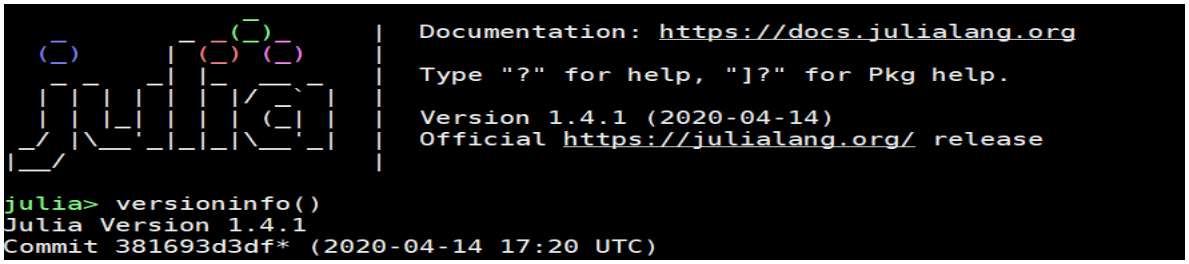

I met the following issues while integrating Julia 1.4.1 to Linux. Here below are the issues with solution.

Issues.

Artifact: Zstd and artifact: MbedTLS installation failure when adding packages MAT and IJulia respectively

PyPlot get installed very well but **MAT** and **IJulia** are failed. Any idea?

julia> versioninfo()
Julia Version 1.4.1
Commit 381693d3df* (2020-04-14 17:20 UTC)
Platform Info:
OS: Linux (x86_64-pc-linux-gnu)
CPU: Intel® Xeon® CPU L5408 @ 2.13GHz
WORD_SIZE: 64
LIBM: libopenlibm
LLVM: libLLVM-8.0.1 (ORCJIT, penryn)
Environment:
JULIA_HOME = /home/hadoop/opt/julia-1.4.1/

(**@v1.4**) pkg> add PyPlot
Updating registry at ~/.julia/registries/General
Updating git-repo https://github.com/JuliaRegistries/General.git
Resolving package versions…
Updating ~/.julia/environments/v1.4/Project.toml
[no changes]
Updating ~/.julia/environments/v1.4/Manifest.toml
[no changes]

(**@v1.4**) pkg> add IJulia
Resolving package versions…
Downloading artifact: MbedTLS
100.0%-=O#-# # # curl: (28) Failed to connect to github-production-release-asset-2e65be.s3.amazonaws.com 2 port 443: Connection timed out
ERROR: Unable to automatically install ‘MbedTLS’ from

‘/home/hadoop/.julia/packages/MbedTLS_jll/hP8le/Artifacts.toml’
Stacktrace:
[1] error(::String) at ./error.jl:33
[2] ensure_artifact_installed(::String, ::Dict{String,Any}, ::String;
platform::Pkg.BinaryPlatforms.Platform, verbose::Bool, quiet_download::Bool) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/Artifacts.jl:89
4
[3] ensure_all_artifacts_installed(::String; platform::Pkg.BinaryPlatforms.Platform,
pkg_uuid::Nothing, include_lazy::Bool, verbose::Bool, quiet_download::Bool) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/Artifacts.jl:95
8
[4] download_artifacts(::Pkg.Types.Context, ::Array{String,1};
platform::Pkg.BinaryPlatforms.Linux, verbose::Bool) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/Operations.jl:6
09
[5] download_artifacts(::Pkg.Types.Context, ::Array{Pkg.Types.PackageSpec,1};
platform::Pkg.BinaryPlatforms.Linux, verbose::Bool) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/Operations.jl:5
88
[6] add(::Pkg.Types.Context, ::Array{Pkg.Types.PackageSpec,1}, ::Array{Base.UUID,1};
preserve::Pkg.Types.PreserveLevel, platform::Pkg.BinaryPlatforms.Linux) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/Operations.jl:1
084
[7] add(::Pkg.Types.Context, ::Array{Pkg.Types.PackageSpec,1};
preserve::Pkg.Types.PreserveLevel, platform::Pkg.BinaryPlatforms.Linux,
kwargs::Base.Iterators.Pairs{Union{},Union{},Tuple{},NamedTuple{(),Tuple{}}}) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/API.jl:159
[8] add(::Pkg.Types.Context, ::Array{Pkg.Types.PackageSpec,1}) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/API.jl:112
[9] #add#27 at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/API.jl:109
[inlined]
[10] add(::Array{Pkg.Types.PackageSpec,1}) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/API.jl:109
[11] do_cmd!(::Pkg.REPLMode.Command, ::REPL.LineEditREPL) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/REPLMode/R
EPLMode.jl:403
[12] do_cmd(::REPL.LineEditREPL, ::String; do_rethrow::Bool) at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/REPLMode/R
EPLMode.jl:381
[13] do_cmd at
/buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/REPLMode/R
EPLMode.jl:376 [inlined]
[14]

(::Pkg.REPLMode.var"#24#27"{REPL.LineEditREPL,REPL.LineEdit.Prompt})(::REPL.LineEdit.MIState, ::Base.GenericIOBuffer{Array{UInt8,1}}, ::Bool) at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/Pkg/src/REPLMode/REPLMode.jl:542
[15] #invokelatest#1 at ./essentials.jl:712 [inlined]
[16] invokelatest at ./essentials.jl:711 [inlined]
[17] run_interface(::REPL.Terminals.TextTerminal, ::REPL.LineEdit.ModalInterface, ::REPL.LineEdit.MIState) at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/REPL/src/LineEdit.jl:2354
[18] run_frontend(::REPL.LineEditREPL, ::REPL.REPLBackendRef) at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/REPL/src/REPL.jl:1055
[19] run_repl(::REPL.AbstractREPL, ::Any) at /buildworker/worker/package_linux64/build/usr/share/julia/stdlib/v1.4/REPL/src/REPL.jl:206
[20] (::Base.var"#764#766"{Bool,Bool,Bool,Bool})(::Module) at ./client.jl:383
[21] #invokelatest#1 at ./essentials.jl:712 [inlined]
[22] invokelatest at ./essentials.jl:711 [inlined]
[23] run_main_repl(::Bool, ::Bool, ::Bool, ::Bool, ::Bool) at ./client.jl:367
[24] exec_options(::Base.JLOptions) at ./client.jl:305
[25] _start() at ./client.jl:484

(**@v1.4**) pkg>

Finding:

Julia downloads packages from Github repository by default (git-repo: https://github.com/JuliaRegistries/General.git). This repository is hosted on Amazon github-production-release-asset-2e65be.s3.amazonaws.com port 443 and keeps breaking connections, therefore, unreliable.

Solution:

To solve this issue, change Julia to download packages from Julia package server JULIA_PKG_SERVER=https://geo.pkg.julialang.org

This can be done by adding a path in your profile file located in your home directory as shown in Fig. 1 and Fig. 2 below.

```
(base) hadoop@hadoopSvr2:~$ vi .profile
```

Fig. 1

```
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin directories
PATH="$HOME/bin:$HOME/.local/bin:$PATH"

JULIA_PKG_SERVER="https://geo.pkg.julialang.org"
export JULIA_PKG_SERVER
~
```

Fig. 2

NB: Do not forget to source the file .profile after making the change.

```
(base) hadoop@hadoopSvr2:~$ source .profile
```

Fig. 3

julia> versioninfo()

Julia Version 1.4.1
Commit 381693d3df* (2020-04-14 17:20 UTC)
Platform Info:

OS: Linux (x86_64-pc-linux-gnu)
CPU: Intel® Xeon® CPU L5408 @ 2.13GHz
WORD_SIZE: 64
LIBM: libopenlibm
LLVM: libLLVM-8.0.1 (ORCJIT, penryn)
Environment:
JULIA_PKG_SERVER = https://geo.pkg.julialang.org
JULIA_HOME = /home/hadoop/opt/julia/

(**@v1.4**) pkg> add IJulia
Cloning default registries into ~/.julia
##
100.0%
Added registry General to ~/.julia/registries/General
Resolving package versions…
Installed SoftGlobalScope ─ v1.0.10
Installed ZMQ ──────────────── v1.2.0
Installed Parsers ──────────── v1.0.4
Installed MbedTLS_jll ──────── v2.16.0+2
Installed VersionParsing ─── v1.2.0
Installed MbedTLS ──────────── v1.0.2
Installed IJulia ──────────── v1.21.2
Installed Conda ──────────── v1.4.1
Installed ZeroMQ_jll ──────── v4.3.2+3
Installed JSON ──────────── v0.21.0
Downloading artifact: MbedTLS
Downloading artifact: ZeroMQ
Updating ~/.julia/environments/v1.4/Project.toml
[7073ff75] + IJulia v1.21.2
Updating ~/.julia/environments/v1.4/Manifest.toml
[8f4d0f93] + Conda v1.4.1
[7073ff75] + IJulia v1.21.2
[682c06a0] + JSON v0.21.0
[739be429] + MbedTLS v1.0.2
[c8ffd9c3] + MbedTLS_jll v2.16.0+2
[69de0a69] + Parsers v1.0.4
[b85f4697] + SoftGlobalScope v1.0.10
[81def892] + VersionParsing v1.2.0
[c2297ded] + ZMQ v1.2.0
[8f1865be] + ZeroMQ_jll v4.3.2+3
[2a0f44e3] + Base64

~

(@v1.4) pkg> add PyPlot

```
    Resolving package versions...
    Installed Reexport ──────────────── v0.2.0
    Installed LaTeXStrings ──────────── v1.1.0
    Installed PyPlot ────────────────── v2.9.0
    Installed FixedPointNumbers ─ v0.8.0
    Installed MacroTools ────────────── v0.5.5
    Installed PyCall ────────────────── v1.91.4
    Installed ColorTypes ────────────── v0.10.3
    Installed Colors ────────────────── v0.12.1
      Updating `~/.julia/environments/v1.4/Project.toml`
    [d330b81b] + PyPlot v2.9.0
      Updating `~/.julia/environments/v1.4/Manifest.toml`
    [3da002f7] + ColorTypes v0.10.3
    [5ae59095] + Colors v0.12.1
    [53c48c17] + FixedPointNumbers v0.8.0
    [b964fa9f] + LaTeXStrings v1.1.0
    [1914dd2f] + MacroTools v0.5.5
    [438e738f] + PyCall v1.91.4
    [d330b81b] + PyPlot v2.9.0
    [189a3867] + Reexport v0.2.0
      Building PyCall → `~/.julia/packages/PyCall/zqDXB/deps/build.log`

(@v1.4) pkg> add MAT
    Resolving package versions...
    Installed Zstd_jll ──────────────── v1.4.5+0
    Installed HDF5_jll ──────────────── v1.10.5+5
    Installed Blosc ─────────────────── v0.7.0
    Installed HDF5 ──────────────────── v0.13.2
    Installed TranscodingStreams ─ v0.9.5
    Installed MAT ───────────────────── v0.8.0
    Installed CodecZlib ─────────────── v0.7.0
    Installed Blosc_jll ─────────────── v1.14.3+1
    Installed Compat ────────────────── v3.10.0
    Installed Lz4_jll ───────────────── v1.9.2+0
    Installed BufferedStreams ──────── v1.0.0
    Installed Zlib_jll ──────────────── v1.2.11+10
Downloading artifact: Zstd
Downloading artifact: Lz4
Downloading artifact: HDF5
Downloading artifact: Blosc
Downloading artifact: Zlib
      Updating `~/.julia/environments/v1.4/Project.toml`
    [23992714] + MAT v0.8.0
      Updating `~/.julia/environments/v1.4/Manifest.toml`
```

[a74b3585] + Blosc v0.7.0
[0b7ba130] + Blosc_jll v1.14.3+1
[e1450e63] + BufferedStreams v1.0.0
[944b1d66] + CodecZlib v0.7.0
[34da2185] + Compat v3.10.0
[f67ccb44] + HDF5 v0.13.2
[0234f1f7] + HDF5_jll v1.10.5+5
[5ced341a] + Lz4_jll v1.9.2+0