「お名前.com のDDNSをRubyスクリプトで更新

関連記事で「お名前.com の独自ドメインを使った。「お名前.com にはダイナミックDNS機能があり、お名前.comで無料配布されているクライアントソフトを使って更新もできるがWindowsソフト限定である。そこで Rubyスクリプトでで DDNS更新するスクリプトを、Webサーバを動かしている Ubuntu 18.04 LTS Server 上で、cronで定期的に実行して自動更新できるようにしたので備忘録を残す。

準備

Rubyが入っていなければ、下記コマンドでインストール。

```
$ sudo apt install ruby
```

Rubyスクリプト

下記のスクリプトを任意の場所に作成。（例：/opt/script/onamae-update-ddns.rb

UserID,Password,Domainは自分の情報に書き換える。このスクリプトは、サーバパソコンの「ホスト名」でDNSレコード登録するので、「お名前.com の方にDNSレコードを登録しておく。

onamae-update-ddns.rb

```
#!/usr/bin/ruby
# coding: utf-8
require 'socket'
require 'openssl'
require 'date'
#require 'resolv'

#実行時の引数をなしにする場合は、次の行をコメントアウト
# abort "usage: #$0 userid password {'*'|''|subdomain} example.com" if
ARGV.size != 4

#実行時の引数を無効にする場合は、次の行をコメントアウトすると警告が表示されない
#UserID, Password, Subdomain, Domain = ARGV

#引数を有効にする場合は、以下のパラメータをコメントアウト
#引数に指定したパラメータは、以下のパラメータで上書き
UserID = "<user_id>"
Password = "<user_password>"
Domain = "<hogehoge.com>"

CacheFile = "/tmp/_onamae_cache.txt"

#Subdomain = "server-1"
Subdomain = Socket.gethostname          #このパソコンのhostnameを取得
```

```
#現在登録されているIPアドレスの取得
if Subdomain == "" then
  DomainName = Domain
else
  DomainName = Subdomain + "." + Domain
end

if File.exist?(CacheFile) == true then
  file = File.open(CacheFile, "r")
  file.each do |text|
    RegisteredIP = text.chomp
  end
  file.close
else
  RegisteredIP = "."
end

#パブリックキーが変更された場合は、次の行を編集
OnamaePubkey =
'af4447e1ca6e6525561904eb420503d9b353e29c7e9efc3479c03ca79cb71b4e'
DDNSServer = 'ddnsclient.onamae.com'

#静的IPを使う場合は、次の行をコメントアウトすると警告が表示されない
IP = TCPSocket.open(DDNSServer, 65000) {|socket| socket.gets }[6..-2]
#静的IPを設定する場合は、次の行で設定
# IP = "0.0.0.0"

time = DateTime.now
# print(time, ": IP=[" + IP + "], RegisterdIP=[" + RegisteredIP +
"]\n")

# DNSの正引きIPと登録したいIPが同じ場合は処理を中止
exit if IP == RegisteredIP
print(time, ": '" + DomainName + "'のIPアドレスを'" + IP + "'に更新登録の処理を
実行します。\n" )

TCPSocket.open(DDNSServer, 65010) do |socket|
  context = OpenSSL::SSL::SSLContext.new
  context.set_params
  context.verify_mode = OpenSSL::SSL::VERIFY_NONE
  ssl = OpenSSL::SSL::SSLSocket.new(socket, context)
  ssl.connect
  raise "#{ssl.verify_result}" if ssl.verify_result !=
OpenSSL::X509::V_OK
  ssl.post_connection_check(DDNSServer)
  #パブリックキーの確認を無効化するには、次の行をコメントアウト
  #逆に、チェックを有効にする場合は、アンコメント
  #raise "public key is not right" if
OpenSSL::Digest::SHA256.hexdigest(ssl.peer_cert.public_key.to_der) !=
```

```
OnamaePubkey

  begin
    ssl.puts *%W[LOGIN USERID:#{UserID} PASSWORD:#{Password} .]
    response = ssl.gets("\n.\n")
    raise response if response.to_i != 0

    ssl.puts *%W[MODIP HOSTNAME:#{Subdomain} DOMNAME:#{Domain}
IPV4:#{IP} .]
    response = ssl.gets("\n.\n")
    raise response if response.to_i != 0
  ensure
    ssl.puts *%W[LOGOUT .]
  end
end

# print( "'" + DomainName + "'のIPアドレスを" + IP + "'に更新登録の処理を実
行しました□\n" )

file = File.open(CacheFile, "w") do |text|
  text.puts(IP)
end
```

このスクリプトでは、取得した現在のIPアドレスを「/tmp/_onamae_cache.txt」に記録しており、実行時に一致する時にはDNS更新実行をしない。

cronで自動実行

---

cronで定期的に、上記Rubyスクリプトを自動実行する（この例では5分間隔）

```
$ sudo corntab -e
```

下記行を追加。

```
5 * * * * /opt/scripts/onamae-update-ddns.rb >> /var/log/onamae-uamae-
update.log
```

関連記事

---

* □RTX1200□ヤマハルータで「お名前.com□の独自ドメイン名を使う * □Ubuntu□RubyでSMTP over SSLメールを送る * 【お名前.com□IPv4(Aレコード)とIPv6(AAAAレコード)の両方に対応させる

参考

---

1. DNSレコードとは？ - お名前.com 2. “rubyスクリプト”による”お名前.com”のDDNS更新 3. snipsnipsnip/update-onamae-ddns.rb - GitHub 4. お名前.com DDNS

From:
https://yokoweb.net/dokuwiki/ - **新石器Wiki**

Permanent link:
https://yokoweb.net/dokuwiki/programing/ruby-language/onamae-update-ddns?rev=1576978842

Last update: **2019/12/22 10:40**