

# 趣味開発のための クラウド/VPS活用術

2014-08-30 YAPC::Asia 2014

# @mackee\_w a.k.a マコピー

- 面白法人カヤック 技術部
- サーバサイドエンジニア(Perl)
- ソーシャルゲームの開発/運用
- 今作っているゲームは事前登録受付中

# 今日話すこと

- VPS/クラウドサービス比較
  - PaaSは話しません
- 今、自宅サーバを使う理由
  - Raspberry Piを自宅サーバに使う

この発表には独断と偏見が  
かなり含まれているので  
この後HUBとかで僕を倒してください

何故PaaSの話をしていないか

# 大体僕の趣味です

- インフラから構築するのが好き
- Perlが動くPaaSは限られている
  - DotCloud
  - Heroku
  - OpenShift
- PaaSは割高なことが多い



今回話題に上げるサービス

**linode**やその他**VPS**がないのは  
使っていないからです><  
レポートお待ちしております  
**HUB**で僕を倒して (r y



とりあえず各インスタンスで  
ベンチ(unixbench)取ってみる

# 各サービスのスペック

- Amazon AWS - t2.micro \$15.60/month(※)
- Google Compute Engine - \$7.48/month(※)
- さくらのVPS - 934円(税抜き)/month
- DigitalOcean - \$5/month
- VULTR - \$5/month

時間がないのでサービスの  
特徴に結果は含める

さていくで

# いきなり復権したt2.micro

## スコア:1839.5

- t1.microショボイし割高だなって思ってたけれど今ならメツチャええやんってなる
- ただし、常時起動ではなく都度起動用途
- AWSのサービスは個人で有用なものも多い

# t2系インスタンスのBurst

- t2系のインスタンスは性能がBurstする
- 一時的に能力がアップ
- 使っていないと貯金が貯まる
- Webサービスのときに便利（構築時とか

# GCE f1-micro スコア:437.1

- 個人ではあまり使わない/使えないサービス
  - ロードバランサー
  - BigQueryもすごいんだけど.....
- App EngineのDocker対応に期待
- 1ヶ月使い続けると安くなるシステム

# 普段使いではVULTR 768MB

## スコア: 2328.2

- かなりお手軽だし安い そこそこ早い
- 最新OSがいち早く試せる
- 知らないうちに機能が追加されている
- CDインストールやPXEブートが可能
- 東京リージョン



# Vagrant連携ができるDO

スコア: 1460.3

- Vagrant便利最高
- 価格はVULTRと同じぐらい
- ただし一番近いのはシンガポールリージョン
- 使っている人が多いと思われる

疑問

ところで俺は一体  
なんでこんなに  
サーバを立ててるんだ

# 普段の用途

- 作って見たミドルウェアのベンチ
- 流行っているらしいミドルウェアを使ってみる
- 手元のMacがぶんぶん鳴るのが気に食わない
- 実はあんまりサービスとか作ってない





## Kuiperbelt

Your synchronous application plug in  
Asynchronous Protocol

# kuiperbelt

ネットワーク帯域の増幅率ヤバイ



# ネットワークの課金形態

使った分だけ払う勢



上限まではOK



無制限



# 転送量を気にかけるべき用途

- 動画配信
- 動画取り込み
- スクレイピングしてデカいXMLとかを定期的に取り込むとか
- アップローダー

そういう時にはさくらのVPS

スコア: 2080.1(2 cores)

もちろん業務ではAWS S3だったり

CloudFrontだったり使い分けるだるうけれど

AWSは課金額が見えないのが趣味でやるには

ツライ



# さくらのVPS

- 立てるのがAPIで出来ない
- 1ヶ月単位の課金
- 常時起動でデータ置くところとかに使えばいいのでは？
- [ssig33.com](http://ssig33.com) - Docker 運用しまくって得られた  
しよぼい知識

# VPSは起動が遅い

- だいたいそれぞれ1分から5分ぐらいかかります
- t2.microとかはgp2をつけると若干早い
- CoreOSで立ち上げると早いとかある
- スナップショット

# 提案 - ひとつ常時立てておく

- プライベートPaaSのようにCoreOSを立てておく
- そのCoreOSに対してデプロイする

# メリット

- (既にビルド済みのコンテナであれば)早い
- お金が余計にかからない
- プライベートPaaS持ってるって言える
- プライベートビーチみたいな

# デメリット

- ネットワーク性能ギリギリまで使うベンチやCPUをいっぱいまで使うベンチには使えない
- kuiperbeltはそういうことあるので複数インスタンス立ち上げてベンチ取りました

[← Back to Documentation](#)

## Running CoreOS on a Vultr VPS

These instructions will walk you through running a single CoreOS node. This guide assumes:

- You have an account at [Vultr.com](#).
- The location of your iPXE script (referenced later in the guide) is located at `http://example.com/script.txt`
- You have a public + private key combination generated. Here's a helpful guide if you need to generate these keys: [How to set up SSH keys](#).

The simplest option to boot up CoreOS is to load a script that contains the series of commands you'd otherwise need to manually type at the command line. This script needs to be publicly accessible (host this file on your own server). Save this script as a text file (.txt extension).

### Choosing a Channel

CoreOS is designed to be [updated automatically](#) with different schedules per channel. You can [disable this feature](#), although we don't recommend it. Read the [release notes](#) for specific features and bug fixes.

Stable Channel

Beta Channel

Alpha Channel

The Stable channel should be used by production clusters. Versions of CoreOS are battle-tested within the Beta and Alpha channels before being promoted. Current version is CoreOS 367.1.0.

# VULTRにCoreOSを載っける

<https://coreos.com/docs/running-coreos/cloud-providers/vultr/>

# もう一つの提案 レシピを持つ

- いちいちyumとかaptとかやってられない
- chefでもansibleでもシェルスクリプトでも  
いいから自分用のレシピを持っておく
- 各々のスタイルが良い

# VULTRでのStartup Scripts

- 公開鍵の流し込み(VULTRだけかも)
  - githubから
- よく使う系のモジュールのインストールとか
  - build-essential / mosh / tmux / vim
- githubのdotfilesから\*rcを取り込む



# 今の時代に自宅サーバの意味

# 自宅サーバのデメリット

- 電気代
- イニシャルコスト
- 性能を上げるにはまた初期投資が必要
- 場所代
- 睡眠を妨げられることがある

普通に何か実験やサービスを  
やる分にはVPSなどに  
頼ったほうが良い

# 自宅にサーバを置くところなる

- 雷が落ちてもるもる飛ぶ
- 引っ越しとかしにくくなる
- 電気代 バカにならない

それでも

まだ自宅サーバが必要な用途

# 例えばGPGPUの場合

- AWSのGPUインスタンス(g2.2xlarge)は価格が\$0.898。1時間90円は不安感ある(※)
- 性能が固定なのでもっとスペックがほしいとかだと選択肢がない

# 例えばTV番組録画環境とか

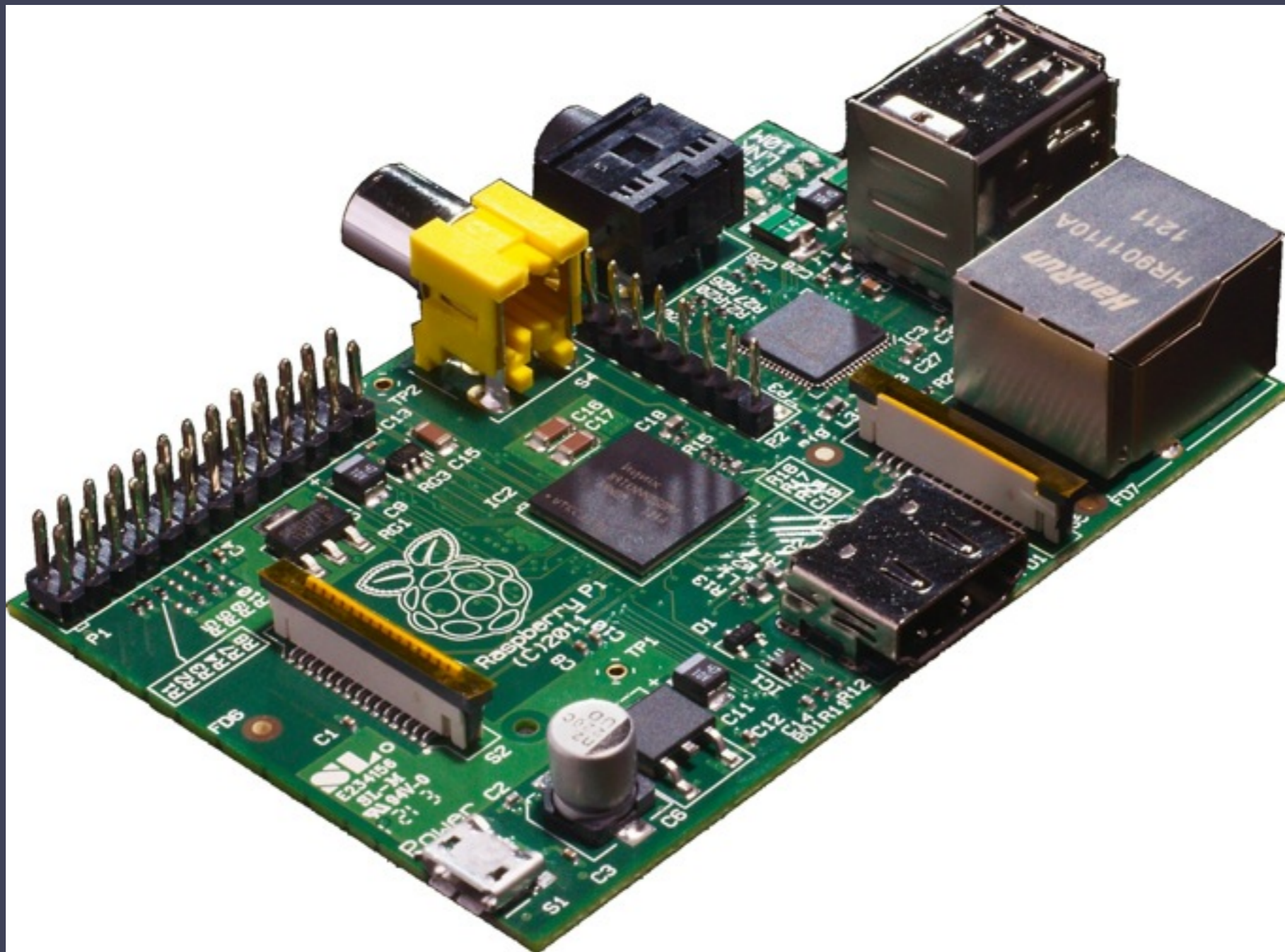
- 録画は(今のところ)自前でやらないといけない
- WOLやタイマーで録画時間に起動して録画してエンコしてシャットダウンとか涙ぐましい努力が必要

# ストレージがめっちゃ欲しい

- テラバイトクラスのストレージをクラウドに求めると大変なことになる
  - S3で低冗長化で\$24.6/month + a
- 冗長化をあまり気にしなくて良いならHDD買ってきてそれ使ったほうが良さそう



# 自宅サーバとしての Raspberry Pi



[http://commons.wikimedia.org/wiki/File:Raspberry\\_Pi\\_Photo.jpg](http://commons.wikimedia.org/wiki/File:Raspberry_Pi_Photo.jpg)

# Raspberry Pi

# メリット

- 低消費電力(普通に使えば最大0.5W程度)
- 小さい
- 必要最低限の機能は備えている
  - ネットワーク
  - USB

# 使ってみた感想

- 熱系トラブルが有る
  - ケースに入れている人に多い
- 外付けHDDにガンガン書き込む系だとめっちゃ遅い 帯域が足りない
- 軽いアプリケーションならOK t1.micro程度

# Raspberry Piは Immutable Infrastructure(物理)

# SDカードがイメージ移動や！！

- それぞれの環境は一緒なのでSDカードを抜き差しするだけで環境を変えられる
- 新しいSDカードにイメージ書き込めばすぐにまっさら環境
- 安いのでいっぱい買える = 複数台の試験とか

# まとめ

- 各サービスは一長一短。使い分けよう
- 時代はクラウド。それでも自宅サーバが必要になる場合があるし、自宅サーバはロマン
- サーバは立てて捨てる時代なので皆立てて捨てまくって試せば良いと思う。怒られない
- 今年のYAPCはRasPiがトレンドでは！？！？