NAME

Alacritty - TOML configuration file format.

SYNTAX

Alacritty's configuration file uses the TOML format. The format's
specification can be found at https://toml.io/en/v1.0.0.

GENERAL

This section documents the root level of the configuration file.

import = ["<string>",]

   Import additional configuration files.

   Imports are loaded in order, skipping all missing files, with the
   importing file being loaded last. If a field is already present in a
   previous import, it will be replaced.

   All imports must either be absolute paths starting with /, or paths
   relative to the user's home directory starting with ~/.

   Example:

       import = [
       "~/.config/alacritty/base16-dark.toml",
       "~/.config/alacritty/keybindings.toml",
       ]

shell = "<string>" | { program = "<string>", args = ["<string>",] }

   You can set shell.program to the path of your favorite shell, e.g.
   /bin/zsh. Entries in shell.args are passed as arguments to the shell.

   Default:

       Linux/BSD/macOS: $SHELL or the user's login shell, if $SHELL
       is unset
       Windows: "powershell"

   Example:

       [shell]
       program = "/bin/zsh"
       args = ["-l"]

working_directory = "<string>" | "None"

   Directory the shell is started in. When this is unset, or "None", the
   working directory of the parent process will be used.

   Default: "None"

live_config_reload = true | false

   Live config reload (changes require restart)

   Default: true

ipc_socket = true | false # (unix only)

   Offer IPC using alacritty msg

   Default: true

ENV

All key-value pairs in the [env] section will be added as environment
variables for any process spawned by Alacritty, including its shell. Some
entries may override variables set by alacritty itself.

Example:

   [env]
   WINIT_X11_SCALE_FACTOR = "1.0"

WINDOW

This section documents the [window] table of the configuration file.

dimensions = { columns = <integer>, lines = <integer> }

   Window dimensions (changes require restart).

```
                    Number of lines/columns (not pixels) in the terminal. Both lines and
                    columns must be non-zero for this to take effect. The number of
                    columns must be at least 2, while using a value of 0 for columns and
                    lines will fall back to the window manager's recommended size

                    Default: { columns = 0, lines = 0 }

           position = "None" | { x = <integer>, y = <integer> }

                    Window startup position.

                    Specified in number of pixels.

                    If the position is "None", the window manager will handle placement.

                    Default: "None"

           padding = { x = <integer>, y = <integer> }

                    Blank space added around the window in pixels. This padding is scaled
                    by DPI and the specified value is always added at both opposing sides.

                    Default: { x = 0, y = 0 }

           dynamic_padding = true | false

                    Spread additional padding evenly around the terminal content.

                    Default: false

           decorations = "Full" | "None" | "Transparent" | "Buttonless"

                    Window decorations.

                    Full

                        Borders and title bar.

                    None

                        Neither borders nor title bar.

                    Transparent (macOS only)

                        Title bar, transparent background and title bar buttons.

                    Buttonless (macOS only)

                        Title bar, transparent background and no title bar buttons.

                    Default: "Full"

           opacity = <float>

                    Background opacity as a floating point number from 0.0 to 1.0. The
                    value 0.0 is completely transparent and 1.0 is opaque.

                    Default: 1.0

           blur = true | false # (works on macOS/KDE Wayland)

                    Request compositor to blur content behind transparent windows.

                    Default: false

           startup_mode = "Windowed" | "Maximized" | "Fullscreen" | "SimpleFullscreen"

                    Startup mode (changes require restart)

                    Windowed

                        Regular window.

                    Maximized

                        The window will be maximized on startup.

                    Fullscreen

                        The window will be fullscreened on startup.

                    SimpleFullscreen (macOS only)

                        Same as Fullscreen, but you can stack windows on top.

                    Default: "Windowed"

           title = "<string>"

                    Window title.

                    Default: "Alacritty"

           dynamic_title = true | false

                    Allow terminal applications to change Alacritty's window title.

                    Default: true

           class = { instance = "<string>", general = "<string>" } # (Linux/BSD only)

                    Window class.
                    On Wayland, general is used as app_id and instance is ignored.

                    Default: { instance = "Alacritty", general = "Alacritty" }
```

```
decorations_theme_variant = "Dark" | "Light" | "None"

    Override the variant of the System theme/GTK theme/Wayland client side
    decorations. Set this to "None" to use the system's default theme
    variant.

    Default: "None"

resize_increments = true | false

    Prefer resizing window by discrete steps equal to cell dimensions.

    Default: false

option_as_alt = "OnlyLeft" | "OnlyRight" | "Both" | "None" # (macos only)

    Make Option key behave as Alt.

    Default: "None"

Example:

    [window]
    padding = { x = 3, y = 3 }
    dynamic_padding = true
    opacity = 0.9
```

SCROLLING

This section documents the [scrolling] table of the configuration file.

```
history = <integer>

    Maximum number of lines in the scrollback buffer.
    Specifying 0 will disable scrolling.
    Limited to 100000.

    Default: 10000

multiplier = <integer>

    Number of line scrolled for every input scroll increment.

    Default: 3
```

FONT

This section documents the [font] table of the configuration file.

```
normal = { family = "<string>", style = "<string>" }

    Default:

        Linux/BSD: { family = "monospace", style = "Regular" }
        Windows: { family = "Consolas", style = "Regular" }
        macOS: { family = "Menlo", style = "Regular" }

bold = { family = "<string>", style = "<string>" }

    If the family is not specified, it will fall back to the value
    specified for the normal font.

    Default: { style = "Bold" }

italic = { family = "<string>", style = "<string>" }

    If the family is not specified, it will fall back to the value
    specified for the normal font.

    Default: { style = "Italic" }

bold_italic = { family = "<string>", style = "<string>" }

    If the family is not specified, it will fall back to the value
    specified for the normal font.

    Default: { style = "Bold Italic" }

size = <float>

    Font size in points.

    Default: 11.25

offset = { x = <integer>, y = <integer> }

    Offset is the extra space around each character. y can be thought of
    as modifying the line spacing, and x as modifying the letter spacing.

    Default: { x = 0, y = 0 }

glyph_offset = { x = <integer>, y = <integer> }

    Glyph offset determines the locations of the glyphs within their cells
    with the default being at the bottom. Increasing x moves the glyph to
    the right, increasing y moves the glyph upward.

builtin_box_drawing = true | false
```

When true, Alacritty will use a custom built-in font for box drawing
characters (Unicode points U+2500 - U+259F) and powerline symbols
(Unicode points U+E0B0 - U+E0B3).

Default: true

COLORS

This section documents the [colors] table of the configuration file.

Colors are specified using their hexadecimal values with a # prefix: #RRGGBB.

primary

This section documents the [colors.primary] table of the configuration
file.

foreground = "<string>"

Default: "#d8d8d8"

background = "<string>"

Default: "#181818"

dim_foreground = "<string>"

If this is not set, the color is automatically calculated
based on the foreground color.

Default: "#828482"

bright_foreground = "<string>"

This color is only used when draw_bold_text_with_bright_colors
is true.

If this is not set, the normal foreground will be used.

Default: "None"

cursor = { text = "<string>", cursor = "<string>" }

Colors which should be used to draw the terminal cursor.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected cell.

Default: { text = "CellBackground", cursor = "CellForeground" }

vi_mode_cursor = { text = "<string>", cursor = "<string>" }

Colors for the cursor when the vi mode is active.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected cell.

Default: { text = "CellBackground", cursor = "CellForeground" }

search

This section documents the [colors.search] table of the configuration.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected cell.

matches = { foreground = "<string>", background = "<string>" }

Default: { foreground = "#181818", background = "#ac4242" }

focused_match = { foreground = "<string>", background = "<string>" }

Default: { foreground = "#181818", background = "#f4bf75" }

hints

This section documents the [colors.hints] table of the configuration.

start = { foreground = "<string>", background = "<string>" }

First character in the hint label.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected
cell.

Default: { foreground = "#181818", background = "#f4bf75" }

end = { foreground = "<string>", background = "<string>" }

All characters after the first one in the hint label.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected
cell.

Default: { foreground = "#181818", background = "#ac4242" }

line_indicator = { foreground = "<string>", background = "<string>" }

Color used for the indicator displaying the position in history during
search and vi mode.

Setting this to "None" will use the opposing primary color.

Default: { foreground = "None", background = "None" }

footer_bar = { foreground = "<string>", background = "<string>" }

Color used for the footer bar on the bottom, used by search regex
input, hyperlink URI preview, etc.

Default: { foreground = "#181818", background = "#d8d8d8" }

selection = { text = "<string>", background = "<string>" }

Colors used for drawing selections.

Allowed values are hexadecimal colors like #ff00ff, or
CellForeground/CellBackground, which references the affected cell.

Default: { text = "CellBackground", background = "CellForeground" }

normal

This section documents the [colors.normal] table of the configuration.

black = "<string>"

Default: "#181818"

red = "<string>"

Default: "#ac4242"

green = "<string>"

Default: "#90a959"

yellow = "<string>"

Default: "#f4bf75"

blue = "<string>"

Default: "#6a9fb5"

magenta = "<string>"

Default: "#aa759f"

cyan = "<string>"

Default: "#75b5aa"

white = "<string>"

Default: "#d8d8d8"

bright

This section documents the [colors.bright] table of the configuration.

black = "<string>"

Default: "#6b6b6b"

red = "<string>"

Default: "#c55555"

green = "<string>"

Default: "#aac474"

yellow = "<string>"

Default: "#feca88"

blue = "<string>"

Default: "#82b8c8"

magenta = "<string>"

Default: "#c28cb8"

cyan = "<string>"

Default: "#93d3c3"

white = "<string>"

Default: "#f8f8f8"

dim

This section documents the [colors.dim] table of the configuration.

If the dim colors are not set, they will be calculated automatically
based on the normal colors.

black = "<string>"

Default: "#0f0f0f"

red = "<string>"

Default: "#712b2b"

```
            green = "<string>"

                Default: "#5f6f3a"

            yellow = "<string>"

                Default: "#a17e4d"

            blue = "<string>"

                Default: "#456877"

            magenta = "<string>"

                Default: "#704d68"

            cyan = "<string>"

                Default: "#4d7770"

            white = "<string>"

                Default: "#8e8e8e"

    indexed_colors = [{ index = <integer>, color = "<string>" },]

            The indexed colors include all colors from 16 to 256. When these are
            not set, they're filled with sensible defaults.

            Default: []

    transparent_background_colors = true | false

            Whether or not window.opacity applies to all cell backgrounds, or only
            to the default background. When set to true all cells will be
            transparent regardless of their background color.

            Default: false

    draw_bold_text_with_bright_colors = true | false

            When true, bold text is drawn using the bright color variants.

            Default: false
```

BELL

```
    This section documents the [bell] table of the configuration file.

    animation = "Ease" | "EaseOut" | "EaseOutSine" | "EaseOutQuad" |
    "EaseOutCubic" | "EaseOutQuart" | "EaseOutQuint" | "EaseOutExpo" |
    "EaseOutCirc" | "Linear"

            Visual bell animation effect for flashing the screen when the visual
            bell is rung.

            Default: "Linear"

    duration = <integer>

            Duration of the visual bell flash in milliseconds. A `duration` of `0`
            will disable the visual bell animation.

            Default: 0

    color = "<string>"

            Visual bell animation color.

            Default: "#ffffff"

    command = "<string>" | { program = "<string>", args = ["<string>",] }

            This program is executed whenever the bell is rung.

            When set to "None", no command will be executed.

            Default: "None"
```

SELECTION

```
    This section documents the [selection] table of the configuration file.

    semantic_escape_chars = "<string>"

            This string contains all characters that are used as separators for
            "semantic words" in Alacritty.

            Default: ",│`|:\"' ()[]{}<>\t"

    save_to_clipboard = true | false

            When set to true, selected text will be copied to the primary
            clipboard.

            Default: false
```

CURSOR

This section documents the [cursor] table of the configuration file.

style = { <shape>, <blinking> }

    shape = "Block" | "Underline" | "Beam"

        Default: "Block"

    blinking = "Never" | "Off" | "On" | "Always"

        Never

            Prevent the cursor from ever blinking

        Off

            Disable blinking by default

        On

            Enable blinking by default

        Always

            Force the cursor to always blink

        Default: "Off"

vi_mode_style = { <shape>, <blinking> } | "None"

    If the vi mode cursor style is "None" or not specified, it will fall
    back to the active value of the normal cursor.

    Default: "None"

blink_interval = <integer>

    Cursor blinking interval in milliseconds.

    Default: 750

blink_timeout = <integer>

    Time after which cursor stops blinking, in seconds.

    Specifying 0 will disable timeout for blinking.

    Default: 5

unfocused_hollow = true | false

    When this is true, the cursor will be rendered as a hollow box when
    the window is not focused.

    Default: true

thickness = <float>

    Thickness of the cursor relative to the cell width as floating point
    number from 0.0 to 1.0.

    Default: 0.15

TERMINAL

This section documents the [terminal] table of the configuration file.

osc52 = "Disabled" | "OnlyCopy" | "OnlyPaste" | "CopyPaste"

    Controls the ability to write to the system clipboard with the OSC 52
    escape sequence. While this escape sequence is useful to copy contents
    from the remote server, allowing any application to read from the
    clipboard can be easily abused while not providing significant
    benefits over explicitly pasting text.

    Default: "OnlyCopy"

MOUSE

This section documents the [mouse] table of the configuration file.

hide_when_typing = true | false

    When this is true, the cursor is temporarily hidden when typing.

    Default: false

bindings = [{ <mouse>, <mods>, <mode>, <action> | <chars> },]

    See keyboard.bindings for full documentation on mods, mode, action,
    and chars.

    When an application running within Alacritty captures the mouse, the
    `Shift` modifier can be used to suppress mouse reporting. If no action

is found for the event, actions for the event without the `Shift`
modifier are triggered instead.

mouse = "Middle" | "Left" | "Right" | "Back" | "Forward" | <integer>

    Mouse button which needs to be pressed to trigger this
binding.

action = <keyboard.bindings.action> | "ExpandSelection"

    ExpandSelection

        Expand the selection to the current mouse cursor
location.

Example:

```
[mouse]
bindings = [
{ mouse = "Right", mods = "Control", action = "Paste" },
]
```

HINTS

This section documents the [hints] table of the configuration file.

Terminal hints can be used to find text or hyperlinks in the visible part of
the terminal and pipe it to other applications.

alphabet = "<string>"

    Keys used for the hint labels.

    Default: "jfkdls;ahgurieowpq"

enabled = [{ <regex>, <hyperlinks>, <post_processing>, <persist>, <action>,
<command> <binding>, <mouse> },]

Array with all available hints.

Each hint must have at least one of regex or hyperlinks and either an action
or a command.

    regex = "<string>"

        Regex each line will be compared against.

    hyperlinks = true | false

        When this is true, all OSC 8 escape sequence hyperlinks will
be included in the hints.

    post_processing = true | false

        When this is true, heuristics will be used to shorten the
match if there are characters likely not to be part of the
hint (e.g. a trailing .). This is most useful for URIs and
applies only to regex matches.

    persist = true | false

        When this is true, hints remain persistent after selection.

    action = "Copy" | "Paste" | "Select" | "MoveViModeCursor"

        Copy

            Copy the hint's text to the clipboard.

        Paste

            Paste the hint's text to the terminal or search.

        Select

            Select the hint's text.

        MoveViModeCursor

            Move the vi mode cursor to the beginning of the hint.

    command = "<string>" | { program = "<string>", args = ["<string>",] }

        Command which will be executed when the hint is clicked or
selected with the binding.

        The hint's text is always attached as the last argument.

    binding = { key = "<string>", mods = "<string>", mode = "<string>" }

        See keyboard.bindings for documentation on available values.

        This controls which key binding is used to start the keyboard
hint selection process.

    mouse = { mods = "<string>", enabled = true | false }

        See keyboard.bindings for documentation on available mods.

        The enabled field controls if the hint should be underlined
when hovering over the hint text with all mods pressed.

```
        Default:

            [[hints.enabled]]
            command = "xdg-open" # On Linux/BSD
            # command = "open" # On macOS
            # command = { program = "cmd", args = [ "/c", "start", "" ] }
            # On Windows
            hyperlinks = true
            post_processing = true
            persist = false
            mouse.enabled = true
            binding = { key = "U", mods = "Control|Shift" }
            regex = "
            (ipfs:|ipns:|magnet:|mailto:|gemini://|gopher://|https://|http://|news:|file:|git://|ssh:|ftp://)
            [^\u0000-\u001F\u007F-\u009F<>\"\\s{-}\\^()`]+"
```

KEYBOARD

This section documents the [keyboard] table of the configuration file.

bindings = [{ <key>, <mods>, <mode>, <action> | <chars> },]

> To unset a default binding, you can use the action "ReceiveChar" to
> remove it or "None" to inhibit any action.
>
> Multiple keybindings can be triggered by a single key press and will
> be executed in the order they are defined in.

key = "<string>"

> The regular keys like "A", "0", and "Я" can be mapped directly
> without any special syntax. Full list of named keys like "F1"
> and the syntax for dead keys can be found here:
>
> https://docs.rs/winit/latest/winit/keyboard/enum.NamedKey.html
> https://docs.rs/winit/latest/winit/keyboard/enum.Key.html#variant.Dead
>
> Numpad keys are prefixed by Numpad: "NumpadEnter" |
> "NumpadAdd" | "NumpadComma" | "NumpadDivide" | "NumpadEquals"
> | "NumpadSubtract" | "NumpadMultiply" | "Numpad[0-9]".
>
> The key field also supports using scancodes, which are
> specified as a decimal number.

mods = "Command" | "Control" | "Option" | "Super" | "Shift" | "Alt"

> Multiple modifiers can be combined using |, like this:
> "Control | Shift".

mode = "AppCursor" | "AppKeypad" | "Search" | "Alt" | "Vi"

> This defines a terminal mode which must be active for this
> binding to have an effect.
>
> Prepending ~ to a mode will require the mode to not = be
> active for the binding to take effect.
>
> Multiple modes can be combined using |, like this:
> "~Vi|Search".

chars = "<string>"

> Writes the specified string to the terminal.

action

> ReceiveChar
>
> > Allow receiving char input.
>
> None
>
> > No action.
>
> Paste
>
> > Paste contents of system clipboard.
>
> Copy
>
> > Store current selection into clipboard.
>
> IncreaseFontSize
>
> > Increase font size.
>
> DecreaseFontSize
>
> > Decrease font size.
>
> ResetFontSize
>
> > Reset font size to the config value.
>
> ScrollPageUp
>
> > Scroll exactly one page up.
>
> ScrollPageDown
>
> > Scroll exactly one page down.

ScrollHalfPageUp

    Scroll half a page up.

ScrollHalfPageDown

    Scroll half a page down.

ScrollLineUp

    Scroll one line up.

ScrollLineDown

    Scroll one line down.

ScrollToTop

    Scroll all the way to the top.

ScrollToBottom

    Scroll all the way to the bottom.

ClearHistory

    Clear the display buffer(s) to remove history.

Hide

    Hide the Alacritty window.

Minimize

    Minimize the Alacritty window.

Quit

    Quit Alacritty.

ClearLogNotice

    Clear warning and error notices.

SpawnNewInstance

    Spawn a new instance of Alacritty.

CreateNewWindow

    Create a new Alacritty window.

ToggleFullscreen

    Toggle fullscreen.

ToggleMaximized

    Toggle maximized.

ClearSelection

    Clear active selection.

ToggleViMode

    Toggle vi mode.

SearchForward

    Start a forward buffer search.

SearchBackward

    Start a backward buffer search.

Vi mode actions:

Up

    Move up.

Down

    Move down.

Left

    Move left.

Right

    Move right.

First

    First column, or beginning of the line when already
    at the first column.

Last

    Last column, or beginning of the line when already at
    the last column.

FirstOccupied

First non-empty cell in this terminal row, or first
non-empty cell of the line when already at the first
cell of the row.

High

Move to top of screen.

Middle

Move to center of screen.

Low

Move to bottom of screen.

SemanticLeft

Move to start of semantically separated word.

SemanticRight

Move to start of next semantically separated word.

SemanticLeftEnd

Move to end of previous semantically separated word.

SemanticRightEnd

Move to end of semantically separated word.

WordLeft

Move to start of whitespace separated word.

WordRight

Move to start of next whitespace separated word.

WordLeftEnd

Move to end of previous whitespace separated word.

WordRightEnd

Move to end of whitespace separated word.

Bracket

Move to opposing bracket.

ToggleNormalSelection

Toggle normal vi selection.

ToggleLineSelection

Toggle line vi selection.

ToggleBlockSelection

Toggle block vi selection.

ToggleSemanticSelection

Toggle semantic vi selection.

SearchNext

Jump to the beginning of the next match.

SearchPrevious

Jump to the beginning of the previous match.

SearchStart

Jump to the next start of a match to the left of the
origin.

SearchEnd

Jump to the next end of a match to the right of the
origin.

Open

Launch the URL below the vi mode cursor.

CenterAroundViCursor

Centers the screen around the vi mode cursor.

InlineSearchForward

Search forward within the current line.

InlineSearchBcakward

Search backward within the current line.

InlineSearchForwardShort

Search forward within the current line, stopping just
short of the character.

InlineSearchBackwardShort

    Search backward within the current line, stopping
    just short of the character.

InlineSearchNext

    Jump to the next inline search match.

InlineSearchPrevious

    Jump to the previous inline search match.

Search actions:

SearchFocusNext

    Move the focus to the next search match.

SearchFocusPrevious

    Move the focus to the previous search match.

SearchConfirm

    Confirm the active search.

SearchCancel

    Cancel the active search.

SearchClear

    Reset the search regex.

SearchDeleteWord

    Delete the last word in the search regex.

SearchHistoryPrevious

    Go to the previous regex in the search history.

SearchHistoryNext

    Go to the next regex in the search history.

macOS exclusive:

ToggleSimpleFullscreen

    Enter fullscreen without occupying another space.

HideOtherApplications

    Hide all windows other than Alacritty.

CreateNewTab

    Create new window in a tab.

SelectNextTab

    Select next tab.

SelectPreviousTab

    Select previous tab.

SelectTab1

    Select the first tab.

SelectTab2

    Select the second tab.

SelectTab3

    Select the third tab.

SelectTab4

    Select the fourth tab.

SelectTab5

    Select the fifth tab.

SelectTab6

    Select the sixth tab.

SelectTab7

    Select the seventh tab.

SelectTab8

    Select the eighth tab.

SelectTab9

    Select the ninth tab.

SelectLastTab

Select the last tab.

                Linux/BSD exclusive:

                CopySelection

                        Copy from the selection buffer.

                PasteSelection

                        Paste from the selection buffer.

Default: See alacritty-bindings(5)

Example:

        [keyboard]
        bindings = [
        { key = "N", mods = "Control|Shift", action = "CreateNewWindow" },
        { key = "L", mods = "Control|Shift", chars = "l" },
        ]

DEBUG

This section documents the [debug] table of the configuration file.

Debug options are meant to help troubleshoot issues with Alacritty. These can
change or be removed entirely without warning, so their stability shouldn't be
relied upon.

render_timer = true | false

        Display the time it takes to draw each frame.

        Default: false

persistent_logging = true | false

        Keep the log file after quitting Alacritty.

        Default: false

log_level = "Off" | "Error" | "Warn" | "Info" | "Debug" | "Trace"

        Default: "Warn"

        To add extra libraries to logging ALACRITTY_EXTRA_LOG_TARGETS variable
        can be used.

        Example:

            ALACRITTY_EXTRA_LOG_TARGETS="winit;vte" alacritty -vvv

renderer = "glsl3" | "gles2" | "gles2_pure" | "None"

        Force use of a specific renderer, "None" will use the highest
        available one.

        Default: "None"

print_events = true | false

        Log all received window events.

        Default: false

highlight_damage = true | false

        Highlight window damage information.

        Default: false

prefer_egl = true | false

        Use EGL as display API if the current platform allows it. Note that
        transparency may not work with EGL on Linux/BSD.

        Default: false

SEE ALSO

alacritty(1), alacritty-msg(1), alacritty-bindings(5)

BUGS

Found a bug? Please report it at
https://github.com/alacritty/alacritty/issues.

MAINTAINERS

        · Christian Duerr <contact@christianduerr.com>

        · Kirill Chibisov <contact@kchibisov.com>