Tools, Workflow and Resources for Effective Development

Live Online Workshop for HSRW and ULSA by Harley Lara - 2022-10-06

Workshop details:

- **Topic**: Tools, workflow and resources for effective development
- **Description**: Hardware and/or Software project development can be a long and complex process, but with the right set of tools everything can be faster and more efficient from the developer's point of view. This workshop is based on providing tools for developers to have a seamless process, producing quality software and hardware. In collaboration with ULSA Nicaragua and EO lab HSRW.
- **Date**: 06 October 2022
- **Time**: from 16:00 to 18:00 (CEST)
- **Location**: IoT Lab, room 02 02 510 HSRW. Zoom meeting for ULSA.
- **Instructor**: Harley Lara
- **Contact**: harley-nelson<DOT>lara-alonso<AT>hsrw<DOT>org

Demo files used during the presentation



Please change the `.doc` extension to the correct extension.
The Bash script uses `.sh` and ansible-playbooks `.yml`.

- Hello bash script

: Print message in the terminal

- Tmux and fzf bash script

: fuzzy search over a group of specified directories

- Software installation bash script

: Basic bash script to install a set of packages

- Software installation ansible playbook

: Basic playbook to install a set of packages

- Tags implementation in bash

: Minimal and basic tag implementation to emulate ansible tags using bash script

bash Demo

hello-bash.sh

```
#!/bin/bash
#
```

```
# Script used as demo during the workshop
# "Tools, Workflows and Resources"
# 06 Oct. 2022, HSRW Germany
#
# Copyright (c) 2022 Harley Lara
#
# Permission is hereby granted, free of charge, to any person obtaining
# a copy of this software and associated documentation files (the
# "Software"), to deal in the Software without restriction, including
# without limitation the rights to use, copy, modify, merge, publish,
# distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so, subject to
# the following conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
# MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE
# LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
# WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

sudo apt-get update && \
apt-get install -y \
    build-essential \
    cmake \
    pkg-config \
    libpthread-stubs0-dev

sudo apt-get install -y \
    curl \
    htop \
    python3-pip
```

tmux Demo

tmux-fzf.sh

```
#!/bin/bash
#
# Script used as demo during the workshop
```

```
# "Tools, Workflows and Resources"
# 06 Oct. 2022, HSRW Germany
#
# Copyright (c) 2022 Harley Lara
#
# Permission is hereby granted, free of charge, to any person obtaining
# a copy of this software and associated documentation files (the
# "Software"), to deal in the Software without restriction, including
# without limitation the rights to use, copy, modify, merge, publish,
# distribute, sublicense, and/or sell copies of the Software, and to
# permit persons to whom the Software is furnished to do so, subject to
# the following conditions:
#
# The above copyright notice and this permission notice shall be
# included in all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
# EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
# MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
# NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE
# LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
ACTION
# OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
# WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.



# Used the next line to set the list of directories
DIRECTORIES="~/ ~/Documents ~/Desktop"
# Uncommend the next line if you are using a external file
# for track the list of directories
# DIRECTORIES=$(cat ~/EXTERNAL-FILE | tr "\n" " ")
# Note about the external file
#       Format of the file:
#           ~/
#           ~/Documents
#           ~/Desktop
#       One directory per line

if [[ $# -eq 1 ]]; then
    SELECTED=$1 && [[ "$SELECTED" == '.' ]] && SELECTED="$PWD"
else
    SELECTED=$(find $DIRECTORIES -mindepth 1 -maxdepth 1 -type d | fzf)
fi

if [[ -z $SELECTED ]]; then
    exit 0
fi

SELECTED_NAME=$(basename "$SELECTED" | tr . _)
```

```
SELECTED_NAME=${SELECTED_NAME:0:8}

if [[ -n $TMUX ]]; then
    # inside tmux
    tmux switch-client -t "$SELECTED_NAME" || \
    tmux new-session -ds "$SELECTED_NAME" -c "$SELECTED" && \
    tmux switch-client -t "$SELECTED_NAME"
elif [[ -z $TMUX ]]; then
    # outside tmux
    tmux new-session -s "$SELECTED_NAME" -c "$SELECTED" || \
    tmux attach -t "$SELECTED_NAME"
fi
```

Encryption and Decryption with Ansible

```
# Encryption:
ansible-vault encrypt FILE_NAME

# Decryption:
ansible-vault decrypt FILE_NAME
```

Full list of software in the presentation

Note 1: if the platform is not specified as Windows, Linux or MacOS, consider the software mentioned as Cross-platform (available on all).

Note 2: Linux and MacOS are Unix-based operating systems

- **Shells (For Unix-based OS)**:
  - Bash (RECOMMENDED)
  - Fish
  - Ksh
  - Zsh
- **Terminal Emulators** (matter of personal choice):
  - Konsole (for Linux): Default in KDE's desktop environment
  - kitty: GPU based terminal emulator
  - Alacritty: GPU based terminal emulator
  - Terminator (for Linux): Terminal and terminal multiplexer
  - PuTTY (for Windows): Mainly ssh, serial and telnet connection
  - Windows Terminal (for Windows): "Terminal application for users of command-line tools and shells like Command Prompt, PowerShell, and WSL" (not installed by default)
  - Find more here
- **Terminal Multiplexers** (for Unix-based OS):
  - Terminator: (Requires a graphical environment to run)

    - Screen: Installed by default in many Linux distributions
    - Tmux: “It lets you switch easily between several programs in one terminal” (RECOMMENDED)
        - Article How does Tmux works
        - Tmux Cheatsheet

- **Automation**:
    - Ansible: “Ansible is a suite of software tools that enables infrastructure as code.”
    - Shortcuts and Macros
        - AutoHotkey (for Windows): “The ultimate automation scripting language for Windows.”
        - AutoKey (for Linux): “A desktop automation utility for Linux and X11.”
        - BetterTouchTools (for MacOS)

- **Tiling Window Managers**:
    - i3 (for Linux)
    - bug.n (for Windows)
    - Amethyst (for MacOS)
    - More about tiling window managers here

- **Lightweight Markup Languages**:
    - Markdown and flavors
    - MediaWiki
    - DokuWiki
    - reStructuredText
    - AsciiDoc
    - Others options here

- **Hosting Docs**:
    - Markdown file in git repositories
    - Personal Wikis: DokuWiki, MediaWiki …
    - SSGs (Static Site Generator):
        - Github Pages
        - Gitlab Pages

- **Diagrams**:
    - Mermaid markdown extension for text-based diagrams.
    - Draw.io graph drawing software

- **Screenshots**:
    - Flameshot “powerful, yet simple to use open-source screenshot software”

From:
https://wiki.eolab.de/ - **HSRW EOLab Wiki**

Permanent link:
https://wiki.eolab.de/doku.php?id=capacity_building:workshops:tooling&rev=1665222972

Last update: **2022/10/08 11:56**