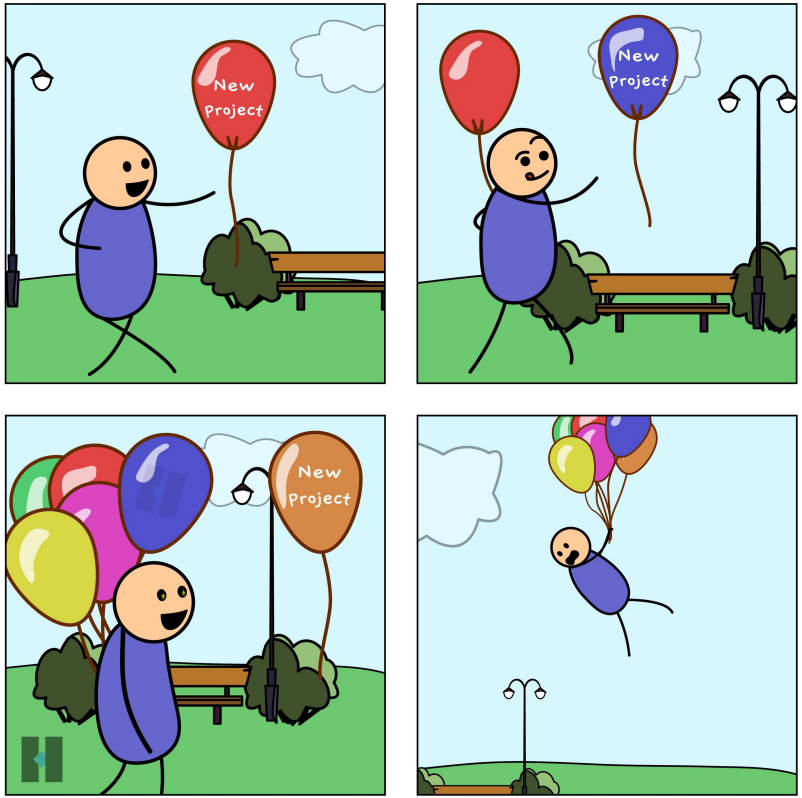
Personal Infrastructure

RITLUG

ritlug.com

Jeffery R, Henry K, Tim Z



Please sign in!

http://bit.ly/ritlug-fall

Keep up with RITlug outside of meetings:

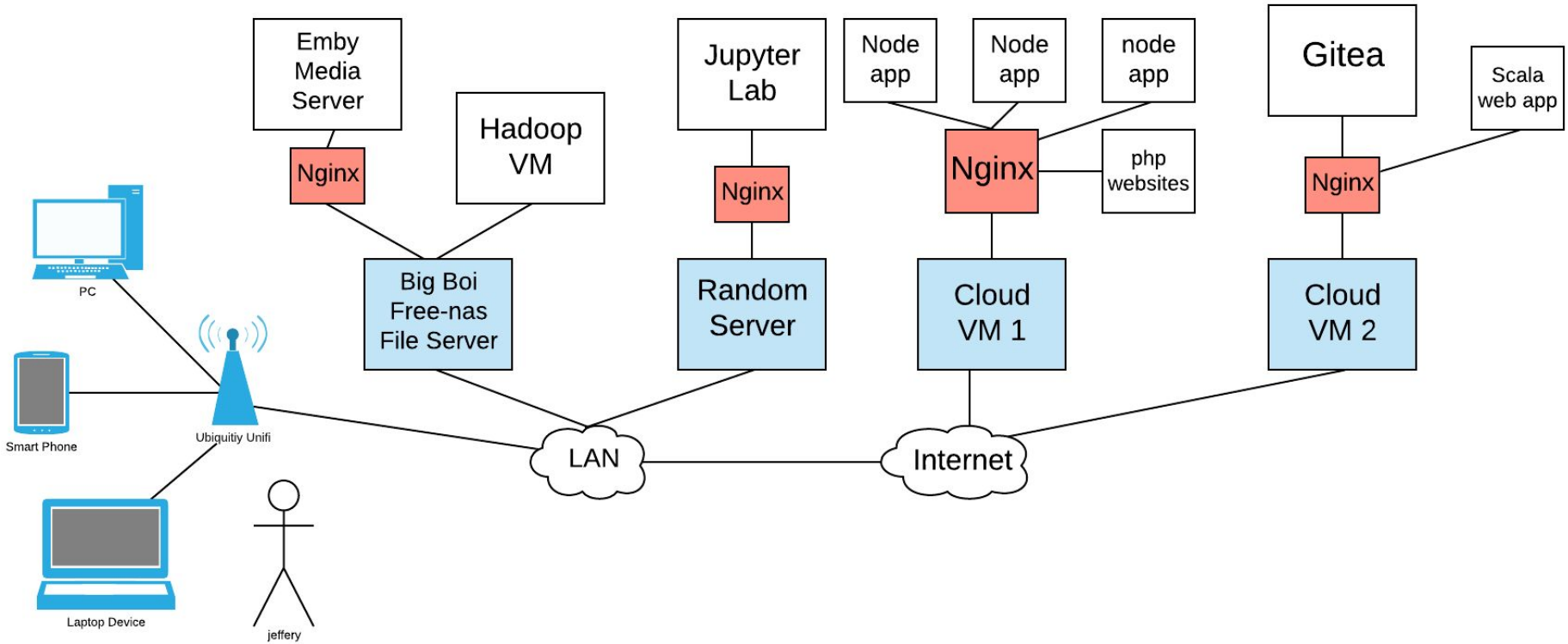
ritlug.com/get-involved,

https://discord.gg/xev2W62

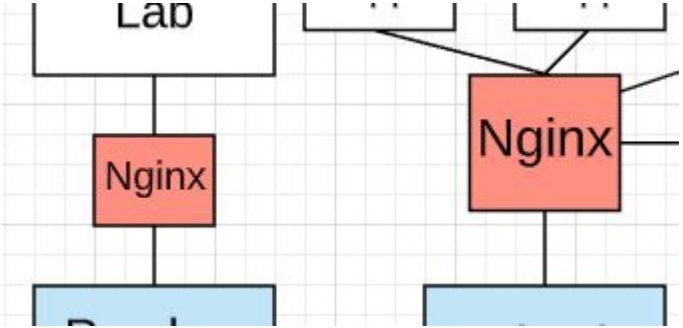
[biggest_words]

The Personal Infrastructure of Jeffery Russell

Jeffery


Emby
Media
Server
Nginx
Hadoop
VM
Big Boi
Free-nas
File Server
Jupyter
Lab
Nginx
Random
Server
Node
app
Node
app
node
app
Nginx
php
websites
Cloud
VM 1
Gitea
Nginx
Scala
web app
Cloud
VM 2
PC
Smart Phone
Ubiquitiy Unifi
LAN
Internet
Laptop Device
jeffery

What is Nginx? And why do I use it everywhere?

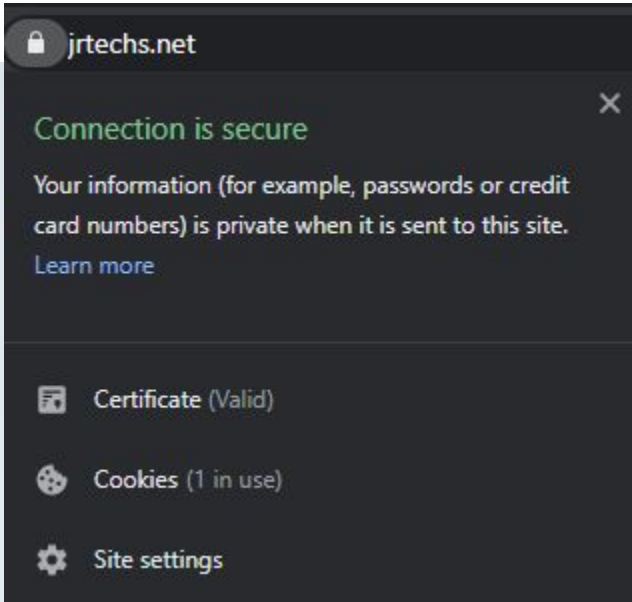


Nginx

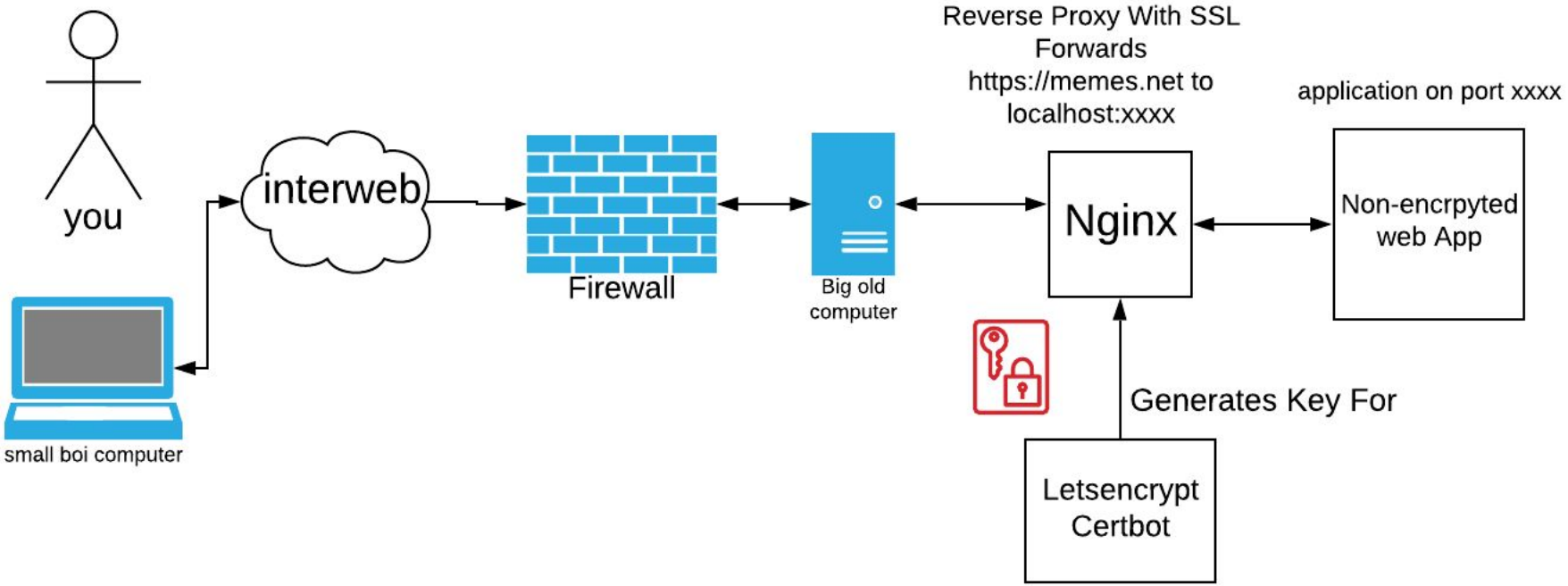
NGINX is open source software for web serving, reverse proxying, caching, load balancing, media streaming, and more. It started out as a web server designed for maximum performance and stability. In addition to its HTTP server capabilities, NGINX can also function as a proxy server for email (IMAP, POP3, and SMTP) and a reverse proxy and load balancer for HTTP, TCP, and UDP servers.

Let’s Encrypt

- Enables you to obtain SSL certificates free of cost
- Traditionally SSL Certs have been expensive.
  - Prices vary, but you usually pay over 2x on your SSL cert than on your domain registration



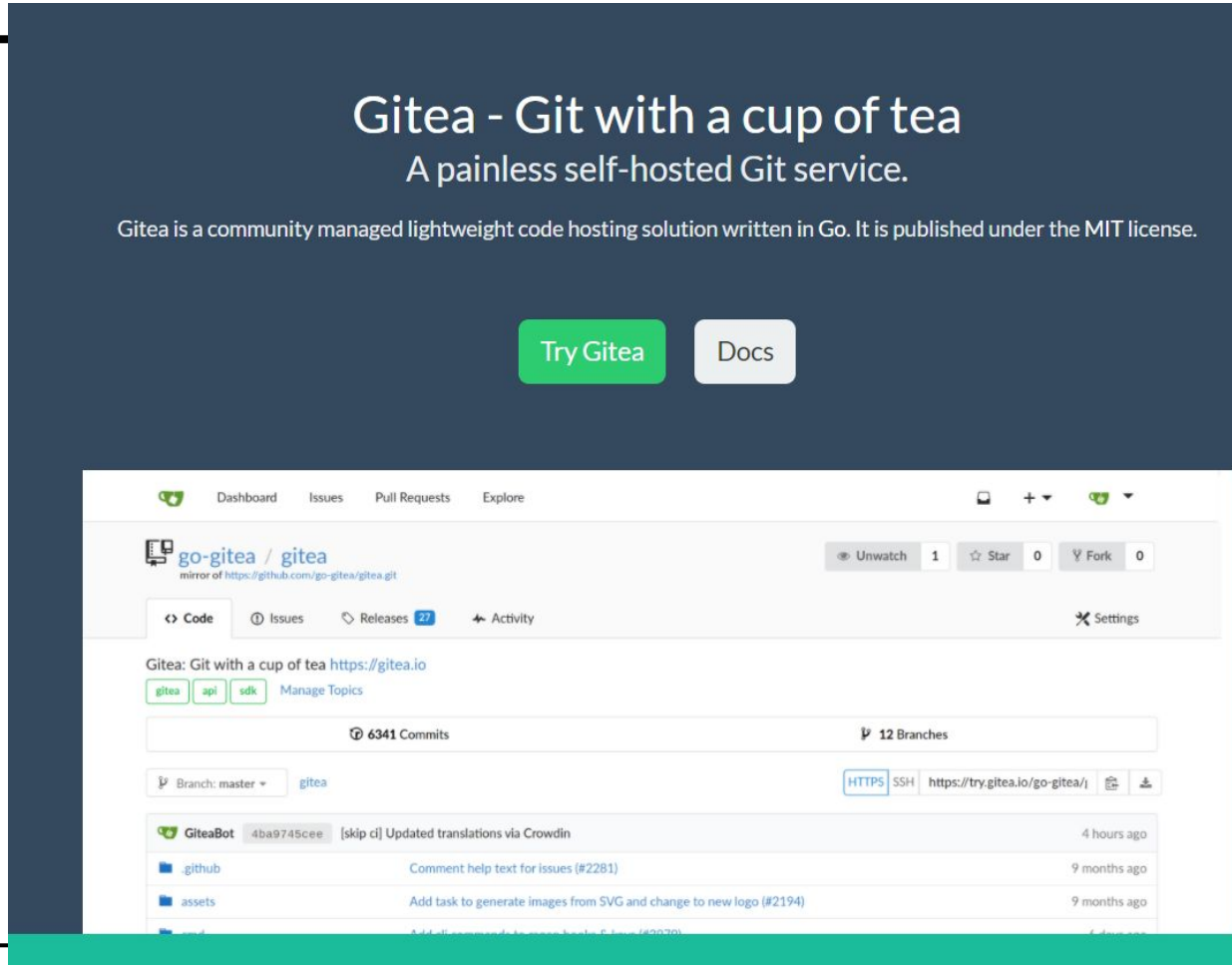
Letsencrypt + Nginx + Unencrypted App= Good



Gitea

I'm using a docker image to run a Gitea server and then a nginx server w/ letsencrypt to provide encrypted traffic.
https://git.jrtechs.net

Installation Instructions:
https://jrtechs.net/open-source/hosting-your-own-gitea-server



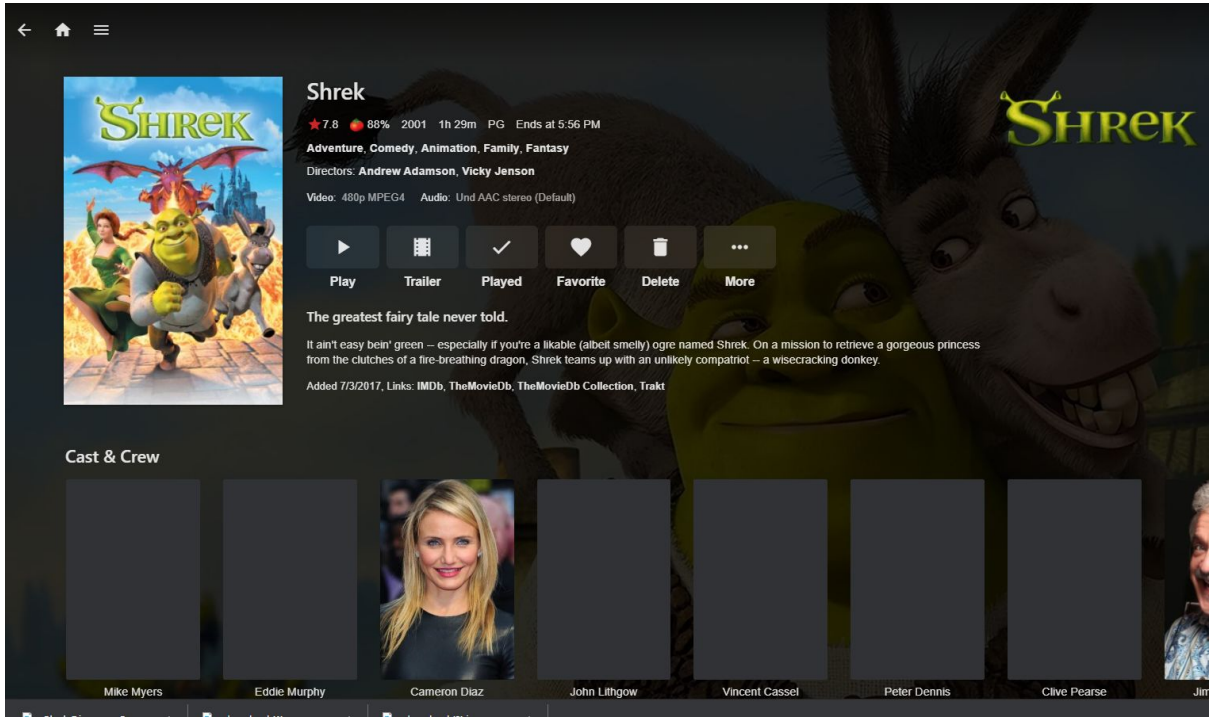
Jupyter Lab

- TLDR Jupyter lab is the cat's meow when it comes to programming environments for Data Scientists.
- Ability to create notebooks where you can run code blocks, embed markdown, and text.
- Check out my blogpost "Jupyter Will Change Your Life" for more information.

JellyFin/Emby

Media server specialized for streaming movies and tv shows. Similar to Kodi.

Check out JellyFin if you are interested in running a server. Note: Emby is closed source where JellyFin is the open-source fork.



[small_words]

“.dotfiles”

.dotfiles

- “Dotfiles” are files on your system that are used for various configuration/automation of other services/jobs/tools on your system, ie.
  - .gitignore
  - .bashrc
  - .bash_history
  - .functions
- Get your .dotfiles in order, get your system in order
- Symlink go brrrrr

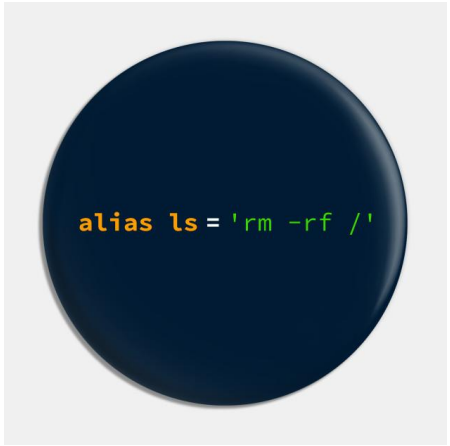
.dotfiles and GitHub

- ***Backup***, ***restore***, and ***sync*** the prefs and settings for your toolbox. Your dotfiles might be the most important files on your machine.
- ***Learn*** from the community. Discover new tools for your toolbox and new tricks for the ones you already use.
- ***Share*** what you've learned with the rest of us.

- https://dotfiles.github.io/

Example .dotfile Git Repo

```
.
├── git
│   ├── .gitconfig
│   └── .gitignore_global
├── install.sh
├── osxdefaults.sh
├── runcom
│   ├── .bash profile
│   └── .inputrc
└── system
├── .alias
├── .env
├── .function
├── .path
└── .prompt
```



Automate your Automation

- “We aren’t cavemen Spongebob, we have technology.”
- You can write some easy and quick scripts to get all your .dotfiles and such from your git repo on a new system, there are also tools for this:
  - GNU Stow
  - dotbot
  - homesick
- Simple Automation Fun:
  - Aliasing = a fun and smart way to keep things working quickly

[smaller_words]

“Systemd Timers”

Systemd

- Software suite aimed to unify service configuration and behavior on Linux
- Primary component is an init system
  - bootstraps user space & manages user processes

Systemd Services

- Specific processes including:
  - logging daemon, bluetooth, firewall, networkmanager
- Services act in a dependency-based control logic
- `Systemctl` main method to interact with systemd

Systemd Service Example

[Unit]
Description=*Add a description*
Requires=*New dependency*
After=*New dependency*

[Service]
Type=simple
ExecStart=*/usr/bin/foo*

Systemd Journal

- Main logging component for systemd
  - `journalctl`
- Found in /var/log/journal
- Can use to find logs on specific units
  - `journalctl -u foo[.service]`

Cron

- Time-based scheduler
- Allows you to schedule jobs to occur at specific times
- Commonly automates system administration & maintenance

Crontab format

- *minute hour day_of_month month day_of_week command*

Example:

**/5 9-16 * 1-5,9-12 1-5 ~/bin/i_love_cron.sh*

Executes at 5 minute intervals from 9 AM to 4:55 PM on weekdays, except during the summer months.

Systemd Timers

- Alternative to cron
- Unit files that end in *.timer* that control *.service* files
- Built-in support for calendar time

Timer Units

- Service File
- Timer File
- Script to run (if needed)

Timer Example

Foo.service

```
[Unit]
Description=Foo bar
Requires=network.target
After=network.target
```

Timer Example (Cont.)

Foo.timer

```
[Unit]
Description=foo bar

[Timer]
# Renew cert weekly on Wednesdays at 3AM EDT
OnCalendar=Wed *-*-* 03:00:00
Persistent=true

[Install]
WantedBy=timers.target
```