Sun microsystems

Sun Ray™ Server Software 3.1 Reference Manual

for the Solaris ™ Operating System

Adobe PostScript

Contents

Preface

Both novice users and those familiar with the Sun Ray software can use online man
pages to obtain information about the system and its features A man page is intended
to answer concisely the question "What does it do?" In general, man pages comprise a
reference manual They are not intended to be a tutorial

Format

The following is a generic format for man pages The man pages of command or file
generally follow this order, but include only needed headings For example, if there
are no bugs to report, there is no BUGS section See the man(1) command for more
information about man pages in general

NAME	This section gives the names of the commands or functions documented, followed by a brief description of what they do
SYNOPSIS	This section shows the syntax of commands or functions When a command or file does not exist in the standard path, its full path name is shown Options and arguments are alphabetized, with single letter arguments first, and options with arguments next, unless a different argument order is required
	The following special characters are used in this section:

[ ]	Brackets The option or argument enclosed in these brackets is optional If the brackets are omitted, the argument must be specified.
…	Ellipses Several values may be provided for the previous argument, or the previous argument can be specified multiple times, for example "filename...".
|	Separator Only one of the arguments separated by this character can be specified at one time.
{ }	Braces The options and/or arguments enclosed within braces are interdependent, such that everything enclosed must be treated as a unit.
PROTOCOL	This section occurs only in subsection 3R to indicate the protocol description file.
DESCRIPTION	This section defines the functionality and behavior of the service Thus it describes concisely what the command does It does not discuss OPTIONS or cite EXAMPLES Interactive commands, subcommands, requests, macros, functions and such, are described under USAGE.
IOCTL	This section appears on pages in Section 7 only Only the device class which supplies appropriate parameters to the ioctl(2) system call is called ioctl and generates its own heading ioctl calls for a specific device are listed alphabetically (on the man page for that specific device) ioctl calls are used for a particular class of devices all of which have an io ending, such as mtio(7I)
OPTIONS	This lists the command options with a concise summary of what each option does The options are listed literally and in the order they appear in the SYNOPSIS section Possible arguments to options are discussed under the option, and where appropriate, default values are supplied
OPERANDS	This section lists the command operands and describes how they affect the actions of the command.

OUTPUT	This section describes the output – standard output, standard error, or output files – generated by the command
RETURN VALUES	If the man page documents functions that return values, this section lists these values and describes the conditions under which they are returned If a function can return only constant values, such as 0 or –1, these values are listed in tagged paragraphs Otherwise, a single paragraph describes the return values of each function Functions declared void do not return values, so they are not discussed in RETURN VALUES
ERRORS	On failure, most functions place an error code in the global variable errno indicating why they failed This section lists alphabetically all error codes a function can generate and describes the conditions that cause each error When more than one condition can cause the same error, each condition is described in a separate paragraph under the error code.
USAGE	This section lists special rules, features and commands that require in-depth explanations The subsections listed below are used to explain built-in functionality: Commands Modifiers Variables Expressions Input Grammar
EXAMPLES	This section provides examples of usage or of how to use a command or function Wherever possible a complete example including command line entry and machine response is shown Whenever an example is given, the prompt is shown as example% or if the user must be superuser, example# Examples are followed by explanations, variable substitution rules, or returned values Most examples illustrate concepts from the SYNOPSIS, DESCRIPTION, OPTIONS and USAGE sections

ENVIRONMENT VARIABLES	This section lists any environment variables that the command or function affects, followed by a brief description of the effect
EXIT STATUS	This section lists the values the command returns to the calling program or shell and the conditions that cause these values to be returned Usually, zero is returned for successful completion and values other than zero for various error conditions
FILES	This section lists all filenames referred to by the man page, files of interest, and files created or required by commands Each is followed by a descriptive summary or explanation
ATTRIBUTES	This section lists characteristics of commands, utilities, and device drivers by defining the attribute type and its corresponding value See `attributes`(5) for more information.
SEE ALSO	This section lists references to other man pages, in-house documentation and outside publications
DIAGNOSTICS	This section lists diagnostic messages with a brief explanation of the condition causing the error
WARNINGS	This section lists warnings about special conditions which could seriously affect your working conditions This is not a list of diagnostics.
NOTES	This section lists additional information that does not belong anywhere else on the page It takes the form of an aside to the user, covering points of special interest Critical information is never covered here
BUGS	This section describes known bugs and wherever possible, suggests workarounds

NAME

  utaction - Sun Ray DTU connect/disconnect action

SYNOPSIS

  /opt/SUNWut/bin/utaction [-**c** *ccmd*] [-**d** *dcmd*] [-**e**] [-**D** *display*] [-**i**] [-**t** *sec*]

DESCRIPTION

  The **utaction** program provides a way to execute commands when a Sun Ray DTU session is connected or
  disconnected. The *ccmd* is invoked using **sh**(1) whenever the session is connected to a DTU. Similarly, the
  dcmd is invoked using **sh**(1) whenever the session is disconnected from a DTU. Normally, action is not taken
  on the initial state of the session (when **utaction** is first run) unless the -i option is used.

  Note: In earlier releases, this command resided in **/opt/SUNWut/lib/utaction** ; now, however, it resides in
  /opt/SUNWut/bin/utaction .

OPTIONS

  The following options are supported.

  -**c** *ccmd*
     Run this command when the current session is connected to a DTU.

  -**d** *dcmd*
     Run this command when the current session is disconnected to a DTU.

  -**D** *display*
     This option will set the X display variable that is to be used in determining the Sun Ray DTU session.
     Otherwise the DISPLAY environment variable is used.

  -**e** This option causes utaction to exit after encountering a command.

  -**i** Run the connect or disconnect command immediately, whichever is appropriate.

  -**t** *sec*
     This option specifies a time-delay in seconds for the actions. In that case, the *ccmd* or *dcmd* will not be
     invoked unless the session remains in the connected or disconnected state, respectively, for at least *sec*
     seconds.

EXAMPLES

  Example 1: This command invokes the CDE screen lock whenever the session is disconnected

  % **utaction -d '/usr/dt/bin/dtaction LockDisplay' &**

ATTRIBUTES

  See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

NOTES

  The *ccmd* and *dcmd* are each only one argument to **utaction**. Quotes should be used if a command contains
  multiple words.

NAME

 utaudio - Sun Ray audio services connection utility.

SYNOPSIS

 /opt/SUNWut/bin/utaudio

 csh

 setenv AUDIODEV 'utaudio'

 ksh

 export AUDIODEV='utaudio'

 sh

 AUDIODEV='utaudio';export AUDIODEV

DESCRIPTION

 utaudio enables standard Solaris audio services using the **utadem**(7D) audio device emulator driver. After connecting to a Sun Ray session, **utadem**(7D) creates a new audio device for which **utaudio** creates device files in the **/tmp/SUNWut/dev** directory. **utaudio** then echoes the root device name to standard output, setting the AUDIODEV environment variable. Standard audio applications can then open the new audio pseudo-device and perform audio play and record operations.

OPTIONS

 There are no options for **utaudio**.

API

 Applications that use the **/dev/audio** interface may open the device indicated by the AUDIODEV environment variable and use the AUDIO_GETDEV **ioctl** to determine which audio device is being used. The **utaudio** driver returns the string "SUNW,CS4231" in the name field of the **audio_device** structure to indicate compatibility with other Ultra platforms. The version field contains "a" and the config field contains "pseudo."

 The AUDIO_SETINFO **ioctl** controls device configuration parameters. When an application modifies the **record.buffer_size** field using the AUDIO_SETINFO **ioctl**, the daemon will constrain it to be non-zero and up to a maximum of 8180 bytes.

 Audio Data Formats
 The **utaudio** daemon supports u-law and A-law with 8-bit precision or 16-bit linear PCM at any sample rate from 8000 Hz to 48 kHz for one or two channels. The Sun Ray standard sampling rate is 48 kHz as this yields the best quality. The input and output data formats for playing and recording do not have to match. Some input devices do not provide 2-channel capture, but two channels will be reproduced by duplication in the case where two channels are requested and the device supports only one.

 Audio Ports
 The **record.avail_ports** and **play.avail_ports** fields of the **audio_info** structure report the available input and output ports for the currently connected Sun Ray DTU. Only AUDIO_MICROPHONE and AUDIO_LINE_IN are supported and most devices will have both inputs. The Sun Ray audio model supports individual volume controls for the two, so it is possible that the volume setting will change with input.

 For output, AUDIO_LINE_OUT is always selected and does not have variable gain. AUDIO_SPEAKER and AUDIO_HEADPHONE are supported and they share a level control. In general, comfortable settings for the speaker will also be comfortable for headphone use. Either or both outputs can be selected simultaneously. The Sun Ray specification supports a third, automatic switching mode that is accessed by deselecting both speaker and headphone or by selecting only line out. The **utsettings**(1) command may also be used to control the device's outputs. In automatic mode, the settings track the physical connection of the headphone.

FILES

 The following files are used:

 • **/tmp/SUNWut/dev/utaudio/***n*
 Numbered audio data pseudo-device file nodes.

 • **/tmp/SUNWut/dev/utaudio/***n*ctl
 Matching numbered control pseudo-device file nodes.

ENVIRONMENT VARIABLES

    utaudio requires the DISPLAY environment variable contain an **X11**(7) display for which the user's session has access. This is set-up automatically in the Sun Ray environment.

    An alternate driver emulator or different unit number can be specified in the UT_ADEM environment variable.

    The results of **utaudio** should be placed in the AUDIODEV environment variable.

EXIT STATUS

    The following exit values are returned:

    0 Normal completion -- daemon back grounded

    1 Either the X11 server, or the session could not be contacted, or there was a problem creating the new pseudo audio device.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

    utsettings(1), **X11(7)**, **utadem(7D)**, **audio(7I)**, **steamio(7I)**, **ioctl(2)**, **priocntl(2)**, **attributes(5)**, **environ(5)**

NOTES

    The **audio**(7I) interface does not have an interface for dynamically changing audio devices such as that offered by the Sun Ray software. It is not possible to track the movement of sessions or changes in audio hardware using this device interface. The **utaudio** daemon makes a best-effort attempt to report changes in device control ability and to make the device appear as flexible as possible, matching that ability to the actual Sun Ray hardware being used.

    If a session is disconnected, audio output continues as if there was an actual hardware connection, even though no samples are actually being transmitted or played. Conversely, audio input stops when there is no connected device.

NAME

    utdetach - Detach the current session from the Sun Ray DTU.

SYNOPSIS

    /opt/SUNWut/bin/utdetach

DESCRIPTION

    The **utdetach** command disconnects the current session from its respective Sun Ray DTU. The session is not destroyed, but rather put into a disconnected state. The session can be accessed if the same user token is presented to the Sun Ray server.

    This command is primarily executed by users of the non-smart card mobility feature so as to disconnect their "mobile" sessions.

    The Sun Ray server starts an instance of **utslaunch** (1M) for each session whenever a user logs into a Sun Ray DTU via **dtlogin**. This makes the **utdetach** command available to users as a hotkey sequence. The default hotkey sequence is Shift + Pause and can be configured in the **utslaunch.properties** file.

OPTIONS

    There are no options for **utdetach**.

EXAMPLES

    Example 1: This command disconnects the current session from the DTU the user is currently using.

    % utdetach

FILES

    The following files are used:

    • **/etc/opt/SUNWut/utslaunch_defaults.properties**
      site-wide defaults

    • **~/.utslaunch.properties**
      user's defaults

    • **/etc/opt/SUNWut/utslaunch_mandatory.properties**
      site-wide mandatory defaults

EXIT STATUS

    The following exit values are returned:

    0 Success

    1 Failure

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto
Stability Level	Evolving

SEE ALSO

    utslaunch(1M), **utslaunch.properties(4)**

NAME

    uteject - Sun Ray Mass Storage media eject utility

SYNOPSIS

    /opt/SUNWut/bin/uteject *device_name*

DESCRIPTION

    The **uteject** command has the same functionality as utdiskadm -e. It ejects media from removable media devices associated with the user's current Sun Ray session. If a filesystem is mounted on the device, the eject operation attempts to unmount it first.

OPTIONS

    The following options are supported.

        device_name

        Eject media from device_name

EXIT STATUS

    The following exit codes are returned:

        0 The operation was successful

        1 The operation was unsuccessful

FILES

    The following files are used:

        $UTDEVROOT/dev/dsk

        The directory containing links to block device names for each partition on the device.

        $UTDEVROOT/dev/rdsk

        The directory containing links to raw device names for each partition on the device.

ENVIRONMENT VARIABLES

    UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutsto

SEE ALSO

    utdiskadm(1M), **utmount(1M)**, **utumount(1M)**, **utmountd(1M)**, **utstoraged(1M)**, **utdisk(7D)**

NAME

      utmount - Sun Ray Mass Storage media mount utility

SYNOPSIS

      /opt/SUNWut/bin/utmount -m *partition_name*[-p *mount_path*]

DESCRIPTION

      The **utmount** command has the same functionality as utdiskadm -m. It is used to mount partition_name on either the default mount point in $DTDEVROOT/mnt, or on a user supplied mount point mount_path.

OPTIONS

      The following options are supported.

         -m partition name

      Mount partition partition_name on default mount point in $DTDEVROOT/mnt.

      -m partition_name -p mount_path

      Mount partition partition_name on directory mount_path

EXIT STATUS

      The following exit codes are returned:

      0 The operation was successful

      1 The operation was   unsuccessful

FILES

      The following files are used:

      $UTDEVROOT/dev/dsk
        The directory containing links to block device names for each partition on the device.

      $UTDEVROOT/dev/rdsk
        The directory containing links to raw device names for each partition on the device.

ENVIRONMENT VARIABLES

      UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

      DTDEVROOT points to a temporary directory associated with the user's session. The directory's lifetime is equal to the lifetime of the login session. It is removed, along with its contents, when the user logs out.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutsto

SEE ALSO

      utdiskadm(1M), **uteject(1M)**, **utumount(1M)**, **utmountd(1M)**, **utstoraged(1M)**, **utdisk(7D)**

NAME

     utselect - Sun Ray failover group server selection tool.

SYNOPSIS

     /opt/SUNWut/bin/utselect [-**L**] [-**R**] [-**S**] [-**X**]

DESCRIPTION

     The **utselect** command is a graphical user interface (GUI) to the **utswitch** command. It allows the user to select manually to which Sun Ray server or session the Sun Ray DTU is to connect. The sessions in the GUI are sorted in order of most current. The second item in the list is highlighted by default to allow easy switching between two servers. The Refresh button executes the **utswitch -l** command and updates the information displayed in the GUI. The OK button executes a **utswitch -h** command to the server highlighted.

OPTIONS

     The following options are supported.

     -**L** Configures **utselect** to run in "login" mode before the log in screen is displayed. Where:

        • If only one server is available, the command exits.

        • The current server is set as the default.

        • Selecting the current server causes the command to exit

        • The locale is determined in a manner similar to CDE

        • The screen is centered in the display

     -**R** Remote server selection is enabled. This enables an entry field where a networked server name can be entered.

     -**S** Remote server selection is disabled.

     -**X** Exit after making a selection from the list.

EXAMPLES

     Example 1: This command enables users to select which Sun Ray server or session to connect. The GUI exits after selection

       % **/opt/SUNWut/bin/utselect -X**

FILES

     Two properties in the **auth.props(4)**file impact the operation of utselect:

       selectAtLogin=true
          This setting will cause utselect to run before dtlogin and allow users to start their session on a particular machine. The default value is "false"

       remoteSelect=true
          This setting will cause utselect to behave as if the -R option were specified. This will allow users to input the name of a server outside of the default HA group, if the Sun Ray DTU can connect to it (i.e. the Sun Ray can be routed to the server, such as in a LAN deployment). The default value is false.

ATTRIBUTES

     See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

     SEE ALSO

     utswitch(1), **attributes(5)**, **auth.props(4)**

NAME

   utset - View or change the Sun Ray DTU settings.

SYNOPSIS

   /opt/SUNWut/bin/utset [-**d** [-**f**] *arg[,...]*] [-**i** *arg[,...]*] [-**o** *arg[,...]*] [-**v** *arg[,...]*]

   /opt/SUNWut/bin/utset -**l**

DESCRIPTION

   The **utset** command reports or modifies Sun Ray device settings. **utset** executed with no options prints the
   current status of all device settings to the standard output stream. **utset** executed with the -**d**, -**i**, -**o** or -**v**
   options modifies the values of the specified settings. **utset** executed with the -**l** option prints a list of
   predefined display timing names to the standard output stream.

   Sun Ray device settings are grouped into four categories: Display, Audio Input, Audio Output, and Video
   Input. The setting to be modified is selected by an option letter, which identifies the category, and an
   argument consisting of a list of individual parameter settings within that category and the values that are
   desired for each of those parameters. The argument is a comma separated list of '*name=value*' pairs, where
   name specifies a device parameter and *value* specifies the desired value of that parameter.

   Parameters reported as **read only** by **utset -l** may not be modified. Read-only status usually reflects some
   limitation of a particular model of DTU. For instance, the Sun Ray 150 supports only one display timing and
   therefore reports its display timing setting as a read-only parameter.

   The **utsettings**(1) command provides equivalent device parameter reporting and modification capabilities in
   a GUI format.

   The **xset**(1) command provides control over device parameters such as mouse acceleration which are reported
   by **utset** but may not be modified by **utset**.

OPTIONS

   The following options are supported.

   -**d** [-**f**] *arg[,...]*
        Set the Display timing:

   -**f**   Force the specified Resolution/Refresh Rate setting without prompting for confirmation from the user.

   r=*<timing>*
        Display timing: the name of a timing. The -**l** option gives the list of available timings. Timing names are
        usually of the form *WxH@R* where *W* is the resulting on-screen width (in pixels), *H* is the resulting on-
        screen height (in pixels) and *R* is the resulting refresh rate (in Hz). **utresdef**(1m) provides additional
        information on monitor timings.

        If -**f** is not given then after the new timing is put into effect the user is asked to confirm that the new
        Resolution/Refresh Rate is acceptable. If no confirmation is provided within a certain interval (currently
        15 seconds) then the timing is returned to its previous setting.

   -**i** *arg[,...]*
        Set the Audio Input settings:

   s=[m|l]
        Input Select: microphone, line.

   g=<0:75>
        Microphone Gain: a range of 0 to 75.

   l=<0:15>
        Line in Gain Left: a range of 0 to 15.

   r=<0:15>
        Line in Gain Right: a range of 0 to 15.

   v=<0:64>
        Volume: a range of 0 to 64.

   -**l**   Show the list of predefined display timing names.

   -**o** *arg[,...]*
        Set the Audio Output settings:

   s=[a|s|h]
        Output Select: auto, speaker, headphone.

       v=<0:31>
          Volume: a range of 0 to 31.

       b=<-32:32>
          Balance: a range of -32 to +32.

       m=[on|off]
          Mute: on, off (also: 1, 0 respectively).

       e=[on|off]
          Stereo Enhance: on, off (also: 1, 0 respectively).

       T=<-6:6>
          Treble: a range of -6 to +6.

       B=<-6:6>
          Bass: a range of -6 to +6.

       -v *arg[,...]*
          Set the Video Input settings:

       b=<0:255>
          Brightness: a range of 0 to 255.

       c=<0:63>
          Contrast: a range of 0 to 63.

       C=<0:127>
          Color: a range of 0 to 127.

       t=<0:255>
          Tint: a range of 0 to 255.

       f=<0:3>
          Filter: a range of 0 to 3.

       T=[on|off]
          Color Trap: on, off (also: 1, 0 respectively).

EXAMPLES

       Example 1: This command displays the current device settings for the Sun Ray DTU to which your session is attached:

       % **utset**
       Version: SunRayP3-3.1_10,REV=2005.03.30.15.44 (read only)
       Audio Input Mic Gain: 58
       Audio Input Line In Gain L: 0
       Audio Input Line In Gain R: 0
       Audio Input Select: Microphone
       Audio Input Monitor Volume: 0
       Audio Output Volume: 15
       Audio Output Balance: 0
       Audio Output Treble: 0
       Audio Output Bass: 0
       Audio Output Select: Auto
       Audio Output Mute: off
       Audio Output Stereo Enhance: off
       Mouse Threshold: 4
       Mouse Acceleration: 2.0
       Display Resolution / Refresh Rate: 1024x768@60 (read only)
       Display Blanking: off
       Video Brightness: 128
       Video Contrast: 7
       Video Color: 46
       Video Tint: 128
       Video Filter: 3

Video Color Trap: on

This session happens to be attached to a Sun Ray 150, whose display timing is fixed and is therefore reported as a read-only parameter.

EXIT STATUS

The following exit values are returned:

0 Success

1 Failure

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

utresdef(1m), **utsettings**(1), **xset**(1)

NAME

        utsettings - View or change the Sun Ray DTU settings.

SYNOPSIS

        /opt/SUNWut/bin/utsettings [**-H** [**-k** *hotkey*]]

DESCRIPTION

        The **utsettings** command opens a Sun Ray Settings dialog box that allows the user to view or change audio, visual, and tactile interface settings for the Sun Ray DTU.

        The **utsettings** application connects to the Session Manager, which tells the application which DTU it is being displayed upon. As the user moves a session from one Sun Ray DTU to another, the Session Manager keeps track of the session's current location and instructs the **utsettings** application to follow. With each session move, the **utsettings** application displays the current DTU's configuration.

        By default, the Sun Ray server starts an instance of **utslaunch(1M)** for each session created by the user logging in via **dtlogin**. This enables the Sun Ray Settings dialog box to be available at the press of a hotkey or key combination. Subsequent presses toggle the dialog box on and off.

        Users can initiate similar functionality by running **utsettings** with the **-H** flag. The hotkey can be specified using the **-k** option. Only one instance of **utsettings -H** or **utslaunch** can be running per session.

        Settings selected through **utsettings** apply only to the DTU where **utsettings** is run; hotdesking to another DTU does not bring the new timing along as part of the session. However, the selected timing is retained and used again if a user hot desks back to the original DTU.

        If the session is associated with a personal mobile token (a smart card or an NSCM credential), then **utsettings** offers to make the selected timing permanent. If a user accepts that offer, then the timing is retained and reused on that user's subsequent personal mobile token sessions on the same DTU. For shared session token types, such as *pseudo*, users are not allowed to establish long-term resolution settings because their settings would interfere with other people's use of the DTU.

OPTIONS

        The following options are supported.

        -**H** Start the **utsettings** application in *hotkey* mode. The **utsettings** application starts with the Sun Ray Settings dialog box hidden. Pressing the hotkey toggles the display of the dialog box. The default hotkey key combination is Shift + Props. The hotkey can be user or site defined according to the **utsettings.hotkey** property in the files listed below. See FILES

        -**k** *hotkey*
            Use the specified key or keys as the hotkey combinations when the **-H** option is specified. This option is dependent upon the **-H** option.

EXAMPLES

        Example 1: This command displays the settings for the Sun Ray DTU you are currently logged into.

        % **utsettings**

FILES

        The following files are used:

          • **/etc/opt/SUNWut/utslaunch_defaults.properties**
           site-wide defaults

          • **~/.utslauch.properties**
           user's defaults

          • **/etc/opt/SUNWut/utslaunch_mandatory.properties**
           site-wide mandatory defaults

EXIT STATUS

        The following exit values are returned:

        0 Success

        1 Failure

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

utslaunch(1M), **dtlogin(1X)**, **dtsession(1X)**, **utslaunch.properties(4)**

NAME

      utswitch - Sun Ray server selection and session listing utility.

SYNOPSIS

      /opt/SUNWut/bin/utswitch { -**l** | -**t** | -**h** *hostname* } [-**k** *token*] [-**p** *port*] [-**r**]

DESCRIPTION

      The **utswitch** command allows switching a Sun Ray DTU among Sun Ray servers in a failover group. It can also list the existing sessions for the current token. One of the following option flags must be specified: **-l**, **-t**, or -**h**. The **utselect**(1) command implements a GUI-based interface to this command.

OPTIONS

      The following options are supported.

      -**h** *hostname*
         Force an explicit switch to the server with *hostname*.

      -**k** *token*
         Specify the token ID *token* to be used in collecting session information from the servers in the failover group. The token normally used is the one connected to the current session.

      -**l** List the servers accessible from the current Sun Ray DTU for the current token and show any existing sessions on those servers.

        • The first field of the output is the server name.

        • The second field is the X display number for an active user session. If no active user session exists, then -1 is printed or -2 is printed if the login screen is being displayed.

        • The third field is the last connection time to an existing session, as a time value from the **time(2)** system call. If there is no session, the third field indicates status from the host as:
          -1 Server is up, but there is no session.
          -2 No response received from the server.
          -3 No path from the Sun Ray to the server.

        • The fourth field is 1 if the server is offline and 0 otherwise.

      -**p** *port*
         Sets the port number of the Authentication Manager on the Sun Ray server to *port*, instead of the default 7009.

      -**r** Forces a remote redirection outside of the current failover group to search for an existing session within an external failover group. If no session is available, load balancing is performed. Without this option, the Sun Ray DTU is bound explicitly to the target Sun Ray server, rather than to an appropriate server within the target failover group. This option may only be used with the **-h** option

      -**t** Switch to the server whose session has the latest connection time among the existing sessions for the current token. Normally this would switch to the current session, so it has limited usefulness. However, it is useful in the case of logging out of an existing X session and back to the login screen. The connection time of a logged out session is biased back in time so that the session will not be selected if there is an existing logged-in X session on another server. From a CDE login screen, it is possible to force a call to **utswitch -t** by selecting Reset Login Screen from the Options menu. This allows switching back to a logged-in session from the login screen without having to log in.

FILES

      The following files are used:

        • **/var/opt/SUNWut/displays/***
         X display files

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      utselect(1), **attributes(5)**

NAME

    utxlock - Sun Ray utility for locking a window session.

SYNOPSIS

    /opt/SUNWut/bin/utxlock

DESCRIPTION

    The **utxlock** utility locks the current display in a manner specific to the current windowing environment. If the current environment is Gnome, it uses **xscreensaver-command**; if the current environment is CDE, it uses **dtsession**; otherwise **utxlock** is used.

    Note:

    Although some users may find screen locking an inconvenience, overriding it has security implications that should be obvious. Override at your own risk.

    A user may disable any screen lock behavior by setting the environment variable **SUN_SUNRAY_UTXLOCK_PREF** to **NULL**. Any other value will be used as a command line to use for invoking a screen lock command instead of the default behavior.

    SRSS invokes **utxlock** on any session disconnect. To disable this behavior, add the following line to your **$HOME/.dtprofile**:

    SUN_SUNRAY_UTXLOCK_PREF=; export SUN_SUNRAY_UTXLOCK_PREF

    As another example, if a user had their own screenlock program called mylock, and they wanted to pass it the argument **-l**, they should add the following line to their **$HOME/.dtprofile** :

    SUN_SUNRAY_UTXLOCK_PREF="$HOME/bin/mylock   -l"; export SUN_SUNRAY_UTXLOCK_PREF

OPTIONS

    No options are supported.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutu

NAME

   utadm - Sun Ray network and DHCP configuration utility.

SYNOPSIS

   /opt/SUNWut/sbin/utadm -**a** *interface-name*

   /opt/SUNWut/sbin/utadm -**c**

   /opt/SUNWut/sbin/utadm -**d** *interface-name*

   /opt/SUNWut/sbin/utadm -**f**

   /opt/SUNWut/sbin/utadm -**l**

   /opt/SUNWut/sbin/utadm -**n**

   /opt/SUNWut/sbin/utadm -**p**

   /opt/SUNWut/sbin/utadm -**r**

   /opt/SUNWut/sbin/utadm -**x**

   /opt/SUNWut/sbin/utadm -**A** *subnetwork*

   /opt/SUNWut/sbin/utadm -**D** *subnetwork*

   /opt/SUNWut/sbin/utadm -**L on** | **off**

DESCRIPTION

   The **utadm** command manages the private network and DHCP configuration for the Sun Ray interconnect.
   It configures the name lookup, host, network, netmask, and DHCP database files so that Sun Ray DTUs can
   be connected to a central server host over one or more private subnets.   One of the following option flags
   must be specified: -**a**, -**c**, -**d**, -**f**, -**l**, -**n**, -**p**, -**r**, -**x**, -**A**, or -**D**. The command is run with superuser privileges.

OPTIONS

   The following options are supported.

   -**a**   Configure the network interface specified by *interface-name* as an Sun Ray subnetwork. In the default
          case, an available private subnetwork address is selected from the range 192.168.128.0 to 192.168.254.0.
          If the subnet selected is 192.168.*N*.0, entries for the hosts, networks, and netmasks files are generated
          using the *hostname* of the server and *interface-name*:

```
File              Entry
/etc/hosts        192.168.N.1                 hostname-interface-name
/etc/networks     SunRay-interface-name       192.168.N.0   SunRay
/etc/netmasks     192.168.N.0                 255.255.255.0
```

   Once the appropriate entries are established, the network interface is activated as hostname-interface-name
   using **ifconfig(1M)**. If the interface is already up and configured, the user will be given the option to bypass
   configuration of the network interface and only configure DHCP on the interface. This allows configuration
   of a Sun Ray interconnect on the primary interface of the server. IP addresses on the Sun Ray subnets are
   managed using the DHCP protocol, which requires the addition of several macro entries to the DHCP table
   to control parameters on Sun Ray subnets.A pool of IP addresses for assignment to Sun Ray DTUs is created.
   It is possible to disable offering IP addresses by entering 0 as the first unit address or entering N to the query
   to offer IP addresses for the interface when prompted. Once the interface is configured and activated,
   utfwadm(1M) is invoked to add the current version of the firmware to the DHCP macros for the new
   network. The user is prompted for approval of all the default options, and may change them as desired. The
   auth server list is the list of IP addresses of Sun Ray servers. The value input overrides the default value
   prompted for. The auth server list can also be provided in a file as a list of IP addresses separated by spaces.

   The -**a** option implies the -**c** option if the initial configuration has not yet been performed.

   -**c**   Initialize the basic configuration files for a Sun Ray interconnect without setting up any subnetworks. This
          involves making sure that the network database files and framework for DHCP exist, and setting the **/etc/
          nsswitch.conf** file so that network information for the local Sun Ray subnets is obtained from local files.

   -**d**   Delete the network interface specified by *interface-name* from the list of configured Sun Ray subnetworks.
          The specified interface must have been previously configured using the -**a** option.

   -**f**   Take this server offline, preventing the creation of new sessions on this server when it is within a failover
          group. Existing sessions are not killed, but load balancing does not select this server for new sessions.

Note: The GroupManager updates the host interface list (if there have been any changes) after every keep alive interval. So in order for the changes to be effective, the user has to wait at least the keepalive time interval.

-**l** Print the current configuration for all the Sun Ray subnetworks. This includes remote subnetworks.

-**n** Bring this server back online. This restores normal operation of the server and allows new sessions to be created on this server. Note: The GroupManager updates the host interface list (if there have been any changes) after every keep alive interval. So in order for the changes to be effective, the user has to wait at least the keepalive time interval.

-**p** Print the current Sun Ray interconnect configuration, showing for each interface the hostname, network, netmask, and number of IP addresses out of an available pool of addresses assigned to Sun Ray DTUs by DHCP.

-**r**   Unconfigure all active Sun Ray interfaces and remove all Sun Ray entries from the configuration databases. If LAN connection support is on, will prompt the user if it should be turned off. Default is yes (turn it off).

-**x** Print the current configuration in a machine-readable format

-**A** Configure the *subnetwork* specified as a Sun Ray subnetwork. This option only configures the DHCP service to allocate IP address and/or to provide Sun Ray parameters to Sun ray clients. It also will automatically turn on support for LAN connections from a shared subnetwork.

-**D** Delete the *subnetwork* specified form the list of configured Sun Ray subnetworks.

-**L on**| **off**
   Turn support for LAN connections from a shared subnetwork on or off.

EXAMPLES

Example 1: The following example configures the Sun Ray private network on hme1

 # /opt/SUNWut/sbin/utadm -a hme1

FILES

The following files are used:

- **/etc/nsswitch.conf**
  Name service switch configuration file.

 /var/dhcp/dhcptab

Solaris files or NIS+ table

- **/etc/inet/hosts**

- **/etc/inet/networks**

- **/etc/inet/netmasks**

- **/etc/hostname.***

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

dhtadm(1M), **pntadm(1M)**, **dhcpconfig(1M)**, **dhcp(4)**, **dhcp_network(4)**, **dhcptab(4)**, **networks(4)**, **netmasks(4)**, **ifconfig(1M)**, **syslogd(1M)**, **syslog(3)**, **nsswitch.conf(4)**, **hosts(4)**, **syslog.conf(4)**, **attributes(5)**, **utfwadm(1M)**

Alexander, S., and Droms, R., DHCP Options and BOOTP Vendor Extensions, RFC 1533, Lachman Technology, Inc., Bucknell University, October 1993.

Droms, R., Dynamic Host Configuration Protocol, RFC 1541, Bucknell University, October 1993.

NAME

    utadminuser - Command used to manage authorized user list to administer the Sun Ray services through the Admin GUI.

SYNOPSIS

    /opt/SUNWut/sbin/utadminuser

    /opt/SUNWut/sbin/utadminuser -h

    /opt/SUNWut/sbin/utadminuser -a *username* [*username ...*]

    /opt/SUNWut/sbin/utadminuser -d *username* [*username ...*]

    /opt/SUNWut/sbin/utadminuser -r

DESCRIPTION

    This command is used to add, list and delete authorized users from the list which is stored in the datastore.

    When invoked without any option, it prints the list of authorized users, one name per line.

    The *username* must be a string consisting of characters from the set of alphabetic characters, numeric characters, period (.), underscore (_), and hyphen (-). The first character should be alphabetic.

    When the Administration GUI is configured to use the default Sun Ray Datastore Authentication mechanism, only the "admin" user is supported and it is added automatically to the list of authorized users by utconfig. When the Administration GUI is configured to use the Unix Login Authentication mechanism, you can add any valid Unix username to the authorized user list.

    Only super-user can add and delete enteries.

OPTIONS

    The following options are supported.

    -a Adds specified users to the list

    -d Deletes specified users from the list

    -r Removes all authorized users

    -h Prints the usage

EXAMPLES

    Example 1: List all current authorized users:

    # /opt/SUNWut/sbin/utadminuser

    Example 2: add a specific user "user1" into the list

    # /opt/SUNWut/sbin/utadminuser **-a** user1

    Example 3: removes all authorized users.

    # /opt/SUNWut/sbin/utadminuser **-r**

EXIT STATUS

    0 if operation performed successfully

    1 if error found

SEE ALSO

    useradd(1M), **utconfig**(1M)

NAME

      utauthd - Sun Ray DTU authentication daemon.

SYNOPSIS

      /opt/SUNWut/lib/utauthd **-b** | **-e** | **-n** | **-s**

DESCRIPTION

      The **utauthd** daemon is responsible for authentication and access control for the Sun Ray DTUs attached to a server. This command should not be executed directly. It is invoked by a system startup script.

OPTIONS

      The following options are supported.

      -b Begin execution of the daemon.

      -e End execution of the daemon.

      -n Number of file descriptors to make available

      -s Signal to send to utauthd

      Without arguments, the default is **-b**.

FILES

      The following files are used by the daemon:

        • **/etc/init.d/utsvc**

      This is the system startup script that invokes the daemon. **/opt/SUNWut/utsessiond**, the Session Manager, performs the actual session switching function.

        • **/etc/opt/SUNWut/auth.props**
          The Authentication Manager's configuration file.

        • **/etc/opt/SUNWut/policy/utpolicy**
          This file determines what policy is used by the Sun Ray server.

      To start the authentication manger, it is customary to run **utrestart**; however, to kill existing user sessions and restart services, it is preferable to use **utrestart** *-c*.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      auth.props(4), **utpolicy(1M), utrestart(1M)**

NAME

      utamghadm - Command used to configure Automated Multi-Group Hotdesking (AMGH), also known as Regional Hotdesking.

SYNOPSIS

      /opt/SUNWut/sbin/utamghadm
      /opt/SUNWut/sbin/utamghadm -d
      /opt/SUNWut/sbin/utamghadm -l *<library>*
      /opt/SUNWut/sbin/utamghadm -s *<executable>*
      /opt/SUNWut/sbin/utamghadm -x

DESCRIPTION

      This command is used by administrators to configure or disable the AMGH feature. AMGH can be configured with backend implementations which are either executable scripts or shared libraries, depending on the site specific implementation. This command provides a means to set the appropriate backend to be used. Setting up the backend implementation implicitly enables AMGH.

      The scripts/libraries used to configure AMGH by this command must be owned by root only and should not have write permissions for others or group.

      Only super-user can change the AMGH configurations using this command.

OPTIONS

      The following options are supported.

      -d Disable AMGH configurations

      -l Specify the library to be used as the AMGH backend implementation. This should be the complete path to the library.

      -s Specify the script to be used as the AMGH backend implementation This should be the complete path to the script.

      -h Prints the usage

      -x Prints the AMGH parameters in machine readable output (key/value form).

EXAMPLES

      Example 1: Display current AMGH configuration

      # /opt/SUNWut/sbin/utamghadm

      Example 2: Configure AMGH to use the script backend called utamghref_script which is available in /opt/SUNWutref/amgh directory.

      # /opt/SUNWut/sbin/utamghadm -s /opt/SUNWutref/amgh/utamghref_script

      Example 3: Configure AMGH to use the libray backend called libutamghref_token.so which is available in /opt/SUNWutref/amgh directory.

      # /opt/SUNWut/sbin/utamghadm -l /opt/SUNWutref/amgh/libutamghref_token.so

      Example 4: Disable AMGH.

      # /opt/SUNWut/sbin/utamghadm -d

EXIT STATUS

      0 if operation performed successfully

      1 if error found

SEE ALSO

      ut_amgh_get_server_list(3), **ut_amgh_script_interface**(3)

NAME

utcapture - Capture packet information from the Authentication Manager.

SYNOPSIS

/opt/SUNWut/sbin/utcapture **-h**

/opt/SUNWut/sbin/utcapture [**-r**] [**-s** *server*] [*desktopID1 desktopID2 ...*]

/opt/SUNWut/sbin/utcapture **-i** *filename*

DESCRIPTION

The **utcapture** command connects to the Authentication Manager and monitors latency, packets sent, and packets dropped between the Sun Ray server and the Sun Ray DTU's.

utcapture writes the captured information to **stdout** in the following format:

TERMINALID TIMESTAMP TOTAL PACKET TOTAL LOSS BYTES SENT PERCENT LOSS LATENCY

OPTIONS

The following options are supported.

-**h** Help for using the command.

-**i** *filename*
Use an input file to search for Sun Ray DTU's that experienced dropped packets. A file is created using **utcapture**:
 # /opt/SUNWut/sbin/utcapture -r > /tmp/*filename*
The process is allowed to run for several minutes or hours. The utcapture command is used again:
 # /opt/SUNWut/sbin/utcapture -i /tmp/*filename*

The output is only the DTU's that experienced dropped packets.

-**r** Write captured data to stdout every 15 seconds in raw, continuous format.

-**s** *server*
Specify the Sun Ray *server* from which to capture data. If outside of the domain of the host running **utcapture**, the Sun Ray server hostname must be fully qualified. By default, the server monitored is the host running **utcapture**.

When no option is specified, **utcapture** writes to **stdout** at 15 second intervals if there is any change in packet loss or latency exceeds 10 ms for any Sun Ray DTU.

OPERANDS

The following operands are supported:

desktopID
Capture data for the specified Sun Ray DTU's only. Appliances are specified by their Ethernet address (*desktopID*) separated by spaces. By default, data for all DTU's is displayed.

EXAMPLES

Example 1: This command captures data from the Authentication Manager running on localhost every 15 seconds and then writes it to **stdout** if there is any change in packet loss for any Sun Ray DTU.

% **utcapture**

Example 2: This command captures data from the Authentication Manager running on localhost every 15 seconds and then writes it to **stdout** regardless if there is any change in packet loss.

% utcapture -r

Example 3: This command captures data from the Authentication Manager running on netraj118.eng every 15 seconds and then writes it to **stdout** if there is any change in packet loss for DTU's with the Ethernet address of **080020a893cb** or **080020b34231**.

% utcapture -s netraj118.eng 080020a893cb 080020b34231

EXIT STATUS

> The following exit values are returned:
>
> **0** Successful completion
>
> **1** Error

ATTRIBUTES

> See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

> **utauthd(1M)**, **utdesktop(1M)**

NOTES

> **utcapture** does not report packet information for Sun Ray DTU's using firmware versions of 1.1 or less.
>
> utcapture does not report latency for Sun Ray DTU's using firmware versions of 1.3 or less.
>
> When using **-r** 0.000 will listed for PERCENT LOSS as 0.000 for every interval that has no loss. When not using **-r** this column will be blank. If the output is to be processed by commands that are column oriented, you must use **-r**.This includes using it as input to **utcapture -i**.

NAME

 utcard - Sun Ray server smart card configuration utility

SYNOPSIS

 /opt/SUNWut/sbin/utcard -**a** *filename*

 /opt/SUNWut/sbin/utcard -**d** *name,version*

 /opt/SUNWut/sbin/utcard -**h**

 /opt/SUNWut/sbin/utcard -**l**

 /opt/SUNWut/sbin/utcard -**p** *name,version*

 /opt/SUNWut/sbin/utcard -**r** *name,version,new-position*

 /opt/SUNWut/sbin/utcard -**u**

DESCRIPTION

 The **utcard** command allows configuration of different types of smart cards in the Sun Ray administration database.

 The administrator must first place a configuration file for a specific smartcard in the **/etc/opt/SUNWut/smartcard** directory. This file must have a **.cfg** extension. The smartcard definition in the **.cfg** file is added to the LDAP datastore by using the **-a** option. When a smartcard definition is added, it is automatically assigned the last position in the probe order.To modify the probe order, use the **-r** option.

OPTIONS

 The following options are supported.

 -**a** *filename*
  Add the card specified within *filename* that is in the **/etc/opt/SUNWut/smartcard** directory

 -**d** Delete the card specified with *name, version*.

 -**h** Show usage information

 -**l** List all configured cards

 -**p** Show the standard properties for the card specified with *name*, *version*.

 -**r** Reorder the card specified with *name, version, to new-position*.

 -**u** List unconfigured cards available for configuration as determined by the **.cfg** files in **/etc/opt/SUNWut/smartcard**

USAGE

 Use this command only on a Sun Ray server that has been configured for administration by the **utconfig** command.

ATTRIBUTES

 See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

 utconfig(1M)

NAME

>   **utconfig** - Sun Ray server software configuration utility.

SYNOPSIS

>   **/opt/SUNWut/sbin/utconfig** [**-u**]

DESCRIPTION

>   The **utconfig** command performs initial configuration of Sun Ray server and supporting administration framework software. Before taking any actions the command prompts the user for configuration parameters for each of the supporting software packages. The command must be run with superuser privileges.

OPTIONS

>   The following option is supported.

>   **-u** Unconfigure the Sun Ray server and administration software returning the mode of operation back to the default zero administration mode.

ATTRIBUTES

>   See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

>   Packaging related commands.
>   **utinstall(1M)**

NAME

      utcrypto - Sun Ray privacy administration utility.

SYNOPSIS

      /opt/SUNWut/sbin/utcrypto

      /opt/SUNWut/sbin/utcrypto -a *key=value ...*

      /opt/SUNWut/sbin/utcrypto -d

      /opt/SUNWut/sbin/utcrypto -e [-f *filename*]

      /opt/SUNWut/sbin/utcrypto -h

      /opt/SUNWut/sbin/utcrypto -m *key=value ...*

      /opt/SUNWut/sbin/utcrypto -o [**-f** *filename*]

DESCRIPTION

      The **utcrypto** command allows the administrator to configure privacy options for the Sun Ray server. These settings include upstream/downstream encryption and upstream/downstream authentication. These settings will affect all sessions.

      utcrypto operations that only display information may be run by any user. Operations that change or delete data must be run as superuser.

OPTIONS

      The following options are supported.

      -**a** Add the privacy settings for all sessions. The -**a** option must be followed by a series of *key=value* pairs separated by spaces. Valid *key=value* pairs are described below. If a *value* pair is not passed in, the value for that key will be set to a default value.

      At least one key=value pair must be specified.

      -**d** Delete the privacy settings for all sessions.

      -**e** Take the privacy settings in comma-delimited format from **stdin** and add/modify the settings for all sessions. This will be a space-delimited list of *key=value* pairs. If followed by the -**f** option and a *filename*, it will read the settings from a file.

      To replicate **utcrypto** settings, run **utcrypto** -**o** on the "source" server, then use the stdout string as an argument to **utcrypto** -**e** on the "target" server or servers.

      -**f** Specifies a filename for the -**e** or -**o** option.

      -**h** Displays the usage message.

      -**m** Modify a privacy settings for all sessions. The -**m** option must be followed by a series of *key=value* pairs separated by spaces. Valid *key=value* pairs are described below. If a *value* is not passed in, the value for that key will remain unchanged.

      -**o** Dump all privacy settings in comma-delimited format to stdout. If followed by the -**f** option and a *filename*, the settings will be dumped to the file. If the file specified by *filename* exists, then a warning message is generated and the script exits.
      If an asterisk (*) appears in the Inherited column, then no value pair was specified, so it has been set to a hard-coded default value.

      To replicate **utcrypto** settings, run **utcrypto** -**o** on the "source" server, then use the stdout string as an argument to **utcrypto** -**e** on the "target" server or servers.

      The following are valid *key=value* pairs:

      enc_up_type
          ARCFOUR | **none** | **default**

      enc_down_type
           ARCFOUR | **none** | **default**

      auth_up_type
           none | **default**

auth_down_type
　　simple | **none** | **default**

mode
　　hard | **soft** | **default**

The keyword **default** for all keys is set to the value of the default configuration, if it exists. If there a default configuration has not been set, then the value is set to hard-coded defaults. The hard-coded default values for the first four keys are **none** and **soft** for the **mode** key.

EXAMPLES

This adds the configuration for upstream ARCfour encryption and simple downstream authentication.

　　Note: Since enc_down_typeand **auth_down_type**are not specified, they take on the default values:

/opt/SUNWut/sbin/utcrypto -a enc_up_type=ARCFOUR auth_down_type=simple

This command modifies the default configuration. Upstream encryption is turned off and downstream encryption is set to ARCfour.

/opt/SUNWut/sbin/utcrypto -m enc_up_type=none enc_down_type=ARCFOUR

This command removes the default configuration.

/opt/SUNWut/sbin/utcrypto -d

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

NAME

      utdesktop - Sun Ray DTU administration utility.

SYNOPSIS

      /opt/SUNWut/sbin/utdesktop **-a** "*desktopID,location,otherinfo*"

      /opt/SUNWut/sbin/utdesktop **-a** **-f** *filename*

      /opt/SUNWut/sbin/utdesktop **-d** *desktopID*

      /opt/SUNWut/sbin/utdesktop **-d** **-f** *filename*

      /opt/SUNWut/sbin/utdesktop **-e** "*desktopID,location,otherinfo*"

      /opt/SUNWut/sbin/utdesktop **-e** **-f** *filename*

      /opt/SUNWut/sbin/utdesktop **-h**

      /opt/SUNWut/sbin/utdesktop **-l** [**-c** | **-g** | **-w** [**-t** *timeout*]]

      /opt/SUNWut/sbin/utdesktop **-L** { **-c** | **-w** [**-t** *timeout*] }

      /opt/SUNWut/sbin/utdesktop **-l** **-i** *substring*

      /opt/SUNWut/sbin/utdesktop **-o**

      /opt/SUNWut/sbin/utdesktop **-p** *desktopID*

DESCRIPTION

      The **utdesktop** command allows the user to manage Sun Ray DTU's connected to the Sun Ray server the command is run on. The information that **utdesktop** displays and allows the user to add, edit, or delete is stored in the Sun Ray administration database. Other information is obtained from the Sun Ray authentication manager.

      utdesktop operations that only display information may be run by any user. Operations that add, edit, or delete data must be run by the superuser.

OPTIONS

      The following options are supported.

      -a *desktopID,location,otherinfo*
          Add DTU with the specified desktop-ID, location, and other information properties. Note that the 3 comma-delimited values should be enclosed within quotes. You must be root to use this option.

      -a **-f** *filename*
          Batch add multiple DTU's using input from the specified filename. The format of each line in the input file is: desktop-ID,location,other-info. You must be root to use this option.

      -d *desktopID*
          Delete the DTU with the specified desktop-ID. You must be root to use this option.

      -d **-f** *filename*
          Batch delete multiple DTU's using input form the specified filename. The format of each line in the input file is: desktop-ID. You may use the output of the **-o** option to feed this option as all arguments after the first comma are ignored. You must be root to use this option.

      -e *desktopID,location,otherinfo*
          Edit properties for the specified DTU by changing the location and other information properties to the specified values. Note that the 3 comma-delimited values should be enclosed within quotes. You must be root to use this option.

      -e **-f** *filename*
          Batch edit properties for multiple DTU's using input from the specified filename. The format of each line in the input file is: desktop-id,location,other-info You must be root to use this option.

      -h Show usage information (this message).

      -l List all DTU's currently registered in the admin database.

      -l **-c**
          List all DTU's that are currently connected to the server (and note any that have been deleted with question marks in the Location field).

      -L **-c**
          List all DTU's that are currently connected (long format).

-l -g
  List all currently connected DTU's and the servers they are connected to.

-l -**w** [-**t** *timeout*]
  List all DTU's waiting for a session during the set *timeout* (short format). The default value of the timeout is 60 seconds.

-**L** -**w** [-**t** *timeout*]
  List all DTU's waiting for a session during the set *timeout* (long format). The default value of the timeout is 60 seconds.

-**l** -**i** *substring*
  List all DTU's with desktop IDs that contain the specified substring.

-**o** Dump DTU list in comma-delimited format. The format of each line output by this option is: desktop-id,location,other-info

-**p** Show desktop properties for the DTU with the specified ID.

EXAMPLES

  Example 1: This command clears the location and the other information properties for DTU 080020a85112:

  # **utdesktop -a "080020a85112,,"**

  Example 2: This command changes the location and the other information properties for DTU 080020a85112 to "SFO12-2103" and "John's Office", respectively:

  # **utdesktop -e "080020a85112,SFO12-2103,John's Office"**

  Example 3: This command edits the properties of multiple DTU's using input from the file /tmp/ desktops:

  # **utdesktop -e -f /tmp/desktops**

  Example 4: This command displays all DTU's that contain "a851" in their desktop IDs:

  % **utdesktop -l -i a851**

  Example 5: This command lists all DTU's in an error state without sessions within the default timeout:

  % **utdesktop -l -w**

  Example 6: For a busy or slow network, this command lists (in long format) all DTU's in an error state without sessions for at least five minutes:

  % **utdesktop -L -w -t 300**

  Example 7: This command displays the current properties for DTU's 080020a85112:

  % **utdesktop -p 080020a85112**

FILES

  The following file is used:

  /etc/opt/SUNWut/utadmin.conf

ATTRIBUTES

  See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

      utuser(1M), **utadmin.conf(4)**, the *Sun Ray Server Software 1.3 Administrator's Guide*

NOTES

      The **-G** option has been deprecated. Use the **-l -g** option pair instead.

NAME

      utdevadm - Enable or disable Sun Ray device services.

SYNOPSIS

      /opt/SUNWut/sbin/utdevadm

      /opt/SUNWut/sbin/utdevadm [-e | -d] -s *service*

      /opt/SUNWut/sbin/utdevadm -h

DESCRIPTION

      The utdevadm command is used to enable/disable Sun Ray device services. This includes USB devices connected through USB ports, Embedded serial ports and Internal smartcard readers on the Sun Ray DTU.

      After Sun Ray Server Software installation, by default all the device services will be enabled. utdevadm command can be used to enable/disable device services only in the configured mode, i.e. it is available only after **utconfig**(1M) has been run to activate Sun Ray data store. This is a site-wide property. When configured, it affects all units connected to the failover group.

      It is required to do a cold restart of Sun Ray services before the change can take effect. When a successful change is made, the command reminds the administrator to restart services using the following command: /opt/SUNWut/sbin/utrestart **-c**

      utdevadm command with no options or arguments shows the current state of the device services.

OPTIONS

      The following options are supported:

      -e Enables specific *service*.

      -d Disables specific *service*.

      -s Specify a *service*.
            internal_serial | internal_smartcard_reader | usb | all

            When internal_serial *service* is disabled, users will not be able to access embedded serial ports on the Sun Ray DTU. Sun Ray 170 has two embedded serial ports.

            When internal_smartcard_reader *service* is disabled, users will not be able to access internal smartcard reader through PC/SC or SCF interface for reading/writing. It does not affect session access, hotdesking using unauthenticated smartcards.

            When usb *service* is disabled, users will not be able to access any devices connected to USB ports. It does not affect HID devices such as keyboard, mouse, barcode reader.

            Specifying all will enable/disable all device services.

      -h Print the usage.

EXAMPLES

      Example 1: The following command shows the current state of the device services.

            # /opt/SUNWut/sbin/utdevadm

```
Sun Ray Device Service         Status
------------------------------ ------------
internal_serial                enabled
internal_smartcard_reader      enabled
usb                            enabled
```

      Example 2: The following command disables usb service.

            # /opt/SUNWut/sbin/utdevadm -d -s usb

Example 3: The following command enables usb service.

 # /opt/SUNWut/sbin/utdevadm -e -s usb

ENVIRONMENT VARIABLES

None

EXIT STATUS

The following exit values are returned:

0 on success

1 on error

FILES

None

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto
Interface Stability	Public Evolving

SEE ALSO

utrestart(1M), **utconfig**(1M)

NAME

   utdevmgrd - Sun Ray device manager daemon.

SYNOPSIS

   /opt/SUNWut/lib/utdevmgrd [**-a** *authlist*] [**-c** *authfile*] [**-d**] [**-k** *authprops*] [**-o** *optroot*] [**-p** *port*]
   [**-r**] [**-s** *sigfile*] [**-t**]

DESCRIPTION

   The **utdevmgrd** daemon is responsible for brokering devices that are attached to Sun Ray DTU's on the
   interconnect fabric for the purpose of remotely accessing the devices for various services. It is also
   responsible for approving services, keeping an inventory of devices and their controlling services, and
   locating devices on the interconnect.

   If either the **-a** or the **-c** option is specified, the device manager daemon operates exclusively in call-back
   mode. In this mode, the device manager only communicates to authentication managers that are explicitly
   enabled by *authlist* or *authfile* and that have requested a call-back. The call-back feature provides a
   mechanism by which the device manager and the authentication manager establish each other's identity.

   The *optroot* directory (default **/tmp/SUNWut**) is shared with other Sun Ray server components. Primarily it
   provides the location for a compatible device tree for each Sun Ray DTU's devices in the **sessions** and **units**
   subdirectories.

   The **units** subdirectory contains a directory for each DTU on the interconnect named by the DTU's serial
   number. Within an DTU's directory, there are the familiar **dev** and **devices** directories that list logical names
   for devices and geographically hierarchical names for devices.

   The **sessions** directory contains symbolic links into the **devices** directory that indicate which sessions are
   connected to which Sun Ray DTU's. The symbolic links are named after the X-Windows server display
   corresponding to a user's session by display number only (in other words, after removing the server name,
   which is always a name local to the current host, and the screen numbers). The user's **DISPLAY** environment
   variable can then be used to find the devices on the 'current' DTU. The user's **UTDEVROOT** environment
   variable achieves this, and can be used to find devices that are 'currently' accessible. The *optroot* directory
   also includes the named pipe with which the device manager communicates to device driver services and the
   session_info directory, which contains user information important to internal workings of the device
   manager.

   The device manager works within a Sun Ray server group environment, which enables rapid switching to
   other servers and user load distribution. In order for device managers on each server in a group to
   communicate, the device manager must gain access to the group signature file. If the signature does not match
   the one used by other device managers in the group, then grouping will fail and not all devices on all DTU's
   on the interconnect will be available on the server, including devices on some DTU's being used by users on
   the server.

   Normally, the device manager finds the group signature file by looking into the authentication manager's
   configuration file (**/etc/opt/SUNWut/auth.props**), but this can be changed by using the **-s** and **-k** options. If
   -s is specified, then *sigfile* is read and used as the group signature. If **-k** is specified, then the *authprops* file
   is scanned for the **gmSignatureFile** key and the listed file is used for the group signature.

   Error messages from **utdevmgrd** are logged using **syslog**(3), with a facility value of LOG_DAEMON.

OPTIONS

   The following options are supported.

   -a *authlist*
       Add the host and port pairs specified in *authlist* to the list of allowed authentication managers. The format
       of *authlist* is a comma separated list of *hostname*:*port* pairs.

   -c *authfile*
       Add the host and port pairs specified in the ASCII file *authfile* to the list of allowed authentication
       managers. The file contains a list of authentication manager specifications, one per line. The
       specifications take the form of hostname followed by port number, separated by white space. Blank lines
       and any line whose first printable character is ''**#**'' are ignored.

   -d Enable debugging output.

   -k *authprops*
       Set the location for the authentication manager's configuration file to *authprops*. This file is used to find
       the group signature file in case the *sigfile* key was not specified. The default for this parameter is **/etc/opt/
       SUNWut/auth.props**. The key in this file that specifies the group signature is **gmSignatureFile**.

-**o** *optroot*
>   Set the device information root directory to *optroot*. This directory contains the service named **pipe**, and the **units**, **sessions**, and **session_info** directories. *optroot* is generally shared with other Sun Ray server components.

-**p** *port*
>   Set the device manager's listen port to the specified port value. The device manager defaults to port 7011. This is the port by which device services and authentication managers contact the device manager.

-**r**   Automatically restart the device manager daemon if it exits. With this option, the device manager daemon creates two processes: a child that performs all the actual work and a parent monitoring process. The parent process will restart a child if the previous one exits. This enables existing services to re-attach to a new child device manager.

-**s** *sigfile*
>   Set the path of the group signature file to *sigfile*.

-**t**   Test mode. Relax checking for error returns for files that are root access. Could cause unpredictable results on an operational device manager in case of true failure.

FILES

The following files are used:

/etc/opt/SUNWut/auth.permit
>   The customary location of the *authfile* for a system.

/tmp/SUNWut
>   The customary location for temporary files used by Sun Ray enterprise server managers, designated by *optroot*.

/tmp/SUNWut/.utdevmgr
>   The named pipe used for communication between the device manager and device driver services.

/tmp/SUNWut/units
>   The directory containing device directories for each DTU. The directory names are after the DTU's serial numbers. Each directory contains a **dev** directory and a **devices** directory.

/tmp/SUNWut/sessions
>   The directory containing links to DTU's in the **units** directory, named by X-Windows display number for each session. These links change as users move from one Sun Ray DTU to another.

/tmp/SUNWut/session_info
>   The directory containing information internal to the device manager for handling session mobility.

/etc/opt/SUNWut/auth.props
>   The customary location of the *authprops* file containing the authentication manager settings. The device manager looks for the gmSignatureFile key to extract the location of the group signature file.

/etc/opt/SUNWut/gmSignature
>   The customary location of the *sigfile* file containing the group signature.

ENVIRONMENT VARIABLES

The following environment variables are used:

DISPLAY
>   Use to get the default X-Windows display number from within the user's environment.

UTDEVROOT
>   Use to get the devices for the current session from within the user's environment.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

utauthd(1M), **syslog(3)**, **syslogd(1M)**, **syslog.conf(4)**

NAME

       utdiskadm - Sun Ray Mass Storage device management utility

SYNOPSIS

       /opt/SUNWut/bin/utdiskadm -c *device_name*

       /opt/SUNWut/bin/utdiskadm -e *device_name*

       /opt/SUNWut/bin/utdiskadm -h

       /opt/SUNWut/bin/utdiskadm -l [-a]

       /opt/SUNWut/bin/utdiskadm -m *partition_name* [-p *mount_path*]

       /opt/SUNWut/bin/utdiskadm -r *device_name*

       /opt/SUNWut/bin/utdiskadm -s [-a]

       /opt/SUNWut/bin/utdiskadm -u *mount_point*

DESCRIPTION

       The **utdiskadm** command allows the user to perform administration tasks on mass storage devices connected to the Sun Ray DTU with which the current login session is associated. This command does not work on devices connected to any other Sun Ray DTU, nor would it work on devices connected directly to the Sun Ray server.

OPTIONS

       The following options are supported.

          -c *device_name*

          Check device for existence of media

          -e *device_name*

          Eject media from devices with removable media

          -h

          Show usage information

          -l

          List all storage devices of current session and their mount points

          -l -a

          List all storage devices of on system. This option can be used by the super-user only

          -m *partition_name*

          Mount partition partition_name on default mount point in **$DTDEVROOT/mnt**.

          -m *partition_name* -p *mount_path*

          Mount partition *partition_name* on directory *mount_path*

          -r *device_name*

          Prepare device *device_name* for removal by unmounting all its partitions

          -s

          List stale mount points for which physical devices do not exist

          -s -a

          List stale mount points on entire system All stale mount points on the Sun Ray server are displayed. Only the super-user is allowed to use this option.

          -u *mount_point*

          Unmount *mount_point*

EXIT STATUS

       The following exit codes are returned:

          0 The operation was successful

          1 The operation was   unsuccessful

FILES

      The following files are used:

          $UTDEVROOT/dev/dsk

          The directory containing links to block device names for each partition on the device.

          $UTDEVROOT/dev/rdsk

          The directory containing links to raw device names for each partition on the device.

ENVIRONMENT VARIABLES

      UTDEVROOT points to a symbolic link of the device root for the Sun Ray appliance associated with a user's session.

      DTDEVROOT points to a temporary directory associated with the user's session. The directory's lifetime is equal to the lifetime of the login session. It is removed, along with its contents, when the user logs out.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      utmountd(1M), **utstoraged(1M)**, **utdisk(7D)**

NAME

 utdssync - Converts and synchronizes Sun Ray datastore service port.

SYNOPSIS

 /opt/SUNWut/sbin/utdssync [**-v**]

DESCRIPTION

 The **utdssync** command converts the Sun Ray datastore service port on the primary datastore server to the default port 7012. It also synchronizes all the secondary servers to use the same port number.

 utdssync is intended for use after software upgrades to Sun Ray Server Software 3.0 (SRSS 3) have been completed on all the servers within the data sharing group.

 Note:

 The command must be run with superuser privileges on the primary datastore server.

OPTIONS

 The following option is supported.

 -v verbose mode

ATTRIBUTES

 See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

 utreplica(1M), **utrcmd(1M)**, **utinstall(1M)**

NAME

   utfwadm - Sun Ray DTU firmware version management.

SYNOPSIS

   /opt/SUNWut/sbin/utfwadm -A { -a | -e *enetAddr* } [-f *firmware*] { -n *intf* | -N *subnetwork* | -V } [-d]
   [-F] [-u] [-i *filename*] ...

   /opt/SUNWut/sbin/utfwadm -D { -a | -e *enetAddr*} { -n *interface* | -N *subnetwork* | -V } ...

   /opt/SUNWut/sbin/utfwadm   -P [-V]

   /opt/SUNWut/sbin/utfwadm   -R

DESCRIPTION

   The **utfwadm** command manages firmware upgrades to Sun Ray DTUs. The DTUs are capable of loading
   firmware upgrades and programming new firmware into their flash PROM (Programmed Read-Only
   Memory).

   When a DTU is powered on, the firmware obtains an IP address and other configuration information using
   the DHCP protocol. Part of the configuration information is a firmware version identifier. If this identifier
   does not match the DTU's existing firmware, the DTU initiates an upgrade which replaces the current
   firmware with the new version.

   The **utfwadm** command must be run when a new firmware version is installed to update the firmware version
   identifier and force the DTUs to load the new version on their next power cycle. utfwadm allows firmware
   identifiers to be set on either a per-network or per-unit basis enabling firmware upgrades to be targeted at
   entire Sun Ray subnetworks or individual DTUs.

   In the SRSS 3 update 1 release, a new mechanism has been introduced to manage firmware without using
   DHCP alone. With increased deployment of Sun Rays remote from their servers, it became difficult to supply
   DHCP parameters from the server to the Sun Ray. Thus, a different mechanism that uses TFTP to access a
   firmware configuration file on the firmware server was developed. This mechanism involves the creation of
   Model.parms or *Model.<MAC>*.parms files for the firmware files corresponding to each Sun Ray *Model*
   (e.g., ''CoronaP1'') that **utfwadm** installs in the **/tftpboot** directory. The files containing *<MAC>* are unit-
   specific, where *<MAC>* is the value defined using the -**e** option.
   See the **ut_parms(8M)** page for a more complete description of the **.parms** file format.

   The modifications to **utfwadm** to support this have been made, and a few additional options have been
   defined to support control of firmware downloads through the creation of these configuration files. These
   options allow the forcing of a firmware downgrade where that would normally be prevented, the generation
   of only these files, without updating DHCP, the use of the frame buffer to hold the downloaded firmware file,
   and the inclusion of an additional file that specifies parameters other than firmware related ones, such as a list
   of servers to choose from.

   It is possible to determine the firmware versions that are available and in use by each Sun Ray connected to
   the current server. See the **utfwload(1M)** command.

   • In the **/tftpboot** directory are the firmware files. To verify the version of a firmware file, type:
     /opt/SUNWut/lib/lzd < /tftpboot/*firmware-filename* **| what**

   • To identify the version of firmware that a particular Sun Ray DTU is using, type:
     /opt/SUNWut/sbin/utdesktop -p *desktopID*

   Where *desktopID* is the full MAC address. To display the MAC address, press the three audio keys on the
   Sun Ray DTU keyboard simultaneously.

   The Sun Ray subnetworks must have been previously set up using the **utadm(1M)** command. The **utfwadm**
   command is run under super-user privileges.

OPTIONS

   The following options are supported.

   -**A** Add a unit or units to the list of DTUs to be upgraded. Please note: The -**A** option must be followed by
        either the -**a** or -**e** suboption.

   -**D** Remove the defined DTUs from the list of units to be upgraded. This option causes the firmware version
        identifier to be unset. Please note: The -**D** option must be followed by either the -**a** or -**e** suboption.

-**a** Add or delete all units from the list of DTUs to be upgraded.

-**e** *enetAddr*
Apply the operation to only the specified unit with Ethernet address given by *enetAddr*, where all six hex bytes of the address are specified.

-**f** *firmware*
This option gives the pathname for the firmware to be downloaded to the DTUs. If *firmware* refers to a file, the hardware version is extracted from the version string within the file, and the file is copied to the **/tftpboot** directory to be downloaded only to that version of the hardware. If *firmware* refers to a directory, then all files named ''Corona*'' in the directory are copied to the **/tftpboot** directory with their version strings appended. If the **-f** option is not given, a default location is used.

-**N** *subnetwork*
Apply the given operation to units attached on the specified *subnetwork*. Multiple subnetworks may by given. It can also specify the special keyword **all** to apply the operation to all configured Sun Ray subnetworks.

-**N all**
Apply the given operation to all subnetworks.

-**n** *intf*
Apply the given operation to units connected to the Ethernet interface *intf*. Multiple interfaces may be given, or the special keyword **all**, which applies the operation to all configured Sun Ray interfaces.

-**V** With -**A**, generate the **.parms** files but do not update DHCP. Note that this option should not be used if DHCP is still being used to provide firmware parameters to DTUs with old firmware.With -**P**, include a listing of the versions specified in the **.parms** files.

-**F** Force the Sun Rays to download this firmware, even though the *barrier* for this firmware is less than the *barrier* defined in the current firmware.

-**u** Turn on firmware load through the frame buffer. This is primarily for debugging purposes, to show the progression of the firmware loading process.

-**i** *parmfile*
Include the values contained in *parmfile* at the end of the **.parms** file for each firmware model. The key/value pairs are currently restricted to the specification of a servers list as ''**servers=**<*name list*>'' and a selection order as ''**select=inorder**'' or ''**select=random**''. These values are not saved across successive invocations of this command, so to retain them, the -**P** option must be specified each time.

-**d** Explicitly disable firmware download by creating **.parms** files with **version=_NONE_**, used with -**A**. This is different from -**D** which removes the **.parms** files completely. This is most useful with -**e** to prevent a particular Sun Ray from being upgraded. It also makes sense to use this to disable firmware upgrade completely, but provide a server list using the -**i** option.

-**n all**
Apply the given operation to all interfaces.

-**P** This variant of the command prints out the version to which each domain should be upgraded on the next power cycle. A domain may be either an interconnect subnet or individual DTU. If it is a subnet, then the **Intf** column lists the interface device. If it is an individual DTU, then its Ethernet address is given in the **Domain** column, and the **Intf** column contains the interface name. If -**V** is specified, the versions from the **.parms** files are printed.

-**R** Remove the firmware files that were copied into the boot directory.

The **-z** option is RESERVED for use by the Sun Ray server software and should not be used.

FILES

The following files are used:

- **/var/dhcp/dhcptab**
File or NIS+ table

- **/tftpboot**
Default location of firmware boot file

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

dhtadm(1M), **dhcpconfig(1M)**, **what(1)**, **dhcp(4)**, **dhcp_network(4)**, **dhcptab(4)**, **attributes(5)**, **utadm(1M)**, **utdesktop(1M)**

NAME

    utfwload - Provide a compact summary of sessions and firmware maintenance.

SYNOPSIS

    /opt/SUNWut/sbin/utfwload [-**a**] [-**l**] [-**L**] [-**H**]

DESCRIPTION

    Without an argument, the **utfwload** command provides the display number of the current session for each connected Sun Ray in use, as well as the logged-in user, IP address, and firmware revision level. Options allow forcing the download of firmware to Sun Rays that are not running the current version supplied by the Sun Ray server.

    Note:

    utfwload upgrades Sun Rays that have been loaded with firmware from the SRSS 2.0 114880-04 patch or later or from SRSS 3.0 only. This command cannot force an upgrade of earlier firmware.

OPTIONS

    The following options are supported.

    -**a** In combination with other options, this option controls the selection of Sun Rays or sessions to display. Without-*a* , only sesssions that have logged in users are displayed. With -*a*, all sessions or Sun Rays are displayed, and those with no logged in user have a user id field of "????".

    -**l**,-**L**

        With **utfwload**, force Sun Rays that are not running the version of firmware installed on the current server to upgrade. The command compares the "System Version" as provided by running **utfwadm** -**P** with the value that would be displayed with -**f**. If the version is different, a download is forced. The -**l** form of the command forces loading only on Sun Rays that are connected to sessions with no user logged in, while the -**L** form forces a download on all Sun Rays that are out of date. This option is available only to users with root privileges.

    -**H** Output column headings above the regular output.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

    utwho(1M), **attributes(5)**

NAME

      utfwsync - Synchronizes Sun Ray DTU firmware downloads.

SYNOPSIS

      /opt/SUNWut/sbin/utfwsync [**-d**] [**-v**]

DESCRIPTION

      The **utfwsync** command refreshes the firmware level on the Sun Ray DTUs (DTUs) with the default firmware version for the current Sun Ray Server Softrware release and patch level. It then forces all the Sun Ray DTUs connected to that Sun Ray server to restart. If the Sun Ray server is part of a failover group, then all Sun Ray DTUs within the failover group are forced to restart. The result is that each DTU tries to download the latest firmware offered by the primary Sun Ray server as it restarts, as described in the **utfwadm(1M)**man page.

      This command is intended for use after software upgrades or after new firmware has been installed on all hosts as part of a patch.

      As the command executes, access to user sessions is interrupted, but the sessions are not lost and are returned after the command completes.

      The command must be run with superuser privileges.

OPTIONS

      The following option is supported.

      -d Disables system reconfiguration and forces all connected DTUs to load the currently configured version of the firmware. The currently configured version of the firmware may or may not be the default version for the current release and patch level.

      -v Verbose mode. Additional messages regarding what is being done are written to **stdout**.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

      utgstatus(1M), **utauthd(1M)**, **utfwadm(1M)**, **utinstall(1M)**

NAME

      utgroupsig - Sets the group signature for Sun Ray servers in a failover group.

SYNOPSIS

      /opt/SUNWut/sbin/utgroupsig

DESCRIPTION

      The **utgroupsig** command sets the failover group signature.

      The **utgroupsig** command prompts for the new signature twice. The group signature file is at least 8 bytes long and has similar content diversity characteristics as required by **passwd(1)**.

      The signature is stored in clear in the location specified in the **auth.props** file with the **gmSignatureFile** property. The group signature file is created with owner **root** and mode 600 (read-write by root).

OPTIONS

      There are no options for this command.

FILES

      The following files are used:

         • **/etc/opt/SUNWut/gmSignature**
           Sun Ray group signature default file.

         • **/etc/opt/SUNWut/auth.props**
           Sun Ray authentication properties file.

EXIT STATUS

      The following exit values are returned:

      0 Success

      1 Invalid input to command.

      2 Unexpected failure. Signature file unchanged.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

      utrcmd(1M), **passwd(1M)**, **auth.props(4)**

NAME

      utgstatus - Display failover group status.

SYNOPSIS

      /opt/SUNWut/sbin/utgstatus [-**s** *hostname*]

DESCRIPTION

      The **utgstatus** command allows the user to view the failover group status information for the local server or for the named server. The information that the command displays is specific to that server at the time the command is run.

      utgstatus displays information only and so can be run by any user.

OPTIONS

      The following option is supported.

      -**s** *hostname*
         Display all the failover group status information for the specified *hostname*.

EXAMPLES

      Example 1: This command displays the failover group status for the local Sun Ray server

      % **/opt/SUNWut/sbin/utgstatus**

      Example 2: This command displays the failover group status for the server sunray3:

      % **/opt/SUNWut/sbin/utgstatus -s sunray3**

      Information returned from this command looks like the following for a typical LAN-based configuration:

      Or like this for a typical interconnect-based configuration:

```
host        flags     interface      flags
                      192.24.0.0/24

-------------     ------------------
sunray3     TN    192.24.0.136   UAM
sunray1     T-    192.24.0.93    UA-
sunray2     TN    192.24.0.95    UAM
sunray-sras TN    192.24.0.96    U--
```

      For proper viewing, make sure the window is wide enough.

      Explanation of **utgstatus** information:

      The Network/Netmask values are denoted in CIDR (Classless Inter Domain Routing) network address notation, where the initial value (192.24.0.0) is the network address itself and the '/24' suffix signifies the number of bits that are the network identifier of the address. The remaining 8 bits are for specific host addresses.

      Host Status Flags

        T Trusted — The trusted hosts are members of this failover group because they share the same group signature.

        N Nline — The server is configured to participate in load balancing (see the utadm man page for a description of the -n option).

      Interface Status Flags

        U Up — The interface is currently reachable by this host.

        A Available — The interface is available for Sun Rays to connect to it and get service.

        M Managing — The interface is configured to manage Sun Rays on its local subnet (in other words, utadm -a was run to configure the interface for Sun Ray service).

In the first example above, all hosts are part of the same failover group. All hosts but sunray1 are "online", which means they will participate in creating in normal session creation. sunray1 is "offline", which means it will not participate in session creation during load balancing for this failover group, although sessions can still be created on it, either explicitly through the use of **utswitch** or **utselect -R**, or implicitly if all other servers are down. The LAN interfaces for all of the hosts are Up, and all but sunray-sras are Available to Sun Rays (sunray-sras did not have **utadm -a** run to configure its interface for Sun Ray service, and it does not have allowLANConnections=true set in auth.props. It is a dedicated SRAS server for the failover group). Both sunray2 and sunray3 are Managing Sun Rays, because **utadm -a** was run for their LAN interfaces. They will offer DHCP parameters and possibly addresses during the Sun Ray bootup phase for Sun Rays on their local subnets.

In the second example above, all of the hosts are Trusted and Online, meaning that they will all participate in failover and load balancing for their Available interfaces. 193.25.0.0/24 is the LAN network, and the other networks are Sun Ray interconnects. All of the LAN interfaces are Up, meaning that they are reachable, but none are Available for Sun Ray service, and none are Managing Sun Rays on the local subnet. All of the interconnect interfaces are Up and Available, and all are Managing Sun Rays.

SEE ALSO

> **utadm**, **auth.props**, **utswtich**, **utselect**

ATTRIBUTES

> See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

NAME

    utinstall - Sun Ray Server Software installation, upgrade, and removal utility.

SYNOPSIS

    /cdrom/cdrom0/utinstall [**-a** *admin-file*] [**-d** *media-dir*] [**-u**] [**-j** *jre*]

    /opt/SUNWut/sbin/utinstall [**-a** *admin-file*] [**-d** *media-dir*] [**-u**] [**-j** *jre*]

DESCRIPTION

    The **utinstall** command installs, upgrades, and removes Sun Ray Server Software. All software required to support the Sun Ray server is installed.

    The **utinstall** command is run under superuser privileges and prompts the user before taking any action. The use of defaults is recommended.

OPTIONS

    -a *admin-file*
        Instead of the default, use *admin-file* as the admin file for **pkgadd** operations (see **pkgadd(1M)**).

    -d *media-dir*
        Instead of the default, use *media-dir* as the installation media root directory.

    -j *jre*
        Instead of the default, use the particular jre specified in the argument.

    -u Remove previously installed Sun Ray Server Software.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuti

SEE ALSO

    Solaris:
    patchadd(1M), **patchrm(1M)**, **pkgadd(1M)**, **pkgrm(1M)**, **admin(4)**

NAME

      utkiosk - Sun Ray script to update kiosk configuration locally and within a failover group.

SYNOPSIS

      /opt/SUNWut/sbin/utkiosk { -**e kiosk**|-**i kiosk**|-h| }

DESCRIPTION

      The **utkiosk** script is used to import and export kiosk configuration information into the LDAP database. utkiosk also updates local configuration files in the **/var/opt/SUNWut/kiosk** directory and application working files in the **/var/opt/SUNWut/kiosk/config** directory. **utkiosk** is used primarily in failover groups.

OPTIONS

      The following options are supported.

      -**e kiosk**
        Export kiosk configuration from the datastore (LDAP database) to the local configuration file. Customarily, this option is run on a standalone server or on the secondary servers in a failover group; however, it can be run on the primary server if necessary.

      If the LDAP server is running, this option synchronizes the secondary server's LDAP database with that of the primary server, and the configuration is exported from LDAP, replacing the local configuration file. The application working files are updated for the local configuration file.

      -**i kiosk**
        Import kiosk configuration to the datastore (LDAP database) from the local configuration file. This option is only run on a standalone server or on the primary server in a failover group. It updates the local kiosk configuration file and imports that file into the LDAP database if the LDAP server is running.

      -**h** Print this message.

EXAMPLES

      Example 1: This command imports the configuration to LDAP and updates all local files.

        # **utkiosk -i kiosk**

      Example 2: This command exports the configuration from LDAP and updates all local files

        # **utkiosk -e kiosk**

FILES

      The following files are used:

      /var/opt/SUNWut/kiosk/kiosk.conf

      All application working files in **/var/opt/SUNWut/kiosk/config**

EXIT STATUS

      The following exit values are returned:

      0 Success

      1 Failure

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

NAME

    utmhadm - Sun Ray DTU multihead group configuration utility.

SYNOPSIS

    /opt/SUNWut/sbin/utmhadm [*groupname*]

    /opt/SUNWut/sbin/utmhadm -**a** *groupname* -**g** *COLSxROWS* -**p** *primaryCID* -**l** *CID1,CID2,...,CIDn*

    /opt/SUNWut/sbin/utmhadm -**d** *groupname*

    /opt/SUNWut/sbin/utmhadm -**e** [-**f** *filename*]

    /opt/SUNWut/sbin/utmhadm -**o** [-**f** *filename*]

    /opt/SUNWut/sbin/utmhadm -**h**

    /opt/SUNWut/sbin/utmhadm

DESCRIPTION

    The **utmhadm** command provides a way to administer Sun Ray server multihead multihead groups. The information that **utmhadm** displays and that is editable is stored in the Sun Ray administration database.

    The **utmhadm** operations that only display information may be run by any user. Operations that change data must be run as superuser.

OPTIONS

    The following options are supported.

    -**a** *groupname*
        Creates a new multihead group having identifier *groupname*. This name must be unique and not already exist on the system.

    -**d** *groupname*
        Removes the multihead group for the specified *groupname*.

    -**e** Populates the system multihead group database with input data of the format produced by **-o**, from standard input.

    -**f** *filename*
        Specify a *filename* for use with **-e** or **-o** instead of standard input or output.

    -**g** *COLSxROWS*
        Specifies the geometry of the multihead group in the form *COLSxROWS*. This number of columns and rows must not exceed the maximum number allowed and must match the number of DTUs specified with **-l**. This option can only be used with **-a**.

    -**h** Prints the usage message.

    -**l** *CID1*, *CID2*,..., *CIDn*
        Specifies the DTU canonical identifiers when creating a group. A canonical identifier has the form IEEE802.*nnnnnnnnnnnn* or *nnnnnnnnnnnn* (the 12-digit hexadecimal MAC address of the DTU) and the list must be comma-separated. The identifiers must be specified in row-major order. The maximum number of DTUs allowed is 16.

    -**o** Dumps all system configured multihead group data, in comma-separated format, to standard output. Intended for subsequent use with **-e**.

    -**p** *primaryCID*
        Identifies which in the list of canonical identifiers, specified with **-l**, is designated as the primary DTU within the group. The primary is repeated in the list specified by **-l**. This option can only be used with **-a**.

    When no options are provided, **utmhadm** lists information about all multihead groups configured on the system.

EXAMPLES

    Example 1: This command list all DTUs that are in the multihead group:

    % **/opt/SUNWut/sbin/utmhadm tera**

Here is sample output:

```
Multihead Group   Geometry        CIDs
----------------- ----------------- ---------------------------------
tera            geometry=2x1     IEEE802.080020b538dc (P)
                                 IEEE802.080020b56e2d
```

Example 2: This command creates a terminal group having two terminals with the first one being the primary:

 # /opt/SUNWut/sbin/utmhadm -a srgroupA -g 2x1 -p IEEE802.080020b0562f -l IEEE802.080020b0562f,IEEE802.080020b64574

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

utxconfig(1), *Sun Ray Server Software Administrator's Guide*

NAME

    utmhconfig - Sun Ray mulithead GUI configuration utility.

SYNOPSIS

    /opt/SUNWut/sbin/utmhconfig

DESCRIPTION

    The **utmhconfig** utility allows the administrator to list, add, or delete multihead groups easily. The initial screen lists any existing multiheaded groups and allows the administrator to select those to delete. The utility can also be used to create a new group. To do this, the administrator starts the utility on the Sun Ray DTU that is to become the "primary" of the group (it has the keyboard, mouse, and all the devices for the group). The administrator selects "Create New Group" and follows the instructions in the wizard to identify all of the terminals in the new multihead group. The administrator run the utmhconfig command as superuser and must have a recognized smart card available.

OPTIONS

    There are no options for **utmhconfig**

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

    utmhadm(1M), **utxconfig(1)**

NAME

 utmhscreen - Sun Ray multihead GUI screen display tool.

SYNOPSIS

 /opt/SUNWut/lib/utmhscreen [**-l**]

DESCRIPTION

 The **utmhscreen** tool provides a window showing the respective displays location in the multihead group. The display showing the widow is highlighted in white, while the other displays are darkened. The window is located in the upper right corner of the display.

 This tool is automatically launched for users during the X server startup process (session creation). If the X server is not running in a multihead environment, the tool immediately exits.

OPTIONS

 utmhscreen accepts the following option:

 -l Indicates to **utmhscreen** that it is being auto launched by the windowing system. The use of this option is beyond the scope of this manual.

RESOURCES

 The tool understands all of the core X Toolkit and Motif resource names and classes as well as:

 enableAutoLaunch (class EnableAutoLaunch)
   Specifies whether or not **utmhscreen** should be launched automatically during X session startup. The default is "true".

EXAMPLES

 Example 1: To disable automatic launching of utmhscreen for a user, set the following X resource in their **$HOME/.Xdefaults** file:

 Utmhscreen*enableAutoLaunch: false

ATTRIBUTES

 See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta
Interface Stability	Evolving

```
tab(!) box;
cw(2.750000i)| cw(2.750000i)
lw(2.750000i)| lw(2.750000i).
Attribute Types!Attribute Values
Availability!SUNWuta
Interface Stability!Evolving
```

SEE ALSO

 utmhadm(1M). **utmhconfig(1M)**, **utxconfig(1)**

NAME

utmountd - Sun Ray Mass Storage media mounter daemon

SYNOPSIS

/opt/SUNWut/bin/utmountd -m [-D *debug-level*] [-p *poll_interval*] [-t *max_threads*]

DESCRIPTION

utmountd performs mount and umount operations on Sun Ray mass storage devices managed by utstoraged(1M). Slices or partitions with Solaris recognizable filesystems are mounted on a directory inside the current sessions mount directory, $DTDEVROOT/mnt.

Error messages from utmountd are logged to**/var/opt/SUNWut/log/utmountd.log**.

OPTIONS

The following options are supported.

-D *debug-level*

Debug mode. Use is beyond the scope of this document. If a debug level is set, debug messages are sent to stderr.

-p *poll_interval*

Minimum time in seconds between polling removable media devices for media insertion events

-t *max_threads*

Maximum number of simultaneous service threads allowed. The more active threads, higher the load on the system. A lower number of threads mean slower response times to mounting or unmounting devices if the daemon is overloaded with work.

FILES

The following files are used:

$DTDEVROOT/mnt

A link pointing to the directory containing mount points

ENVIRONMENT VARIABLES

DTDEVROOT points to a temporary directory associated with the user's session. The directory's lifetime is equal to the lifetime of the login session. It is removed, along with its contents, when the user logs out.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutsto

SEE ALSO

utdiskadm(1M), **utstoraged(1M)**, **utdisk(7D)**, **fstyp(1M)**

NAME

    utparalleld - Sun Ray printer service daemon.

SYNOPSIS

    /opt/SUNWut/lib/utparalleld [-**D** *debug-level*] [-**o** *optroot*] [-**r**]

DESCRIPTION

    utparalleld provides printer support for Sun Ray DTUs. **utparalleld** supplies driver services for all USB parallel adaptors and USB printers that comply with the USB printer class.

    utparalleld uses theSun Ray parallel I/O driver to provide applications the same interface as standard workstation parallel ports, such as **/dev/ecpp** or **/dev/bpp**. Applications such as the **lp**(1) daemon can use device nodes that are provided by **utparalleld**.

    When a parallel adaptor or a USB printer is attached to a DTU, **utparalleld** creates device nodes in the **$UTDEVROOT/devices** directory. A user can use the device link **$UTDEVROOT/dev/printers/***printer-name* to access a specific printer.

OPTIONS

    The following options are supported.

    -**D** *debug-level*
        Debug mode. Use is beyond the scope of this document.

    -**o** *optroot*
        Use *optroot* as the parallel service's root directory for device node creation. The default value is **/tmp/SUNWut**. *optroot* should be the same directory as the *optroot* directory used by **utdevmgrd**(1M).

    -**r** Automatically restart the printer service daemon if it exits. With this option, the printer service daemon creates two processes: a child that performs all the actual work, and a parent monitoring process. The parent process restarts a child if the previous one exits.

FILES

    The following files are used:

    $UTDEVROOT**/dev/printers**
        The directory containing links to parallel device names for each DTU..

ENVIRONMENT VARIABLES

    UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

    utdevmgrd(1M), utseriald(1M)

NAME

    utpolicy - Sun Ray authentication manager policy management command.

SYNOPSIS

    /opt/SUNWut/sbin/utpolicy -**a** [-**g**] [-**p**] [-**r** *type*] [-**s** *type*] [-**z** *type*]

    /opt/SUNWut/sbin/utpolicy

    /opt/SUNWut/sbin/utpolicy -**h**

    /opt/SUNWut/sbin/utpolicy -**g**

    /opt/SUNWut/sbin/utpolicy -**k** {*card|pseudo*|both}

    /opt/SUNWut/sbin/utpolicy -**d**

    /opt/SUNWut/sbin/utpolicy -**M**

    /opt/SUNWut/sbin/utpolicy -**S**

DESCRIPTION

    The **utpolicy** command simplifies and writes the policy configuration of the Sun Ray authentication manager, **utauthd(1M)**.

OPTIONS

    The following options are supported.

    POLICY SETTING

    The specified Policy Setting arguments completely replace the current active authentication policy. Only arguments that are specified become active. Policy Setting and Card Reader Assignment arguments can be specified together

    -**a** This option, followed by valid Policy Setting, or Card Reader Assignment arguments, applies these arguments to the active authentication policy for the system. This option is not valid by itself.

    -**d** Disable the Exit menu option from the Non-smart card mobility GUI. The Exit option is enabled by default.

    -**g** Turn on session selection within a server group. Allows the user to select on which server the user's session is run.

    -**k** {card|pseudo|both}
        Enables Controlled Access Mode (CAM) for the specified session type. Selecting *card* enables CAM for card sessions, selecting *pseudo* for terminal sessions; and selecting **both** enablesCAM for all types of sessions. For functionality, at least one of the -**r**, -**s**, or -**z** options must be invoked with the same argument as the -k option. The -k option is not considered until the **utconfig** application has configured for Controlled Access Mode.

    -**M** Enable non-smart card mobile sessions.

    -**m** Enable multihead session capability, allowing multiple terminals to act as display devices for a single user session.

    -**p** This option changes the behavior of the self-registration application so that it does not require a name and password before registering a token. Note that the self-registration application only verifies the name and password. They are not stored.

    -**r {card|pseudo|both}**
        Specify the token types that must be registered in the administrative database in order to be granted access to a login screen. Policy looks up and uses token database entry

    -**S** *smartcard_type*
        Specify the smart card type that should cause the smart card login GUI (utsclogin) to be invoked when a card of the type in the list is inserted into the DTU. This is the smart card token type (without the token ID) as provided by the Sun Ray software. If more than one smart card type is to be specified, multiple -S options must be specified.

    -**s {card|pseudo|both}**
        Specify the token types that will be presented with a registration screen if they do not have an entry in the administrative database. Policy allows self-registration of tokens.

-z {card|pseudo|both}
　　　Specify the token types that do not require an entry in the administrative database in order to be granted access to a login screen. Policy grants access to tokens without database entry.

With the **-h** option, the **utpolicy** command prints out the usage message.

With no options, the **utpolicy** command prints out the policy in effect.

The following options are RESERVED for use by the Sun Ray Server Software and should not be used:

-G, -P, -Q, -b, -f, -l, -u, -x, +x

EXAMPLES

Example 1: This command configures the policy so that all access via smart card requires a valid administrative database entry before access is granted. If a database entry has not been created for a smart card, then a registration session is presented on the DTU. If no smart card is used, then the normal login screen is presented.

 # /opt/SUNWut/sbin/utpolicy -a -r card -s card -z pseudo

FILES

The following files are used:

• **/etc/opt/SUNWut/policy/utpolicy**
　　　The policy configuration file

• **/etc/opt/SUNWut/auth.props**
　　　Sun Ray authentication manager's configuration file

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

utauthd(1M), **utreader(1M)**, **utrestart(1M)**, **auth.props(4)**, **pam(3PAM)**

NAME

      utpreserve - Sun Ray configuration file preservation utility.

SYNOPSIS

      /cdrom/cdrom0/utpreserve [**-d** *preserve-directory*]

DESCRIPTION

      The **utpreserve** command stops Sun Ray services, terminates user sessions, and saves existing Sun Ray server configuration data into a compressed tar file, **/var/tmp/SUNWut.upgrade/preserve_3.1.tar.gz**.

OPTIONS

      The following option is supported.

      -d *preserve-directory*
        Save the compressed tar file into the *preserve-directory*.

SEE ALSO

      utinstall(1M), **utconfig(1M)**

NAME

>   **utpw** - Sun Ray administration password change utility.

SYNOPSIS

>   **/opt/SUNWut/sbin/utpw**

DESCRIPTION

>   The **utpw** command changes the Sun Ray administrator password (also known as the "UT admin" password). This password is entered by the administrator when logging into the Administration Tool and is used to make a privileged connection to the LDAP server.

>   **utpw** changes the password both in the administration database, and the password file on the local server.

>   In a failover group, **utpw** also affects the administration database of the secondary servers, but only the password file on the local server. The administrator must log into the secondary servers and run **utpw** on them to change the password files.

OPTIONS

>   There are no options for **utpw**.

EXAMPLES

>   **Example 1: This command changes the administration password:**

>   **# /opt/SUNWut/sbin/utpw**
>   Enter new UT admin password:
>   Re-enter new UT admin password:
>   Enter old UT admin password:

>   Changing LDAP password...
>   Done.
>   Changing password file...
>   Done.

FILES

>   The following files are used:

>   • **/etc/opt/SUNWut/utadmin.pw**

>   • **/etc/opt/SUNWut/utadmin.conf**

EXIT STATUS

>   The following exit values are returned:

>   **0** Success

>   1 Error

ATTRIBUTES

>   See **attributes(5)** for descriptions of the following attributes:

> | Attribute Type | Attribute Value |
> | --- | --- |
> | Availability | SUNWuta |

SEE ALSO

>   **utdesktop(1M)**, **utuser(1M)**, *Sun Ray Server Software Administrator's Guide*

NOTES

>   The **-f** option has been deprecated. Use **utpw** instead. If you use the **-f** option, you must supply the Sun Ray administration password though there is no prompt for it displayed.

NAME

utquery - query Sun Ray Desktop Units current parameter values

SYNOPSIS

/opt/SUNWut/sbin/utquery [-d] *IP_address*

/opt/SUNWut/sbin/utquery -h

DESCRIPTION

The **utquery** command allows administrators to query the current parameter values on a Sun Ray Desktop Unit (DTU). The *IP_address* may specify the network IP address of a single DTU to query, a subnet broadcast address to query all DTUs on the subnet or the broadcast address to query all DTUs associated with this Sun Ray server.

The utquery command can be used to aid in diagnosing problems when a DTU is unable to successfully connect to an authentication manager or when a Multihead group is redirected outside their "home" failover group.

OPTIONS

The following options are supported.

-d Report the DHCP parameters the DTU obtained at boot-up.

-h Print the usage.

EXAMPLES

Example 1: To display the DHCP values for the DTU at IP address 129.146.58.182

% **utquery -d 129.146.58.182**

The following is an example of the output for the command above:

```
terminalID=080020d15f23
        terminalIPA=10.6.102.224
        model=CoronaP3
        Subnet=255.255.255.0
        Router=10.6.102.1
        Broadcst=10.6.102.255
        LeaseTim=86400
        DHCPServer=10.6.102.3
        INFORMServer=10.6.102.113
        AuthSrvr=10.6.102.113
        AuthPort=7009
        LogHost=10.6.102.113
        FwSrvr=10.6.102.113
        NewTVer=3.1_12,REV=2005.04.13.06.03
        parmsVersion=3.1_12,REV=2005.04.13.06.03
        parmsBarrier=310
        parmsBarrierLevel=310
        parmsServers=
        parmsSelect=default
        currentAuth=10.6.102.111
        currentFW=3.1_12,REV=2005.04.13.06.03
        currentBarrierLevel=310
        dnsList=129.147.5.51,129.145.155.32,129.145.155.42
        dname=SFBay.Sun.COM
```

In the output above, there can be a number of values with *Barrier* in the name, and understanding the differences is subtle. In general,the *Barrier* value is the value associated with a firmware file, and its corresponding version. So, *currentBarrier* is the barrier value for the current firmware running on the DTU, and *parmsBarrier* is the barrier level of the file residing on the firmware server. In the absence of any *BarrierLevel* values, the firmware file on the server will not be loaded if its cc value is less than the currently running firmware. For example, in the output above, the *currentBarrier* is 310, so any attempt to load SRSS 2.0 firmware, which has a barrier level of 200, would fail. In order to permit the forced loading of firmware with a lower barrier level, the *BarrierLevel* values are used. The *BarrierLevel* is either the *currentBarrier* value of the resident firmware, the *BarrierLevel* value provided from DHCP, or the *parmsBarrierLevel*

obtained from the *.parms* file. Later values in this list take precedence if present. It is the *currentBarrierLevel* that is compared with the barrier level of the server's file to determine if the download is permitted or not.

Example 2: To display the DHCP values for all DTUs on subnet 129.146.58:.

utquery -d 129.146.58.255

Example 3: To display the DHCP values for all DTUs on this server:

utquery -d 255.255.255.255

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

NOTES

The utquery command will only report on DTUs that are operating with firmware version 2.0 or newer.

The utquery command will only report on DTUs that have successfully acquired DHCP parameters from a DHCP server.

The**NewTVer** value reported is the firmware revision level supplied by DHCP, which is used to determine whether a firmware upgrade is required. The **currentFW** value is the firmware revision level of the current firmware running in the Sun Ray device.

The **AuthSrvr** and **AltAuth** values reported are those provided by DHCP parameters at DTU boot time. The **currentAuth** value is the IP address of the Sun Ray server to which the Sun Ray is currently connected.

In firmware version 3.1 and newer, additional values obtained via TFTP from the *.parms* file are also printed. (See **utfwadm(1m)**.) If **parmsVersion** is defined, it overrides **NewTVer**.

The **utquery** command with a broadcast address does not work on remote subnets because most routers do not forward the **broadcast-address** packets.

NAME

      utrcmd - Sun Ray remote administration utility.

SYNOPSIS

      /opt/SUNWut/lib/utrcmd [**-n**] *hostname command* [*args*]

DESCRIPTION

      The **utrcmd** program provides a way to run certain Sun Ray administration commands remotely. It contacts the **in.utrcmdd**daemon on the remote *hostname* and executes the specified *command* with the specified arguments *args* (if any).

      utrcmd copies its standard input to the specified command, the standard output of the command to **utrcmd**'s standard output, and the standard error of the command to **utrcmd**'s standard error. Interrupt, quit, and terminate signals are propagated to the specified command; **utrcmd** terminates normally when the command does.

OPTIONS

      The following option is supported.

      -n Redirect the input of **utrcmd** to **/dev/null**. This option prevents interactions between **utrcmd** and the shell which invokes it. For example, if you are running **utrcmd** and invoke a **utrcmd** in the background without redirecting its input away from the terminal, it will block even if no reads are posted by the specified command. The **-n** option prevents this behavior.

USAGE

      Official hostnames or nicknames may be given as the *hostname*.

      The **utrcmd** and **in.utrcmdd** programs use the Sun Ray failover group configuration to perform a set of checks before allowing the specified command to proceed.

      The program **utrcmd** runs with set-user-ID permission of root or superuser. The **utrcmd** command proceeds only if all of the following are true (on the initiating system):

      • The user's real user-ID is superuser, or the user has membership rights in the **utadmin** group.

      • The **/etc/opt/SUNWut/auth.props** file is owned by superuser and is not writable by anyone other than superuser.

      • The **gmSignatureFile** property of **auth.props** specifies a group signature file.

      • The group signature file exists and is owned by superuser and is not readable, writable, or executable by anyone other than superuser.

      • The group signature file is at least 8 bytes long and has similar content diversity characteristics as required by **passwd**(1).

      • The **utrcmd/tcp** service is enabled.

      The **in.utrcmdd** program will accept the connection only if all of the following are true on the remote system:

      • The **utrcmd/tcp** service is enabled and matches the configuration on the initiating system.

      • The **in.utrcmdd** program is enabled in **/etc/inetd.conf**.

      • The **utadmin** group is configured on the system.

      • The **/etc/opt/SUNWut/auth.props** file is owned by superuser and is not writable by anyone other than superuser.

      • The **gmSignatureFile** property of **auth.props** specifies a group signature file.The group signature file exists and is owned by superuser, and is not readable, writable, or executable by anyone other than superuser.

      • The group signature file is at least 8 bytes long and has similar content diversity characteristics as required by **passwd**(1).

      If the connection is accepted, the **utrcmd** program begins a challenge-response handshake with the **in.utrcmdd** program, using the contents of the group signature file to sign messages (without revealing the contents of the signature file). Either **utrcmd** or **in.utrcmdd** will reject the transaction if the handshake fails. The specified command will not be run if the contents of the group signature files on the two systems differ.

      Finally, **in.utrcmdd** rejects the specified command if it is not recognized. Specified commands always run in group **utadmin**.

The following commands are allowed and are always run in group **utadmin**:

- **/opt/SUNWut/sbin/utreplica**
- **/opt/SUNWut/sbin/utpolicy**
- **/opt/SUNWut/sbin/utfwadm**
- **/usr/sbin/dhtadm**
- **/usr/sbin/pntadm**
- **/opt/SUNWut/lib/utauthd**
- **/etc/init.d/utsvc**
- **/opt/SUNWut/sbin/utsession**
- **/opt/SUNWut/sbin/utreader**
- **/opt/SUNWut/sbin/utrestart**

Impact of disabling **utrcmd**:

utrcmd is used to run certain Sun Ray administration commands on remote systems. Hence disabling **utrcmd** means that the following functionality will not work as expected:

- Admin GUI
- Restart Services
- Sun Ray Session management
- CLI
- **/opt/SUNWut/sbin/utreplica** to setup a failover group
- **/opt/SUNWut/sbin/utdssync** to convert and synchronize the Sun Ray data store
- **/opt/SUNWut/sbin/utfwsync** to synchronize firmware levels

EXAMPLES

Example 1: This command lists the configured token readers on a remote Sun Ray server

   # **/opt/SUNWut/lib/utrcmd sun5 /opt/SUNWut/sbin/utreader**

FILES

The following files are used:

- **/etc/hosts**
  Internet host table

- **/etc/group** Group file

- **/etc/inet/services**
  Internet services table

- **/etc/inetd.conf**
  Internet services daemon configuration table

- **/etc/opt/SUNWut/auth.props**
  Sun Ray authentication properties file

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      utauthd(1M), **inetd(1M)**, **group(4)**, **auth.props(4)**, **hosts(4)**, **nsswitch(4)**, **passwd(1)**, **rsh(1)**, **attributes(5)**

NOTES

      utrcmd works in a manner similar to **rsh(1)**. However, it imposes multiple restrictions to maintain system security.

NAME

   utreader - Sun Ray utility for configuring a terminal as a token reader.

SYNOPSIS

   /opt/SUNWut/sbin/utreader

   /opt/SUNWut/sbin/utreader -a

   /opt/SUNWut/sbin/utreader -c

   /opt/SUNWut/sbin/utreader -d

   /opt/SUNWut/sbin/utreader -h

DESCRIPTION

   The **utreader** command is used to configure Sun Ray terminals as token card readers.

   When utreader is run with no options, the list of terminals configured as token readers is displayed. The token reader list can be viewed by all users, but any changes made to the configuration must be made by the super-user.

   Note:

   Whenever a change to a the token reader configuration been made, a restart of services has to be initiated in order for the new configuration to take effect.

OPTIONS

   The following options are supported.

   -a *<terminalId>*
      Configures the terminal with the specified terminal ID as a token reader.

   -c Unconfigures all token readers and restores them as normal Sun Ray terminals.

   -d *<terminalID>*
      Unconfigures the token reader with the specified terminal ID and restores it as a normal Sun Ray terminal.

   -h Displays the usage message.

EXAMPLES

   Example 1: This command configures a terminal as a token reader where the terminal ID is AAAABBBBCCCC.

      % **utreader -a AAAABBBBCCCC**

   Example 2: This command unconfigures all token readers.

      utreader -c

EXIT STATUS

   The following exit values are returned:

   0 Success

   1 Failure

ATTRIBUTES

   See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

NAME

    utreplica - LDAP replication utility for Sun Ray servers.

SYNOPSIS

    /opt/SUNWut/sbin/utreplica **-p [-a** | **-d]***secondary-server1* [*secondary-server2* ... "]"

    /opt/SUNWut/sbin/utreplica **-s** *primary-server*

    /opt/SUNWut/sbin/utreplica **-l**

    /opt/SUNWut/sbin/utreplica **-u**

    /opt/SUNWut/sbin/utreplica **-z [***port#*]

DESCRIPTION

    The **utreplica** command configures the Sun Ray LDAP server to replicate data from the primary server to each secondary server in a failover group. The command must be run with superuser privileges on the Sun Ray server to be configured.

OPTIONS

    The following options are supported.

    -l List the current failover administration status.

    -p [-a|**-d]** *secondary-server*
        Configure the primary server. *secondary-server* is the host name of the secondary server. List all seondary servers within the failover group.
        -a is used to add the specified secondary servers to the current list of secondary servers.
        -dis used to delete the specified secondary servers from the current list of secondary servers.

    -s *primary-server*
        Configure a secondary server. *primary-server* is the host name of the primary server.

    -u Unconfigure this Sun Ray server for LDAP database replication.

    -z[*port#*]
        Update the port number with the one specified for the datastore service. If run on the primary server without specifying the port number, it simply updates all the necessary configuration files on the primary with the default port for the datastore service. If run on the secondary server without specifying the port number, it resyncs all necessary configuration files on the secondary server with the port number currently configured on the primary server.

USAGE

    utreplica is used only on Sun Ray servers in a failover group. Configure the primary Sun Ray server first, then the secondary servers.

FILES

    The following files are configured on the primary Sun Ray server:

      • **/etc/opt/SUNWut/srds/current/utdsd.conf**

      • **/etc/opt/SUNWut/srds/current/utdsd.ini**

      • **/etc/services**

      • **/etc/opt/SUNWut/utadmin.conf**

    The following files are configured on the secondary Sun Ray server:

      • **/etc/opt/SUNWut/srds/current/utdsd.conf**

      • **/etc/opt/SUNWut/utadmin.conf**

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      utconfig(1M)

NOTES

      To replicate the LDAP information properly, all Sun Ray servers in the failover group must have the same group signature.

      Use of the **-p**, **-s**, and **-u** options on a Sun Ray server interrupts all active sessions on that server.

      Please note that "interrupt" in this context means that, while the screen will flash for approximately 30 seconds, all current sessions are preserved, not destroyed. Active users may need to remove and re-insert their smart cards to return to their sessions. If a CDE login screen is presented, the user should select Options/Reset Login Screen to return to an existing session.

NAME

      utresadm - manage explicit monitor resolution settings for Sun Ray.

SYNOPSIS

      /opt/SUNWut/sbin/utresadm

      /opt/SUNWut/sbin/utresadm -a -**c** *CID* -**t** *token resname*

      /opt/SUNWut/sbin/utresadm -d -**c** *CID* -**t** *token*

      /opt/SUNWut/sbin/utresadm -p [-**c** *CID*] [-**t** *token*]

      /opt/SUNWut/sbin/utresadm -o

      /opt/SUNWut/sbin/utresadm -i

DESCRIPTION

      The **utresadm** command allows an administrator to create, delete, and view an explicit resolution for a monitor attached to a Sun Ray desktop unit. Resolutions established through **utresadm** take precedence over any resolution discovered through a DDC exchange between the Sun Ray DTU and the monitor. Resolutions may be specified for a particular combination of Sun Ray DTU and access token, for a particular Sun Ray independent of access token, and for all Sun Ray DTUs controlled by servers in a failover group. In cases where multiple configuration records might apply to a given session, then the most specific match is applied.

      utresadm with no options or arguments shows the resolution that has been configured for the current access token in the current Sun Ray DTU.

      Available resolution names are reported by **utresdef(1M)**.

OPTIONS

      The following options are supported.

      -**a** establishes the resolution *resname* as the preferred resolution for the given *CID* and *token*. If *token* is specified as **default** then *resname* becomes the preferred resolution for all tokens at the given *CID* for which no explicit resolution has been configured. If both *CID* and *token* are specified as **default** then *resname* becomes the preferred resolution for all CIDs and tokens for which no explicit resolution has been configured. This option is available only to the superuser. Changes become effective only after services are restarted.

      -**d** deletes an explicit resolution previously configured for the given *CID* and *token*. This option is available only to the superuser. Changes become effective only after services are restarted.

      -**p** shows the explicit resolution configured for the given *CID* and *token*. If no token is specified then all configuration records for the given *CID* are shown. If no *CID* is specified then all configuration records for the given *token* are shown. If neither *CID* nor *token* is specified then all configuration records are shown.

      -**o** shows all known resolution configuration records in a format suitable for consumption by **utresadm -i**.

      -**i** reads (from standard input) a list of resolution configuration records in the form emitted by utresname -o and configures the corresponding resolutions. This option is available only to the superuser.

      -**c** the canonical ID of the Sun Ray desktop unit, or the word **default** if this operation is to apply to all desktop units.

      -**t** the access token for the session, or the word **default** if this operation is to apply to all tokens.

OPERANDS

      resname
         The desired resolution for the given *token* at the given *CID*.

EXIT STATUS

      The following exit values are returned:

      0 if the desired activity was completed without error.

      1 if the command terminated because of a command line syntax problem.

      2 if the Sun Ray data store was inaccessible.

      3 if the resolution definition (provided on standard input) was unacceptable.

FILES

> **/etc/opt/SUNWut/utadmin.conf**
>
> **/etc/opt/SUNWut/utadmin.pw**

ATTRIBUTES

> See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

> **utsettings(1)**, **utset(1)**, **utresdef(1M)**, **utconfig(1M)**

NOTES

> **utresadm** provides no assurance that the configured resolution timing can be understood by the monitor attached to a Sun Ray unit; it is possible to configure a resolution that can not be understood by the monitor. **utsettings(1)** provides a safer interactive means of establishing an explicit resolution for the current access token on the current Sun Ray unit.
>
> **utresadm** is available only after **utconfig(1M)** has been run to activate the Sun Ray data store.
>
> As with anything that must go through the LDAP database, there is a slight lag between the time an update is applied on the primary server and its propagation to the secondary server or servers; however, the length of the interval would normally be measured in seconds or fractions of a second.
>
> In addition, once a Sun Ray session, which can persist across many logins, is attached to a given DTU, the monitor timing persists for the life of the session and can be changed only by modifying the timing explicitly from within that session while it remains attached to that DTU.External activities, such execution of **utresadm** -**a** or **utresadm** -**d** after the session has been associated with the DTU, have no effect on any existing association. However, **utrestart** -**c** destroys all existing sessions and thereby makes all previously existing associations obsolete. As DTUs reconnect after the restart and are granted new sessions, the newly configured timings are applied to the new sessions as they are created.

NAME

  utresdef - manage monitor resolution definitions for Sun Ray.

SYNOPSIS

  /opt/SUNWut/sbin/utresdef

  /opt/SUNWut/sbin/utresdef *resname*

  /opt/SUNWut/sbin/utresdef -a [-**c** *comment*] *dimensions resname*

  /opt/SUNWut/sbin/utresdef -d *resname*

  /opt/SUNWut/sbin/utresdef -o

  /opt/SUNWut/sbin/utresdef -i

  /opt/SUNWut/sbin/utresdef -h

DESCRIPTION

  The **utresdef** command allows an administrator to create, delete and view resolution definitions for monitors attached to Sun Ray DTUs.

  Resolutions are associated with specific Sun Ray units by **utresadm(1M).** Users may configure resolutions for their own personal access tokens through **utsettings(1)** or **utset(1)**.

  Available resolution names and their definitions are reported by **utresdef(1M)** when nothing but the command is entered. If a *resname* is specified without any options, its definition is reported.

OPTIONS

  The following options are supported.

  -**a** defines the resolution *resname*. This option is available only to the superuser.

  -**d** deletes the resolution definition named *resname*. This option is available only to the superuser.

  -**o** shows all known resolution definitions in a format suitable for consumption by **utresdef -i**.

  -**i** reads (from standard input) a list of *resname*s, *comment*s, *dimension*s and resolution definitions in the form emitted by **utresdef -o** and configures the corresponding resolutions. This option is available only to the superuser.

  -**c** a **comment** explaining the nature and purpose of the resolution being defined.

  -**h** Displays the usage message for this command.

OPERANDS

  dimensions
    the dimensions, in the form *width*x*height*, of the on-screen display after this resolution definition has been applied to the Sun Ray desktop unit. The dimensions are often implied by *resname* but this is not guaranteed. *resname* may take some unconventional form (it might perhaps be named for a particular model of monitor) from which the dimensions can not be inferred, so the resulting dimensions must be provided explicitly.

  resname
    the name of the resolution being defined, deleted or displayed. *resname* is conventionally given in the form *width*x*height*@*rate* (e.g. 640x480@60) but this convention is not enforced.

EXIT STATUS

  The following exit values are returned:

  0 if the desired activity was completed without error.

  1 if the command terminated because of a command line syntax problem.

  2 if the Sun Ray data store was inaccessible.

  3 if the resolution definition (provided on standard input) was unacceptable.

FILES

  /etc/opt/SUNWut/utadmin.conf

  /etc/opt/SUNWut/utadmin.pw

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

utsettings(1), **utset(1)**, **utresadm(1M)**, **utconfig(1M)**

NOTES

utresdef is available only after **utconfig(1M)** has been run to activate the Sun Ray data store.

NAME

      utrestart - Sun Ray utility for resetting and restarting services.

SYNOPSIS

      /opt/SUNWut/sbin/utrestart

      /opt/SUNWut/sbin/utrestart -c

      /opt/SUNWut/sbin/utrestart -h

DESCRIPTION

      The **utrestart** command is used for resetting and restarting Sun Ray services. It replaces the **utpolicy -i** option which has been deprecated in 2.0.

      utrestart can only be run by the super-user.

      The utrestart command without options causes a "warm" restart: Sun Ray services are restarted and existing sessions are preserved.

OPTIONS

      The following options are supported.

      -**c** Restarts Sun Ray services. Sessions will be lost.

      -**h** Prints the usage for this command.

EXAMPLES

      Example 1: This resets services.

      # **/opt/SUNWut/sbin/utrestart**

      Example 2: This restarts services.

      # **/opt/SUNWut/sbin/utrestart** -c

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

NAME

   utseriald - Sun Ray DTU serial services daemon.

SYNOPSIS

   /opt/SUNWut/bin/utseriald [-**D** *debug-level*] [-**o** *optroot*] [-**r**]

DESCRIPTION

   utseriald provides serial support for Sun Ray DTUs through driver services for USB serial adaptors. It also provides support for the built-in, non-USB serial ports on some Sun Ray models. For a list of supported adaptors, see the web site:

   http://www.sun.com/sunray

   utseriald uses the Sun Ray serial I/O driver on to provide applications the same interface as standard workstation serial ports, such as **/dev/term/a** or **/dev/term/b** on Solaris.

   When a USB serial adaptor is attached to a DTU, **utseriald** creates device nodes in the **$UTDEVROOT/devices** directory. Sun Ray models with permanent, built-in serial ports will also have corresponding device nodes created in this directory.

   When a serial adaptor is attached to a DTU, **utseriald** creates device nodes in the **$UTDEVROOT/devices** directory.

   A user can use the device link **$UTDEVROOT/dev/term/***terminal-name* to access the serial ports on the Sun Ray DTU.

OPTIONS

   The following options are supported.

   -**D** *debug-level*
      Debug mode. Use is beyond the scope of this document.

   -**o** *optroot*
      Use *optroot* as the serial service's root directory for device node creation. The default value is **/tmp/ SUNWut**. *optroot* should be the same directory as the *optroot* directory used by **utdevmgrd**(1M).

   -**r** Automatically restart the serial service daemon if it exits. With this option, the serial service daemon creates two processes: a child that performs all the actual work, and a parent monitoring process. The parent process restarts a child if the previous one exits.

FILES

   The following files are used:

   $UTDEVROOT/**dev/term**
      The directory containing links to serial device names for each DTU.

ENVIRONMENT VARIABLES

   UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

ATTRIBUTES

   See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

   utdevmgrd(1M), **utparalledl(7D)**

NAME

      utsession - List and manage the Sun Ray sessions on the local Sun Ray server.

SYNOPSIS

      /opt/SUNWut/sbin/utsession -**p** [-**x**] [-**d** *disp#*] [-**u** *unix*] [-**t** *token*] [-**n** *name*]

      /opt/SUNWut/sbin/utsession { -**k** | -**s** } [-**a**] [-**x**] { [-**d** *disp#*] [-**u** *unix*] [-**t** *token*] [-**n** *name*] }

      /opt/SUNWut/sbin/utsession -**r** [-**a**] [-**x**] [-**d** *disp#*] [-**u** *unix*] [-**t** *token*] [-**n** *name*]

      /opt/SUNWut/sbin/utsession -**h**

      /opt/SUNWut/sbin/utsession -**l**

DESCRIPTION

      The first synopsis (**-p**) is used to print Sun Ray sessions for the specified user or token on the current server. For Solaris this is both active and suspended sessions. When the **-u**, **-t**, **-n**, and **-d** options are not used, **utsession** prints all of the Sun Ray sessions on the current server.

      When listing the Sun Ray sessions, **utsession** also lists the state for each session:

      D Disconnected — The session is currently not attached to any Sun Ray. The session is considered connected if this flag is omitted.

      I Idling — A **dtlogin** session that is currently waiting for user login (ie. **dtgreet**). User has already logged into the **dtlogin** session if the flag is omitted.

      S Suspended — Solaris only. The session is currently suspended. The session is considered running if this flag is omitted.

      The second synopsis (**-k** | **-s**) is used to kill (both Linux and Solaris) or suspend (Solaris only) Sun Ray sessions on the current server. At least one of the **-d**, **-u**, **-t**, or **-n** options must be specified. Unless the **-a** option is specified, more than one session matching the specified criteria will return an error.

      When suspending a session, **utsession** uses **dterror.ds(1)** to display a message on the user's screen to indicate that the session is being suspending by the administrator.

      The third synopsis (**-r**) is used to resume Sun Ray Solaris sessions on the current server. If **-d**, **-u**, **-t**, or **-n** option is not specified, **utsession** will resume all the suspended Sun Ray sessions on the current server. Unless the **-a** option is specified, more than one session matching the specified criteria will return an error.

      The fourth synopsis (**-h**) displays the usage of this command.

      Note:

      This command must be run as root.

OPTIONS

      The following options are supported:

      -**a** Apply the operation to all matching sessions if more than one matches the search criteria. If **-a** is not specified, multiple matching sessions return an error.

      -**d** *disp#*
      Specify the X display number for search.

      -**h** Display the usage of this command.

      -**k** Kill the sessions matching the search criteria. You must also specify at least one of the **-d**, **-u**, **-t**, or **-n** options.

      -**l** Gives detailed security status, normally a list of terminal CIDs currently used by the session, listed under the terminalCIDs key. For multihead sessions, it also displays the primary terminal CID under the primaryCID key.

      -**n** *name*
      Specify the registered Sun Ray username for search. Sessions belonging to users matching the username are listed. It is a case sensitive, exact match.

      -**p** Print the sessions belonging to the specified user or token. If any matching criteria is specified, it will include the matched suspended sessions.

      -**r** Make the sessions matching the search criteria active. **utsession** will resume all specified Sun Ray sessions if **-d**, **-u**, **-t**, or **-n** options are not specified. This option will be deprecated in the 3.0 release.

-**s** Suspend the sessions matching the search criteria. You must also specify at least one of the **-d**, **-u**, **-t**, or **-n** options. This option will be deprecated in the 3.0 release.

-**t** *token*

Specify the Sun Ray token for search. The token is in one of the following forms:

- Raw token form for unregistered users (**MicroPayflex.####**)

- Pseudo token form for terminal users (**pseudo.***macaddr*)

- Logical token form for registered users (**user.#####**)

- Mobile token form for NSC mobile users (**mobile.***username*)

-**u** *unix*

Specify the UNIX login name for search.

-**x** RESERVED for special handling. This is invoked by **utrcmd(1M)** to support remote operation of the Administration Tool interface.

EXAMPLES

Example 1: This command lists all sessions on the current server.

utsession -p

Example 2: This command finds the sessions for the UNIX user "jdoe".

utsession -p -u jdoe

Example 3: This command terminates a registered Sun Ray user's (john doe's) session.

utsession -k -n "john doe"

Example   4:   This   command   suspends   the   session   belonging   to   pseudo   token MicroPayflex.000105d665000100 on display **10**.

utsession -s -d 10 -t MicroPayflex.000105d665000100

EXIT STATUS

The following exit values are returned:

0 Command completed successfully.

1 Entry not found.

-1 An error occurred.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

utuser(1M), **auth.props(4)**, **utdesktop(1M)**

NAME

utsessiond - Sun Ray session manager daemon.

SYNOPSIS

/opt/SUNWut/lib/utsessiond [**-a** *authlist*] [**-c** *authfile*] [**-d**] [**-h** *hostname*] [**-p** *port*] [**-P** *nport*] [**-r**] [**-t**]

DESCRIPTION

The **utsessiond** daemon provides a reliable rendezvous point for services in a Sun Ray session. It acts as an intermediary to forward session connection and disconnection messages from the Sun Ray authentication manager to the services and provides facilities for supporting distributed synchronization of clip-lists for the services.

If either the -**a** or the -**c** options is specified, the session manager daemon operates exclusively in call-back mode. In this mode, the session manager only takes session connect and disconnect commands from authentication managers that are explicitly enabled by *authlist* or *authfile* and that have requested a call-back. The call-back feature provides a mechanism by which the session manager and the authentication manager may establish each other's identity.

Error messages from **utsessiond** are logged using **syslog**(3) with a facility value of LOG_DAEMON.

OPTIONS

The following options are supported.

-**a** *authlist*
Add the host and port pairs specified in *authlist* to the list of permitted Authentication Managers. The format of *authlist* is a comma separated list of *hostname:port* pairs.

-**c** *authfile*
Add the host and port pairs specified in the ASCII file *authfile* to the list of permitted Authentication Managers. The file contains a list of Authentication Manager specifications, one per line. The specifications take the form of *hostname* followed by *port* number, separated by white-space. Blank lines and any line whose first printable character is ''#'' are ignored.

-**d** Enable debugging output.

-**h** *hostname*
Set the *hostname* portion of the session IDs generated by the Session Manager to the specified *hostname* value. By default this is set to the machine's node name. This option can be used to handle servers supporting multiple IP addresses as part of a clustering solution.

-**p** *port*
Set the Session Manager's listen port to the specified *port* value. The default is port 7007. This is the port by which session services and Authentication Managers contact the Session Manager.

-**P** *nport*
This option is no longer used. Retained only for backward compatibility.

-**r** Automatically restart the Session Manager daemon if it exits. With this option the Session Manager daemon will create two processes: a child that performs all the actual work and parent monitoring process. The parent process will restart a child if the previous one exits.

-**t** Test mode. Use is beyond the scope of this document.

FILES

The following file is used:

• **/etc/opt/SUNWut/auth.permit**
The customary location of the **authfile** for a system.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

        utauthd(1M), **syslog(3)**, **syslogd(1M)**, **syslog.conf(4)**

NAME

utslaunch - Sun Ray DTU launch application.

SYNOPSIS

/opt/SUNWut/lib/utslaunch

DESCRIPTION

The **utslaunch** application is used to launch various Sun Ray applications via a "hotkey" key combination. The applications are enabled when the key combination is pressed.

The **utslaunch** application provides hotkey functionality while consuming fewer system resources.

Hotkey key combinations are defined in the **utslaunch.properties** files.

OPTIONS

There are no options for **utslaunch**.

EXAMPLES

Example 1: This command starts the utslaunch daemon in the background.

utslaunch &

FILES

The following files are used for hotkey configuration:

- **/etc/opt/SUNWut/utslaunch_defaults.properties**
  site-wide defaults

- **~/.utslaunch.properties**
  user's defaults

- **/etc/opt/SUNWut/utslaunch_mandatory.properties**
  site-wide mandatory defaults

The following file is used:

/usr/dt/config/Xsession.d/0100.SUNWut

ENVIRONMENT VARIABLES

utslaunch uses the **DISPLAY** environment variable to get the user's X display number.

It also uses the **HOME** environment variable to get the user's home directory to be able to use user's hotkey settings.

EXIT STATUS

The following exit values are returned:

0 Success

1 Failure

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

utslaunch.properties(4), **utsettings(1)**, **utdetach(1)**

NAME

       utstoraged - Sun Ray Mass Storage service daemon

SYNOPSIS

       /opt/SUNWut/lib/utstoraged [-D *debug-level*] [-o *optroot*] [-r]

DESCRIPTION

       utstoraged provides mass storage support for Sun Ray DTUs. **utstoraged** supplies driver services for USB mass storage devices that comply with the USB Bulk Only Mass Storage Specification 1.0.

       utstoraged uses the **utdisk**(1M) loopback driver to provide the **dkio**(7I) interface to Solaris applications. Applications can access storage devices through block and raw device links created in the $**UTDEVROOT/ dev/dsk** and $**UTDEVROOT/dev/rdsk** directories respectively. **utstoraged** interacts with the mounter daemon, **utmountd**(1M), to mount devices containing Solaris-recognizable filesystems.

       Users have access rights only to storage devices connected to the Sun Ray DTU on which their session is active and only for as long as the session is active. When the session is disconnected from the Sun Ray DTU through logout, hotdesking, server switching, or any other means, ownership of the storage device is lost and all pending data transfers are aborted. This can result in corruption of the filesystem on the media and loss of data.

       Note:

       Users are strongly advised to use the **utdiskadm**(1M) command to prepare the device for removal before disconnecting a session.

       Error messages are logged to **/var/opt/SUNWut/log/utstoraged.log**

OPTIONS

       The following options are supported:

           -D *debug-level*

           Debug mode. Use is beyond the scope of this document. If a debug level is set, debug messages are sent to stderr.

           -o *optroot*

           Use **optroot** as the parallel service's root directory for device node creation. **optroot** should be the same directory as the **optroot** directory used by **utdevmgrd**(1M).

           -r

           Automatically restart the storage service daemon if it exits. With this option, the storage service daemon creates two processes: a child that performs all the actual work, and a parent monitoring process. The parent process restarts a child if the previous one exits.

FILES

       The following files are used:

           $UTDEVROOT/dev/dsk

           The directory containing links to block device names for each slice or partition on the device.

           $UTDEVROOT/dev/rdsk

           The directory containing links to raw device names for each slice or partition on the device.

ENVIRONMENT VARIABLES

       UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

ATTRIBUTES

       See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

        utdiskadm(1M), **utmountd(1M)**, **utdevmgrd(1M)**, **utdisk(7D)**

NAME

> **utsunmc** - Sun Ray server software module for the Sun Management Center, providing addition, load, and removal utilities.

SYNOPSIS

> **/opt/SUNWut/sbin/utsunmc** [**-u**]

DESCRIPTION

> The **utsunmc** command adds the Sun Ray server software module to the Sun Management Center (SunMC) and loads it to permit monitoring of the Sun Ray software. The **utsunmc** command can also remove the Sun Ray server software module from the SunMC.

> The **utsunmc** command is run with superuser privileges.

OPTIONS

> The following option is supported.

> **-u** Remove the previously added and loaded Sun Ray server software module.

> Without arguments, addition and load of the Sun Ray server software module is performed.

ATTRIBUTES

> See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutesa

SEE ALSO

> Sun Management Center 3.0 Software User's Guide

NOTES

> The **utsunmc** command requires the Sun Management Center 3.0 or Sun Management Center 2.1.1 to be installed.

> The Sun Management Center agent is stopped while this command runs and restarted after the command completes. The agent may not properly restart on Sun Management Center 2.1.1. In this case, the command **/opt/SUNWsymon/sbin/es-start -a** should be run.

NAME

    utumount - Sun Ray Mass Storage unmount utility

SYNOPSIS

    /opt/SUNWut/bin/utumount -u *mount_path*

DESCRIPTION

    The **utumount** command has the same functionality as utdiskadm -u. It attempts to unmount the filesystem on *mount_path* if the related device is a Sun Ray storage device belonging to the user.

OPTIONS

    The following options are supported.

        -u *mount_path*

      Unmount *mount_path*

EXIT STATUS

    The following exit codes are returned:

    0 Suncess.

    1 Failure.

FILES

    The following files are used:

    $UTDEVROOT/dev/dsk
      The directory containing links to block device names for each partition on the device.

    $UTDEVROOT/dev/rdsk
      The directory containing links to raw device names for each partition on the device.

ENVIRONMENT VARIABLES

    UTDEVROOT points to a symbolic link of the device root for the Sun Ray DTU associated with a user's session.

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutsto

SEE ALSO

    utdiskadm(1M), **uteject(1M)**, **utmount(1M)**, **utmountd(1M)**, **utstoraged(1M)**, **utdisk(7D)**

NAME

> **/opt/SUNWut/sbin/utusbadm** - Enable or disable USB device services

SYNOPSIS

DESCRIPTION

> Note that \fButusbadm\fR command is deprecated. Use the **/opt/SUNWut/sbin/utdevadm** command to enable/disable USB services.

> The **utusbadm** ommand is used to disable/enable access to all USB ports on SunRay devices. It does not affect HID devices such as the keyboard and mouse; however, it does affect all other devices attached to the USB ports, which will not be accessible if the site is so configured.

> This is a site-wide property. When set, it affects all units connected to the failover group.

> Changing this configuration requires a cold system restart before it can take effect. When a successful change is made, the command reminds the administrator to restart the services.

OPTIONS

> The following options are supported.

> -**h** Print the current usage state.

> -**e** Enable all USB device services.

> -**d** Disable all USB device services.

ENVIRONMENT VARIABLES

> None

EXIT STATUS

> The following exit values are returned:

> **0** on success

> **1** on error

FILES

> None

ATTRIBUTES

> See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto
Interface Stability	Public Evolving

SEE ALSO

> **utrestart(1M), utdevadm(1M)**

NAME

> **utusbdevs** - Simple utility to display USB devices connected to a Sun Ray.

SYNOPSIS

> /opt/SUNWut/sbin/utusbdevs

> /opt/SUNWut/sbin/utusbdevs -**V** -**W** -**d** -**u** -**s**
> -**t** <**terminal ID**> -**c** <**USB class**> -**f** <**filter**>

> /opt/SUNWut/sbin/utusbdevs **-h**

DESCRIPTION

> The utusbdevs command is a simple utility to display all the USB devices connected to Sun Ray's.

> utusbdevs command with no options or arguments displays all USB devices of all classes on all Sun Rays connected to the server or server group.

OPTIONS

> The following options are supported:

> **-V - Verbose mode**
> > Displays lots of debugging info

> **-W - Wait**
> > Loops forever, waiting for Device Manager events

> **-d - only show devices**

> **-u - only show units**

> **-s - observe current Sun Ray**
> > Displays all USB devices of all classes for the current Sun Ray

> **-t <terminal ID>**
> > Displays all USB devices of all classes for the Sun Ray with the terminal ID specified.

> **-c <class>**
> > Displays all USB devices that are of class and all subclasses of class on all Sun Rays connected to the server or server group.

> **-f <filter>**
> > Displays all USB devices of all classes on all Sun Rays connected to the server or server group. Displays only the specified filter values. This filter determines which fields for each USB device or unit are displayed on the output.

> Filter (-f) specification keys:

> > **t: terminal ID**
> >
> > **a: USB bus address**
> >
> > **u: unit ID**
> >
> > **i: terminal IP**
> >
> > **n: number of units**
> >
> > **c: USB class code**
> >
> > **p: USB product code**
> >
> > **U: UID of device owner**
> >
> > **l: device lease time**
> >
> > **M: device manufacturer name**

m: device model name

Default filter specification keys: **tauincpUlmMs**

-h - help
Prints the usage.

EXAMPLES

Example 1:

/opt/SUNWut/sbin/utusbdevs
Displays all USB devices of all classes on all Sun Rays connected
to the server or server group.

Example 2:

/opt/SUNWut/sbin/utusbdevs -t IEEE802.080020010203
Displays all USB devices of all classes for the Sun Ray
with the terminal ID specified.

Example 3:

/opt/SUNWut/sbin/utusbdevs -c 9.0
Displays all USB devices that are of class 9.0 and all subclasses
of class 9.0 on all Sun Rays connected to the server or server group.

Example 4:

/opt/SUNWut/sbin/utusbdevs -t IEEE802.080020010203 -c 9.0
Displays all USB devices that are of class 9.0 and all subclasses
of class 9.0 for the Sun Ray with the terminal ID specified.

Example 5:

/opt/SUNWut/sbin/utusbdevs -f tuc
Displays all USB devices of all classes on all Sun Rays connected
to the server or server group. Display only the terminal ID,
the unit ID and the USB class code.

ENVIRONMENT VARIABLES

None

EXIT STATUS

The following exit values are returned:

0 on success

1 on error

FILES

None

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto
Interface Stability	Public Evolving

NAME

   utuser - Sun Ray user administration utility.

SYNOPSIS

   /opt/SUNWut/sbin/utuser -**a** "*tokenID,server-name,server-port,name,other-info*" [-**r** *tokenreader*]

   /opt/SUNWut/sbin/utuser -**a** -**f** *filename* [-**r** *token-reader*]

   /opt/SUNWut/sbin/utuser -**a** -**i** *current-tokenID new-tokenID* [-**r** *token-reader*]

   /opt/SUNWut/sbin/utuser -**d** *tokenID*

   /opt/SUNWut/sbin/utuser -**d** -**f** *filename*

   /opt/SUNWut/sbin/utuser -**d** -**i** *current-tokenID*

   /opt/SUNWut/sbin/utuser -**e** "*tokenID,server-name,server-port,name,other-info*"

   /opt/SUNWut/sbin/utuser -**e** -**f** *filename*

   /opt/SUNWut/sbin/utuser -**e** -**i** *current-tokenID* [**enabledisable**]

   /opt/SUNWut/sbin/utuser -**h**

   /opt/SUNWut/sbin/utuser -**l**

   /opt/SUNWut/sbin/utuser -**l** -**c**

   /opt/SUNWut/sbin/utuser -**l** -**i** *substring*

   /opt/SUNWut/sbin/utuser -**l** -**n** *substring*

   /opt/SUNWut/sbin/utuser -**L**

   /opt/SUNWut/sbin/utuser -**L** -**c**

   /opt/SUNWut/sbin/utuser -**L** -**i** *substring*

   /opt/SUNWut/sbin/utuser -**L** -**n** *substring*

   /opt/SUNWut/sbin/utuser -**L** -**g**

   /opt/SUNWut/sbin/utuser -**o**

   /opt/SUNWut/sbin/utuser -**p** *tokenID*

   /opt/SUNWut/sbin/utuser -**r** *token-reader*

DESCRIPTION

   The **utuser** command allows the administrator to manage users registered on the Sun Ray server which the
   command is run. The information that **utuser** provides is from the Sun Ray administration database and the
   Sun Ray Authentication Manager.

   utuser operations that only display information may be run by any user. Operations that change or delete data
   are run under superuser privileges.

OPTIONS

   The following options are supported.

   -**a** Add user with the specified *tokenID*, *servername*, *serverport*, *name* and *otherinformation* properties.
        The 5 comma-delimited values should be enclosed within quotes. The *other-information* property is
        optional.

   -**a** -**f** Batch add multiple users using input from the specified *filename*. The format of each line in the input
        file is: *tokenID,server-name,serverport,name,other-info*

   -**a** -**i** Add the specified *new-tokenID* to the user that currently has token *current-tokenID*.

   -**d** Delete the user with the specified *tokenID*. This command deletes the user and all of the user's tokens. (To
        delete a single token without deleting the user, use the **-di** option.)

   -**d** -**f** Batch delete multiple users using input from the specified *filename*. The format of each line in the input
        file is: *tokenID*. However, you may use the output of the **-o** option as input to this option as all arguments
        after the first comma are ignored. For each token-id specified in the filename, this command deletes the
        associated user and all of the user's tokens. (To delete a single token without deleting the user, use the **-
        di** option.)

-**d** -**i** Delete token *current-tokenID* from the user that currently has ownership of it. The token to be deleted must not be the user's only token. This command does not delete the user or any of the user's other tokens. (To delete the user and all the user's tokens, use the **-d** option.)

-**e** Edit properties for the user with the specified *tokenID* by changing the *server-name*, *server-port*, *name* and *other-information* properties to the specified values. Note that the 5 comma-delimited values should be enclosed within quotes. The other information property is optional.

-**e** -**f** Batch edit multiple users using input from the specified *filename*. The format of each line in the input file is: *tokenID,server-name,server-port,name,other-info.*

-**e** -**i** Enable or disable the specified *current-tokenID*.

-**h** Show usage information (this message).

-**l** List all users registered in the admin database..

-**l** -**c** List all users registered in the admin database that are currently logged in.

-**l** -**i** List all users registered in the admin database with token-ids that contain the specified substring.

-**l** -**n** List all users registered in the admin database with names that contain the specified substring.

-**L** List all users registered in the admin database (long format).

-**L** -**c** List all users registered in the admin database that are currently logged in (long format).

-**L** -**i** List all users registered in the admin database with token-ids that contain the specified *substring* (long format).

-**L** -**n** List all users registered in the admin database with names that contain the specified *substring* (long format).

-**L** -**g** List all users registered in the admin database currently logged in and the servers into which they are logged in.

-**o** Dump user list in comma-delimited format. The format of each line output by this option is: *tokenID*, *server-name*, *server-port*, *name*, and *other-info*.

-**p** Show user properties for user with the specified *tokenID*.

-**r** When specified alone, this option reads a token-id from the specified token reader. When specified with the **-a**, **-af** or **-ai** options, the **-r** flag instructs **utuser** to use the specified token reader to assist in adding users or tokens whenever the character "x" is used in place of a token-id. The command will prompt you to insert the token into the specified reader when its ready to read the token.

For the **-l -i**, **-l -n**, **-L -i**, and **-L -n** options, the substring comparisons are case-insensitive.

EXAMPLES

Example 1: This command displays all users that have "parker" in their usernames:

% **/opt/SUNWut/sbin/utuser -l -n parker**

Example 2: This command adds a user with unknown token-ID, server name "localhost", server port "7007", user name "John Anderson", and other information "C987" by using the token reader 08002086e18f to read the token-ID:

/opt/SUNWut/sbin/utuser -a "x,localhost,7007,John Anderson,C987"
-r 08002086e18f

Example 3: This command adds multiple users using input from the /tmp/users file:

/opt/SUNWut/sbin/utuser -a -f /tmp/users

Example 4: This command reads a token from token reader 08002086e18f:

/opt/SUNWut/sbin/utuser -r 08002086e18f

FILES

The following file is used:

• **/etc/opt/SUNWut/utadmin.conf**

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta

SEE ALSO

utdesktop(1M), **utadmin.conf(4)**

NOTES

The **-G** option has been deprecated in favor of using **utuser -L -g**.

The **-k** option has been deprecated. Use **utsession -k** instead.

NAME

utwall - Sun Ray user notification utility.

SYNOPSIS

/opt/SUNWut/sbin/utwall -a *aufile* [-**r** *n*] [-**v**] { **ALL** | user [**:***display*] | **:***display* ... }

/opt/SUNWut/sbin/utwall [-**d**] [-**m "***subject*"] [-**t "***message-text*"] [-**v**] { **ALL** | user [**:***display*] | **:***display*
... "}"

/opt/SUNWut/sbin/utwall [-**u "***message-text*"]

DESCRIPTION

utwall sends a message or an audio file to users having an **Xserver** process. The messages can be sent in email and/or displayed in a pop-up window. When sent to a multihead session, the pop-up window will appear on all displays for that session.

Options **-a** and **-d** require superuser privileges.

OPTIONS

The following options are supported.

-**a** *aufile*
   Annunciate mode. Plays the audio file *aufile* on the specified user's X session. Audio files of type **.au** can be found at **/usr/demo/SOUND/sounds**.

-**d** Pop up a window with the supplied message on each Xserver instance.
   .IP "-**m " "***subject*"""" "6"
   Send mail with the given subject "*subject*" and supplied message. If the text has white space, use single or double quotes. Substitution is supported.

-**r** *n* Repeat the annunciation *n* times. This option can only be used with **-a**. Default is 1.
   .IP "-**t " "***message-text*"""" "6"
   Message text. Alternatively, the message can be supplied as **stdin**. If the text has white space, use single or double quotes. Substitution is supported.

-**v** Verbose mode.

OPERANDS

The following operands are supported:

ALL
   Action is performed on all user having an Xserver process.

user**:***display*
   Action is performed on the given users (optional display number *display*) having an Xserver process.

:*display*
   Action is performed on the users having display number *display*.

EXAMPLES

Example 1: This command sends email to all users:

   # **/opt/SUNWut/sbin/utwall -m 'System policy change...' -t
   'Tonight\nPlease log off' ALL**

The email reads:

   Subject: System policy change...
   Tonight
   Please log off

Example 2: This command pops a window up on all sessions stating "Log off now!"

   # **/opt/SUNWut/sbin/utwall -d -t "Log off now!" ALL**

Example 3: This command pops a window up on jsmith's session on display 26 with the text from *messagefile*

> # **/opt/SUNWut/sbin/utwall -d jsmith:26 <** *messagefile*

Example 4: This command pops a window up with a greeting to the user on display number 10

> # **/opt/SUNWut/sbin/utwall -d -t "Hello" :10**

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

wall(1M), **mailx(1M)**, **utaudio(1)**

NOTES

When Sun Ray DTUs are configured for Xinerama, only the origin screen displays the **utwall** message.

NAME

   utwho - Provide a compact summary of sessions and firmware maintenance.

SYNOPSIS

   /opt/SUNWut/sbin/utwho -**c** [-**a**] [-**H**]

DESCRIPTION

   The **utwho** command is a script that assembles information about display number, token, and logged-
   in user, and displays that information in a compact format. It can also display the IP address, Sun Ray
   model, and MAC address of a Sun Ray that is connected to a given session.

OPTIONS

   The following options are supported.

   -**c** With **utwho**, show connected Sun Rays with the display number, session token, logged in user, IP address,
       Sun Ray model, and Sun Ray MAC address. The display number is presented in the form d.m, where d
       is the X display number of the session, and m is the index within a multihead group of that particular Sun
       Ray.The session token is the value of **$SUN_SUNRAY_TOKEN** within a session. The model and MAC
       address are output as PxB.MAC, where Px is the ending part of the model type, e.g., P4, and MAC is the
       6-byte ethernet MAC address in hexadecimal format. Without the -**c** option, the   command displays only
       session information, including the X display number, token, and logged in user. In this mode, sessions
       are displayed even if they don't have a Sun Ray connected to them.

   -**a** In combination with other options, this option controls the selection of Sun Rays or sessions to display.
       Without -**a**, only sesssions that have logged in users are displayed. With -**a**, all sessions or Sun Rays are
       displayed, and those with no logged in user have a user id field of "????".

   -**H** Output column headings above the regular output.

ATTRIBUTES

   See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

   utfwadm(1M), **attributes(5)**

NAME

      utxconfig - Sun Ray DTU X server configuration utility.

SYNOPSIS

      /opt/SUNWut/bin/utxconfig [-**a**] [-**c** *config-file*] [-**d** *display*] [-**D**] [-**k** *xkb-value*] [-**l**] [-**L** *geom*] [-**m** *multihead*] [-**p** *pcolor*] [-**r** *res*] [-**R** *geom*] [-**S** *screen-order*] [-**t** *token*] [-**x** *xinerama*]

      /opt/SUNWut/bin/utxconfig -**e** [-**d** *display*] [-**t** *token*]

      /opt/SUNWut/bin/utxconfig [-**o**] [-**f** *file*]

      /opt/SUNWut/bin/utxconfig [-**i**] [-**f** *file*]

      /opt/SUNWut/bin/utxconfig [-**X** *xserver*]

DESCRIPTION

      utxconfig displays and configures X server parameters for Sun Ray sessions. The changes to the X server are not evident until a restart of the X server process. For example, log out, then log in.

OPTIONS

      The following options are supported.

      -**a** Allows the setting or listing of the default values. Only superuser may change the default settings.

      -**c** *config-file*
         Sets a specific *configfile* to use. The usage of this option is beyond the scope of this manual.

      -**d** *display*
         Sets the X display variable used to determine the Sun Ray DTU session. Otherwise, the DISPLAY environment variable is used. Users must have access to an X server attached to their session before they can read or change the settings for that session.

      -**D** Debug flag.

      -**e** Erases all specific settings for the session. All settings return to their default values.

      -**f** *file*
         Specifies a file to be used in conjunction with **-o** or **-i**.

      -**i** Populates the system settings database from a comma delimited text record such as the one produced by **-o**. Input is taken from the standard input unless **-f** is specified. You must be root to use this option.

      -**k** *xkb-value*
         Enable or disable the XKB extension. To enable use of XKB specify "on". To disable XKB specify "off". To revert to the system default specify "reset". Enabling XKB is not recommended on Solaris 8 or Solaris 9.

      -**l** Lists out the current settings for the session. If no specific values have been set for the session, the default values are printed.

      -**L** *geom*
         Lists out the X server screen device start-up arguments for the user preferred geometry set with **-R** or for *geom* if none is set. The use of this option is beyond the scope of this manual.

      -**m** *multihead*
         Enables or disables multihead mode for X session startup. By default, if a session is started on a multihead terminal group, then the session starts in multihead mode to match the terminal group with an appropriate number of screens and geometry. Specify "off" to disable this behavior and the session starts on a single terminal with one screen. The keyword "reset" may be specified to reset to the system default.

      -**o** Output all system settings in a comma-delimited text record. Intended for use with **-i**. Outputs to standard output unless **-f** is specified.

      -**p** *pcolor*
         Parameter that specifies the level of support for the PseudoColor (8-bit) visual in the X server. The PseudoColor visual is not enabled by default. The accepted values for *pcolor* are "off", "on", "default", and "reset". A *pcolor* value of "off" will disable the PseudoColor visual. A *pcolor* value of "on" will enable the PseudoColor visual, but the TrueColor visual (24-bit) will remain the default. A *pcolor* value of "default" will enable the PseudoColor visual and make it the default visual, although the TrueColor visual will still be available. A *pcolor* value of "reset" will revert to the system default.

      -**r** *res*
         Parameter that specifies a resolution (number of pixels) that the X server should provide for the session.

The format of *res* is *WIDTHxHEIGHT*, for example 1280x1024. **utxconfig** enforces restrictions on the possible widths and heights that can be specified. The keyword "auto" may also be specified, which will enable selection of a resolution that best matches the resolution capabilities of the terminal on which the X session is started. The keyword "reset" may be specified to revert to the system default.

-**R** *geom*
Specify a preferred screen geometry in the form *COLSxROWS*. At X server startup this geometry overrides the terminal group geometry on which the session is started. See -**m**. The keyword "auto" may also be specified as a value, which will cause the geometry of the terminal group on which the X session is started to be used. The keyword "reset" may be specified to revert to the system default.

-**S** *screen-order*
Specify a preferred screen number order for the session's screen group. The order must be a legal for standard Xsun (Sun Ray X server) screen placement, or the keyword "auto" which will cause a default screen ordering to be chosen. Manipulating the geometry via the -R option will automatically change the screen order to an appropriate value.

-**t** *token*
Allows the setting of a specific token to use. The use of this option is beyond the scope of this manual.

-**x** *xinerama*
Enable or disable XINERAMA extension (not supported under the Solaris 2.6 operating environment; see man page for **Xserver(1)**). To enable use of XINERAMA under Solaris 8 or 9 operating environment, specify "on". To disable XINERAMA, specify "off". To revert to the system default specify "reset".

-**X** *xserver*
Specify which X server to output options for.

EXAMPLES

Example 1: This command enables PseudoColor visual on a 1024 x 768 screen:

% **/opt/SUNWut/bin/utxconfig -r 1024x768 -p on**

Example 2: This command configures a multihead group with two screens, right and left:

% **/opt/SUNWut/bin/utxconfig -m on -R 2x1 -S 0,1**

Example 3: This command sets the screen geometry and ordering to automatic:

% **/opt/SUNWut/bin/utxconfig -R auto**

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuta
Interface Stability	Evolving

SEE ALSO

Xserver(1)

NOTES

The settings are actually maintained on the basis of an authentication token and do not remain specific to a single X display number.

NAME

      libusbut.so.1 - libusb plugin

SYNOPSIS

DESCRIPTION

      The Sun Ray **libusb** plugin provides Sun Ray-specific support for **libusb**.

      The **SUNWlibusbut** package delivers the Sun Ray libusb plugin **libusbut.so.1** in **/opt/SUNWut/lib**. This plugin is loaded by the **libusb** wrapper library **libusb.so**, which is delivered as part of the **SUNWlibusb** package.

      For the wrapper library to be able load the plugin the following link must exist in /usr/sfw/lib/libusb_plugins:

      libusbut.so.1 -> /opt/SUNWut/lib/libusbut.so.1

      If the **SUNWlibusb** package is installed in the system prior to Sun Ray Server Software installation, then the required **symlink** is created automatically.

      If the **SUNWlibusb** package is not installed, the administrator needs to install the **SUNWlibusb** package and then create the symlink as follows:

      # cd /usr/sfw/lib/libusb_plugins

      # ln -s /opt/SUNWut/lib/libusbut.so.1

      See also **/usr/sfw/share/doc/libusb/libusb.txt**.

NOTES

      The Sun Ray **libusb** plugin currently does not support:

      usb_interrupt_write

      timeouts for the I/O calls.

      Note: Do not fork processes to do I/O. Use threads instead.

      The **SUNWlibusb** package can be found in the Supplemental area of the Sun Ray Server Software.

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWlibusbut
Interface Stability	External

SEE ALSO

      intro(3), **attributes(5)**

NAME

ut_amgh_get_server_list, ut_amgh_free_server_list - APIs for Automated Multi-Group Hotdesking also known as Regional Hotdesking.

SYNOPSIS

cc [flag ...] file ... [library ...]

#include <utamgh.h>

int ut_amgh_get_server_list(const struct ut_amghargs *amghargs, struct ut_amghret *amghret);
int ut_amgh_free_server_list(struct ut_amghret *amghret);

DESCRIPTION

The **ut_amgh_get_server_list**() function is called for retrieving a list of servers associated with the input parameters for the purposes of Automated Multi-Group Hotdesking. The underlying AMGH PAM services will invoke this function to obtain a list of servers which the Sun Ray should be redirected to. The first server in the list which is alive and online will be utilised.

The ut_amghargs structure has the following entries:

```
struct ut_amghargs {
  int  amghversion;       /* the version of the customer API */
  char *username;         /* the user name e.g. 'fred'*/
  char *token;            /* the token id e.g. 'Payflex.50045fda00130100'*/
  char *terminal_cid;     /* the canonical terminal id e.g. 'IEEE802.0800201e2347'*/
  char *terminal_ip_addr; /* the terminal ip address e.g. '10.23.146.12'*/
  char *insert_token;     /* the token id inserted  e.g. 'Payflex.50045fda00130100'*/
  char *display;          /* the display number e.g. ':12'*/
};
```

The ut_amghret has the following entries:

```
struct ut_amghret {
  char **server_list;   /* the reference to a server array */
  char *username;       /* iff the implementation returns one else NULL */
  char *errorstr;       /* the error message */
  int  chain_amgh;      /* RESERVED for future use */
  int  use_firstserver; /* Use firstServer prop as the host */
};
```

The amghversion field is filled by the underlying PAM service before **ut_amgh_get_server_list**() is called. **ut_amgh_get_server_list**() should verify that amghversion is >= UT_AMGH_VERSION to protect itself from being deployed on old and potentially incompatible versions of AMGH. If the test fails it should return UT_AMGH_INCOMPATIBLE_VERSION before accessing other fields in the ut_amghargs or ut_amghret structures.

ut_amgh_get_server_list() may use the values of the ut_amghargs structure to determine the list of hosts to return to the AMGH service. It should set server_list to point to storage containing an array of pointers to fully-qualified hostnames or IP address strings expressed in the IPv4 standard '.' notation, with a NULL pointer as its last element. If no suitable hosts are found, **ut_amgh_get_server_list**() should assign NULL to the server_list field in the structure ut_amghret. If an error is encountered it should set errorstr in the ut_amghret structure to a suitable error string, set server_list to NULL and return one of the valid error codes listed below. **ut_amgh_get_server_list**() may allocate dynamic storage for use in the ut_amghret structure, which will be freed by a later call to **ut_amgh_free_server_list**().

ut_amgh_get_server_list() may also return a username. If a NULL username is supplied by the PAM service and a username is returned it will be used to initialize PAM_USER in the PAM context.

ut_amgh_free_server_list() frees up any memory dynamically allocated in the ut_amghret structure. It is the responsibility of the   calling application to invoke **ut_amgh_free_server_list**() to   free the memory returned by **ut_amgh_get_server_list**().

RETURN VALUES

Upon successful completion, UT_AMGH_SUCCESS must be returned. In addition, the following values may be returned :

UT_AMGH_INCOMPATIBLE_VERSION
>   The version filled in struct ut_amghargs field amghversion by the customer is not compatible with UT_AMGH_VERSION value.

UT_AMGH_NO_PERMISSION
>   No permission to access the back-end database for the server lookups.

UT_AMGH_DB_ERROR
>   Error accessing the back-end database.

UT_AMGH_NULL_USERNAME
>   The implementation requires a username to perform its lookup, but the AMGH system was not configured to pass in a username and hence the ut_amghargs username field was NULL.

UT_AMGH_ERROR
>   Unknown error encountered by the implementation.

UT_AMGH_NOMEM
>   Unable to allocate memory.

UT_AMGH_USERNAME_CONFLICT
>   More than one non-NULL user names found for the token.

NOTES

A reference Makefile and source code for a file-based implementation of the AMGH APIs are available under /opt/SUNWutref/amgh/, as a learning aid for potential implementors.

SEE ALSO

ut_amgh_script_interface(3)

NAME

    ut_amgh_script_interface - executable API for Automated Multi-Group Hotdesking (also known as Regional Hotdesking)

SYNOPSIS

    <user_defined_executable_name> < input > output

    input:
    [username=*username*]
    [token=*token*]
    [terminal_cid=*macaddr*]
    [terminal_ip_addr=*dtu_IP*]
    [insert_token=*token*]
    [display=*display_number*]

    output:
    [username=*username*]
    [host=*host*]
    [use_firstserver=true/false]

DESCRIPTION

    The Sun Ray Server Software Automated Multi-Group Hotdesking (AMGH) system utilizes an API to obtain information about a username and/or a set of servers for a Sun Ray to be directed to. See the ut_amgh_get_server_list man page for more details.

    AMGH also can be configured to run a customer-supplied executable to produce information for this API. This man page documents the interface that such an executable should implement.

    The executable will be executed with no arguments. On its standard input stream (see **intro**(3)), stdin, data representing the ut_amghargs structure contents will be written. This data will be in the form of key/value pairs, one per line on stdin. The key name will be from the beginning of the line to the first '=' "character," "and" "the" "value" "string" "will" "begin" "after" "the" character and continue to the next newline.

    Valid lookup keys are :

```
username          /* the user name  e.g. 'fred' */
token             /* the token id    e.g. 'Payflex.50045fda00130100'*/
terminal_cid      /* the canonical terminal id  e.g. 'IEEE802.0800201e2347'*/
terminal_ip_addr /* the terminal ip address  e.g. '10.23.146.12'*/
display           /* the display number e.g. ':12'*/
insert_token      /* the token id inserted  e.g. 'Payflex.50045fda00130100'*/
                  /* This is useful in cases where inserted token is different */
                  /* from the session token, for eg when the registered policy */
                   /* is in effect */
```

    Lines containing keys which are unrecognized should simply be discarded from stdin and the executable should continue parsing the remaining input until EOF is reached.

    At that point the executable should perform its lookup. If a username is to be returned, it should write a key/value to standard output (stdout) of the form:

    username=USERNAME

    where USERNAME is the value returned by the lookup. It may also return any number of lines of the form:

    host=HOST

    where each HOST represents either a fully-qualified hostname or an IP address.

    The script could also return the following values :

    chain_amgh=[true | false]

    use_firstserver=[true | false]

The usage of these options is beyond the scope of this man page and will be discussed in details in the software documentation. These are documented here for completeness. The return values for these options can also be a non-zero number (true) or 0 (false).

ERRORS

If an error is encountered during the lookup, the executable may write a descriptive error message to standard error (stderr), which will be included in the error log and exit with a suitable error code (see EXIT VALUES).

EXIT VALUES

Exit values should correspond to the error codes in /opt/SUNWut/include/utamgh.h.

If no errors are encountered during the lookup, the executable should exit with a status of 0 (UT_AMGH_SUCCESS).

Note that if the lookup does not produce a match, this is not considered an error. In this case, no stdout should be generated, and the executable should exit with a status of 0 (UT_AMGH_SUCCESS).

If the executable exits with a non-zero exit status, an error will be logged corresponding to the value of the exit status.

SEE ALSO

ut_amgh_get_server_list(3)

NAME

    auth.props - Sun Ray authentication daemon configuration file.

SYNOPSIS

    /etc/opt/SUNWut/auth.props

DESCRIPTION

    The **auth.props** file contains the Sun Ray Authentication Manager's configuration properties. Changes to many of these properties are not supported and should not be set to other than the default values.

OPTIONS

    The following options are available:

    adminConfigFile=*filename*
      This file contains the administrative database configuration information.

    allowAnnotations=*boolean*
      UNSUPPORTED. When set true, any application can connect from any IP address and annotate a session. Annotations are restricted to keywords prefixed by "x_". Values are not restricted.

    allowFWLoad=*boolean*
      Specifies whether or not the **utload** command is allowed to download firmware to DTUs connected to this Authentication Manager.

    allowLANConnections=*boolean*
      When set true the DTU connections are allowed from public LAN interfaces, as well as from private Sun Ray interconnect interfaces.

    cbport=*portNumber*
      UNSUPPORTED. The Authentication Manager listens on this port for connections from the **utsessiond** daemon and other programs, such as **utload**.

    cbtimeout=*seconds*
      UNSUPPORTED. Specifies the read timeout in seconds for programs that connect to the cbport.

    controllers=*maximum*
      UNSUPPORTED. Specifies the maximum number of spare threads that are available for handling new connections from applications such as **utload**(1M).

    enableGroupManager=*boolean*
      UNSUPPORTED. Flag to turn on the group manager function.

    enableLoadBalancing=*boolean*
      Flag to turn on group manager load balancing.

    enableMulticast=*boolean*
      UNSUPPORTED. Flag to enable/disable use of multicast in group manager. If disabled, group manager will use broadcast.

    forceSessionLocation=*boolean*
      UNSUPPORTED. Flag to force use of **sessionHost** and **sessionPort** settings from this file regardless of the various authentication modules.

    gmDebug=*level*
      UNSUPPORTED. Group manager debugging level.

    gmKeepAliveInterval=*seconds*
      UNSUPPORTED. The group manager uses this as the time in seconds between broadcast keepalive messages.

    gmport=*port*
      UNSUPPORTED. The group manager uses this port to send and receive keepalive/discovery messages from other Authentication Managers.

    gmSignatureFile=*file*
      The group manager can "sign" messages to other group managers based on the contents of a signature file. Other group managers with the same signature file contents are "trusted". To be usable, the file must be owned by 'root' and must not be readable, writable, or executable by anyone else; it must contain at least 8 bytes, at least two of which are letters and at least one which is a non-letter printable character.

log=*filename*
> UNSUPPORTED. This option specifies a file that contains the log messages.

logAddTimeStamp=*boolean*
> UNSUPPORTED. Add your own timestamp to syslog messages. This may be appropriate for debugging or in cases where a remote syslog server is being used and higher resolution timestamps are required.

logFacility=*value*
> The logFacility can be one of the following:kern, user, mail, daemon, auth, syslog, lpr, news, uucp, cron, local0, local1, local2, local3, local4, local5, local6, local7

Log files
> Log priorities for different **utauthd** message classes can be one of the following: emerg, alert, crit, err, warning, notice, info, debug, OFF.
> The message classes are:
> **logPriClientError**=*value*
> **logPriDebug**=*value*
> **logPriNotice**=*value*
> **logPriWarning**=*value*
> **logPriConfigError**=*value*
>
> **logPriUnexpectedError**=*value*

maxStarting=*maximum*
> UNSUPPORTED. Specifies the maximum number of threads that can be simultaneously initiating a session. Additional threads wanting to start or verify a session wait for previous threads to finish starting or verifying a session.

moduleDir=*directoryName*
> UNSUPPORTED. Specifies the location of the authentication modules.

multicastTTL=*integer*
> UNSUPPORTED. Time-to-live parameter for forwarding multicast packets. If set above one, keepalive messages can pass through routers.

noClaimSleepTime=*seconds*
> UNSUPPORTED. The amount of time in seconds to sleep after a token has been offered to all of the authentication modules and before notifying the DTU that the authentication failed.

policy=*filename*
> Specifies the location of the authentication policy specification.

port=*portNumber*
> The **utauthd** daemon listens on this port for connections from Sun Ray DTUs.

remoteSelect=*boolean*
> If true, the remote server selection option of the **utselect**(1) command is enabled by default.

reportAllDesktopEvents=*boolean*
> UNSUPPORTED. When true, all desktop events are reported instead of being filtered to just those events that change the "exists" state of the DTU.

selectAtLogin=*boolean*
> If true, activates a **utselect -L** GUI allowing the user to select a Sun Ray server before logging into CDE. If only one server is available, the GUI exits automatically. Refer to the **utselect** man page for more information on the **-L** option.

sessionHost=*hostname*
> UNSUPPORTED. Specifies the host name of the server that is running the default **utsessiond** for this Authentication Manager.

sessionPort=*portNumber*
> UNSUPPORTED. Specifies the port number of the server that is running the default **utsessiond** for this Authentication Manager.

sessionTypesFile=*filename*
> Specifies a file that contains mappings from session types to the associated session startup and shutdown commands.

smartcardConfigSource = *<list of space-separated sources>*
> The keys in this property specify the order in which to search for the configuration files. The special

reserved key **LDAP** means go to the configured LDAP database. Any other value refers to a local **probe order** file.

smtimeout=_seconds_
UNSUPPORTED. Specifies the read timeout in seconds for reading messages from the **utsessiond** daemon.

termAddrIsSecret=_boolean_
UNSUPPORTED. When true, the IP address and port of DTUs are not reported in the dynamic status information provided on port cbport in response to the string.

terminalokens = <_list of space-separated tokens_>
UNSUPPORTED. Define the types of tokens that are handled by the terminal rather than by the auth manager. If a token of this type is seen, the auth manager will use the ID value generated by the terminal rather than trying to determine the ID on the server.

terminateEnable=_boolean_
UNSUPPORTED. Enable the cleanup of empty sessions based on notification from the session manager. The default is enabled.

timeout=_seconds_
UNSUPPORTED. DTUs are required to send a message to the Authentication Manager at least once every time period specified by _seconds_.

tokenDir=_directory_
UNSUPPORTED. Specifies a directory that contains the mappings from logical token names to session identifiers. The persistent storage of these mappings allows the **utauthd** daemon to recover its state after restarting. This state is reset on reboot of the system.

token.equiv=_filename_
UNSUPPORTED. Specifies a file that contains mappings from one raw token name to another.

workers=_maximum_
UNSUPPORTED. Specifies the maximum number of spare threads that are available for handling new connections from Sun Ray DTUs.

FILES

The following files are used:

/etc/init.d/utsvc
This is the system startup script that invokes the daemon **/opt/SUNWut/lib/utsessiond**. The session manager performs the actual session switching function.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutr

SEE ALSO

utauthd(1M), **utpolicy(1M)**, **utsessiond(1M)**, **utselect(1)**

NAME

    utadmin.conf - Sun Ray server administration configuration file.

SYNOPSIS

    /etc/opt/SUNWut/utadmin.conf

DESCRIPTION

    The **utadmin.conf** file is a contains configuration parameters for the Sun Ray server administration database.

    The **admin.defaultlocale** parameter (see below) is the only parameter that should be changed once the Sun Ray server is configured and in use. All other parameters are reserved.

PROPERTIES

    The supported configuration parameters are listed below. For each one, the name, description, and an example are given.

    Name
        Description

    admin.defaultlocale
        The default locale for Web-based Administration Tool. Supported values:
        "en_US" (US English), "fr" (French), "ja" (Japanese), "zh" (Simplified Chinese).

    admin.dstatus.dbfile
        The name of the NDBM data files where the desktop status is stored.

    admin.http.cfile
        Configuration file for the Sun Ray Administration Server.

    admin.http.port
        The webserver port used by the Administration Tool.

    admin.server.name
        The name of the server where the administration database LDAP server process is running. This is usually the host name of the Sun Ray server.

    admin.subtree
        The subtree within the LDAP hierarchy where Sun Ray administration data for this server resides. This is an entry under the UT root entry that was specified by **utconfig**.

    admin.user.name
        The LDAP user that the administration clients should bind as to perform privileged operations.

    admin.ustatus.dbfile
        The name of the NDBM data files where the user status is stored.

EXAMPLES

    Example 1: Configuration parameters for the LDAP and NDBM databases:

```
admin.server.name     = sray-139
admin.server.port     = 7012
admin.user.name       = cn=utadmin,utname=sray-139,o=v1,o=utdata
admin.subtree         = utname=sray-139,o=v1,o=utdata
admin.defaultlocale   = en_US
admin.dstatus.dbfile  = /var/opt/SUNWut/ndbm/dstatus
admin.ustatus.dbfile  = /var/opt/SUNWut/ndbm/ustatus
admin.http.port       = 1660
admin.ssl.enable      =
admin.http.cfile      = /usr/apache/httpd.conf
```

ATTRIBUTES

    See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutr

SEE ALSO

> **utinstall(1M)**, **utconfig(1M)**, **utuser(1M)**, **utdesktop(1M)**
> *Sun Ray Server Software 3 Administrator's Guide*

NAME

utslaunch.properties - Default hotkey key combinations for various applications supported by **utslaunch**.

SYNOPSIS

/etc/opt/SUNWut/utslaunch_defaults.properties

~/.utslaunch.properties

/etc/opt/SUNWut/utslaunch_mandatory.properties

DESCRIPTION

The files listed above are standard Java properties files that can contain defaults which customize the operation of the **utslaunch** application. Each file contains entries in the format of:

name=value

where *name* is the property name and *value* is the value set.

EXTENDED DESCRIPTION

When the **utslaunch** application starts, it looks for and reads the properties files in the order listed below. Note that a hotkey key combination specified in a file can be overridden by a file read later in the search order.

1. **/etc/opt/SUNWut/utslaunch_defaults.properties**
   This file contains site-wide default properties which are used if the user has not specified any. These properties override any application defaults.

2. **~/.utslaunch.properties**
   This file contains the user's default properties. These properties override application and site-wide default properties.

3. **/etc/opt/SUNWut/utslaunch_mandatory.properties**
   This file contains site-wide mandatory default properties which supersede any application, site-wide, or user defaults.

PROPERTIES

The supported application properties are listed below. For each property, the name, description, application default, and some examples are given.

Name — **utdetach.hotkey**

Description — Specifies the hotkey or key combination that disconnects the current session from the DTU the user is currently using. The value is a valid X keysym name preceded by one or more of the supported modifiers (Ctrl, Shift, Alt, Meta).

Application Default — Shift Pause (Hold down Shift and press the Pause key)

Name — **utsettings.hotkey**

Description — Specifies the hotkey or key combination that invokes the Sun Ray Settings dialog box. The value is a valid X keysym name preceded by one or more of the supported modifiers (Ctrl, Shift, Alt, Meta).

Application Default — Shift SunProps (Hold down Shift and press the Props key)

Examples:

- F3
- Shift F4
- Ctrl Shift Alt F5

EXAMPLES

Example 1: The following is a sample of the contents of a properties file.

    utdetach.hotkey=Shift Pause
    utsettings.hotkey=Shift SunProps

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutr
Stability Level	Evolving

SEE ALSO

utslaunch(1M), **utdetach(1M)**

NAME

      sunray - Sun Ray virtual device driver.

SYNOPSIS

      /dev/sunray

DESCRIPTION

      The **/dev/sunray** file refers to a pseudo-device driver that provides frame-buffer compatible information for configuring the **Xsun**(1) X server. The **sunray** driver's only function is to properly respond to the VIS_GETIDENTIFIER **ioctl**(2).

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWuto

SEE ALSO

      visual_io(7I)

NAME

utadem - Sun Ray audio driver emulator.

SYNOPSIS

/dev/utadem

DESCRIPTION

utadem provides a generic virtual audio interface to Sun Ray DTU's. The actual interface to the DTU is through a daemon that is session-aware. The daemon connects to **utadem** through a master port and is responsible for creating the slave device nodes which connect to normal audio applications.

API

The exact capabilities of the audio device emulated depend on the OS and audio daemon.

Solaris

Applications that normally open **/dev/audio** may use **utadem** as long as they have some way of selecting the audio device, such as through the **-d** *device* switch, or the AUDIODEV environment variable. Compliance to the standard **audio**(7I) interface is handled in the following manner:

Audio Data Formats

The data formats supported depend on the daemon. Please refer to the daemon documentation for its capabilities.

Audio Ports

Input and output audio ports are directly dependent on the Sun Ray DTU and not on the daemon. The daemon is capable of discovering the type and quantity of input ports available and report them in the **record.avail_ports** and **play.avail_ports** fields of the **audio_info** structure. Although the ports can be controlled directly, the actual audio output is generally a mix of multiple services, so the **play.gain** setting is the contribution of this audio device to the total experience. Since recording is exclusive of a single service, the **record.gain** and **record.balance** controls directly affect the hardware gain.

Sample Granularity

Since the **utadem** driver is working through a daemon which transfers the audio data over an interconnect, larger granularities and jitter in the reporting of sample counts is possible. At any given time, the reported input and output sample counts will vary from the actual sample count by no more than the size of the buffers it is transferring. Programs should not rely upon the absolute accuracy of the **play.samples** and **record.samples** fields of the **audio_info** structure.

Audio Status Change Notification

As described in **audio**(7I), it is possible to request asynchronous notification of state changes in an audio device.

The following OSS features are unsupported:

SNDCTL_DSP_REALTIME
Realtime capabilities are not supported.

SNDCTL_CAP_MMAP
Device DMA memory mapping is not supported.

/dev/audio
/dev/audio interface is not currently supported. Use /dev/dsp with appropriate format settings instead

ERRORS

Solaris

utadem errors are defined in the **audio(7I)** man pages. If the daemon has exited, further audio operations are no longer possible on the slave ports. Audio programs must exit in order to clear this error. New opens will return ENODEV. Data writes and **ioctl** operations will return ENXIO. Data reads will complete normally and then return end-of-file.

FILES

The following file is used:

• **/dev/utadem**
Master port for daemons.

The logical device name of the slave port depends on the daemon.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Solaris Availability	SUNWutu
MT-Level	Safe

SEE ALSO

utaudio(1), ioctl(2). attributes(5), audio(7I), streamio(7I)

NAME

      utparallel, utserial - Sun Ray serial and parallel port device driver emulators.

SYNOPSIS

      #include <sys/types.h>

      #include <fcntl.h>

   utserial

      #include <sys/termios.h>

      #include <termio.h>

   utparallel

      #include <sys/ecppio.h>

DESCRIPTION

      utserial is a tty-style interface that provides a generic virtual interface to USB serial adaptors connected to the Sun Ray DTU.

      utparallel is a parallel-style interface that provides a generic virtual interface to USB parallel adaptors connected to the Sun Ray DTU.

      utserial and **utparallel** are each loadable **STREAMS** drivers.

EXTENDED DESCRIPTION

      The actual interface to the DTU for each of these drivers is through the Sun Ray interconnect via either the **utseriald** daemon or the **utparalleld** daemon, each of which is session-aware. The daemons are connected to either **utserial** or **utparallel** through a master port and each is responsible for creating the slave device nodes through which normal applications will connect.

API

      Applications open a device file created by either **utseriald** or **utparalleld**. Device files created by **utseriald** comply to the **termio**(7I) interface and device files created by **utparalleld** comply to the **ecpp**(7D) interface. Hardware limitations in USB adaptors might prevent compliance with these interfaces.

FILES

      The following files are used:

        • **/dev/utserial**
          Master port for **utserial**

        • **/dev/utparallel**
          Master port for **utparallel**

ATTRIBUTES

      See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutu
MT-Level	Safe

SEE ALSO

      utseriald(1M), **utparalleld(1M)**, **termio(7I)**, **ecpp(7D)**

NAME

    utdisk, **utdiskctl** - Sun Ray Mass Storage device and controller drivers.

SYNOPSIS

    $UTDEVROOT/dev/dsk/*partition_name*

    $UTDEVROOT/dev/rdsk/*partition_name*

DESCRIPTION

    The **utdisk** and **utdiskctl** drivers provide the **dkio(7I)** interface to mass storage devices connected to Sun Ray appliances (DTUs).

    The actual interface to the DTU for each of these drivers is through the Sun Ray interconnect via the **utstoraged(1M)** daemon.

    The **utmountd** daemon mounts devices with Solaris-recognizable filesystems on them. See **utmountd(1M)** for more details.

API

    Applications open a device link created by **utstoraged**. Links to raw device nodes are created in the **$UTDEVROOT/dev/rdsk** directory, and links to block device nodes are created in the **$UTDEVROOT/ dev/dsk** directory.

    Device nodes created by **utstoraged** comply with the **dkio(7I)** interface. Hardware limitations in mass storage devices may prevent compliance with these interfaces.

    I/O requests to the device must be aligned on a 512-byte boundary, and all I/O request lengths must be in multiples of 512 bytes. Requests that do not meet these requirements will trigger an **EINVAL** error.

DEVICE STATISTICS SUPPORT

    Each device maintains I/O statistics for all partitions allocated for that device. For each partition, the driver accumulates reads, writes, bytes read, and bytes written. The driver also initiates hi-resolution time stamps at queue entry and exit points to enable monitoring of residence time and cumulative residence-length product for each queue. Statistics are disabled by default but may enabled in the utdiskctl.conf configuration file.

IOCTLS

    Refer to **dkio(7I)**.

ERRORS

    EACCES
        Permission denied.

    EBUSY
        The partition was opened exclusively by another thread.

    EFAULT
        The argument features a bad address.

    EINVAL
        Invalid argument.

    ENOTTY
        The device does not support the requested **ioctl()** function (see **dkio(7I)**).

    ENXIO
        The device did not exist when attempting operation.

    EAGAIN
        Resource temporarily unavailable.

    EINTR
        A signal was caught during the execution of the **ioctl()** function.

    ENOMEM
        Insufficient memory.

    EPERM
        Insufficient access permission.

    EIO
        An I/O error occurred.

CONFIGURATION

The **utdisk** and **utdiskctl** drivers can be configured by defining properties in the **utdiskctl.conf** file. The following properties are supported:

utdebug
The **utdisk** and **utdiskctl** drivers log errors to the system messages file. The value of the **utdebug** property determines the level of detail in the messages that are logged.

utdebug can accept the following values:

0 log errors only
1 log warnings
2 log error and warning details
3 log instance and device info
4 log operational sequence info
5 log everything else

The default value of this property is 2. Logging cannot be disabled.

utkstats
Setting **utkstats=1** causes **utdiskctl** to maintain I/O statistics for partitions. Set this value to zero to prevent the driver from recording partition statistics. This slightly reduces the CPU overhead for I/O, minimizes the amount of **sar(1)** data collected, and makes these statistics unavailable for reporting by **iostat(1M)**, even though the **-p/-P** option is specified. The default value for this property is 0.

FILES

The following files are used:

- **/usr/kernel/drv/utdiskctl.conf**
  Driver configuration files.

- **/var/adm/messages**
  System messages file.

- **$UTDEVROOT/dev/dsk/name**
  Block interface to disk or partition.

- **$UTDEVROOT/dev/rdsk/name**

  Raw interface to disk or partition.

  Where *name* is a string descriptive of the device type with a suffix denoting the partition number or UNIX slice number.

EXAMPLES

Example 1: Zip disk with PCFS filesystem (SPARC Platform)

The block device link for the Zip disk is given by **$UTDEVROOT/dev/dsk/zip1s2**.

The raw device link for the Zip disk is given by **$UTDEVROOT/dev/rdsk/zip1s2**.

Example 2: Hard disk with 1 UFS slice numbered 6 (SPARC Platform)

The block device link for the backup slice is given by **$UTDEVROOT/dev/dsk/disk1s2**.

The raw device link for the backup slice is given by **$UTDEVROOT/dev/rdsk/disk1s2**.

The block device link for the slice 6 is given by **$UTDEVROOT/dev/dsk/disk1s6**.

The raw device link for the slice 6 is given by **$UTDEVROOT/dev/rdsk/disk1s6**.

Example 3: Zip disk with PCFS filesystem in the first partition (IA platform)

The block device link for the Zip disk is given by **$UTDEVROOT/dev/dsk/zip1p0**.

The raw device link for the Zip disk is given by **$UTDEVROOT/dev/rdsk/zip1p0**.

The block device link for the first partition is given by **$UTDEVROOT/dev/dsk/zip1p1**.

The raw device link for the first partition is given by **$UTDEVROOT/dev/rdsk/zip1p1**.

Example 4: Hard disk with 1 UFS slice numbered 6 in the third partition (IA Platform)

The block device link for the full disk is given by **$UTDEVROOT/dev/dsk/disk1p0**.

The raw device link for the full disk is given by **$UTDEVROOT/dev/rdsk/disk1p0**.

The block device link for the third partition is given by **$UTDEVROOT/dev/dsk/disk1p3**.

The raw device link for the third partition is given by **$UTDEVROOT/dev/rdsk/disk1p3**.

The block device link for the backup slice is given by **$UTDEVROOT/dev/dsk/disk1s2**.

The raw device link for the backup slice is given by **$UTDEVROOT/dev/rdsk/disk1s2**.

The block device link for the slice 6 is given by **$UTDEVROOT/dev/dsk/disk1s6**.

The raw device link for the slice 6 is given by **$UTDEVROOT/dev/rdsk/disk1s6**.

ENVIRONMENT VARIABLES

UTDEVROOT points to a symbolic link of the device root for the Sun Ray appliance associated with a user's session.

ATTRIBUTES

See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutstk
MT-Level	Safe

SEE ALSO

utstoraged(1M), **utmountd(1M)**, **utdiskadm(1M)**, **dkio(7I)**, **iostat(1M)**, **sar(1)**

NOTES

Users have access rights only to storage devices connected to the Sun Ray DTU on which their session is active and only for as long as it is active. When the session is disconnected from the Sun Ray DTU whether through logout, hotdesking, server switching, or any other reason, ownership of the storage device is lost and all pending data transfers are aborted. This can result in the filesystem on the media getting corrupted and data getting lost.

Note: Users are strongly advised to use the **utdiskadm(1M)** command to prepare the device for removal before disconnecting their session.

On the SPARC platform, only the first partition on disks with multiple FAT partitions is mounted.

NAME

   utserial, utparallel - Sun Ray serial and parallel port device driver emulators.

SYNOPSIS

   #include <sys/types.h>
   #include <fcntl.h>

   utserial

   #include <sys/termios.h>
   #include <termio.h>

   utparallel

   #include <sys/ecppio.h>

DESCRIPTION

   utserial is a tty-style interface that provides a generic virtual interface to USB serial adaptors connected to the Sun Ray DTU.

   utparallel is a parallel-style interface that provides a generic virtual interface to USB parallel adaptors connected to the Sun Ray DTU.

   utserial and **utparallel** are each loadable **STREAMS** drivers.

EXTENDED DESCRIPTION

   The actual interface to the DTU for each of these drivers is through the Sun Ray interconnect via either the **utseriald** daemon or the **utparalleld** daemon, each of which is session-aware. The daemons are connected to either **utserial** or **utparallel** through a master port and each is responsible for creating the slave device nodes through which normal applications will connect.

API

   Applications open a device file created by either **utseriald** or **utparalleld**. Device files created by **utseriald** comply to the **termio**(7I) interface and device files created by **utparalleld** comply to the **ecpp**(7D) interface. Hardware limitations in USB adaptors might prevent compliance with these interfaces.

FILES

   The following files are used:

   • **/dev/utserial**
     Master port for **utserial**

   • **/dev/utparallel**
     Master port for **utparallel**

ATTRIBUTES

   See **attributes(5)** for descriptions of the following attributes:

Attribute Type	Attribute Value
Availability	SUNWutu
MT-Level	Safe

SEE ALSO

   utseriald(1M), **utparalleld(1M)**, **termio(7I)**, **ecpp(7D)**