or Remote Supervisor Adapter II. The **generate** command requires an Extensible Markup Language (XML) file that contains the certificate information that you want in the directory from which the ASU is running. When you unpack the ASU files, a template file (template.xml) is unpacked. This file provides an XML file with the correct syntax, which you should modify with the information for generating the selected certificate.

> **Note:** The XML file supports the self-signed certificate request and the certificate signing request (CSR) request. The start and end tag for the self-signed certificate is new_key_and_self_signed_cert_info. The start and end tag for CSR is new_key_and_cert_sign_req_info.

Template.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<asu version="2.1">
 <new_key_and_self_signed_cert_info>
   <item type="Required">
    <vectorID>0001</vectorID>
    <name>countryName</name>
    <value minlen="2" maxlen="2">xx</value>
   </item>
   <item type="Required">
    <vectorID>0001</vectorID>
    <name>stateOrProvinceName</name>
    <value minlen="1" maxlen="30">xx</value>
   </item>
   <item type="Required">
    <vectorID>0001</vectorID>
    <name>localityName</name>
    <value minlen="1" maxlen="50">xx</value>
   </item>
   <item type="Required">
    <vectorID>0001</vectorID>
    <name>organizationName</name>
    <value minlen="1" maxlen="60">xx</value>
   </item>
   <item type="Required">
    <vectorID>0001</vectorID>
    <name>commonName</name>
    <value minlen="1" maxlen="60">xx</value>
   </item>
   <item type="Optional">
    <vectorID>0001</vectorID>
    <name>Name</name>
    <value minlen="1" maxlen="60">xx</value>
   </item>
   <item type="Optional">
    <vectorID>0001</vectorID>
    <name>emailAddress</name>
    <value minlen="1" maxlen="60">xx</value>
   </item>
   <item type="Optional">
    <vectorID>0001</vectorID>
    <name>validityPeriod</name>
```

```xml
     <value minlen="0" maxlen="2">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>organizationalUnitName</name>
     <value minlen="0" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>Surname</name>
     <value minlen="0" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>givenName</name>
     <value minlen="0" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>Initials</name>
     <value minlen="0" maxlen="20">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>dnQualifier</name>
     <value minlen="0" maxlen="60">xx</value>
    </item>
</new_key_and_self_signed_cert_info>
<new_key_and_cert_sign_req_info>
    <item type="Required">
     <vectorID>0001</vectorID>
     <name>countryName</name>
     <value minlen="2" maxlen="2">xx</value>
    </item>
    <item type="Required">
     <vectorID>0001</vectorID>
     <name>stateOrProvinceName</name>
     <value minlen="1" maxlen="30">xx</value>
    </item>
    <item type="Required">
     <vectorID>0001</vectorID>
     <name>localityName</name>
     <value minlen="1" maxlen="50">xx</value>
    </item>
    <item type="Required">
     <vectorID>0001</vectorID>
     <name>organizationName</name>
     <value minlen="1" maxlen="60">xx</value>
    </item>
    <item type="Required">
     <vectorID>0001</vectorID>
     <name>commonName</name>
     <value minlen="1" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>Name</name>
     <value minlen="1" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>emailAddress</name>
     <value minlen="1" maxlen="60">xx</value>
    </item>
    <item type="Optional">
     <vectorID>0001</vectorID>
     <name>organizationalUnitName</name>
```

```
          <value minlen="0" maxlen="60">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0001</vectorID>
         <name>Surname</name>
         <value minlen="0" maxlen="60">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0001</vectorID>
         <name>givenName</name>
         <value minlen="0" maxlen="60">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0001</vectorID>
      </item>
        <item type="Optional">
         <vectorID>0001</vectorID>
         <name>Initials</name>
         <value minlen="0" maxlen="20">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0001</vectorID>
         <name>dnQualifier</name>
         <value minlen="0" maxlen="60">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0002</vectorID>
         <name>challengePassword</name>
         <value minlen="6" maxlen="30">xx</value>
        </item>
        <item type="Optional">
         <vectorID>0002</vectorID>
         <name>unstructuredName</name>
         <value minlen="1" maxlen="60">xx</value>
        </item>
     </new_key_and_cert_sign_req_info>
    </asu>
```

Notes:

1. *xx* is a user input field, the minimum length for each vector (item) is identified by minlen=, and the maximum length is identified by maxlen=. For example, for the vector named stateOrProvinceName, the minlen is 1, the maxlen is 30, and a valid *xx* value is Vermont.

2. Items that are identified as "Required" have to be updated with user data. Items that are identified as "Optional" do not have to be updated. If the optional items are not updated, remove them from the XML file.

3. The ASU requires you to provide the XML file with the correct data for the **generate** command to run correctly.

Syntax

The syntax of the **generate** command is

```
asu generate setting xml_file [-nx]
```

where *setting* is the name of a valid Remote Supervisor Adapter or Remote Supervisor Adapter II setting and *xml_file* is the name of an XML file that contains valid information.

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **generate** command is a message that indicates that the Remote Supervisor Adapter or Remote Supervisor Adapter II has completed the command successfully.

Examples

The **generate** command for a self-signed certificate and corresponding output are shown in the following example.

Command line:

```
asu generate RSA_Generate_SSL_Client_Certificate asu.xml
```

Output:

```
Certificate was generated successfully!
```

The **generate** command for a CSR certificate and corresponding output are shown in the following example:

Command line:

```
asu generate RSA_Generate_SSL_Client_CSR asu.xml
```

Output:

```
Certificate was generated successfully!
```

Help command

Use the **help** command to view help for one or more settings. For BIOS settings, the help that is output by this command is the same help that you access when you press F1 during startup.

Syntax

The syntax of the **help** command is

```
asu help [setting | class] [connect_options]
```

where *setting* is the name of an ASU setting and *class* is the name of an ASU class of settings.

Note: The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **help** command shows the help text for one or more settings. The name and description of the setting are followed by the help title and the help text.

```
<setting 1>: <setting description 1>
<help title 1>
--------------
<help text 1>

<setting 2>: <setting description 2>
<help title 2>
--------------
<help text 2>

.
.
.
<setting n>: <setting description n>
<help title n>
--------------
<help text n>
```

Examples

The **help** command and corresponding output are shown in the following examples.

Command line:

```
BootOrder.BootOrder: Boot Order

Help for Boot Order
-------------------
```

Output:

```
Specify, from the list of bootable devices, the desired order in which
to search for bootable media.  One or more items from the list may be
specified.
```

Command line:

```
asu help CMOS_PrimaryBootDevice3
```

Output:

```
CMOS_PrimaryBootDevice3: Third Startup Device

Help for Startup Device
-----------------------
The system uses a startup sequence to determine which device
will be the startup device. The startup device is the
diskette drive, hard disk, or network adapter which will be used
to load the operating system. This field specifies the third
device for which a system start will be attempted. If the
start from this device fails, the system will attempt to
start from the fourth startup device.
```

Command line:

```
asu help bios
```

Output:

```
CMOS_DisketteA: Diskette Drive A

Help for Diskette Drive
-----------------------
If you change or add a diskette drive, you might need to
use this option to set the correct type.

CMOS_CRTRequired: Displayless Operation

Help for Displayless Operation
------------------------------
This option suppresses the error messages that
normally occur when no video device is present.

.
.
.

CMOS_OSUSBControl: OS USB Selection

Operating System USB Support
----------------------------
Choose which operating system you will be using
for RSA II USB support.
```

Import command

Note: This command is for BIOS-based servers only.

The **import** command is a Remote Supervisor Adapter and Remote Supervisor Adapter II command only. Use the command to import a certificate into the Remote Supervisor Adapter or Remote Supervisor Adapter II command. The **import** command requires a binary file that is in the same directory from which the ASU is running.

Syntax

The syntax of the **import** command is

```
asu import setting certificate_binary_file [-nx]
```

where *setting* is the name of an ASU setting and *certificate_binary_file* is the name of a file that is generated with the valid certificate information.

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **import** command is a message that indicates that the Remote Supervisor Adapter or Remote Supervisor Adapter II has completed the command successfully.

Examples

The **import** command and corresponding output are shown in the following example.

Command line:

```
asu import RSA_Import_Trusted_Certificate_1 asu.cert
```

Output:

```
Certificate was imported successfully!
```

Loaddefault command

Use the **loaddefault** command to load default values for one or more settings.

Limitations:
- Not all settings in the definition or configuration file have assigned default values.
- The **loaddefault** command sets the default settings only for the settings that have defined default values in the configuration file.
- On BIOS-based servers, the **loaddefault** command should not be viewed or used as a general restore factory defaults command.
- On IMM-based servers, when the **loaddefault** command is run remotely (using the **--host** connectivity option) the authentication and password class settings are not applied.
- On IMM-based servers that are running the ASU 3.00, the **loaddefault** command is not functionally equivalent to the "Restore Defaults" function that is defined in the IMM Web interface. You have to use the IMM Web interface or the server Setup utility settings to restore the IMM settings to the factory defaults.
- On IMM-based servers that are running the ASU 3.01 or later (and have the required IMM firmware that is described in "Using the ASU to configure settings in IMM-based servers" on page 15), the **loaddefault** command is now functionally equivalent to the "Restore Defaults" function that is defined in the IMM Web interface. Settings from other groups are still not the equivalent of the 'restore factory defaults' function that is available through the server Setup utility (F1 Setup).

Note: For the ASU 3.01, some instances are deleted when you run the **loaddefault** command. To determine which instances are deleted, use the **showdefault** command. Instances that have a default value of "remove" are deleted.

The following example shows using the **showdefault** command on a setting that has a default action of remove and then performing the **loaddefault** command on the setting.

Command line:

```
asu showdefault IMM.LoginId.7
```

Output:

```
IMM.LoginId.7=<remove>
```

Command line:

```
asu loaddefault IMM.LoginId.7
```

Output:

```
Could not find setting IMM.LoginId.7
```

Syntax

The syntax of the **loaddefault** command is

```
asu loaddefault [setting | class] [-v] [-nx] [connect_options]
```

where *setting* is the name of an ASU setting and *class* is the name of an ASU class of settings.

Notes:

1. If the optional **-v** parameter is specified, the output is verbose.

2. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

3. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **loaddefault** command is displayed if a setting is changed to the default value. If a setting is already set to the default value, no output is displayed. If a setting is not already set to the default value, the value is changed, and the output is shown as the output of the **set** command.

Note: Starting with the ASU 3.01, there are exceptions on IMM-based servers. Performing the **loaddefault** command on a group of settings (for example, IMM, UEFI, SYSTEM_PROD_DATA, BOOT_ORDER, and all) can trigger a reset to factory defaults. If the group is being reset to factory defaults, the setting names and their new values are not displayed. Instead, the ASU displays the message `Issuing reset of IMM`.

Without the **-v** parameter:

```
<setting 1>=<default value 1>
<setting 2>=<default value 2>

.
.
.

<setting n>=<default value n>
```

With the **-v** parameter:

```
<setting 1>: <setting 1 description> = <default value 1>
<setting 2>: <setting 2 description> = <default value 2>

.
.
.

<setting n>: <setting n description> = <default value n>
```

Examples

The **loaddefault** command and corresponding output are shown in the following examples.

Command line (IMM-based servers):

```
asu loaddefault uEFI.Com1BaudRate
```

Output:

```
uEFI.Com1BaudRate=115200
```

Command line:

```
asu loaddefault CMOS_CRTRequired
```

Output:

```
CMOS_CRTRequired=Enabled
```

Command line:

```
asu loaddefault CMOS_KbdRequired —v
```

Output:

```
CMOS_KbdRequired: Keyboardless Operation = Enabled
```

Command line:

```
asu loaddefault bios
```

Output:

```
CMOS_DisketteA=1.44 MB 3.5"
CMOS_CRTRequired= Enabled
CMOS_WakeOnLAN=Enabled
```

Command line (IMM-based servers with ASU 3.01 or later):

```
asu loaddefault IMM
```

Output:

```
Issuing reset of IMM
The IMM has started the reset. Waiting for the reset to complete.
Connected to IMM at IP address 169.254.95.118
Reset completed successfully
```

Command line (IMM-based servers with ASU 3.01 or later):

```
asu loaddefault all
```

Output:

```
Connected to IMM at IP address 9.5.107.158
SYSTEM_PROD_DATA.SysInfoProdName=
SYSTEM_PROD_DATA.SysInfoProdIdentifier=
SYSTEM_PROD_DATA.SysInfoSerialNum=
SYSTEM_PROD_DATA.SysInfoUUID=
SYSTEM_PROD_DATA.SysEncloseAssetTag=
uEFI.TurboModeEnable=Disable
uEFI.OperatingMode=Custom Mode
.
.
.
Issuing reset of IMM
The IMM has started the reset. Waiting for the reset to complete.
Connected to IMM at IP address 169.254.95.118
Reset completed successfully
```

Nodes command

Use the **nodes** command to detect the available nodes in the current system.

Syntax

The syntax of the **nodes** command is

```
asu nodes
```

Note: The **nodes** command is applicable on a multi-node system or single-node system. On a single-node system, the command always reports a 1. On a multi-node system, the number reported is based on how many nodes are on the system.

Output

The output of the **nodes** command is a string with the number of nodes currently available on the system.

Examples

The **nodes** command and corresponding output are shown in the following example.

Command line:

```
asu nodes
```

Output:

```
System Nodes: 2
```

Patchadd command

Note: This command is for BIOS-based servers only.

Use the **patchadd** command to add support for a particular firmware setting to the ASU. Depending on the system from which you are running this command, a BIOS patchadd might not occur because the BIOS patch is already contained in the BIOS ROM.

Syntax

The syntax of the **patchadd** command is

```
asu patchadd patch_filename
```

where *patch_filename* is the name of a patch file. For a description of the patch file format, see "Using ASU patches" on page 10.

Output

The output of the **patchadd** command shows the success or failure of adding a patch. If the patch succeeded, a message is shown indicating that a new patch was written to the executable file, and information about the patch is provided. If the patch failed, a message is shown indicating why the patch failed.

If the **patch** command is successful, the output looks similar to this example:

```
Wrote new patch <<patch identification>> to <executable>
Wrote patch footer to <executable>
```

If the **patch** command is not successful (duplicate), the output looks similar to this example:

```
<<patch identification>> already patched.
Wrote patch footer to <executable>
```

Examples

The **patchadd** command and corresponding output are shown in the following examples.

Successful patch:

Command line:

```
asu patchadd T2C125A.def
```

Output:

```
Wrote new patch <T2[25->25] (BIOS)> to ./asu
Wrote patch footer to ./asu
```

System BIOS already has patch:

Command line:

```
asu patchadd T2C125A.def
```

Output:

```
BIOS def file already defined in BIOS ROM!
```

Patchextract command

Note: This command is for BIOS-based servers only.

Use the **patchextract** command to extract a patch from the ASU to a patch file. Then, you can patch the extracted patch file to another version of the ASU by using the **patchadd** command.

Syntax

The syntax of the **patchextract** command is

```
asu patchextract patch_number patch_filename
```

where *patch_number* is the patch number to extract. To show the patch number for each patch, use the **patchlist** command. *patch_filename* is the name of the patch file that is extracted.

Output

The output of the **patchextract** command shows the success or failure of the extraction operation. If the extraction is successful, a message is displayed indicating which patch was extracted and the file name to which it was extracted.

```
Extracted patch <patch number>: <<patch identification>> to <patch filename>
```

Example

The **patchextract** command and corresponding output are shown in the following example.

Command line:

```
asu patchextract 1 T2.def
```

Output:

```
Extracted patch 1: <T2[25->25] (BIOS)> to T2.def
```

Patchlist command

Note: This command is for BIOS-based servers only.

Use the **patchlist** command to display the patches that are applied to the ASU.

Syntax

The syntax of the **patchlist** command is

```
asu patchlist
```

Output

The output of the **patchlist** command is a list of patches. Each patch has a patch number and patch identification.

```
Patch <patch number 1>: <<patch identification 1>>
Patch <patch number 2>: <<patch identification 2>>

:
:

Patch <patch number n>: <<patch identification n>>
```

Example

The **patchlist** command and corresponding output are shown in the following example.

Command line:

```
asu patchlist
```

Output:

```
Patch 1: <T2[25->25] (BIOS)>
Patch 2: <GE[00->99] (RSA)>
Patch 3: <GE[46->46] (BIOS)>
```

Patchremove command

Note: This command is for BIOS-based servers only.

Use the **patchremove** command to remove a patch from the ASU. Depending on the system from which you are running the command, the BIOS patch might not be removed because the patch is contained in the BIOS ROM.

Syntax

The syntax of the **patchremove** command is

```
asu patchremove patch_number
```

where *patch_number* is the patch number to extract. You can use the **patchlist** command to show the patch number for each patch.

Output

The output of the **patchremove** command shows the success or failure of the removal operation. If the removal is successful, messages are displayed that indicate the removal of a patch and the copying of every other patch to the temporary executable file.

Output:

```
Copied patch <<patch identification>> to <temporary executable>
Removing patch <<patch identification>> from <executable>
```

Example

The **patchremove** command and corresponding output are shown in the following example.

Command line:

```
asu patchremove 2
```

Output:

```
Copied patch <T2[25->25] (BIOS)> to smep2tmp-9yFPOa
Removing patch <GE[00->99] (RSA)> from ./asu
```

The **patchremove** command that is attempting to remove a BIOS patch that is in BIOS ROM and the corresponding output are shown in the following example:

Command line:

```
asu patchremove 3
```

Output:

```
Can not remove patch in BIOS ROM, patch <T2[25->25] (BIOS)> is not removed.
```

Readraw command

Note: This command is for BIOS-based servers only.

Use the **readraw** command to read raw CMOS data and save it in a file that you can use on other systems by using the **writeraw** command.

Syntax

The syntax of the **readraw** command is

```
asu readraw filename [-nx]
```

where *filename* is the name of a file to which the raw CMOS data is saved.

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **readraw** command is a message that indicates that the raw read is completed.

Examples

The **readraw** command and corresponding output are shown in the following example.

Command line:

```
asu readraw CMOSraw.dat
```

Output:

```
Raw CMOS read from CMOS, written to CMOSraw.dat
```

Rebootbmc command

Note: This command is for BIOS-based servers only.

Use the **rebootbmc** command to restart the baseboard management controller. This is useful after you make changes to baseboard management controller settings for which a restart is necessary for the changes to take effect.

Syntax

The syntax of the **rebootbmc** command is

```
asu rebootbmc [-nx]
```

Note: If the optional -nx parameter is specified, the ASU performs the operation for node x, where x is the selected node in a multi-node system. x can be a number from 1 through 8. If the -nx parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **rebootbmc** command is a message that indicates that the restart of the baseboard management controller is completed.

Example

The **rebootbmc** command and corresponding output are shown in the following example.

Command line:

```
asu rebootbmc
```

Output:

```
Rebooting BMC...done
```

Rebootimm command

Note: This command is for IMM-based servers only.

Use the **rebootimm** command to restart the integrated management module (IMM). This is useful after you make changes to IMM settings for which a restart is necessary for the changes to take effect. This command takes approximately 4 minutes to complete.

Syntax

The syntax of the **rebootimm** command is

```
asu rebootimm [-nx] [connect_options]
```

Notes:

1. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

2. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **rebootimm** command is a message that indicates that the restart of the IMM is completed.

Examples

The **rebootimm** command and corresponding output are shown in the following example.

Command line:

```
asu rebootimm
```

Output:

```
Connected to IMM at IP address 169.254.95.118
Issuing reset command to IMM.
The IMM has started the reset. Waiting for the reset to complete.
Connected to IMM at IP address 169.254.95.118
Reset completed successfully.
```

Rebootrsa command

Note: This command is for BIOS-based servers only.

Use the **rebootrsa** command to restart the Remote Supervisor Adapter or Remote Supervisor Adapter II. This is useful when you must restart a Remote Supervisor Adapter or Remote Supervisor Adapter II after you make changes to Remote Supervisor Adapter settings for which a restart is necessary for the changes to take effect. This command takes approximately 30 seconds to be completed.

Syntax

The syntax of the **rebootrsa** command is

```
asu rebootrsa [-nx]
```

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **rebootrsa** command is a message that indicates that the restart of the Remote Supervisor Adapter is completed.

Example

The **rebootrsa** command and corresponding output are shown in the following example.

Command line:

```
asu rebootrsa
```

Output:

```
Rebooting RSA/RSA2...done
```

Replicate command

Use the **replicate** command to replicate all settings in the input `update config` file.

On IMM-based servers, the **replicate** command skips the settings that are defined as "noreplicate." To see the settings that are defined as "noreplicate," use the **show** command and specify the **noreplicate** class.

Syntax

The syntax of the **replicate** command is

```
asu replicate file_name [connect_options]
```

where *file_name* is the name of the file that was created by a previous **asu save** command or by redirecting the output of a previous **asu show** command to a file.

Note: The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **replicate** command is a list of outputs from set commands.

```
<setting 1>=<value 1>
<setting 2>=<value 2>

:
:

<setting n>=<value n>
```

Examples

The **replicate** command and corresponding output are shown in the following examples.

Command line (IMM-based servers):

```
asu replicate rep.data
```

rep.data file (Show or save output file):

```
uEFI.Com1BaudRate=115200
uEFI.Com1DataBits=8
uEFI.Com1Parity=None
uEFI.Com1StopBits=1
uEFI.Com1TextEmul=VT100
uEFI.Com1ActiveAfterBoot=Enable
uEFI.Com1FlowControl=Disable
```

Output:

```
uEFI.Com1BaudRate=115200
uEFI.Com1DataBits=8
uEFI.Com1Parity=None
uEFI.Com1StopBits=1
uEFI.Com1TextEmul=VT100
uEFI.Com1ActiveAfterBoot=Enable
uEFI.Com1FlowControl=Disable
```

Command line (BIOS-based servers):

```
asu replicate rep.data
```

rep.data file (Show output file):

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

.
.
.

CMOS_OSUSBControl=Other OS
```

Output:

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

.
.
.

CMOS_OSUSBControl=Other OS
```

Note: The output is identical to the show output file that is used as input to the **replicate** command.

Resetrsa command

Note: This command is for BIOS-based servers only.

Use the **resetrsa** command to reset the Remote Supervisor Adapter or Remote Supervisor Adapter II to the default settings and then restart it. This command takes approximately 30 seconds to be completed.

Syntax

The syntax of the **resetrsa** command is

```
asu resetrsa [-nx]
```

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **resetrsa** command is a message that indicates that the restart of the Remote Supervisor Adapter is completed.

Example

The **resetrsa** command and corresponding output are shown in the following example.

Command line:

```
asu resetrsa
```

Output:

```
Rebooting RSA/RSA2...done
```

Restore command

Use the **restore** command to restore all settings that are defined in the update configuration file to the server.

In IMM-based servers, the backup control settings are *not* restored by default. To restore the backup control settings in an IMM-based server, you must specify the **-incbackupctl** modifier (see the specific syntax in Table 2 on page 25).

Limitations:

Settings that are defined in the password and authentication class are not restored during a restore operation. To list the settings that belong to the password or authentication class, use the following command:

```
asu showvalues authentication
```

To list only the password class settings, use the following command:

```
asu showvalues password
```

During a restore operation, the ASU does not delete settings that might exist on the target server that is being restored and that are not included in the restore file.

The restore command simply restores the values that are defined in the restore file for those settings that exist in the target system. Therefore, the restore operation should be viewed as restoring values and not as a system settings restore command.

Syntax

The syntax of the **restore** command is

```
asu restore file_name [--incbackupctl ] [-nx] [connect_options]
```

where *file_name* is the name of the file that was created by a previous **asu save** command or by redirecting the output of a previous **asu show** command to a file.

Notes:

1. The optional **–incbackupctl** parameter is used on a restore operation to specify whether the settings that are defined by the backupctl class are to be included. To list the backcupctl class, use the **show** command and specify the backupctl class.
2. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

Each setting and the restored value are displayed.

```
<setting 1>=<value 1>
<setting 2>=<value 2>

.
.
.

<setting n>=<value n>
```

Examples

The **restore** command and corresponding output are shown in the following examples.

Command line (IMM-based servers):

```
asu restore rep.data
```

rep.data file (Show or save output file):

```
uEFI.Com1BaudRate=115200
uEFI.Com1DataBits=8
uEFI.Com1Parity=None
uEFI.Com1StopBits=1
uEFI.Com1TextEmul=VT100
uEFI.Com1ActiveAfterBoot=Enable
uEFI.Com1FlowControl=Disable
```

Output:

```
uEFI.Com1BaudRate=115200
uEFI.Com1DataBits=8
uEFI.Com1Parity=None
uEFI.Com1StopBits=1
uEFI.Com1TextEmul=VT100
uEFI.Com1ActiveAfterBoot=Enable
uEFI.Com1FlowControl=Disable
```

Command line (BIOS-based servers):

```
asu restore rep.data
```

rep.data file (Show or save output file):

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

:
:

CMOS_OSUSBControl=Other OS
```

Output:

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

:
:

CMOS_OSUSBControl=Other OS
```

Note: The output is identical to the save or show output file that is used as input to the **restore** command.

Save command

Use the **save** command to save all settings into a file. By default, backup control settings (settings in the class backupctl) are saved unless an optional modifier is specified. The supported modifiers include --group, --setlist, and --excbackupctl (see the specific syntax in Table 2 on page 25).

Limitations:

Settings that are defined in the password class or authentication settings (user IDs) class settings are not saved during a save operation. To list the settings that belong to the password class, type the following command:

```
asu showvalues password
```

Syntax

The syntax of the **save** command is

```
asu save file_name [--group group_name | --setlist set_name1..set_nameN]
[--excbackupctl] [-nx] [connect_options]
```

where *file_name* is the name of the file to which the saved settings are written.

Notes:
1. If the optional **--group** parameter is specified, only settings that belong to the specified group are saved.
2. If the optional **--setlist parameter** is specified, only settings that belong to the specified list of settings are saved.
3. If the optional **--excbackupctl** parameter is specified, backup control settings (settings in the class backupctl) are not saved.
4. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
5. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **save** command is a message that indicates that the settings are saved to the file name that is specified in the command.

Examples

The **save** command and corresponding output are shown in the following examples.

Command line to save all settings:

```
asu save save.txt
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

:
:

BMC_CRTRequired=Enabled
BMC_KbdRequired=Enabled

:
:
```

Command line to save all BIOS settings:

```
asu save save.txt --group bios
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled

:
:
```

Command line to save all IMM settings:

```
asu save save.txt --group IMM
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
IMM.PowerRestorePolicy=Last state
IMM.PowerOnAtSpecifiedTime_Year=0

:
:
```

Command line to save only the IMM.LockoutPeriod and UEFI.rehook19 settings:

```
asu save save.txt --setlist IMM.LockoutPeriod uEFI.rehook19
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
IMM.LockoutPeriod=2
uEFI.rehook19=CMOS_KbdRequired=Enabled
```

Command line to save all settings except for those in the backupctl class:

```
asu save save.txt --excbackupctl
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
uEFI.TurboModeEnable=Enable
uEFI.ProcessorEistEnable=Enable
```

.
.
.

Command line to save all BIOS settings from node 2:

```
asu save save.txt --group bios –n2
```

Output:

```
Settings saved to save.txt
```

save.txt file (Save output file):

```
CMOS_CRTRequired=Enabled
CMOS_KbdRequired=Enabled
```

.
.
.

Set command

Use the **set** command to change the value of a setting or to list a setting. Starting with the ASU 3.01, the **set** command also creates an instance if the instance number does not exist, and the instance value is less than or equal to the maximum allowed instances for the setting. For more information about instances, see "Instances of settings" on page 18.

Syntax

The syntax of the **set** command is either:

```
asu set setting value [-v] [-nx] [connect_options]
```

where *setting* is the name of a setting to change.

Use the command **asu show all** to show a list of available settings; *value* is the exact value string to set for *setting*.

For settings with a single value, the **asu showvalues** command output is *setting_name=value*.

Or

```
asu set setting value1=value2=valueN [-v] [-nx] [connect_options]
```

where *setting* is the name of a setting to change that can accept a list of values.

Use the **asu showvalues** *setting* command to show a list of all values that are available for the *setting*.

For settings that allow a list of values, the **asu showvalues** command output syntax is *setting_name==value1=value2=valueN*. The double equal sign (==) shows that the setting can accept single or multiple values in an ordered list.

Limitations:

Settings that are defined in the password class cannot be set remotely. To list the settings that belong to the password class, use the command:

```
asu showvalues password
```

Notes:

1. Values that contain spaces must be enclosed in quotation marks ("). If a value contains quotation marks, add a backslash before each quotation mark (\") in the value.
2. If the optional **-v** parameter is specified, the output is verbose.
3. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
4. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The output of the **set** command when the **–v** parameter is not specified is the setting name and the new value. When the **–v** parameter is specified, the description of the setting is also shown.

Setting with a single value without the **-v** parameter:

```
<setting>=<new value>
```

Setting with a single value with the **-v** parameter:

```
<setting>: <setting description> = <new value>
```

Setting with multiple values without the **-v** parameter:

```
<setting>=<new value1>=<new value2>=<new valueN>
```

Setting with multiple values with the **-v** parameter:

```
<setting>: <setting description>=<new value1>=<new value2>=<new valueN>
```

Examples

The **set** command and corresponding output are shown in the following examples.

Command line:

```
asu set CMOS_CRTRequired Disabled
```

Output:

```
CMOS_CRTRequired=Disabled
```

Command line:

```
asu set CMOS_DisketteA "1.44 MB 3.5\""
```

Output:

```
CMOS_DisketteA=1.44 MB 3.5"
```

Command line:

```
asu set RSAIP_HostIPAddress1 192.168.0.100
```

Output:

```
RSAIP_HostIPAddress1=192.168.0.100
```

Command line:

```
asu set RSAString_LoginId2 rsauser
```

Output:

```
RSAString_LoginId2=rsauser
```

Command line to set the boot order to be CD/DVD ROM, then diskette, and then Hard Disk 0:

```
asu set BootOrder.BootOrder "CD/DVD Rom=Floppy Disk=Hard Disk 0"
```

Output:

```
BootOrder.BootOrder=CD/DVD Rom=Floppy Disk=Hard Disk 0
```

To create a new record instance and set the record key setting:

Command line:
```
asu set iSCSI.AttemptName.2 "MyAttempt2Name"
```

Output:
```
iSCSI.AttempName.2=MyAttempt2Name
```

Note: All other settings in this record (for example, iSCSI.LocalIp.2 and iSCSI.SubnetMask.2) are set to default values.

To set the other settings for this instance in this record:

Command line:
```
asu set iSCSI.LocalIp.2 "9.5.107.170"
```

Output:
```
iSCSI.LocalIp.2=9.5.107.170
```

Which is followed by the command line:
```
asu set iSCSI.SubnetMask.2 "255.255.255.0"
```

Output:
```
iSCSI.SubnetMask.2="255.255.255.0"
```

Show command

Use the **show** command to see the current value of one or more settings.

Syntax

The syntax of the **show** command is

```
asu show [all |  --group group_name |  setting_name |
--setlist name1..nameN | class] [-v] [-nx] [connect_options]
```

If no setting command modifier or class setting is specified, all settings and their current values are displayed.

Notes:
1. If the optional **all** parameter is specified, all settings are displayed.
2. If the optional --group *group_name* is specified, only settings in the group *group_name* are displayed.
3. If the optional --setlist *name1..nameN* is specified, only the settings that are specified in *name1..nameN* are displayed.
4. If the optional *class* setting is specified, only settings that belong to the specific class are displayed.
5. If the optional **-v** parameter is specified, the output is verbose.
6. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
7. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

If the **–v** parameter is not specified, the setting and the current value are displayed. If the **–v** parameter is specified, the description of the setting is displayed as well as an indicator that the value is the default value.

Without the **-v** parameter:

```
<setting>=<current value>
```

With the **-v** parameter:

```
<setting>: <setting description> = <current value> [(default)]
```

Examples

The **show** command and corresponding output are shown in the following examples.

Command line:

```
asu show RSAIP_HostIPAddress1
```

Output:

```
RSAIP_HostIPAddress1=192.168.0.100
```

Command line:

```
asu show CMOS_WakeOnLAN -v
```

Output:

```
CMOS_WakeOnLAN: Wake On Lan = Enabled (default)
```

Command line:

```
asu show bios
```

Output:

```
CMOS_DisketteA=1.44 MB 3.5"
CMOS_CRTRequired=Disabled

:
:

CMOS_OSUSBControl=Other OS
```

Showdefault command

Use the **showdefault** command to show the default value for one or more settings.

Syntax

The syntax of the **showdefault** command is

```
asu showdefault [all | --group group_name | setting_name |
--setlist name1..nameN | class] [-v] [-nx] [connect_options]
```

If no setting command modifier or class setting is specified, all settings and their current values are displayed.

Notes:

1. If the optional **all** parameter is specified, all setting are displayed.
2. If the optional --group *group_name* is specified, only settings in the group *group_name* are displayed.
3. If the optional --setlist *name1..nameN* is specified, only the settings that are specified in *name1..nameN* are displayed.
4. If the optional *class* setting is specified, only settings that belong to the specific class are displayed.
5. If the optional **-v** parameter is specified, the output is verbose.
6. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
7. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

If the **–v** parameter is not specified, the setting and the default value are displayed. If the **–v** parameter is specified, the description of the setting is also displayed.

Without the **-v** parameter:

setting=default value

With the **-v** parameter:

setting: *setting description = default value*

For instance settings, the default state can be that the instance does not exist. If this is the case, the *default value* that is output by the **showdefault** command is "delete." This indicates that all instances of the setting are deleted if the **loaddefault** command is performed on the setting. For more information about instances, see "Instances of settings" on page 18.

Examples

The **showdefault** command and corresponding output are shown in the following examples.

Command line:

```
asu showdefault CMOS_WakeOnLAN –v
```

Output:

```
CMOS_WakeOnLAN: Wake On Lan = Enabled
```

Command line:

```
asu showdefault bios
```

Output:

```
CMOS_DisketteA=1.44 MB 3.5"
CMOS_CRTRequired=Disabled

:
:

CMOS_OSUSBControl=Other OS
```

Command line:

```
asu showdefault iSCSI.AttemptName.1
```

Output:

```
iSCSI.AttemptName.1=<remove>
```

Showgroups command

Use the **showgroups** command to list the setting groups that are available on the server. The settings are organized into groups. All UEFI settings belong to the UEFI group, and all BIOS settings belong to the BIOS group. The listed groups can be used as a class of commands that support the class modifier, or they can be used with the --group *group* option on commands that support this option.

Syntax

The syntax of the **show** command is

```
asu showgroups [-nx] [connect_options]
```

Notes:

1. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

2. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

The setting groups that are available on the server are displayed.

Examples

The **showgroups** command and corresponding output are shown in the following example.

Command line:

```
asu showgroups
```

Output on an IMM-based server:

```
IMM
SYSTEM_PROD_DATA
uEFI
BootOrder
```

Output on a BIOS-based server:

```
bios
bmc
rsa
```

Showlocation command

Note: This command is for BIOS-based servers only.

Use the **showlocation** command to show the location of one or more settings. This shows where the actual data for the setting is stored.

Syntax

The syntax of the **showlocation** command is

```
asu showlocation [all | --group group_name | setting_name |
--setlist name1..nameN | class] [-v] [-nx] [connect_options]
```

If no setting command modifier or class setting is specified, all settings and their current values are displayed.

Notes:

1. If the optional **all** parameter is specified, all setting are displayed.
2. If the optional --group *group_name* is specified, only settings in the group *group_name* are displayed.
3. If the optional --setlist *name1..nameN* is specified, only the settings that are specified in *name1..nameN* are displayed.
4. If the optional *class* setting is specified, only settings that belong to the specific class are displayed.
5. If the optional **-v** parameter is specified, the output is verbose.
6. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
7. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.

Output

If the **–v** parameter is not specified, the setting and its location are displayed. If the **–v** parameter is specified, the description of the setting is also displayed.

Without the **-v** parameter:

```
<setting>=<location>[<extra location info>]

if <location> is CMOS, <extra location info> is of the form
        <byte offset>","<bit offset>","<number of bits>
if <location> is SP, <extra location info> is of the form
        <SP dot byte 1>"."<SP dot byte 2>"," ... "."<SP dot byte n>
if <location> is SP6, <extra location info> is of the form
        ">"<write command info>  ">"<read command info> and <write command info>
        and <read command info> are of the form
        <read command byte>["@"<data offset>]
        ("."<command data>)*["|"<request data length>]
```

Without the **-v** parameter:

```
<setting>: <setting description> {
          <location>[<extra location info>]
}

if <location> is CMOS, <extra location info> is of the form
          <byte offset>","<bit offset>","<number of bits>
if <location> is SP, <extra location info> is of the form
          <SP dot byte 1>"."<SP dot byte 2>"." ... "."<SP dot byte n>
if <location> is SP6, <extra location info> is of the form
          ">"<write command info>  "<"<read command info> and <write command info>
          and <read command info> are of the form
          <read command byte>["@"<data offset>]
          ("."<command data>)*["|"<request data length>]
```

Examples

The **showlocation** command and corresponding output are shown in the following examples.

Command line:

```
asu showlocation CMOS_SerialA
```

Output:

```
CMOS_SerialA=CMOS[70,00,03]
```

Command line:

```
asu showlocation CMOS_SerialA —v
```

Output:

```
CMOS_SerialA: Serial Port A {
     CMOS[70,00,03]
}
```

Command line:

```
asu showlocation RSA_Network1
```

Output:

```
RSA_Network1=SP[04.09.01.01.02]
```

Command line:

```
asu showlocation RSA_Network1 -v
```

Output:

```
RSA_Network1: Network Interface 1 {
     SP[04.09.01.01.02]
}
```

Showvalues command

Use the **showvalues** command to list all possible values for one or more settings. This is useful for finding the **value** parameter that is used for the **set** command.

Syntax

The syntax of the **showvalues** command is

```
asu showvalues [all | --group group_name | setting_name |
--setlist name1..nameN | --instances | class] [-v | -t] [-nx] [connect_options]
```

Notes:

1. If the optional **all** parameter is specified, all setting are displayed.
2. If the optional --group *group_name* is specified, only settings in the group *group_name* are displayed.
3. If the optional --setlist *name1..nameN* is specified, only the settings that are specified in *name1..nameN* are displayed.
4. If the optional *class* setting is specified, only settings that belong to the specific class are displayed.
5. If the optional **-v** parameter is specified, the output is verbose.
6. If the optional **-t** parameter is used, the output includes the raw values.
7. If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).
8. The connect options are defined for only IMM-based servers. The --host *ip_address* connect option is required if you connect remotely to the IMM. The --user *user_id* and --password *password* connect options are required only if the default user ID and password settings are changed in the IMM for either a local or remote LAN connection. The --user *user_id* and --password *password* connect options are not required if you are using the local KCS interface.
9. If the optional --instances parameter is specified, only settings that can have instances are displayed. The minimum and maximum number of instances allowed for the settings is also displayed. For more information about instances, see "Instances of settings" on page 18.

Output

If the **–v** parameter is not specified, the setting and its values are displayed. If the **–v** parameter is specified, the description of the setting is also displayed.

If the setting is an enumerated type:

–v and –t not specified:

```
<setting>=<value 1>=<value 2>=...=<value n>
```

–v specified:

```
<setting>: <setting description> {
        <value 1>
        <value 2>
        .
        .
        .
        <value n>
}
```

–t specified:

```
<setting>=<value 1>[<raw 1>]=<value 2>[<raw 2>]=...=<value n>[<raw n>]
```

If the setting is a string type:

–v not specified:

```
        <setting>=char[<length>]
                <length> is the max length string that can be entered.
                If <length> is omitted, there is no maximum.
```

–v specified:

```
        <setting>: <setting description> {
                char[<length>]
    }
                <length> is the max length string that can be entered.
                If <length> is omitted, there is no maximum.
```

If the setting is an IP address type:

–v not specified:

```
        <setting>= x.x.x.x where (0 <= x <= 255)
```

–v specified:

```
        <setting>: <setting description> {
                A string formatted x.x.x.x, where x is an integer from 0 to 255
    }
```

If the setting is a MAC address type:

–v not specified:

```
        <setting>= x:x:x:x where (0 <= x <= FF)
```

–v specified:

```
        <setting>: <setting description> {
                A string formatted x:x:x:x, where x is a hex integer from 0 to FF
    }
```

If the setting is a keystroke sequence type:

–v not specified:

```
        <setting>=(c )* where c in [0x01-0xFF, 'ESC', ^A-^Z, ^[-^_, ' '-'~']
```

–v specified:

```
        <setting>: <setting description> {
        A space-separated sequence of characters where each of the characters is:
                0x01-0xFF, 'ESC', ^A-^Z, ^[-^_, ' ', '!', '#', '$', '%', '&',
                '(', ')', '*', '+', ',', '-', '.', '/', '0'-'9', ':', ';', '<',
                '=', '>', '?', '@', A'-'Z', '[', '\', ']', '^', '_', 'a'-'z',
                '{', '|', '}', or '~'
```

If the --instances parameter is specified, the output for a setting that is not part of a record is:

-v not specified:

```
        <setting>= numeric type=dec min=0 max=65535 default=3260 [min=0, max=256]
                The output for the setting indicates that it can have anywhere from
                0 to 256 instances. This is indicated by '[min=0, max=256]'
```

-v specified:

```
<setting>: T <setting description> {
    numeric data
    numeric type = dec
    minimum value = 0
    maximum value = 65535
    default value = 3260
    min instances = 0
    max instances = 256
  }
```

The verbose output for the setting also indicates that it can have anywhere from 0 to 256 instances. In this case this is indicated by 'min instances = 0' and 'max instances = 256'.

If the --instances parameter is specified, the output for a settings that are part of a record is:

-v not specified:

```
setting1=char[] default="" [min=0, max=12] recordKey
setting2=char[] default="" [min=0, max=12] recordKey="setting1"
setting3=char[] default="" [min=0, max=12] recordKey="setting1"
```

The output for all of the settings indicate that they can have anywhere from 0 to 12 instances. *setting1* is the recordKey and *setting2* and *setting3*: are part of a record, where *setting1* is the key setting. For more information about records, see "Record management" on page 19.

-v specified:

```
<setting1>: <setting1 description> {
    char[]
    default =
    min instances = 0
    max instances = 12
    Record Key
}
<setting2>: <setting2 description> {
    char[]
    maximum characters = 16
    pattern = ^(.{4,16})?
    default =
    Record Key = <setting1>
}
<setting3>: <setting3 description> {
    char[]
    maximum characters = 16
    pattern = ^(.{4,16})?
    default =
    Record Key = <setting1>
}
```

The verbose output for *setting1* also indicates that it can have anywhere from 0 to 256 instances. In this case this is indicated by 'min instances = 0' and 'max instances = 256'. *setting1* is the recordKey and *setting2* and *setting3* are part of the same record as *setting1*, where *setting1* is the key setting. For more information about records, see "Record management" on page 19.

Examples

The **showvalues** command and corresponding output are shown in the following examples.

Command line:

```
asu showvalues CMOS_SerialA
```

Output:

```
CMOS_SerialA=PnP=Auto-configure=<Port 3F8, IRQ 4>=Port 2F8,
IRQ 3=Port 3E8, IRQ 4=Port 2E8, IRQ 3=Disabled
```

Command line:

```
asu showvalues CMOS_SerialA –v
```

Output:

```
CMOS_SerialA: Serial Port A {
PnP
Auto-configure
Port 3F8, IRQ 4 (default)
Port 2F8, IRQ 3
Port 3E8, IRQ 4
Port 2E8, IRQ 3
Disabled
}
```

Command line:

```
asu showvalues CMOS_SerialA -t
```

Output:

```
CMOS_SerialA=PnP=Auto-configure=<Port 3F8, IRQ 4>=Port 2F8,
IRQ 3=Port 3E8, IRQ 4=Port 2E8, IRQ 3=Disabled
```

Command line:

```
asu showvalues -–instances
```

Output:

```
IMM.LoginId=char[] maxchars=16 pattern=^(.{4,16})? default=<remove>
[min=0, max=12] recordKey
IMM.Password=char[] default="" [min=0, max=12] recordKey="IMM.LoginId"
IMM.AuthorityLevel=<Supervisor>=ReadOnly=Custom [min=0, max=12]
recordKey="IMM.LoginId"
```

Version command

Use the **version** command to show the version and build date of the ASU.

Beginning with ASU 3.00, the version number uses the following standard format:

$w.xy.zzz$

where w is the major revision number (this value changes if there are major new features in the release), x is the minor revision number (this value changes if there are only minor new features in the release), y is the sub-minor revision number (this value changes if there are only fixes in the release), and zzz is the build number.

Syntax

The syntax of the **version** command is

```
asu version
```

Output

The output of the **version** command shows the current version and build date of the ASU.

Example

The **version** command and corresponding output are shown in the following example.

Command line:

```
asu version
```

Output:

```
Advanced Settings Utility 3.00.65A Mar 12 2009
```

Writeraw command

Note: This command is for BIOS-based servers only.

Use the **writeraw** command to read and write CMOS data that is saved in a file through the **readraw** command.

Syntax

The syntax of the **writeraw** command is

```
asu writeraw filename [-nx]
```

where *filename* is the name of a file in which the raw CMOS data is to be read.

Note: If the optional -n*x* parameter is specified, the ASU performs the operation for node *x*, where *x* is the selected node in a multi-node system. *x* can be a number from 1 through 8. If the -n*x* parameter is not specified, the operation is performed on the primary node (node 1).

Output

The output of the **writeraw** command is a file that is generated in the directory where the ASU is running and a message that indicates that the raw write operation is completed.

Examples

The **writeraw** command and corresponding output are shown in the following example.

Command line:

```
asu writeraw CMOSraw.dat
```

Output:

```
Raw CMOS read from CMOSraw.dat, written to CMOS
CMOSraw.dat file generated
```

Appendix A. Getting help and technical assistance

If you need help, service, or technical assistance or just want more information about IBM products, you will find a wide variety of sources available from IBM to assist you. This section contains information about where to go for additional information about IBM and IBM products, what to do if you experience a problem with your system, and whom to call for service, if it is necessary.

Before you call

Before you call, make sure that you have taken these steps to try to solve the problem yourself:

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system and any optional devices are turned on.
- Use the troubleshooting information in your system documentation, and use the diagnostic tools that come with your system. Information about diagnostic tools is in the *Problem Determination and Service Guide* on the IBM *Documentation* CD that comes with your system.
- Go to the IBM support website at http://www.ibm.com/systems/support/ to check for technical information, hints, tips, and new device drivers or to submit a request for information.

You can solve many problems without outside assistance by following the troubleshooting procedures that IBM provides in the online help or in the documentation that is provided with your IBM product. The documentation that comes with IBM systems also describes the diagnostic tests that you can perform. Most systems, operating systems, and programs come with documentation that contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the documentation for the operating system or program.

Using the documentation

Information about your IBM system and preinstalled software, if any, or optional device is available in the documentation that comes with the product. That documentation can include printed documents, online documents, readme files, and help files. See the troubleshooting information in your system documentation for instructions for using the diagnostic programs. The troubleshooting information or the diagnostic programs might tell you that you need additional or updated device drivers or other software. IBM maintains pages on the World Wide Web where you can get the latest technical information and download device drivers and updates. To access these pages, go to http://www.ibm.com/systems/support/ and follow the instructions. Also, some documents are available through the IBM Publications Center at http://www.ibm.com/shop/publications/order/.

Getting help and information from the World Wide Web

On the World Wide Web, the IBM website has up-to-date information about IBM systems, optional devices, services, and support. The address for IBM System x® and xSeries® information is http://www.ibm.com/systems/x/. The address for IBM BladeCenter® information is http://www.ibm.com/systems/bladecenter/. The address for IBM IntelliStation® information is http://www.ibm.com/intellistation/.

You can find service information for IBM systems and optional devices at
http://www.ibm.com/systems/support/.

Software service and support

Through IBM Support Line, you can get telephone assistance, for a fee, with usage,
configuration, and software problems with System x and xSeries servers,
BladeCenter products, IntelliStation workstations, and appliances. For information
about which products are supported by Support Line in your country or region, see
http://www.ibm.com/services/sl/products/.

For more information about Support Line and other IBM services, see
http://www.ibm.com/services/, or see http://www.ibm.com/planetwide/ for support
telephone numbers. In the U.S. and Canada, call 1-800-IBM-SERV
(1-800-426-7378).

Hardware service and support

You can receive hardware service through your IBM reseller or IBM Services. To
locate a reseller authorized by IBM to provide warranty service, go to
http://www.ibm.com/partnerworld/ and click **Find Business Partners** on the right
side of the page. For IBM support telephone numbers, see http://www.ibm.com/
planetwide/. In the U.S. and Canada, call 1-800-IBM-SERV (1-800-426-7378).

In the U.S. and Canada, hardware service and support is available 24 hours a day,
7 days a week. In the U.K., these services are available Monday through Friday,
from 9 a.m. to 6 p.m.

IBM Taiwan product service

台灣 IBM 產品服務聯絡方式：
台灣國際商業機器股份有限公司
台北市松仁路 7 號 3 樓
電話：0800-016-888

IBM Taiwan product service contact information:
    IBM Taiwan Corporation
    3F, No 7, Song Ren Rd.
    Taipei, Taiwan
    Telephone: 0800-016-888

Appendix B. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

> IBM Director of Licensing
> IBM Corporation
> North Castle Drive
> Armonk, NY 10504-1785
> U.S.A.

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

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

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product, and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol ($^{®}$ or $^{™}$), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc., in the United States, other countries, or both and is used under license therefrom.

Intel, Intel Xeon, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc., in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

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

Important notes

Processor speed indicates the internal clock speed of the microprocessor; other factors also affect application performance.

CD or DVD drive speed is the variable read rate. Actual speeds vary and are often less than the possible maximum.

When referring to processor storage, real and virtual storage, or channel volume, KB stands for 1024 bytes, MB stands for 1 048 576 bytes, and GB stands for 1 073 741 824 bytes.

When referring to hard disk drive capacity or communications volume, MB stands for 1 000 000 bytes, and GB stands for 1 000 000 000 bytes. Total user-accessible capacity can vary depending on operating environments.

Maximum internal hard disk drive capacities assume the replacement of any standard hard disk drives and population of all hard disk drive bays with the largest currently supported drives that are available from IBM.

Maximum memory might require replacement of the standard memory with an optional memory module.

IBM makes no representation or warranties regarding non-IBM products and services that are ServerProven®, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. These products are offered and warranted solely by third parties.

IBM makes no representations or warranties with respect to non-IBM products. Support (if any) for the non-IBM products is provided by the third party, not IBM.

Some software might differ from its retail version (if available) and might not include user manuals or all program functionality.

Documentation format

The publications for this product are in Adobe Portable Document Format (PDF) and should be compliant with accessibility standards. If you experience difficulties when you use the PDF files and want to request a web-based format or accessible PDF document for a publication, direct your mail to the following address:

Information Development
IBM Corporation
205/A015
3039 E. Cornwallis Road
P.O. Box 12195
Research Triangle Park, North Carolina 27709-2195
U.S.A.

In the request, be sure to include the publication part number and title.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Index

IBM®

Part Number: 60Y1427

Printed in USA

(1P) P/N: 60Y1427