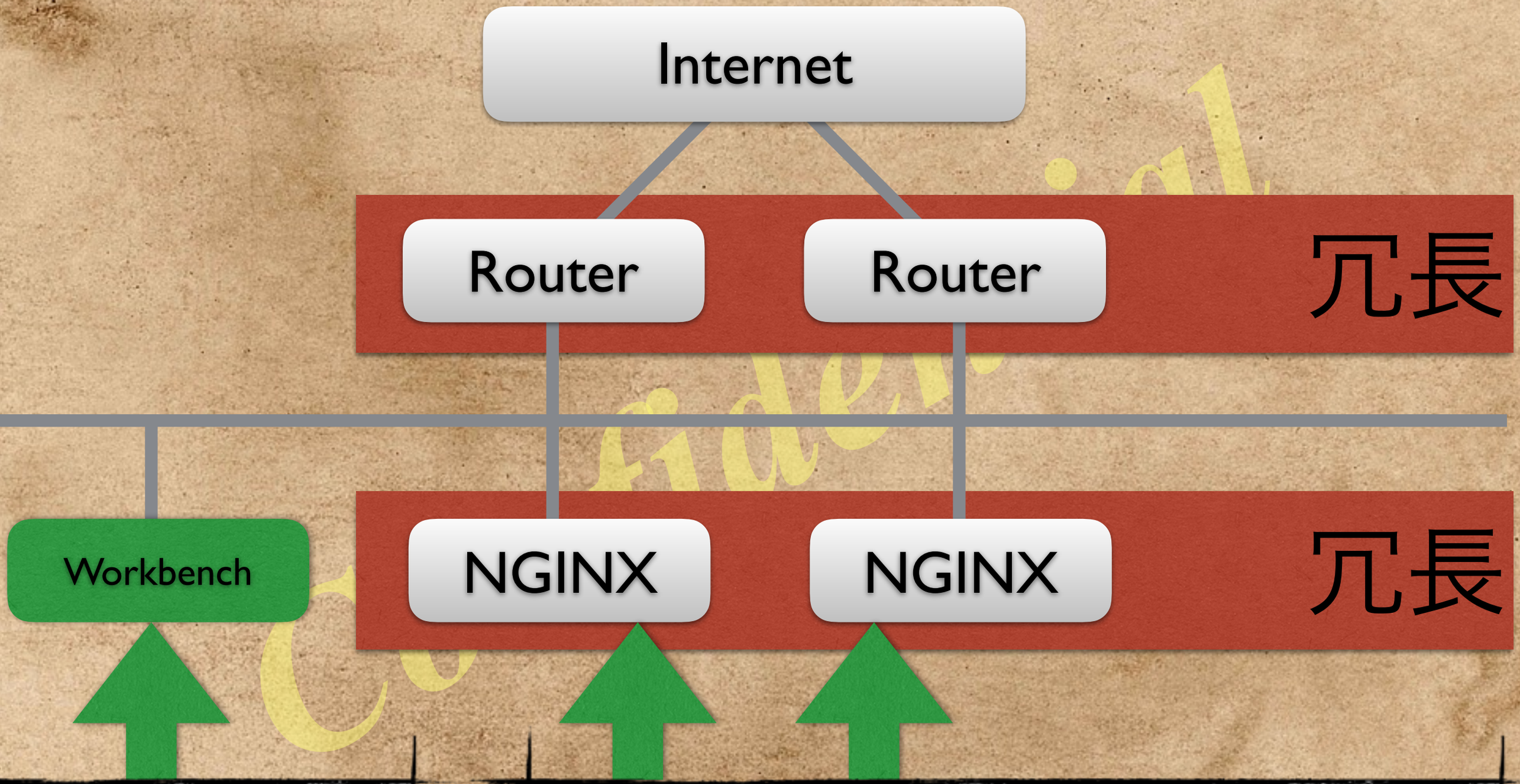
Let’s Encryptの証明書更新

- Let’s Encrypt の証明書更新を思い通りやりたい
- 冗長構成を取っている時の証明書の更新が面倒くさい
- 「おそらく」「ふつー」は、py-certbotを利用？

こんな環境



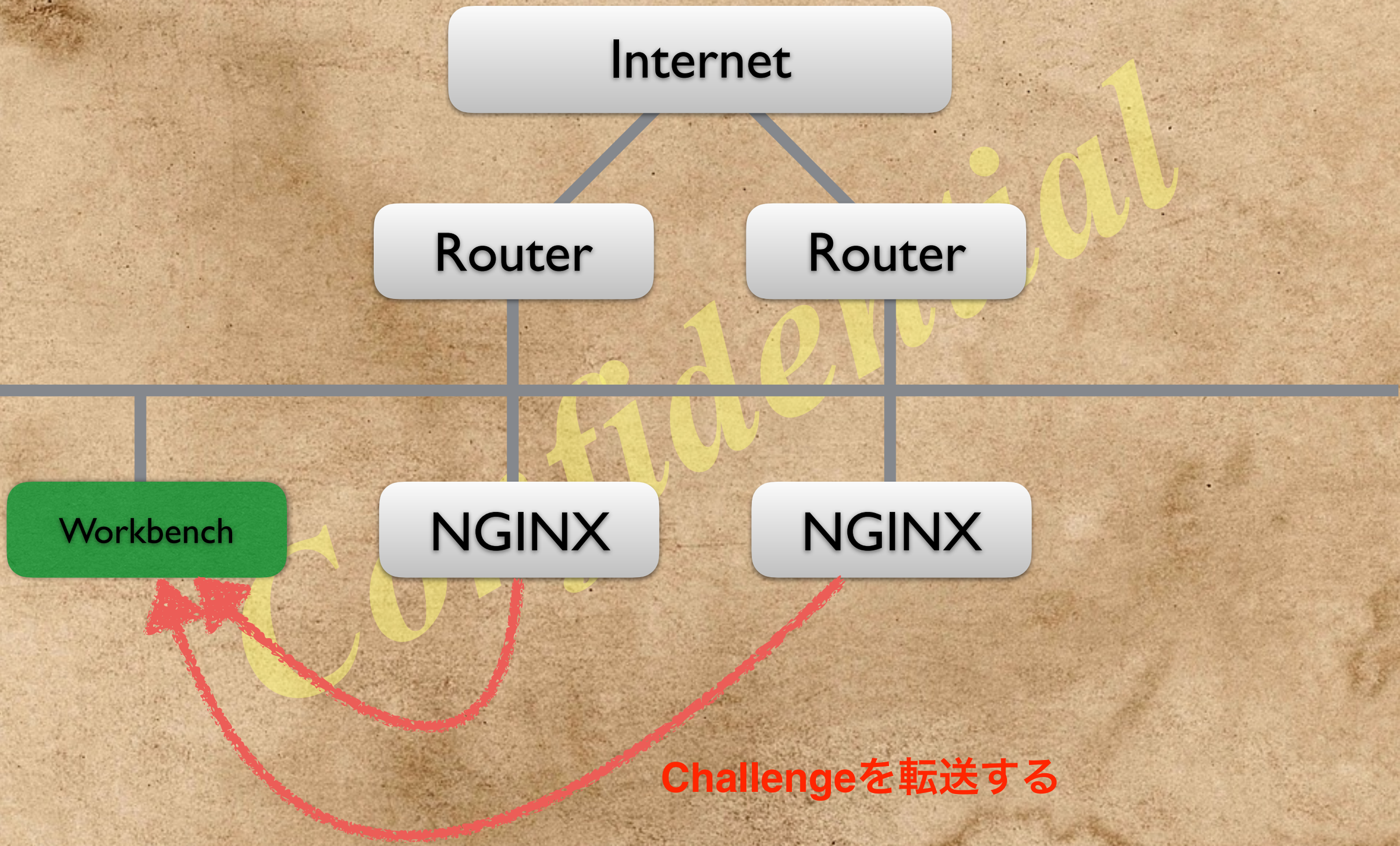
問題

- py-certbotは完全にBlackbox
- 何をやっているかわからない
- NGINXへのchallengeをどうするか

解決

- security/acme-client を使う
- sshでそれぞれに配送する
- challenge通信はworkbenchにproxyで飛ばす

こんな環境



- 豆
- sudoresに色々技を使えます
- 証明書なので、権限大事
- Let’s Encryptの証明書は90日で失効する
- cronとかweeklyとかで頑張ろう

・sudoresの豆

```
account ALL=(ALL) NOPASSWD:chmod,chown
```

NGINX側設定

```
upstream HTTP_ACME {
    server xxx.xxx.xxx.xxx;
}


server {
    ....
    location ^~ /.well-known/acme-challenge/ {
    proxy_pass      http://HTTP_ACME;
}
```

肝は、.well-known/acme-challenge 宛の通信を転送すること

作ったscript

```
#! /bin/sh

# Let's Encrypt Certificate renewal script for
# FreeBSD and acme-client
#   Copyright (C) by seirios@seirios.org
#
# Usage: crt-upd.sh [target domains...]

DEBUG=0
ACME_BASE=/home/seirios/htdocs/acme

ACCKEY=${ACME_BASE}/SSL/privkey.pem
SSL=${ACME_BASE}/SSL
CHALLENGE=${ACME_BASE}/WWW
DOMAINSFILE=${ACME_BASE}/domains.txt

UID=`id -u`
[ ${UID} -ne 0 ] && echo "Must run on root/UID=0" && exit

if [ ${DEBUG} -ne 0 ]; then
  ECHO="/bin/echo"
else
  ECHO=""
fi

if [ ${#} -eq 0 ]; then
  DOMAINS=`cat "${DOMAINSFILE}" | sed 's/[#|].*$//' |
while read DOMAIN line ; do
    echo -n "${DOMAIN} "
  done`
else
  DOMAINS=${@}
fi

[ ${DEBUG} -ne 0 ] && /bin/echo "Target domain: $
{DOMAINS}"

for i in ${DOMAINS}; do
  echo "Getting ${i} Certificates"
  DOMKEY=/home/seirios/htdocs/acme/SSL/$i/privkey.pem
  [ ! -d ${SSL}/${i} ]       && ${ECHO} mkdir ${SSL}/${i}
  [ ! -d ${CHALLENGE}/${i} ] && ${ECHO} mkdir $
{CHALLENGE}/${i}

  ${ECHO} acme-client -bnNv -k ${DOMKEY} -f ${ACCKEY} -C $
{CHALLENGE}/${i} -c ${SSL}/${i} ${i}
done
```

結論

- 意味がわかれば acme-clientはとても便利だ
- ちょっと工夫すると、証明書取得はどこででもできる
- sudoを使う時には、できれば実行コマンド制限をしよう
- Let’s Encryptつかうなら失効に注意