

コンテンツ屋の運用現場

SaaSの活用と運用

株式会社ミクシィ
吉野純平

SaaS利用の前に考えるべきこと

SaaS利用前に考えるべきこと

- 利用規約を熟読しよう
 - そして法務と相談しよう
- 源泉徴収について理解しよう
 - 経理と相談しよう

今日の紹介は解釈等の幅が大きいので、
詳細は述べず、確認する上でのキーワード

その1：よくある利用規約の例

- **（規約の解釈は法務と相談しましょう）**
- 「参考にして開発してはいけません」
 - 証明できないので実質同じ業態の事業ができない
- 契約を解除しても「残存」する項目
- 「適用範囲」 契約主体？グループ会社全体？
- 「税金はすべて利用者が払います」（後述）

その2：海外サービスの場合の国際税務

- (信用せず各自、経理や税理士等に確認してください)
- 源泉徴収をする義務は、支払側にある
 - 指摘を受けるのは、利用者側。追徴もありうる。
- 外国法人や非居住者への支払いが特に注意
- 租税条約があるなしで変わるところがある
- 租税条約があったとしても関連書類が必要な場合も？
- 国内代理店があっても徴収代行の色が強い場合には対象の可能性あり

ここから先は弊社のシステム運用周りの話

我々の環境の特徴：とあるゲームの場合

- おおよそ月に1回、サービス停止を含むメンテナンスがある
- 瞬断が懸念される以上の作業は、メンテ時に実施
 - いかにサービスメンテ時以外にできる作業を増やすかがキモ
 - レイヤ3での迂回は普段から実施
 - L2やVRRP等はメンテ時のみ（いまは存在しないけど）
- SaaSを利用するところが多い
- ネットワーク構成も運用ツールも攻めの姿勢で行った
 - 2014年ごろは今よりも少人数でのNW運用だった
 - 説明や同意を取る人数が少ない=>私の思うとことん良いをやる

設備と運用者

- 利用DC数：4(Network POP 2, Server DC 2)
- センター：都内のみ
- 運用者：自社 6人。委託先 おおよそ7人
 - 普段から触っている人数は実質5人くらい？
 - 基本同じ場所にはいない
 - 全員リモートでも日常業務が可能なようにする
 - 会議で何か決めるのではなく、オンライン上で決める

全ては情報格差の是正のために

- 必ずオンラインで終わること
 - チャットツールやチケットシステム
- 口頭で話したは
 - 全体に伝わらない
 - 検索できない
- インターネットで見られるとベター
 - VPNがないほうがいい

具体的に使っているツールの紹介

使っているSaaS等のツール郡

- slack
- github
- CircleCI
- Pagerduty
- stackdriver logging
- ツールの認証（githubやgsuiteアカウントで認証）

すべてはgitを使って管理してgithub.comに置く

- git => インターネットがなくても作業できる
 - 全てローカルにミラーがある
- githubじゃなくても良い
 - 止まっても即座に運用への影響がない
- 過去のすべてのがある(2013年くらいから)
 - コンフィグの意味、レビュー
 - デザインのコンセプトとレビュー歴

CircleCI

- 開発したソフトウェアの自動テスト
- ネットワーク設定のテストは現時点は自作CIサーバ
 - junoser等でテストをする

Pagerduty

- アラートの通知に利用
- 呼び出される順序等を設定可能
- APIがあり交代等をボットにさせられる
 - 通知が出るような作業をするときに作業者に変更等
- 自社網じゃないところから通知を飛ばしている
- また閉域網を介した別実装も並行運用している

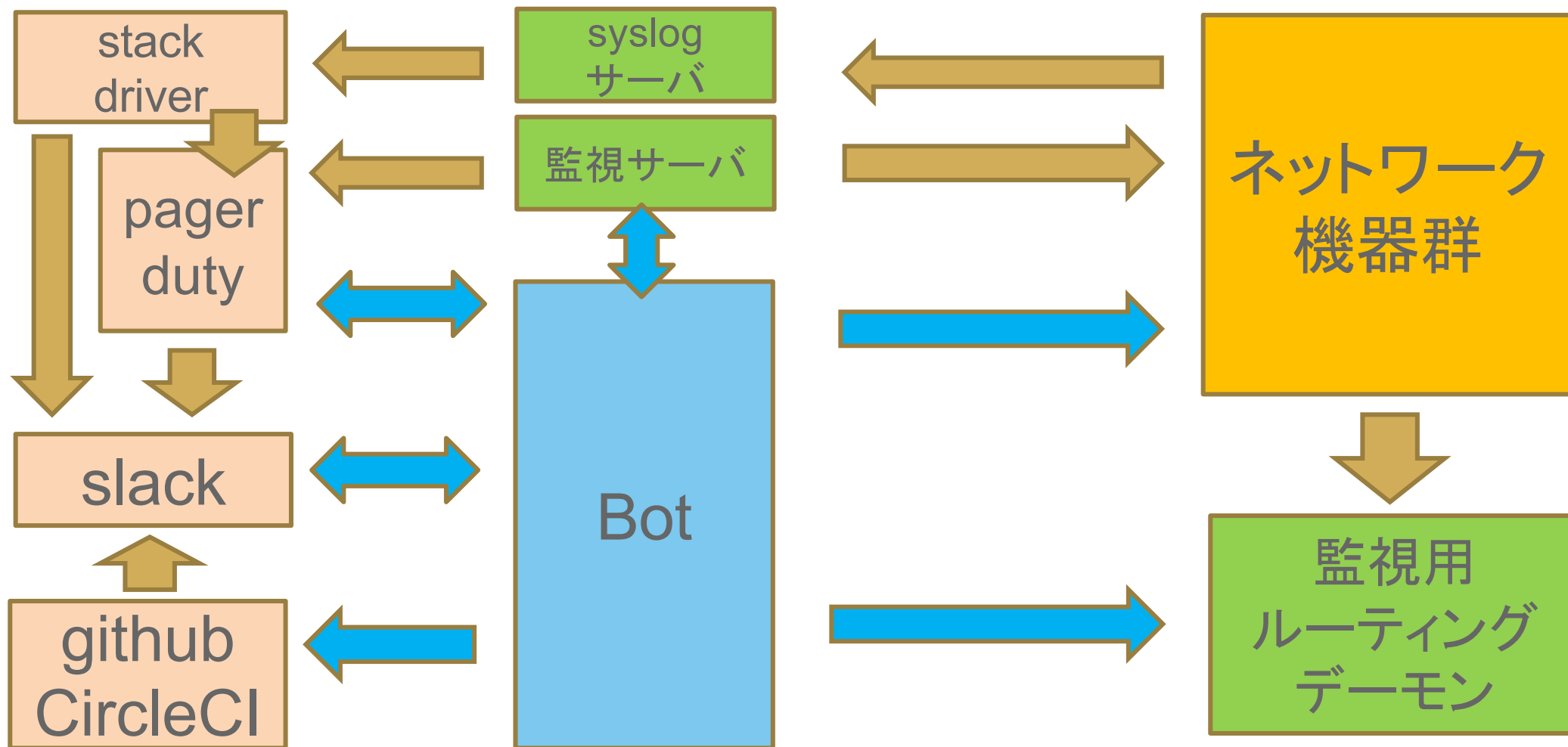
stackdriver logging

- ログのチェックに利用
- syslogサーバはシンプルにデータ投入とローカル保存のみ
 - 昔はここに設定がいろいろあった
- 主な使い方
 - 緊急度等の振り分け
 - 即時対応が必要なもの
 - 翌営業日に対応が望ましいもの（その営業日タイミングで通知）
 - 影響ないが把握のためslackに表示しておくもの
 - ログには出てしまおうが、影響ないと確認済みのものを抑止

各種ツールの認証

- githubやg suiteのアカウントでの認証を実施
- 認証proxy経由でツールアクセス
 - VPNをしなくてもツールに触れる
 - スマホからも状況確認くらいはできる

全体のつながり関係



BotはSaaSじゃありません

- 運用メンバーの自作
- 正常性確認や監視抑止等のツールがかなり強い
 - **自動化するなら自動設定投入よりも正常性確認**
 - 正常性確認ができないのに自動化なんてできない
 - 作業時間ウエイトもかなり大きい

ネットワーク機器のデプロイの実際

- 最新に近いバージョンを追いかけている
- known issueを回避or欲しい機能を使いたい
- L3設計で迂回を楽にすることでなんとかできている
 - ただもうちょっと楽はしたい
- ステップ
 - 事前検証
 - 本番投入（ウエイト大きめにする）
 - 平常運用・横展開

