**Technical white paper**

# Login Screen Customization

## HP ThinPro

## Table of contents

Overview ....................2
  Themes ....................2
Desktop background configuration ....................2
Common attributes ....................2
  name ....................2
  padding ....................2
  color ....................3
  alpha ....................3
  size ....................3
  position ....................3
  alignment ....................3
  context ....................4
Elements ....................4
  global ....................4
  gradient ....................4
  image ....................5
  polygon ....................6
  rectangle ....................6
  text ....................6
  tileset ....................8
Example background ....................9
Dynamic text elements ....................10
  Custom dynamic tokens ....................11
Login dialog box customization ....................11
  Appearance customization ....................11
  Additional customization ....................14
Logoff, sleep, restart, and shutdown dialog box customization ....................17
For more information ....................18

# Overview

HP ThinPro Background Manager can be used to choose between several predefined background themes and further customize the end-user experience, including the desktop background, login, and logout screens on the thin client. This white paper describes options available and how to construct a desktop background and login screen for your needs.

## Themes

The directory `/etc/hptc-zero-login/styles` contains the configuration files for each of the themes shown in Background Manager, with each style in its own subdirectory. For each connection type, there is one directory that specifies the style elements of the connection's background image and login dialog box style.

To create a custom theme, you can either modify the files in one of the theme directories or create a new directory to build your own theme. If you create a new theme directory, it is added to the list of themes visible from Background Manager.

There are up to three relevant files in each theme directory. If any of these files is missing, the default settings are used.

- **bgConfig.rtf** specifies the desktop background
- **desktop.qss** specifies the appearance of the login window
- **login.rtf** specifies other details about the login window, such as its position, buttons, and so on

# Desktop background configuration

Within a style directory, the file bgConfig.rtf specifies the desktop background. The syntax of a bgConfig.rtf file is in a stylesheet-like format with some or all the elements described as follows. Each element begins with the element type and then a set of attributes surrounded by braces, such as the following example:

```
global {
    color: 666666;  # Dark gray
    padding: 20;    # 20 pixels
}
```

Any number of image, polygon, rectangle, or text elements can be specified. If any gradients are specified, only the last gradient element is used to specify the color of the desktop background; otherwise, the color specified in the global section is used. Any line that begins with a number sign (#) is considered a comment and is ignored, as are blank lines. Text that follows a semicolon and begins with # is also treated as a comment, such as the previous example.

Each element is assigned a set of attributes such as size, color, and position. Each attribute is specified by the attribute name followed by a colon, followed by its values, and then followed by a semicolon, all on a single line. Some of these attributes are common to many element types. Those attributes are described in Common attributes.

The advantage of building a desktop background from elements rather than using a single large, scaled image is that this configuration method accommodates a wide variety of monitor sizes, aspect ratios, and configurations, so that it works on a configuration with four 4K monitors at a 16:9 aspect ratio the same as it does on a single 1024×768 4:3 monitor.

# Common attributes

## name

Parameters: A string

Example: `name: itemName;`

Default: none

Use: Specifies a string to associate with the element. It is used only in debugging output, such as when a syntax or value error is found in attribute parsing.

## padding

Parameters: An absolute (pixel) or percentage value

Example: `padding: 20;`

Example: `padding: 4%;`

Default: none

Use: Positions an object on the screen as if the screen were smaller on all sides by the padding value. For instance, if an element would normally be placed at 0,0, with a padding of 20 it would be placed at 20,20 instead. If specified in the global element, the padding value applies to all subsequent elements, leaving an empty gutter around the screen edge, unless those elements override the global padding value with their own padding value.

## color

Parameters: RRGGBB 6-digit hex value or rrr,ggg,bbb 0–255,0–255,0–255 form

Example: `color: ff8800;`

Example: color: `255,204,192;`

Default: 255,255,255 (white)

Use: Specifies the color of the element. Note that no '#' precedes the hex value.

## alpha

Parameter: 0–255 integer

Example: `alpha: 127;`

Default: 255 (fully opaque)

Use: Specifies the opacity of the element. 255 is fully opaque; 0 is fully transparent, rendering the element invisible. Elements are layered over the background in the order that they are defined.

## size

Parameters: WW×HH, where WW is width in absolute pixels or in percentage of screen width and HH is height in absolute pixels or in percentage of screen height.

Example: `size: 400×10%;`

Default: the natural size of the element; that is, the pixel size of an image.

Use: Specifies the size of the element. Elements are scaled to match the specified size.

## position

Parameters: XX,YY, where XX is a position in absolute pixels or in percentage of screen width and height.

Example: `position: 400,90%;`

Default: 0,0 (the upper left)

Use: Specifies the position of the element. Alignment can be used to specify an anchor point. Position then specifies how the element is aligned relative to that anchor point. See alignment.

## alignment

Parameters: [left|hcenter|right] [top|vcenter|bottom]

Example: `alignment: left bottom;`

Default: hcenter vcenter (the element is centered at the given position)

Use: Specifies an anchor point for the element. Position specifies how the element is aligned relative to that anchor point. For instance, an element with a position of 90%,70% and alignment of right bottom is positioned so that its right edge is at 90% of the width of the screen and its bottom edge is at 70% of the height of the screen.

### context

Parameters: [login|desktop|all]

Example: `context: login;`

Default: all

Use: Specifies whether the element should be shown only on the login screen for the protocol, only on the desktop background for the protocol (if any), or on both. Only some protocols (such as Citrix XenDesktop) use a desktop background.

## Elements

Each element is specified by the element type, followed by a left brace. Each attribute is listed on a single line. Finally, the element is closed by a right brace on its own line.

For example:

```
global {

color: 333344;

padding: 16;

}
```

### global

Use: Specifies global background or padding values.

Common attributes recognized: name, color, padding

- **color**—Specifies the solid background color of the screen, if no gradients are specified
- **padding**—Specifies the default padding for all subsequent elements

Custom attributes: none

### gradient

Use: Specifies a full-screen gradient for use in the background.

Common attributes recognized: name, context, position (for radial gradient center)

Custom attributes:

- **type**

  Parameter: [linear|radial]

  Example: `type: linear;`

  Default: `linear`

  Use: Specifies whether a gradient is oriented linearly or radially. Linear gradients can be either horizontal or vertical; the coordinates are a fraction of width or height. Radial gradients are centered on the screen; the coordinates are a fraction of the distance to the screen edge (top and bottom or left and right).

- **axis**

  Parameter: [height|width]

  Example: `axis: width;`

  Default: `height`

  Use: Specifies the direction of a linear gradient (top-to-bottom or left-to-right) or whether the radius of a radial gradient is half the screen height or half the screen width.

- **metric**

  Parameter: [linear|squared]

  Example: `metric: linear;`

Default: `squared`

Use: Specifies whether the color interpolation between points for radial gradients is done with a dx2+dy2 distance calculation (squared) or the square root of the number (linear). Squared interpolation can be drawn more quickly.

- **colors**

  Parameter: A space-separated list of [value, color] pairs, where the value is a 0.0 to 1.0 floating point fraction of the axis of measurement (that is, the width of the screen in a linear width-axis gradient) and the color is the color of the gradient at that point. The value runs top-to-bottom for vertical linear gradients, left-to-right for horizontal linear gradients, and center-to-edge for radial gradients. Colors are specified as either six-digit hex or three 0 to 255 comma-separated values.

  Example: `colors: 0.0,000000 0.5,996600 0.9,255,255,255;`

  Use: Specifies points and colors. Between these points, the respective colors are interpolated along the linear or radial axis. If no values are given, the colors are displayed evenly spaced on the axis between 0.0 and 1.0. If the first fractional value is greater than 0.0, the first color is used in the space between the screen edge and the first value. Likewise, if the last fractional value is less than 1.0, the last color is used between the last value and the screen edge. Values must be in increasing values, although a value can be repeated for a sharp transition.

  For example, the colors attribute 0.0,CCCCCC 0.5,EEEEEE 0.5,660000 1.0,330000 in a vertical linear gradient specifies a gradient between light grays on the upper half of the display and dark reds on the lower half of the display.

- **dithered**

  Parameter: [true|false]

  Example: `dithered: true;`

  Default: `false`

  Use: Eliminates color banding if a gradient displays this visual artifact. Radial gradients with a squared metric attribute do not support dithering.

## image

Use: Specifies an image to overlay a portion of the background.

Common attributes recognized: name, size, alpha, position, alignment, context

Custom attributes:

- **source**

  Parameter: file path

  Example: `source: /writable/misc/Company_logo.png;`

  Use: Specifies the absolute pathname to the image file. Many formats are supported, including PNG, JPG, and GIF. The image can have transparent regions.

- **proportional**

  Parameter: [true|false]

  Example: `proportional: false;`

  Default: `true`

  Use: When true, the image aspect ratio is maintained when the image is scaled to fit within a specified rectangle if any. When false, the image scales non-proportionally to fit exactly the specified size.

- **tiled**

  Parameter: [true|false]

  Example: `tiled: true;`

  Default: `false`

  Use: When true, if the image is positioned and sized normally, it is also duplicated to cover the screen.

## polygon

Use: Specifies a polygon within a rectangular region.

Common attributes recognized: name, size, color, alpha, position, alignment, context

Custom attributes:

- **points**

  Parameter: The points of the polygon, on a 0.0 to 1.0 scale, relative to the region at "position" of size "size".

  Example: `0.2,0.0 0.5,0.5 0.8,0.0 1.0,1.0 0.0,1.0;`

  Use: Specifies the position and size of a polygon. Enter the points in the order of the polygon's location in the rectangular region. 0.0,0.0 would be the region's upper left; 1.0,1.0, the lower right.

- **filled**

  Parameter: true|false

  Example: `filled: true;`

  Use: If true (the default), the polygon is filled in; otherwise, only the outline is drawn.

## rectangle

Use: Specifies a rectangle to overlay a portion of the background.

Common attributes recognized: name, size, color, alpha, position, alignment, context

Custom attributes:

- **filled**

  Parameter: [true|false]

  Example: `filled: true;`

  Use: If true (the default), the rectangle is filled in; otherwise, only the outline is drawn.

## text

Use: Specifies a string of text to overlay a portion of the background.

Common attributes recognized: name, size, color, alpha, position, alignment, context

Custom attributes:

- **text-locale**

  Parameter: A string of localized text

  Example: `text-de_DE: Dieser Text ist auf Deutsch.;`

  Use: Specifies a supported locale and a string to be displayed. The string is displayed when the thin client is in this locale. The supported locales are the same as those supported by HP ThinPro: de_DE (German), en_US (English), es_ES (Spanish), fr_FR (French), ja_JP (Japanese), ko_KR (Korean), ru_RU (Russian), zh_CN (Simplified Chinese), and zh_TW (Traditional Chinese). The file encoding is UTF-8. If no matching text-locale attribute is found, the generic text attribute is used instead. See Dynamic text elements.

- **text**

  Parameter: Default text

  Example: `text: text: This will be shown on the screen.`

  Use: Specifies a text string to be used, if no localized text is specified.

**Note**
The text rendering engine does not support HTML-style markup. It does support using \n to display multiple lines of text in one element. See Dynamic text elements.

- **font-locale**

  Parameter: locale-specific fontName

  Example: `font-ja_JP: kochi-gothic;`

  Use: Specifies a font to be used when the string is rendered in the specified locale. The supported locales are the same as those supported by HP ThinPro: de_DE (German), en_US (English), es_ES (Spanish), fr_FR (French), ja_JP (Japanese), ko_KR (Korean), ru_RU (Russian), zh_CN (Simplified Chinese), and zh_TW (Traditional Chinese). The name must match one of the fonts available in the folder /usr/share/fonts/truetype. For Japanese text, it might be necessary to specify kochi-gothic; for Simplified Chinese text, uming.

- **font**

  Parameter: fontName

  Example: `font: DejaVuSerif-Bold;`

  Default: DejaVuSerif

  Use: Specifies a font to be used, if no localized font is specified. The name must match one of the fonts available in the folder /usr/share/fonts/truetype.

- **font-size**

  Parameter: A pixel value (for example, 20) or percentage of screen height (for example, 5%) or a point size (for example, 12 pt)

  Example: `font-size: 12pt;`

  Example: `font-size: 100;`

  Example: `font-size: 5%;`

  `Use: Specifies the defau`lt size of the font. The text can be further scaled, if size, max-width, and/or max-height are specified.

- **max-width**

  Parameter: A size in pixels or in percentage of screen width

  Example: `max-width: 90%;`

  Example: `max-width: 500;`

  Use: Specifies that the string is scaled down to fit within the width, if the string is wider than the width.

- **max-height**

  Parameter: A size in pixels or in percentage of screen height

  Example: `max-height: 64;`

  Example: `max-height: 200;`

  Use: Specifies that the string is scaled down to fit within the height, if the string is taller than the height.

- **shadow-color**

  **Note**
  Available in HP ThinPro 6.0 or later.

  Parameter: RRGGBB 6-digit hex value or rrr,ggg,bbb 0–255,0–255,0–255 form

  Example: `shadow-color: ff8800;`

  Example: `shadow-color: 255,204,192;`

  Default: 0,0,0 (black)

  Use: Specifies the color of the text shadow. A value for shadow-offset must be set for the shadow to be drawn.

- **shadow-offset**

  **Note**
  Available in HP ThinPro 6.0 or later.

  Parameter: X or X,Y, where the numbers are the offset in percentage of text height

  Example: `shadow-offset: 4;`

  Example: `shadow-offset: 5,3;`

  Default: 0

  Use: Specifies an offset to draws the text in the shadow-color value underneath the main text, if not zero. If only one offset is specified, it is used for both X and Y. If both offsets are zero, no shadow is drawn.

## tileset

**Note**
Available in HP ThinPro 6.2 or later.

Use: Specifies a set of adjacent images to use as the desktop background.

Common attributes recognized: name

- **name**—Specificies the set of images to be scaled and cropped a to the layout and size of the monitors to fill the desktop background without distortion of the images (as best as possible).

  For instance, if _0_0, _1_0, and _2_0 tiles are provided, but only two monitors of roughly the same aspect ratio of the images are connected to the computer, the _0_0 and _1_0 images are conjoined, scaled upward (if necessary), and cropped to match the desktop. A tileset can only be used at the base of the background image; it cannot be superimposed over other content.

Custom attributes:

- **source-list**

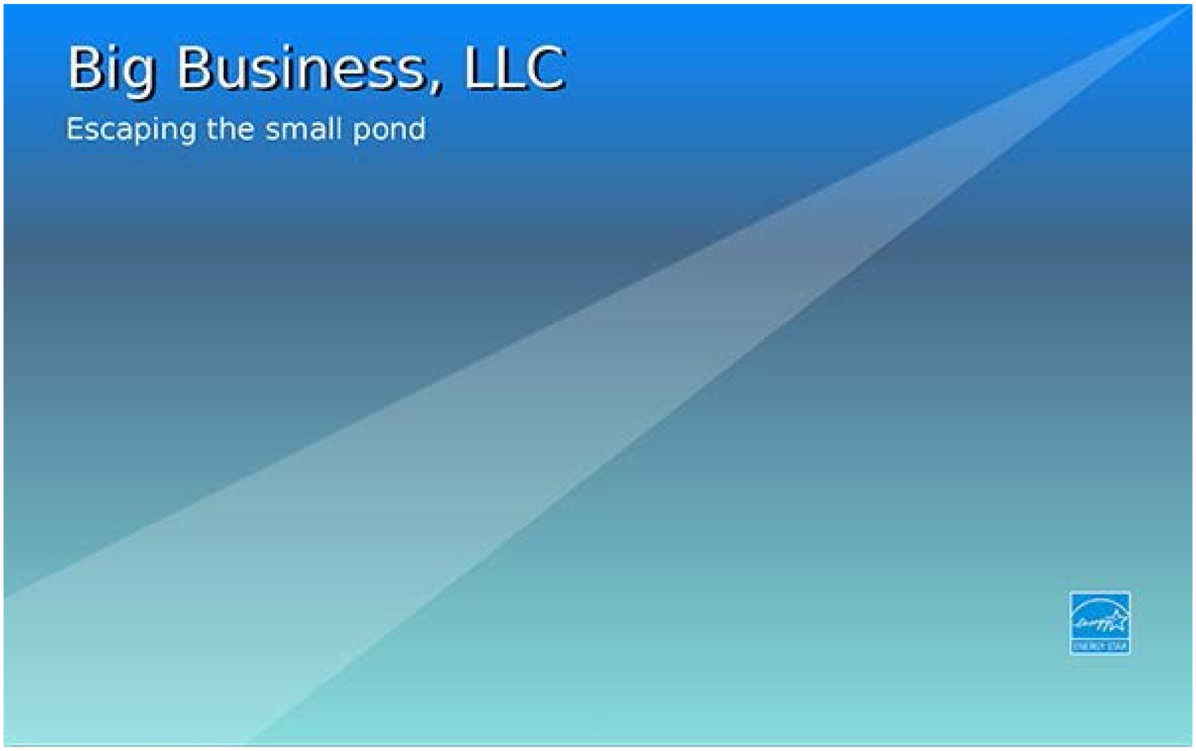
  Parameter: A set of file paths, each with a _x_y-type suffix

  Example:
  ```
  source-list: /usr/share/wallpaper/mybackground_0_0.jpg,
  /usr/share/wallpaper/mybackground_1_0.jpg,
  /usr/share/wallpaper/mybackground_0_1.jpg,
  /usr/share/wallpaper/mybackground_1_1.jpg
  ```

  Use: Specifies the absolute path to the set of image files. Many file formats are supported, including PNG, JPEG, and GIF. Each file must have a _x_y-type suffix, starting with _0_0, to specify the image tile position relative to the other tiles. For example, the suffix _2_1.png means that the file image is displayed as the third tile to the right and the second tile above the _0_0 tile. All images must be of the same type and size.

## Example background

To create the following custom desktop background:



1. Enter the gradient attribute:

   ```
   gradient {
   colors: 0.0,6688ff 0.33,446688 1.0,88dddd;
   }
   ```

2. Enter the two text attributes:

   ```
   text {
   text: Big Business, LLC; color: white;
   font: unfonts-core; max-height: 300;
   max-width: 800; alignment: left vcenter;

   font-size: 300;
   position: 100,100;
   shadow-offset: 5,3;
   }

   text {
   text: Escaping the small pond; color: white;
   font: DejaVuSans-ExtraLight; max-height: 150;
   max-width: 600; alignment: left vcenter; font-size: 200;
   position: 100,200;
   }
   ```

3. Enter the diagonal, partially transparent polygon:

   ```
   polygon {
   size: 100%×100%;
   points: 0.0,0.8 0.0,1.0 0.2,1.0, 1.0,0.0;
   color: ffffff; filled: true; alpha: 44;
   }
   ```

4. Enter the image attribute for the ENERGY STAR logo in the lower-left corner:

   ```
   image {
   source:   /usr/share/pixmaps/energy_star_logo-128x128.png; size: 100×100;
   position: -100,-150; alignment: bottom right;
   }
   ```

## Dynamic text elements

**Note**
Available in HP ThinPro 6.0 or later.

HP ThinPro Background Manager can overlay text information on the desktop background using a simple interface. For more customization, you can use dynamic text elements within the normal text elements in the bgConfig.rtf file.

To add dynamic text elements:

- In a normal text element, add a predefined dynamic text element. Dynamic text elements have two percentage signs on either side.

  For example, the text element “`Your IP Address is %%IP%%`” is resolved to something similar to “`Your IP Address is 192.168.1.1`”.

The following dynamic text elements are predefined:

- **%%hostname%%**—Hostname of the thin client
- **%%hostname_full%%**—Fully qualified domain name of the thin client
- **%%domainname%%**—Domain name of the thin client
- **%%IP%%**—IP address of the thin client
- **%%DHCP_server%%**—DHCP server for the network
- **%%DNS_server%%**—DSN server for the network
- **%%gateway%%**—Gateway machine for the network
- **%%boot_time%%**—Date and time the thin client was last rebooted
- **%%df%%**—Amount of free disk space on the main USB drive partition
- **%%network_speed%%**—Speed of the network interface
- **%%eth0_mac%%**—MAC address of the wired network interface
- **%%wlan0_mac%%**—MAC address of the wireless network interface
- **%%hw_model%%**—Thin client model; for example, t620
- **%%thinpro_version%%**—Thin client software release; for example, 6.0.0
- **%%time%%**—Date and time the system information was last updated

### Custom dynamic tokens

If you to enter dynamic text that does not have a predefined dynamic text element, you can create a new one. Do one of the following:

- Modify the script `/etc/bginfo/bginfo.aux.sh`.
- Add a new executable script or program to the directory /etc/bginfo/commands, with the name matching the string inside the percentage signs.

For instance, you create a new script named /etc/bginfo/commands/timezone. Whenever `%%timezone%%` is added to the text elements, the system's time zone abbreviation ("date+%Z") is displayed.

Verify that the file is marked executable. If not, it is ignored.

## Login dialog box customization

For the Smart Zero configuration, the login dialog box is central to user interaction. For the ThinPro configuration, it is launched only when a connection is launched and the authentication parameters are not already supplied for the connection. In both the ThinPro and Smart Zero configurations, it is possible to customize the login dialog box.

**Note**
In ThinPro 6.2 and later, many of these customization options can also be applied to logout, sleep, restart, and shutdown dialog boxes. The customization files for those dialog boxes are in the theme directory labeled **shutdown**.

### Appearance customization

The .qss files in the theme directory (other than those named *-sample.qss, which are ignored) contain the customization details about the graphics of the login dialog box. The format of these files is a Qt stylesheet. This document does not try to enumerate all the Qt stylesheet options and possibilities. For more information on the Qt stylesheet customization options, see the Qt documentation.

The QDialog and individual items within it can be customized. Some of the more useful or interesting options are as follows:

- `background-color: rgb(255,0,0);`—Specifies that the dialog box background is bright red
- `min-width: 450px;`—Secifies the minimum width of the dialog box
- `border-color: black;`—Specifies that the dialog box has a black border
- `border-width: 3px;`—Specifies that the dialog box border is three pixels wide
- `font-size: 12pt;`—Specifies that the font size is 12 point
- `font-color: blue;`—Specifies that the font color is blue
- `qproperty-text: “Login Here”;`—Specifies that Login Here is displayed as the dialog box header

**Note**
To more easily and reliably customize text, use the .rtf files. See Additional customization.

Individual items within the dialog box can be customized. For example, the following element adds width to the Log In button and specifies that it has a bright blue background with black text:

```
QPushButton#connectButton { background-color: rgb(102, 147, 190); color: black;
padding-left: 160px; padding-right: 160px;
}
```

Other individual customizable items that might be of interest are listed as follows. (Items shown with an asterisk might be available only with HP ThinPro 6.0 and later.)

- QLabel#loginIcon*
- QLabel#loginHeader* (previously QLabel#title)
- QLabel#userLabel*
- QLabel#passwordLabel*
- QPushButton#showPasswordButton
- QLabel#capsLockLabel
- QLabel#domainLabel*
- QCheckBox#smartcardCheck*
- QCheckBox#rememberCheck*
- QPushButton#connectButton or QPushButton#loginButton
- QPushButton#cancelButton*

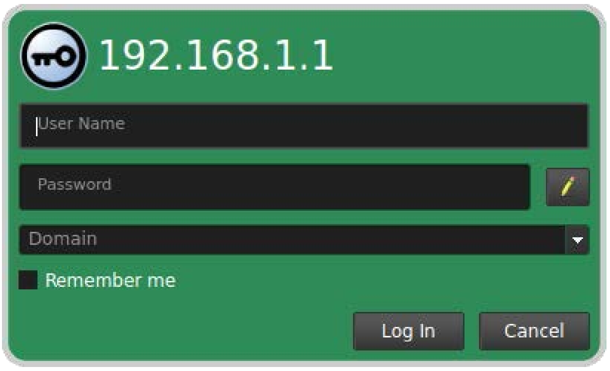
The following items are available with HP ThinPro 6.2 and later:

- QLabel#leftPixmap—Creates a pixmap to the left of the credentials
- QLineEdit#username
- QLineEdit#password
- QLineEdit#domain

**Example**

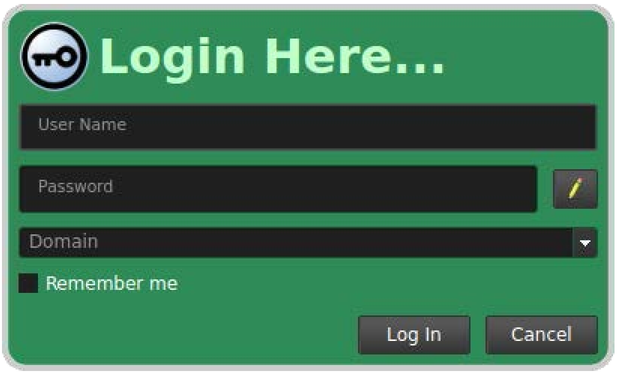
For example, you might customize the center area of the dialog box, as follows:

```
QDialog {

/* Sea green dialog background color */ background-color: rgb(46,139,87);

/* Rounded, thicker borders */ border-width: 6px;

border-radius: 16px; border-color: #CCCCCC; border-style: solid;

/* Make sure it is at least 400 pixels wide */ min-width: 400px;

}
```



Then, you might customize the login header, as follows:

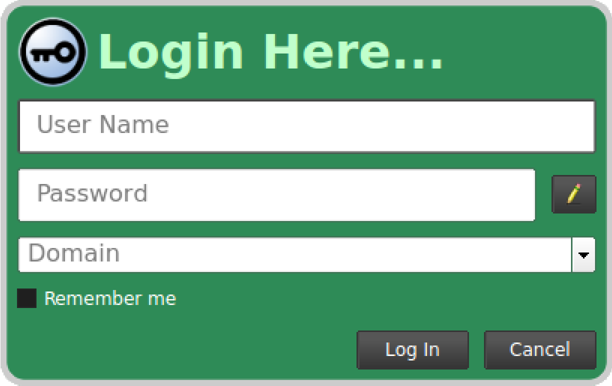
```
QLabel#loginHeader {
/* Change the login text at the top */ qproperty-text: "Login Here…";
color: rgb(192,255,204); font-size: 24pt;
font-weight: bold;
}
```



**Note**
Text that is overridden in the .qss file is not localized when the locale changes. You can manually localize this string. (It is UTF-8 encoded.)

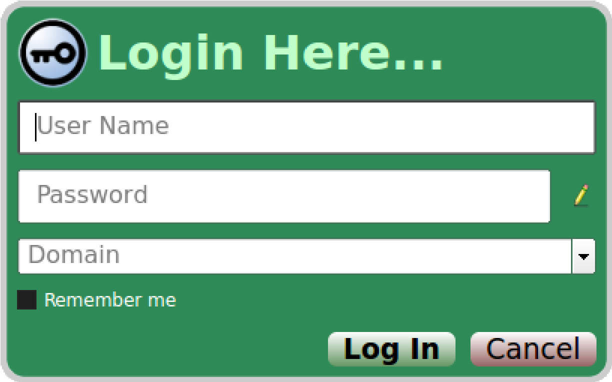
You might customize the background on the text-entry areas, as follows:

```
/* Black text on white background, 12pt font */ QLineEdit, QComboBox{
background-color: white; color: black;
font-size: 12pt;
}
```



You might customize the Connect (or Log In) button and Cancel button, as follows:

```
QPushButton { border-radius: 6; color: black; font-size: 14pt;

}

QPushButton#connectButton {

/* A shaded gradient from white to medium green */ background:
qlineargradient(x1:0.5,y1:0, x2:0.5,y2:1, stop:0 white, stop:1 rgb(100,150,100));

font-weight: bold;

}

QPushButton#cancelButton {

/* A shaded gradient from white to medium red */ background:
qlineargradient(x1:0.5,y1:0, x2:0.5,y2:1, stop:0 white, stop:1 rgb(150,100,100));

}
```



The sample .qss files provide more examples of the use of these options.

## Additional customization

The.rtf files in the theme directory (other than those named *-sample.rtf, which are ignored) provide additional customization options for the login dialog box, as follows.

**Note**
True or false can be used in place of 0 and 1, respectively.

- `window-position: X,Y`—Specifies the position of the login dialog box. The values can either be absolute offsets from the upper-left corner or percentages of the full screen width and height.
- `button-location: [right|left]`—Specifies whether the buttons are left or right aligned.
- `title-enabled: [0|1]`—Specifies whether the title field is displayed.
- `title-text: textString`—Specifies text to be used as the title instead of the default text.
- `icon-enabled: [0|1]`—Specifies whether the dialog box icon is displayed.
- `username-enabled: [0|1]`—Specifies whether the username box is displayed.
- `password-enabled: [0|1]`—Specifies whether the password box is displayed.
- `domain-enabled: [0|1]`—Specifies whether the domain box is displayed.
- `smartcard-enabled: [0|1]`—Specifies whether the smart card checkbox is displayed.
- `rememberme-enabled: [0|1]`—Specifies whether the Remember Me checkbox is displayed.
- `loginbutton-enabled: [0|1]`—Specifies whether the Log In button is displayed.
- `cancelbutton-enabled: [0|1]`—Specifies whether the Cancel button is displayed.
- `icon-path: path`—Specifies the path to the icon to be displayed in the window decorations. This option is available only in HP ThinPro 6.0 or later.

The following options are available only in HP ThinPro 6.2 or later:

**Note**

The login dialog box can be configured to use strings in a language not supported by HP ThinPro. For instance, the login dialog box can be displayed in Dutch if Dutch translations were provided for the *-text fields. The strings provided for a *-text field overwrite the default text, and the text does not change if the system language is modified.

- `decorations-visible: [0|1]`—Specifies whether the dialog box is displayed with window borders and decorations such as expand, iconize, and close.
- `disabled-visible: [0|1]`—If set to true, disabled fields are displayed as read-only.
- `placeholders-enabled: [0|1]`—Specifies whether placeholder text is displayed in any empty fields.
- `left-pixmap-path: path`—Specifies the absolute path to an image to display to the left of the credentials fields.
- `left-pixmap-visible: [0|1]`—Specifies whether the left pixmap is displayed.
- `left-pixmap-size: wxh`—Specifies the size of the left pixmap.
- `pixmap-path: path`—Specifies the absolute path to an image to display to the left of the dialog box title.
- `pixmap-visible: [0|1]`—Specifies whether the pixmap is displayed.
- `pixmap-size: wxh`—Specifies the size of the pixmap.
- `title-visible: l0|1]`—Specifies whether the title field is displayed. This is the same function as title-enabled.
- `username-text: string`—Specifies the text to display to the left of the username entry field or the placeholder text for that field, if placeholders-enabled is set to true.
- `username-enabled: [0|1]`—Specifies whether the username entry field is read-only or editable.
- `username-visible: [0|1]`—Specifies whether the username entry field is displayed.
- `password-text: string`—Specifies the text to display to the left of the password entry field or the placeholder text for that field, if placeholders-enabled is set to true.
- `password-enabled: [0|1]`—Specifies whether the password entry field is read-only or editable.
- `password-visible: [0|1]`—Specifies whether the password entry field is displayed.
- `domain-text: string`—Specifies the text to display to the left of the domain entry field or the placeholder text for that field, if placeholders-enabled is set to true.
- `domain-enabled: [0|1]`—Specifies whether the domain entry field is read-only or editable.
- `domain-visible: [0|1]`—Specifies whether the domain entry field is displayed.
- `domain-label-text: string`—In the system security settings, you can configure the login dialog box to display a domain entry field; otherwise, the user can specify their domain as part of the username entry field. When the domain entry field is available, this string specifies whether the domain field is displayed as read-only.
- `domain-help-text: string`—Specifies the help text that is displayed when the domain **?** button is clicked.
- `smartcard-text: string`—Specifies the string that is displayed next to the smartcard check box.
- `smartcard-enabled: [0|1]`—Specifies whether the smartcard check box is read-only.
- `smartcard-visible: [0|1]`—Specifies whether the smartcard check box is displayed.
- `rememberme-text: string`—Specifies the string that is displayed next to the remember me check box.
- `rememberme-enabled: [0|1]`—Specifies whether the remember me check box is read-only.
- `rememberme-visible: [0|1]`—Specifies whether the remember me check box is displayed.
- `cancelbutton-text: string`—Specifies the text to display on the cancel button.
- `cancelbutton-enabled: [0|1]`—Specifies whether the cancel button is read-only.
- `cancelbutton-visible: [0|1]` —Specifies whether the cancel button is displayed.
- `loginbutton-text: string`—Specifies the text to display on the log in button.
- `loginbutton-enabled: [0|1]`—Specifies whether the log in button is read-only.
- `loginbutton-visible: [0|1]`—Specifies whether the log in button is displayed.

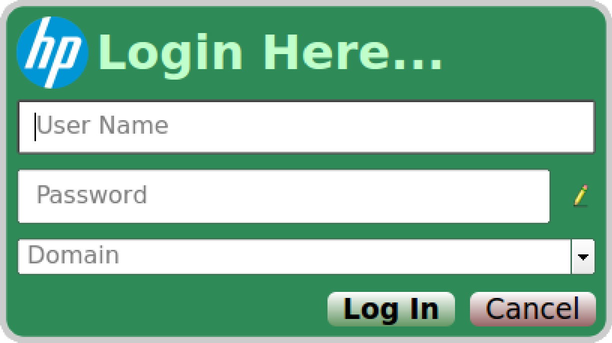
**Example**

This example uses the example created in Appearance customization.

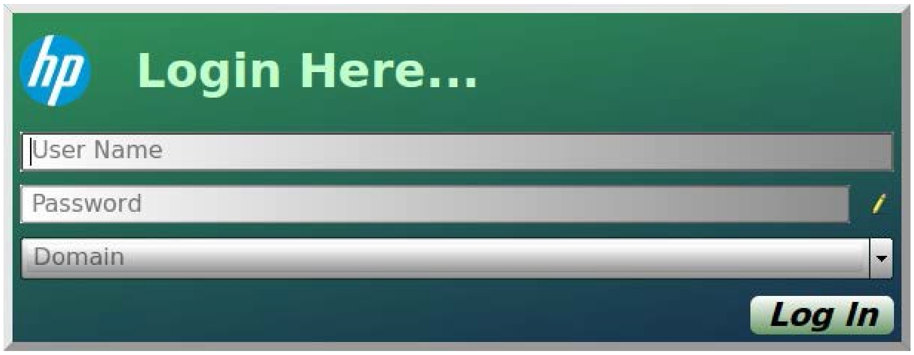
1. Enter these lines into the default login.rtf file:

```
rememberme-enabled: 0
icon-path: /usr/share/pixmaps/HP_Logo_300.png
```

   The Remember Me check box is not displayed and the default login icon is replaced with the HP logo.



2. Configure the following elements in the default.qss and login.rtf files:
   - Gradient backgrounds for the text-entry areas and the dialog box
   - An outset border
   - Padding around the title
   - Extra width to the dialog box
   - Padding and minimum heights to the entry boxes
   - Hide the cancel button
   - Modify the log in button size and style



The sample .rtf files provide more examples of the use of these options.

## Logoff, sleep, restart, and shutdown dialog box customization

You can customize the logoff, sleep, restart, and shutdown dialog boxes using the files in the **shutdown** theme directory.

Each customizable field is preceded by a prefix string specifying the dialog box in question; for example, strings that begin with “logoff-*” modify the behavior of the logoff dialog box. For more information, see the shutdown-sample.rtf file. These prefix strings are represented by *`prefix`* in the following list.

- *`prefix`*`-dialog-visible: [0|1]`—If set to 0, the operation proceeds without prompting the user for confirmation.
- *`prefix`*`-decorations-visible: [0|1]`—Specifies whether window borders and decorations (such as expand, iconize, and close) are displayed.
- *`prefix`*`-icon-path: path`—Specifies the absolute path to an image to use as the window’s icon.
- *`prefix`*`-title-text: string`—Specifies the string to use as the window title.
- *`prefix`*`-pixmap-path: path`—Specifies the absolute path to an image to be displayed next to the prompt text.
- *`prefix`*`-pixmap-visible: [0|1]` —Specifies whether the pixmap is displayed.
- *`prefix`*`-prompt-text: string`—Specifies the text of the prompt displayed to the user.
- *`prefix`*`-skip-visible: [0|1]` —Specifies whether to display the check box that lets the user skip subsequent invocations of the dialog box.
- *`prefix`*`-skip-text: string`—Specifies the string to display next to the skip checkbox.
- `prefix-progress-visible: [0|1]` —Specifies whether a countdown is used. If the countdown expires, the operation proceeds.
- *`prefix`*`-progress-timeout: integer`—Specifies the number of seconds in the countdown.
- *`prefix`*`-yes-visible: [0|1]` —Specifies whether the yes button is displayed.
- *`prefix`*`-yes-text: string`—Specifies the text displayed on the yes button.
- *`prefix`*`-no-visible: [0|1]`—Specifies whether the no button is displayed.
- *`prefix`*`-no-text: string`—Specifies the text displayed on the no button.

## For more information

For more information about HP thin clients, go to the following websites:

- HP thin client software and operating system website: http://www8.hp.com/us/en/thin-clients/software-and-os.html
- HP support website: http://www.hp.com/support (search for the thin client model to find the support page for that particular model)
  - For documentation, see **User Guides** on the thin client's support page.
  - For software updates and add-ons, see **Software and Drivers** on the thin client's support page.

**Sign up for updates**
**hp.com/go/getupdated**

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Document Part Number: 903900-002

Second Edition: December 2017

First Edition: June 2016