2016.07.08 JANOG39@金沢市文化ホール
見つけた！モダンなトラフィック可視化 BoF

データ収集・解析基盤の構築苦労話

NTT コミュニケーションズ 技術開発部
Data Science & AI Technical Unit / Cutting Edge Strategic Unit
亀井聡
<skame@nttv6.jp>

自己紹介

- 亀井聡
- ～2012.06 @ NTT 研究所
  - 品質とかトラヒックとか．ネットワーク装置からP2Pまで．
- 2012.07～ @ NTT コミュニケーションズ 技術開発部
  - インターネット計測と分析
    - ネットワークチームからデータベースチームまで．
    - インフラ系解析チームと解析環境立ち上げ．
    - 気がついたらAIまで…….
    - そしてまたなんか一個増えてる．
  - Data Science & AI Technical Unit (2016/05～)
  - Cutting Edge Strategic Unit (2016/11～)

JANOGとのかかわり

- JANOG13@赤坂プリンス 「広がるP2Pサービスとインターネットインフラへの影響」 2003.10
- JANOG14@宮崎 「オーバーレイネットワークの可能性と. そのインパクト」 2004.07
- JANOG/信学会IA研究会共催@神保町IIJ 「[特別講演]コンテンツ配信を中心とした国内インターネットの構造分析」 2012.09
- JANOG33@別府 「イベントトラフィックに対するトラフィックエンジニアリング」 2014.01
- JANOG38@那覇「できる! データドリブン障害検知」2016.07

モダンなトラフィック可視化って？

- ビッグデータ的な取り組みはWeb界隈の技術が結構進んでいる
  - datadog とか Mackerel とか.
  - AWS とかに放り込んで Redshift とか BigQuery とか.
- 一方運用現場では mrtg でごにょごにょ syslog のフィルタを…….
  - ほら，あの，通秘とかあるから．全部集めるためには信頼性の高い基盤をオンプレで置かないと…….
  - 大量の秘伝のタレ．自由記入形式のチケットの山.
    などなど
- Web方面が頑張ってくれたおかげでオープンソースでも結構ツール揃うよ.
  - とはいえツール多過ぎないか？
  - コード書かないと拡張できないとか，保守つけたい部分もあるよね.
  - 見取図が欲しいけどよくわからない.
- とりあえずざっと動かしてみた経験を共有したい.
  - もちろんいろいろ欠けてますし，網羅性はないですが，ご意見募集.

データ収集と活用の流れ

- データ取得・発生
- データベース/データストア
- 可視化

基本的にはこれだけ.

だが，可視化では気付き，までが主なので分析も必要.

ツール類

- 従来型
  - zabbix / mrtg / cacti / syslog
- データ取得
  - beats / logstash / fluentd / telegraf / pmacct
- キュー
  - kafka / rabbitmq
- データベース/データストア
  - Elasticsearch / influxdb / graphite / cassandra / hdfs
  - csv / tsv / json をストレージに
- 可視化
  - grafana / re:dash / kibana

データの分析・深掘り

- サマライズ(秒単位→時間単位)
- タグ付け(ログイン成功・失敗)
- JOIN(geoipくっつける，顧客IDとIP紐付ける)
- syslog の message を機器毎に別ロジックで parse する
- anomaly detection を時系列で
- 機械学習
- DNN
  :
  :

どのステップでやるかとか，ETL なのか ELT なのか，とか.

Extract Transformation Load

- bigdata 界隈だととりあえず生で突っこんであとで考える
  - IoTではエッジでけずりたい
  - オペレータ業務的にはエッジで削るのが主流か.
    - 有効なアラートだけ精度良く欲しい.

作ってみた(-ing)経験談

- ある程度汎用で作りたいよね，とかいって汎用ログ基盤を作る，といってどんどん仕様がふくらんではや○年，とかいうのも避けたい.
- とりあえずできるところ・必要なところから作ってみながら考えよう(技術開発部隊だからこそできたのかも).
  - まずは csv で貯め込んで R で解析.
  - CPU足りなくなったのでマシン増やしてNAS(nfs)にした.
  - 統計処理つらくなったので elasticsearch に入れてみた.
  - 速度的にはいい感じ．JOIN処理とか無理．あとクエリ言語が独自辛い.
  - hive クラスタ組んだ．マスタデータをHDFSに.
  - presto で JOIN とか快適.
  - リアルタイムっぽい処理もしたい.
  - 自明(?)なタグ付け等の処理はデータ来た時点で終わらせておきたい.
  - kafka クラスタ導入．こいつも presto で刺せる.

残課題(テクニカルなもののみ)

- 監視の監視
  - ツールによってあったりなかったり
- トリガー制御(lambda的な)
  - やりたいけどオンプレだと辛い．Minio に機能はある(ドキュメントがない)
- リアルタイム処理(よく言われるが，マイクロバッチでほとんどok)
  - Storm / spark streaming をちょっと試した．
- ジョブ・フロー制御
  - Airflow とか nifi とか試しているところ．
- デプロイ戦略
  - 今は docker + git で作った Immutable コンテナを swarm にデプロイ．Gitlab-ci での自動化は道半ば．
- 機械学習とDNN
  - ユースケースがなぁ．

などなど

実例コーナー

- テストベッド環境での 監査とか集めてるデータで可視化できない？外向きに開いてる ssh とか.
- と，いきなり言われた.

ということでざっくり方針

- syslog で飛ばしてもらうのが楽だけど，設定換えてもらうのが利用者のレベルのばらつきを考えると難しいかも.
- authlog ぐらい /var/log の下にデフォルトでいるだろう，ということで agent 型で設計.

- logstash でプロト実装．pattern ゴリゴリ書いて目的は達成できそう．あとはどうデプロイしてETL処理をどこでやるか.

- クライアントは軽くする，ということで filebat で読み込むことに.

設計

- filebeat
  - input /var/log/*auth*
  - output kafka-topic1

- logstash1
  - input kafka-topic1
  - output kafka-topic2

基本的な ETL が済んだものが topic2 に貯まる.

- logstash2
  - input kafka-topic2
  - output kafka-topic3

ssh 固有の処理を加えたものを topic3 に．ちょっと容量無駄遣いだが.
ついでに geoip で緯度経度とか ASN 付加している.

- logstash3
  - input kafka-topic3
  - output hdfs
  - output elasticsearch

もうちょっと深掘りしたい(分析)

- フローと突合したいよね
  - pmacct で取ってるデータがある.
- hive に入ってるので比較できる.
  - 認証通ってないアドレスのフローとか大丈夫？サンプリングだから100%ではないが，とか.
- 有効なIDのリストとのJOIN．急に増えた人とかいない？
  - このへんはアドホック分析 w/ python / R using presto で.
- 最初から入れておくべきロジックが洗い出せれば logstash のところに戻す.
- リアルタイム処理したければ，storm / spark streaming / kafka stream が必要だが，さて.

もうちょっと手がかからないようにしたい

- Kafka は結構 HA 強力
  - 並列でデータ取ってもそれほど問題は起きないので，logtash を複数走らせればなんとかなりそう.
- Kafka / confluent のエコシステムをもうちょい掘る
  - hdfs/hive connector とか elasticsearch-connector で logstash 部分を簡素化できる筈.
- そして kafka 依存度が上がっていくという.
- DevOps / ChatOps / CI / CD からの輸入も重要そう.
- 他に使えそうなノウハウはどのへんから引くべき？