



# Utility Commands

---

This module describes the utility commands for Cisco IOS XR software. Utility commands provide CLI equivalents to common UNIX commands.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **universal** keyword can also be entered using the UNIX-equivalent (**-u**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

---

- [utility cut, on page 2](#)
- [utility date, on page 6](#)
- [utility date set, on page 9](#)
- [utility fgrep, on page 11](#)
- [utility find, on page 13](#)
- [utility less, on page 15](#)
- [utility mv, on page 17](#)
- [utility sort, on page 19](#)
- [utility tail, on page 22](#)
- [utility uniq, on page 24](#)
- [utility wc, on page 26](#)

# utility cut

To extract selected characters or fields from standard input or from a file, use the **utility cut** command in XR EXEC mode.

**utility cut** {**list** *character-list* | **fields** *field-list* [**nodelim**] [**delimiter** *delimiter-character*]*WORD*} [**file** *input-file*] | **usage**}

## Syntax Description

<b>list</b> <i>character-list</i>	<p>(-c) Cuts out the characters that are located on each line as specified with the <i>character-list</i> argument.</p> <p>The <i>character-list</i> argument specifies the character positions or range of the characters to be cut.</p> <ul style="list-style-type: none"> <li>Use a comma (,) to indicate more than one character. For example, <b>utility list 1,2,5</b> outputs the first, second, and fifth characters.</li> <li>Use a dash (-) to indicate a range. For example, <b>utility list 1-64</b> outputs the first 64 characters of each line, <b>utility list 5-</b> outputs the fifth character to the end of the line.</li> </ul> <p><b>Note</b> Lines are separated by a delimiter. The default delimiter is tab.</p>
<b>fields</b> <i>field-list</i>	<p>(-f) Cuts out the fields (lines) as indicated with the <i>field-list</i> argument.</p> <p>The <i>field-list</i> argument specifies the field numbers or ranges. For example, <b>utility field 2,9</b> outputs the second and ninth fields, <b>utility field 1-3</b> outputs the first three fields, <b>utility field -6</b> outputs the first six fields.</p> <p><b>Note</b> The fields indicated by the <i>field-list</i> argument are assumed to be separated in the file by a delimiter character. The default delimiter is tab. Use the <b>delimiter delimiter</b> option to specify a delimiter character. Lines without field delimiters are processed unless the <b>nodelim</b> keyword is specified.</p>
<b>nodelim</b>	<p>(Optional) (-s) Ignores lines with no delimiter. Use this optional keyword when the <b>fields field-list</b> keyword and argument is specified.</p>
<b>delimiter</b> <i>delimiter-character</i>	<p>(Optional) (-d) Specifies an alternative delimiter to indicate the end of each field. Replace the <i>delimiter-character</i> argument with the character used as the delimiter.</p>
<i>WORD</i>	<p>(Optional) UNIX command-line option string. The maximum number of characters is 80.</p>
<b>file</b> <i>input-file</i>	<p>(Optional) Storage device and directory path of the text file used instead of the standard input (keyboard input).</p> <p>The syntax of the <i>input-file</i> argument is: <i>device</i> :[/ <i>directory-path</i>]/ <i>filename</i></p> <p>The <i>device</i> argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (?) function to display the available storage devices and network protocols.</p>

---

**usage** (Optional) Displays the UNIX options supported by this command.

---

**Command Default** If no file is specified, the keyboard input (standard input) is used.  
The delimiter is tab.

**Command Modes** XR EXEC

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

**Usage Guidelines** The **utility cut** command cuts out columns, fields, or characters displayed from standard input or from a file. Use the **fields** *field-list* keyword and argument if the fields vary in length from line to line. (The lines must be separated by a delimiter character.) By default, the field delimiter character is the Tab key. Use the **delimiter** *delimiter-character* keyword and argument to specify a different delimiter.

Use the **list** *character-list* keyword and argument only if the fields are of a fixed length. Replace the *character-list* argument with the character positions to be extracted.

For the *character-list* argument, use a comma (,) to indicate more than one character, or use a dash (-) to indicate a range. For example, **utility list 1,2,5** outputs the first, second, and fifth characters, **utility list 1-64** outputs the first 64 characters of each line, **utility list 5-** outputs the fifth character to the end of the line.

You can also use the cut utility as a filter. If no files are specified, the keyboard input (standard input) is used.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **fields** keyword can also be entered using the UNIX-equivalent (**-f**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility cut** command is entered with the **list** *character-list* keyword and argument to display the first 10 characters in each line. The output is from the results of the **show version** command, which is entered with the pipe (|) character:

```
RP/0/RP0/CPU0:router# show version | utility cut list 1-10
```

```
Cisco IOS
Copyright

ROM: Syste

router upt
System ima

cisco CRS-
7457 proce

16 Gigabit
2 Ethernet
```

```

20 Packet
20 SONEt/S
2043k byte
38079M byt
1000592k b
1000640k b

Configurat
Package ac
--More--

```

In the following example, the **utility cut** command is used to extract fields from a file:

```

RP/0/RP0/CPU0:router# utility cut fields 1,5 delimiter : file disk0:/usr/passwd

root:Super-User
daemon:
bin:
sys:
adm:Admin
lp:Line Printer Admin
uucp:uucp Admin
nuucp:uucp Admin
listen:Network Admin
nobody:Nobody

```

In the following example, the **utility cut** command is used with the **delimiter** keyword to specify an alternative field delimiter:

```

RP/0/RP0/CPU0:router# utility cut fields 1,4,5 delimiter : file disk0:/usr/passwd

root:1:Super-User
daemon:1:
bin:2:
sys:3:
adm:4:Admin
lp:8:Line Printer Admin
uucp:5:uucp Admin
nuucp:9:uucp Admin
listen:4:Network Admin

```

In the following example, a range of fields is specified:

```

RP/0/RP0/CPU0:router# utility cut fields 1-4 delimiter : file disk0:/usr/passwd

root:x:0:1
daemon:x:1:1
bin:x:2:2
sys:x:3:3
adm:x:4:4
lp:x:71:8
uucp:x:5:5
nuucp:x:9:9
listen:x:37:4

```

In the following example, the **list** *character-list* keyword and argument are used to specify the character positions to be extracted:

```
RP/0/RP0/CPU0:router# utility cut list 1-30 file disk0:/usr/passwd
```

```
root:x:0:1:Super-User:/:/sbin/
daemon:x:1:1:/:
bin:x:2:2:/:usr/bin:
sys:x:3:3:/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/
uucp:x:5:5:uucp Admin:/usr/lib
nuucp:x:9:9:uucp Admin:/var/sp
listen:x:37:4:Network Admin:/u
nobody:x:60001:60001:Nobody:/:
noaccess:x:60002:60002:No Acce
nobody4:x:65534:65534:SunOS 4.
```

```
=====
```

In the following example, the UNIX equivalent options are used directly. First, the **utility cut** command is entered with the **usage** keyword to display the possible options. Next, the **utility cut** command is entered with the options to extract the desired data.

```
RP/0/RP0/CPU0:router# utility cut usage
```

```
cut -c list [file], cut -f list [-d delim] [-s] [file]
```

```
RP/0/RP0/CPU0:router# utility cut -f 1,4 -d : disk0:/usr/passwd
```

```
root:1
daemon:1
bin:2
sys:3
adm:4
lp:8
```

# utility date

To display the date and time, use the **utility date** command in

XR EXEC

mode.

**utility date** {**format** *word* | **universal** | **usage***WORD*}

## Syntax Description

**format** *word* (Optional) (+) Specifies the format for the date display. Use the online help system to display the available format syntax for the *word* argument.

**universal** (Optional) (-u) Displays the date in Coordinated Universal Time (UTC) instead of local time. UTC is the standard term for Greenwich Mean Time (GMT).

**usage** (Optional) Displays the UNIX options supported by this command.

*WORD* (Optional) UNIX command-line option string. The maximum number of characters is 80.

## Command Default

The date is displayed in local time.

## Command Modes

XR EXEC

## Command History

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

## Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The **utility date** command displays the internal time and date for the router.

### Date Format

Use the **format** *word* option to specify the format and content of the displayed date and time. The format is composed of ASCII characters and field descriptors prefaced with %, in a manner similar to a C-language printf() format specifier. In the output, each field descriptor is replaced by its corresponding value; all other characters are copied to the output without change. The format is specified using the following characters:

**%C**

Century in 'CC' form. For example: 20

**%y**

Year in 'YY' form. For example: 06

- %m**  
Month in 'MM' form. For example: 08
- %d**  
Date in 'DD' form. For example: 28
- %H**  
Hour in 'hh (24 hr.)' form. For example: 18
- %M**  
Minutes in 'mm' form. For example: 55
- %S**  
seconds in 'ss' form. For example: 24



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses ( ) in the syntax description. For example, the **universal** keyword can also be entered using the UNIX-equivalent (**-u**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID	Task ID	Operations
	universal	execute

This example shows how to display the router date and time using the **utility date** command:

```
RP/0/RP0/CPU0:router# utility date
Fri Aug 04 11:53:38 UTC 2006
```

This example shows how to display the router date and time using a variety of options with the **format** keyword:

```
RP/0/RP0/CPU0:router# utility date format "%y%m%d"
060828
RP/0/RP0/CPU0:router# utility date format "%y-%m-%d"
06-08-28
RP/0/RP0/CPU0:router# utility date format "%C%y-%m-%d"
2006-08-28
RP/0/RP0/CPU0:router# utility date format "%C%y-%m-%d:%H:%M:%S"
2006-08-28:02:09:58
RP/0/RP0/CPU0:router# utility date format "DATE: %y-%m-%d %nTIME: %H:%M:%S"
```

DATE: 06-09-17  
TIME: 12:42:24

**Related Commands**

Command	Description
<a href="#">utility date set, on page 9</a>	Sets the internal date and time of the router.



# utility date set

To set the router time, use the **utility date set** command in mode.

**utility date set** *hh:mm:ss*

## Syntax Description

<i>hh</i>	Specifies the hour in 2-digit numerical format. Range is 00 to 23.
<i>mm</i>	Specifies the minutes in 2-digit numerical format. Range is 0 to 59.
<i>ss</i>	Specifies the seconds in 2-digit numerical format. Range is 0 to 59.

## Command Default

None

## Command Modes

System Admin EXEC

## Command History

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

## Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

A colon (:) is required between the entry for hour, minutes, and seconds.



**Note** Generally, if the system is synchronized by a valid outside timing mechanism, such as a Network Time Protocol (NTP) clock source, or if you have a networking device with calendar capability, you need not set the software clock. Use the **date** command or the **clock set** command if no other time sources are available.



**Note** To manually copy the hardware clock (calendar) settings into the software clock, use the **clock read-calendar** command in EXEC mode.

By default, the system makes a “slow adjustment” if the new time is in the range of the following:

- –2.5 minutes + old time
- 5 minutes + old time

In a slow adjustment, the clock speed increases by less than 100 percent or decreases by less than 50 percent over a period of time from 1 second to 5 minutes until the clock catches up with the new time. This slow adjustment does not cause major discontinuities in the time flow. Use the **-S0** option to disable the slow adjustment.

---

**Task ID**

---

**Task ID   Operations**

---

---

universal   execute

---

The following example shows how to set the time using the **utility date set** command:

```
RP/0/RP0/CPU0:router(admin)# utility date set 13:07:00
```

```
  Fri Sep 15 13:07:00 UTC 2006
```

---

**Related Commands**

Command	Description
<a href="#">utility date, on page 6</a>	Displays the internal date and time of the router.

# utility fgrep

To search a file for a fixed character string, use the **utility fgrep** command in

XR EXEC

mode.

**utility fgrep** {**expr** *expression* | **script** *expression-file*} [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

**utility fgrep** *expression* [*WORD*] [**count**] [**linenum**] [**matchfile**] [**matchline**] [**nocase**] [**nofile**] [**reverse**] [**file** *search-file*]

## utility fgrep usage

Syntax Description	
<b>expr</b> <i>expression</i>	(-e) A regular expression, whose type is determined by the -e and -f options. This form is used when only one expression is specified on the command line. Any names specified after this option are treated as input files.
<b>script</b> <i>expression-file</i>	(-f) A file containing a set of regular expressions, each separated by a new line. The type of the expressions is determined by the -e and -f options. This form is used when more than one expression is specified. You can specify more than one -f option.  The syntax of the <i>expression-file</i> argument is: <i>device</i> :[/ <i>directory-path</i> ]/ <i>filename</i>
<i>WORD</i>	(Optional) UNIX command-line option string. The maximum number of characters is 20.
<b>count</b>	(Optional) (-c) Displays a count of selected lines.
<b>linenum</b>	(Optional) (-n) Before each output line, displays the line's line number.
<b>matchfile</b>	(Optional) (-l) ("el") Displays only the names of files containing the selected lines.
<b>matchline</b>	(Optional) (-x) Includes only input lines selected against an entire fixed string or regular expression.
<b>nocase</b>	(Optional) (-i) Ignores uppercase and lowercase distinctions during comparisons.
<b>nofile</b>	(Optional) (-h) Displays results without a filename prefix attached to the matched lines. This option applies only when more than one file is searched.
<b>reverse</b>	(Optional) (-v) Selects only those lines that don't match the specified patterns.
<b>file</b> <i>search-file</i>	(Optional) The file used for the search. Replace the <i>search-file</i> argument with the device and directory path of the file. The syntax for the <i>search-file</i> argument is: <i>device</i> :[/ <i>directory-path</i> ]/ <i>filename</i>
<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

## Command Default

The keyboard input (standard input) is used if no files are specified.

If more than one input file is specified, then the filename is displayed before each line.

---

**Command Modes** XR EXEC

---

Command History	Release	Modification
	Release 3.9.0	No modification.
	Release 5.0.0	This command was introduced.

---



---

**Usage Guidelines** To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The **utility fgrep** command searches files for a fixed character string (as opposed to **grep** and **egrep**, which search for a pattern that matches an expression).

The results are displayed to the standard output (terminal screen).




---

**Note** The **fgrep** utility options are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. For example, the **count** keyword can also be entered using the UNIX-equivalent (**-c**). To display the UNIX-equivalent syntax online, enter the **usage** keyword.

---



---

Task ID	Task ID	Operations
	universal	execute

---

The following example, the **utility fgrep** command is used with the **nocase** and **linenum** keywords:

```
RP/0/RP0/CPU0:router# show version | utility fgrep expr uptime nocase linenum
7:router uptime is 5 days, 20 hours, 10 minutes
```

# utility find

To locate files within one or more directories, use the **utility find** command in

XR EXEC

mode.

**utility find** {**path** *directory-path* {*LINE* | **name** *filename-pattern* | **user** *user-id*} | **usage**}

## Syntax Description

<b>path</b> <i>directory-path</i>	Specifies the storage device and directory for the file search. The search is performed for the specified directory and all subdirectories in that directory tree.  If a directory path is not specified, then the search is performed in the current directory (a path of . [dot] is assumed).
<i>LINE</i>	(Optional) UNIX command-line expressions provided as a string.
<b>name</b> <i>filename-pattern</i>	(Optional) Searches for the name of the file. The <i>filename-pattern</i> argument is a regular expression string.
<b>user</b> <i>user-id</i>	(Optional) Searches for files belonging to a specific user. The <i>user-id</i> argument is the username of the file owner.
<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

## Command Default

If a directory path is not specified, then the search is performed in the current directory.

If a **name** *filename-pattern* is not specified, then the search return all files in the specified directory.

If a user is not specified, then the search is performed for all users.

## Command Modes

XR EXEC

## Command History

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

## Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Use the **utility find** command to locate files within one or more directories. You can perform the search for a specific directory (and its subdirectories). If a directory is not specified, then the search is performed for the current directory.

To search for a regular expression string, use the **name** *filename-pattern* keyword and argument. Replace the *filename-pattern* argument with the regular expression string. If this option is not used, then all files within the specified directory are displayed.

To search for files belonging to a specific user, use the *user-id* argument. If this option is not used, then files belonging to all users are displayed.

---

**Task ID**


---

**Task ID    Operations**


---

 universal    execute
 

---

In the following example, the **utility find** command is used to locate the file named “-fwdg-3.8.0”. The path is the root directory of disk0:.

```
RP/0/RP0/CPU0:router# utility find path disk0: name hfr-fwdg-3.4.0

disk0:/instdb/admin_pkgs_mdata/hfr-fwdg-3.8.0
disk0:/hfr-fwdg-3.8.0
```

In the following example, the **utility find** command is used to locate files matching a pattern. In this example, all files ending in “.txt” are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr name *.txt

disk0:/usr/test2.txt
```

In the following example, the UNIX equivalent option is used to locate files matching a pattern. In this example, all files ending in “.txt” are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0: -name *.txt

disk0:/-base-3.8.0/etc/vim/doc/editing.txt
disk0:/-base-3.8.0/etc/vim/doc/help.txt
disk0:/-base-3.8.0/etc/vim/doc/intro.txt
disk0:/-base-3.8.0/etc/vim/doc/uganda.txt
disk0:/usr/test2.txt
```

In the following example, the files belonging to a specific user are displayed:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr user 0

disk0:/usr
disk0:/usr/passwd
disk0:/usr/test2.txt
```

In the following example, the UNIX equivalent option is used to display files belonging to a specific user:

```
RP/0/RP0/CPU0:router# utility find path disk0:/usr -user 0

disk0:/usr
disk0:/usr/passwd
disk0:/usr/test2.txt
```

# utility less

To display a file page-by-page, use the **utility less** command in

XR EXEC

mode.

**utility less** {[**exitEOF**] [**WORD**] | **nocase** | **position** *line-number* | **startat** *string*} [**file** *source-file*]

## Syntax Description

<b>exitEOF</b>	(Optional) ( <b>-E</b> ) Automatically exits the utility the first time an end-of-file is encountered.
<b>WORD</b>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
<b>nocase</b>	(Optional) ( <b>-i</b> ) Ignores uppercase and lowercase distinctions during comparisons.
<b>position</b> <i>line-number</i>	(Optional) ( <b>-j</b> ) Uses the line at <i>line-number</i> on the screen to position matched lines during a pattern search.
<b>startat</b> <i>string</i>	(Optional) ( <b>-p</b> ) Starts at the first occurrence of the pattern specified by the <i>string</i> argument in the file.
<b>file</b> <i>source-file</i>	(Optional) Specifies the storage device and directory path for the text file to be displayed. The default is standard input.  The syntax for the <i>source-file</i> argument is: <i>device</i> :[/ <i>directory-path</i> ]/ <i>filename</i>

## Command Default

If no text file is specified, standard input is assumed.

## Command Modes

XR EXEC

## Command History

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

## Usage Guidelines

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Use the **utility less** command to display files page by page. You can specify regular expressions for pattern matching using the **startat** keyword. You can scroll up as well as down. When you enter the less mode, commands are similar to the “vi” editor.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

## Task ID

### Task ID Operations

universal execute

The following example, the **utility less** command is used to display the file “config\_store”. Only part of the file is shown here.

```
RP/0/RP0/CPU0:router# utility less file disk0:/usr/config_store
```

```
Last configuration change at Tue Feb 20 18:34:02 2007 by xxx
!
hostname H1
line console
  exec-timeout 600 0
  session-timeout 600
!
line default
  exec-timeout 600 0
  session-timeout 600
!
.
.
.
```



# utility mv

To rename or move a file from one directory to another, use the **utility mv** command in

XR EXEC

mode.

**utility mv** [{*WORD* | **force** | **interactive**}] **source** *source-file* **target** *target-file* | **usage**}

<b>Syntax Description</b>	<b>WORD</b>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
	<b>force</b>	(Optional) ( <b>-f</b> ) Forces an overwrite if the target file already exists. There is no confirmation prompt.
	<b>interactive</b>	(Optional) ( <b>-i</b> ) Specifies to prompt for confirmation before renaming a file.
	<b>source</b> <i>source-file</i>	Specifies the storage device, directory, and filename for the file to be moved.
	<b>target</b> <i>target-file</i>	Specifies the new storage device, directory, and filename for the file.
	<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

**Command Default** No default behavior or values

**Command Modes** XR EXEC

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 3.9.0	No modification.
	Release 5.0.0	This command was introduced.

**Usage Guidelines** To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

<b>Task ID</b>	<b>Task ID</b>	<b>Operations</b>
	universal	execute

In the following example, the **utility mv** command is used to move the file “aaa” from disk0a: to disk1a:

```
RP/0/RP0/CPU0:router# utility mv source disk0a:/aaa target disk1a:/aaa
```

**Related Commands**

Command	Description
<a href="#">utility cut, on page 2</a>	Cuts characters or lines from the output displayed from standard input or a file.
<a href="#">utility sort, on page 19</a>	Sorts, merges, or sequence-checks the output displayed from standard input or a file.
<a href="#">utility tail, on page 22</a>	Copies the end portion of the output displayed from standard input or a file.

# utility sort

To sort, merge, or sequence-check the lines in one or more files, or from the standard input, use the **utility sort** command in

XR EXEC

mode.

**utility sort** {[*WORD*] | [**dict**] [**fieldSep** *character*] [**ignoreblank**] [**key** *key-definition*] [**lowercase**] [**merge**] [**numeric**] [**outfile** *filename*] [**printable**] [**reverse**] [**unique**] ]} [**file** *filename*] | **usage**}

Syntax Description	
<i>WORD</i>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
<b>dict</b>	(Optional) ( <b>-d</b> ) Sorts in dictionary order. Uses only alphanumeric and blank characters in the sort operation.
<b>fieldSep</b> <i>character</i>	(Optional) ( <b>-t</b> ) Specifies a character as the field separator.
<b>ignoreblank</b>	(Optional) ( <b>-b</b> ) Ignores leading blank characters in field comparisons.
<b>key</b> <i>key-definition</i>	<p>(Optional) (<b>-k</b>) Defines a key to be the sort key. The <i>key-definition</i> argument field is defined using the following syntax:</p> <p><i>field_start</i> [<i>type_string</i>] [<i>field_end</i>] [<i>type_string</i>]</p> <ul style="list-style-type: none"> <li>• <i>field_start</i> and <i>field_end</i>—Specifies the beginning and end of the key field.</li> <li>• <i>type_string</i>—Specifies attributes specific to the key.</li> </ul> <p>The <i>field_start</i> and <i>field_end</i> arguments are each specified by a pair of digits of the form m.n, where the m refers to the field starting after the mth field separator in a line. For <i>field_start</i>, the .n refers to the nth character of the specified field, and is taken as zero if not specified. For <i>field_end</i>, the .n refers to the nth character after the last character of the specified field, and is taken as zero if not specified.</p> <p>The <i>type_string</i> argument may be formed from the characters bdfnr, which apply their defined attributes to the determination of the key.</p> <p><b>Note</b> When ordering options appear independent of key field specifications, the requested field ordering rules are applied globally to all sort keys. When attached to a specific key, the specified ordering options override all global ordering options for that key.</p>
<b>lowercase</b>	(Optional) ( <b>-f</b> ) Folds uppercase letters into lowercase (ignores case and treats upper case characters the same as lowercase characters).
<b>merge</b>	(Optional) ( <b>-m</b> ) Merges sorted files. Assumes that the files are already sorted and so does not sort the files.
<b>numeric</b>	(Optional) ( <b>-n</b> ) Interprets the field as numeric and sorts in numeric order. Includes the sign and optional thousands separator. This keyword also ignores leading blank characters in field comparisons (implies the <b>ignoreblank</b> keyword).

<b>outfile</b> <i>filename</i>	(Optional) ( <b>-o</b> ) Writes the results to a file. The <i>filename</i> argument is the destination disk, directory, and filename. The <i>filename</i> argument can be the same as the source file.
<b>printable</b>	(Optional) ( <b>-i</b> ) Ignores all nonprintable characters.
<b>reverse</b>	(Optional) ( <b>-r</b> ) Reverses the sort order. The sort is ascending by default.
<b>unique</b>	(Optional) ( <b>-u</b> ) Suppresses all but one line in each set of lines having equal keys.
<b>file</b> <i>filename</i>	(Optional) Specifies a file to be sorted.
<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

**Command Default**

If no file is specified, then the standard input (keyboard) is used.

If an **outfile** *filename* keyword and argument is not specified, then the standard output (display) is used.

The file is sorted in ascending order.

**Command Modes**

XR EXEC

**Command History**

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

**Usage Guidelines**

**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility sort** command is used to sort the contents of the file “words.txt”:

```
RP/0/RP0/CPU0:router# utility sort file disk0:/usr/words.txt
```

```
The
few
inquires
A
Code.
Date
Done
This
best-selling
bestseller
book
come
concerning
fiction,
have
```

```
its  
list  
muscle  
of  
onto  
our  
the  
way  
way  
work
```

In the following example, only the unique characters in the file “words.txt” are displayed:

```
RP/0/RP0/CPU0:router# utility sort unique file disk0:/usr/words.txt
```

```
Code.  
Date  
best-selling  
book  
concerning  
have  
list  
of  
our  
way  
work
```

# utility tail

To copy the end portion of a file or the standard input, use the **utility tail** command in

XR EXEC

mode.

**utility tail** {[*WORD*] | [**bytes**] [**continuous**] [**count** *number*]} [**file** *input-file*] | **usage**}

## Syntax Description

<b>WORD</b>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
<b>bytes</b>	(Optional) ( <b>-c</b> ) Copies the end of the file measured in bytes. The default is lines.
<b>continuous</b>	(Optional) ( <b>-f</b> ) Continues to copy data from the end of the file after the last line is reached. The operation pauses for 1 second, and then resumes in a continuous loop.  The input file must be a regular file, not a terminal or a FIFO special file (a named pipe).
<b>count</b> <i>number</i>	(Optional) ( <b>-n</b> ) Copies the number of lines (default) or bytes specified with the <i>number</i> argument. The range is 0 to 4294967295. By default, the last 10 lines are copied.  The <i>number</i> argument is a decimal integer that defines the location in the file to begin copying: <ul style="list-style-type: none"> <li>• Include the plus (+) character to copy from the beginning of the file.</li> <li>• Include the minus (-) character to copy from the end of the file.</li> <li>• Do not include a character to copy from the end of the file.</li> </ul> <p><b>Note</b> Select the <b>bytes</b> keyword to copy the information measured in a count of bytes.</p>
<b>file</b> <i>input-file</i>	(Optional) Directory path and filename for the input file. If no file is specified, then the standard input is used.  The syntax for the <i>input-file</i> argument is: <i>device</i> :[/ <i>directory-path</i> ]/ <i>filename</i>  The <i>device</i> argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (?) function to display the available storage devices and network protocols.
<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

## Command Default

If the **utility tail** command is entered without keywords or arguments, the last 10 lines of the standard input are copied.

## Command Modes

XR EXEC

## Command History

Release	Modification
Release 3.9.0	No modification.
Release 5.0.0	This command was introduced.

---

**Usage Guidelines**

Use the **utility tail** command to copy data from the end of a file. By default, the last 10 lines are copied. Use the **bytes** keyword to copy the data measured in bytes. Use the **count** *number* option to define the number of lines or bytes to copy. Use the **file** *filename* option to specify an input file.



---

**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

---

In the following example, the **utility tail** command is used to display the last 10 lines of the output from the **show version** command:

In the following example, the **utility tail** command is used with the bytes keyword to display the last 10 bytes in the output:

```
RP/0/RP0/CPU0:router# show version | utility tail count 10 bytes
.95.3-p8
RP/0/RP0/CPU0:router#
```

# utility uniq

To display or remove repeated lines in a file, use the **utility uniq** command in

XR EXEC

mode.

**utility uniq** [{**WORD**] [**afterChars** *number*] [**afterField** *number*] [**count**] [{**nonrepeating** | **repeating**}] [**infile** *input-file* **outfile** *output-file*] [**usage**]

Syntax Description	
<b>WORD</b>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
<b>afterChars</b> <i>number</i>	(Optional) ( <b>-s</b> ) Ignores the first characters on each line of the input file. Use the <i>number</i> argument to specify the number of characters. The range is 0 to 4294967295.
<b>afterField</b> <i>number</i>	(Optional) ( <b>-f</b> ) Ignores the first fields on each line of the input file. Use the <i>number</i> argument to specify the number of fields. The range is 0 to 4294967295.
<b>count</b>	(Optional) ( <b>-c</b> ) Displays the number of times the line appeared in the input file at the beginning of each output line.
<b>nonrepeating</b>	(Optional) ( <b>-u</b> ) Displays only the nonrepeating lines from the input file (repeating lines are not displayed).
<b>repeating</b>	(Optional) ( <b>-d</b> ) Displays only the repeating lines from the input file (nonrepeating lines are not displayed).
<b>infile</b> <i>input-file</i>	<p>(Optional) Specifies an input file for processing. The <i>input-file</i> argument specifies the device, directory, and filename of the input file. If no input file is specified, then the standard input (keyboard) is used.</p> <p>The syntax of the <i>input-file</i> argument is: <i>device</i> :[/ <i>directory-path</i>]/ <i>filename</i>.</p> <p>The <i>device</i> argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (?) function to display the available storage devices and network protocols.</p>
<b>outfile</b> <i>output-file</i>	<p>(Optional) Specifies an output file. The <i>output-file</i> argument specifies the device, directory, and filename of the output file. If no file is specified, then the standard output (display) is used.</p> <p>The syntax of the <i>output-file</i> argument is: <i>device</i> :[/ <i>directory-path</i>]/ <i>filename</i>.</p> <p>The <i>device</i> argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (?) function to display the available storage devices and network protocols.</p>
<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

## Command Default

If no input file is specified, then the standard input is used.

If no output file is specified, then the standard output is used.



**Command Modes** XR EXEC

Command History	Release	Modification
	Release 3.9.0	No modification.
	Release 5.0.0	This command was introduced.

**Usage Guidelines** Use the **utility uniq** command to display only lines that are repeated in a file, or to display only lines that appear once. This utility compares only adjacent lines, so the file or standard input must be sorted.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

In the following example, the **utility uniq** command is used to display the repeating lines in the output of the **show environment** command:

```
RP/0/RP0/CPU0:router# show environment | utility uniq repeating
host      5V      4500,5500      4250,5750      4000,6000
fabricq 1.25V    1125,1375     1063,1438     1000,1500
fabricq 1.25V    1125,1375     1063,1438     1000,1500
ingress 1.25V    1125,1375     1063,1438     1000,1500
spa5      1.5V      1500,0        1575,1425      0,0
host      5V      4500,5500      4250,5750      4000,6000
fabricq 1.25V    1125,1375     1063,1438     1000,1500
fabricq 1.25V    1125,1375     1063,1438     1000,1500
ingress 1.25V    1125,1375     1063,1438     1000,1500
spa5      1.5V      1500,0        1575,1425      0,0
```

# utility wc

To count words, lines, or bytes in a file, use the **utility wc** command in

XR EXEC

mode.

**utility wc** [{[{*WORD*] | [**bytes**] [**lines**] [**words**]}] [**file** *input-file*] | **usage**}]

<b>Syntax Description</b>	<b>WORD</b>	(Optional) UNIX command-line option string. The maximum number of characters is 80.
	<b>bytes</b>	(Optional) ( <b>-c</b> ) Displays the number of bytes in each input file.
	<b>lines</b>	(Optional) ( <b>-l</b> ) ( <b>-e</b>  -?) Displays the number of lines in each input file.
	<b>words</b>	(Optional) ( <b>-w</b> ) Displays the number of words in each input file.
	<b>file</b> <i>input-file</i>	(Optional) Specifies the input file. The <i>input-file</i> argument specifies the device, directory, and filename of the input file. If no input file is specified, then the standard input (keyboard) is used.  The syntax of the <i>input-file</i> argument is: <i>device</i> :[ / <i>directory-path</i> ] / <i>filename</i> .  The <i>device</i> argument, followed by a colon, indicates the name of the device where the file is located. Use the online help (?) function to display the available storage devices and network protocols.
	<b>usage</b>	(Optional) Displays the UNIX options supported by this command.

**Command Default** Output is displayed in the order bytes, words, and lines, even if the options are entered in a different order.

**Command Modes** XR EXEC

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	Release 5.0.0	This command was introduced.
	Release 3.9.0	No modification.

**Usage Guidelines** To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Output is displayed in the following order:

- When keywords are entered, the output appears in the order bytes, words, and lines.
- When no keyword is entered, the output appears in the order lines, words, and bytes.
- When any UNIX equivalent options are entered, the output appears in the order specified by the options. For example, if the command **utility wc -w -l -c** is entered, the output appears in the order words, lines, and bytes.



**Note** Keywords are entered using the displayed syntax, or with UNIX-equivalent syntax. The UNIX-equivalent syntax is displayed in parentheses () in the syntax description. To display the UNIX-equivalent syntax online, enter the **usage** keyword.

Task ID	Task ID	Operations
	universal	execute

In the following example, the **utility wc** command is issued to display the number of lines, words, and bytes in the output of the **show version** command:

```
RP/0/RP0/CPU0:router# show version | utility wc
221      1160      10820
```

The output displays the following:

- 221 lines
- 1160 words
- 10820 bytes

In the following example, the **utility wc** command is entered with the **words** keyword to display the number of words in the output of the **show version** command:

```
RP/0/RP0/CPU0:router# show version | utility wc words
1160
```

