KOF2013: 関西オープンフォーラム@大阪南港ATC 2013/11/08(金)

JSONでメール送信

できるHTTP API SERVER "Haineko"

@azumakuniyuki Cubicroot Co. Ltd.

自己紹介

@azumakuniyuki

JSONでメール送信

メール送信用HTTP API "Haineko"

Haineko

〝はいねこ〟

HTTP
API
INTO
ESMTP
K=undef
O=undef

Haineko の概要

- Plack/PSGIアプリケーション(Perl)
- リレーサーバ(**キューは持たない**)
- HTTP POSTでメールを送る
- 他のSMTPサーバかEmailクラウドへリレー
- 応答は全てJSON
- Haineko に渡す時のBASIC認証対応
- **$REMOTE_ADDRでリレー制限(relayhosts)**
- **許可された宛先のみに送信(recipients)**

Architecture

Hainekoの処理の流れ、全体図と部分図。

Haineko Flow

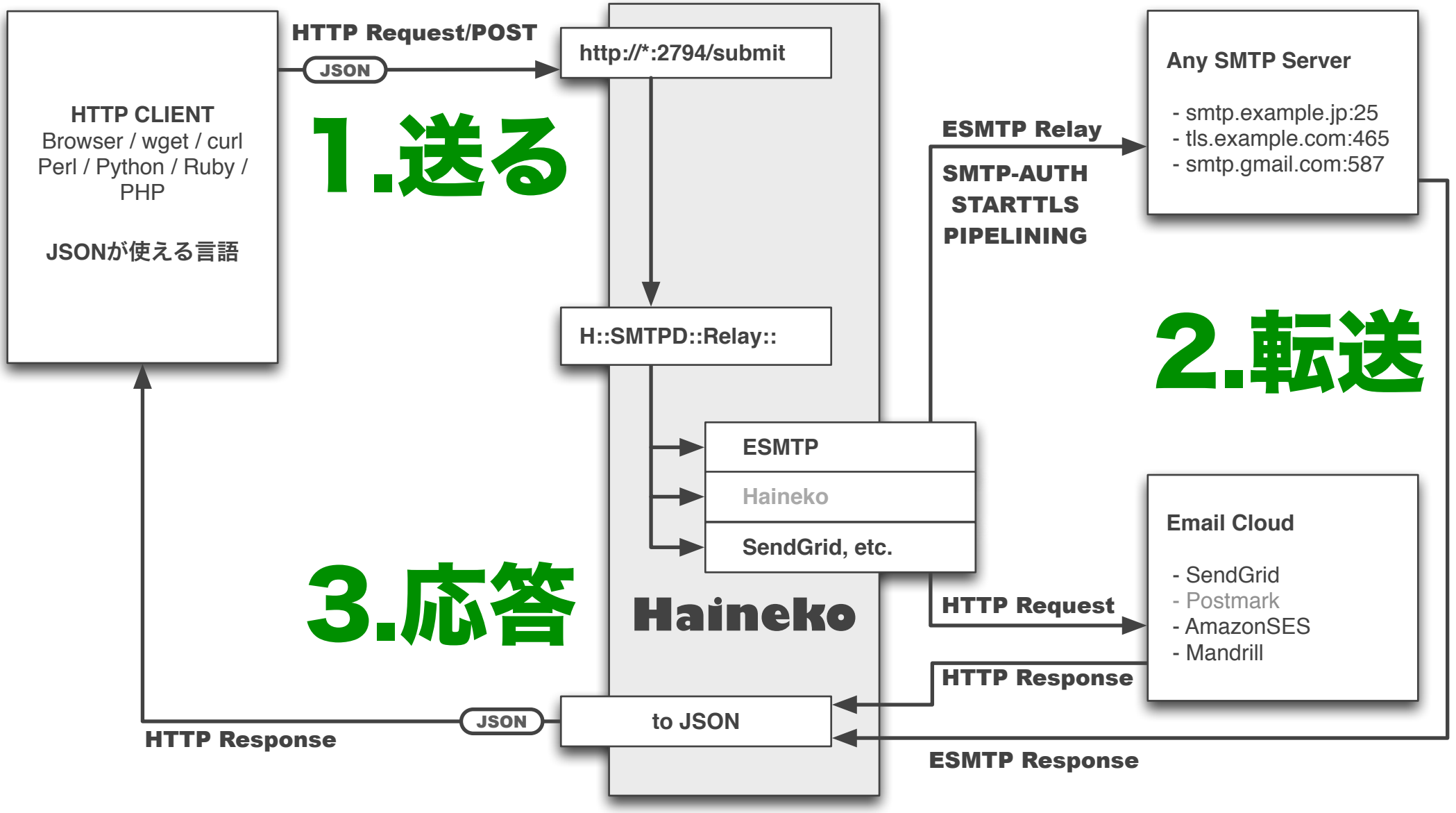

Table Files

各種の制限項目、経路テーブルなど。

etc/mailertable

- 宛先ドメインでのルーティングテーブル(YAML)

```
default:  ← mailertable, sendermtのどちらにも一致しない場合
  mailer: 'ESMTP'
  host: '192.0.2.22' ← このSMTPサーバの25番に接続して送信(リレー)
  port: 25

example.com:   ← 宛先が *@example.comの時は
  mailer: 'ESMTP' ← ESMTPで(Haineko::SMTPD::Relay::ESMTPを使う)
  host: 'smtp.gmail.com'← Gmailの587番ポートに接続
  port: 587
  auth: 'Google'← etc/authinfoの"Google"認証情報を使う
  starttls: 1 ← SMTPセッションでSTARTTLSを使う
```

etc/sendermt

- 発信ドメインでのルーティングテーブル(YAML)
- mailertableと同じ書式

```
example.jp:  ← 発信者が *@example.jpの時は
  mailer: 'ESMTP'
  host: '192.0.2.253'← このSMTPサーバの25番に接続して送信(リレー)
  port: 25

example.org:  ← 発信者が *@example.orgの時は
  mailer: 'SendGrid'← Haineko::SMTPD::Relay::SendGridを使う(Web API)
  auth: 'SendGrid' ← etc/authinfoの"SendGrid"認証情報を使う
```

etc/authinfo

- SMTP-AUTH, EmailクラウドのAPI用認証情報(YAML)
- ファイルのパーミッションに注意

```
Google: ← mailertable, sendermtの``auth''で指定したラベル
  username: '******@gmail.com'
  password: 'nekochan22' ← Gmailのユーザ名とパスワード

SendGrid: ← mailertable, sendermtの``auth''で指定したラベル
  username: 'kijitora' ← SendGrid APIのユーザ名(API_USER)
  password: 'nyanko22' ← SendGrid APIのパスワード(API_KEY)
```

etc/relayhosts

- メールのリレーを許可するクライアントのIPアドレス
- IPアドレスかネットワーク帯域を記述(YAML)
- open-relayは危ないので将来削除するかも

```
open-relay: 0 ← ``1''にするとIPアドレスのチェックをしない(危険)
relayhosts: ← 許可するIPアドレスかネットワークを配列で列挙
  - 127.0.0.1
  - 192.0.2.22/32  ← Net::CIDR::Liteが理解出来る形式で
  - 168.254.0.0/16
```

etc/recipients

- メールを送っても良い宛先のアドレスかドメイン
- このファイルに一致しない宛先には送らない
- open-relayは危ないので将来削除するかも

```
open-relay: 0 ← ``1''にするとどんな宛先にも送れる(危険)
domainpart: ← 送っても良い宛先をドメインで指定(配列で列挙)
  - 'example.co.jp' ← *@example.co.jp宛はなんでもOK!
recipients: ← 送っても良い宛先を個別に指定(配列で列挙)
  - 'neko@cat.example.jp'
  - 'kijitora@example.com'
```

etc/password

- bin/hainekoctl -Aか$HAINEKO_AUTHで要BASIC認証
- export HAINEKO_AUTH=/path/to/password
- BASIC認証なしでも動作します
- BASIC認証が通ったらJSONでPOSTできる

↓ BASIC認証のユーザ名

haineko: '{SSHA}9p2euyteR33mp0TYKjmYhTlIt1ctxuRn'
nekosan: '{SSHA}ai350adgaiADGNkla0p3iNEKOCHAN222'

↑BASIC認証パスワードのハッシュ

Install

Hainekoのインストール

Install Haineko

A. git cloneしたリポジトリから直接起動できます
  - ./bin/hainekoctl --devel start

B. configure && make install (/usr/local/haineko)
```
$ cd /usr/local/haineko
$ ./bin/hainekoctl setup
$ ./bin/hainekoctl start
```

C. cpanm .
```
$ cd /usr/local && ./bin/hainekoctl setup
$ ./bin/hainekoctl start
```

Sending & Reply

メール送信とJSONでの応答

Start Haineko

- **bin/hainekoctl** 制御スクリプト

```
$ bin/hainekoctl -d start (開発モードで起動)
$ bin/hainekoctl start (Productionモードで起動)

$ plackup -o '127.0.0.1' -p 2794 -a haineko.psgi

$ bin/hainekoctl stop (pidファイルを読んで停止)
$ bin/hainekoctl restart
```

メール送信(curl)

```
$ curl 'http://127.0.0.1:2794/submit'
    -X POST
    -d '{ ehlo: "ホスト名",
          mail: "neko@example.jp",
          rcpt: [
            "kijitora@example.org",
            "mikeneko@example.com" ],
          body: "ネコと和解せよ",
          header: { subject: "ニャー!!", … } }'
  | /usr/local/bin/jq .
```

メール送信(LL)

- JSONが使えてHTTPも使える言語ならなんでも
- ソースコードの"eg/sendmail.*"にサンプル

- **Perl:**
  - use Furl, JSON;

- **Python:**
  - import json, httplib, urllib

- **Ruby:**
  - require 'json', 'net/http', 'uri'

正常応答(JSON)

```
{
  "smtp.queueid":"r6DCXsV00649aTl2",
  "smtp.response": {
    "dsn": "2.0.0",
    "error": 0,
    "message":[ "Accept for delivery" ],
    "command": "QUIT",
    "code": "221" },
  "smtp.useragent": "ユーザエージェント(USER_AGENT)",
  "smtp.recipient": [ "kijitora@example.jp"],
  "smtp.addresser": "NNN@example.org",
  "smtp.remotehost": "127.0.0.1", "smtp.remoteport":49989
}
```

エラー応答(JSON)

```
{
  "smtp.queueid":"r6DCXsV00649aTl2",
  "smtp.response": {
    "dsn": "5.7.1",
    "error":1,
    "message":[ "Recipient address not permitted" ],
    "command": "RCPT",
    "code": "533" },
  "smtp.useragent": "ユーザエージェント(USER_AGENT)",
  "smtp.recipient": [ "kijitora@example.gov" ],
  "smtp.addresser": "NNN@example.org",
  "smtp.remotehost": "127.0.0.1", "smtp.remoteport":49989
}
```

Repository

github.com/azumakuniyuki/Haineko

Haineko SMTP 検索

終