EC2のインターネット接続(後編)

まりお

@satensi3103

目次

0. 目的

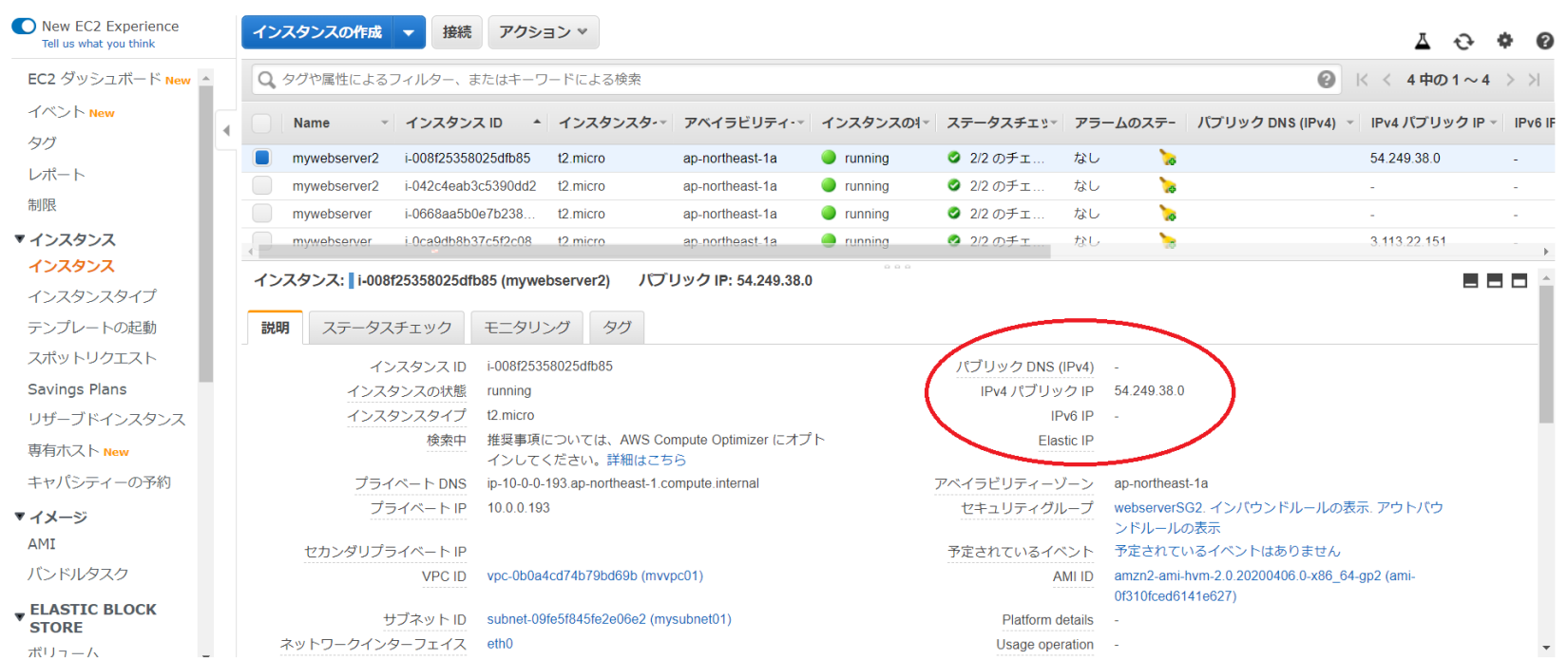
- 前編ではインターネット接続に必要な3つの準備を行った
  ①パブリックIPアドレスの割当て
  ②インターネットゲートウェイの設置
  ③ルートテーブルの変更
- 後編ではEC2インスタンスにSSHでログインする
- ENIの状態確認、パブリックIPアドレス取得も扱う

1. EC2インスタンスにSSHでログイン

- 前章で作成したキーペアファイルを使用するので用意する

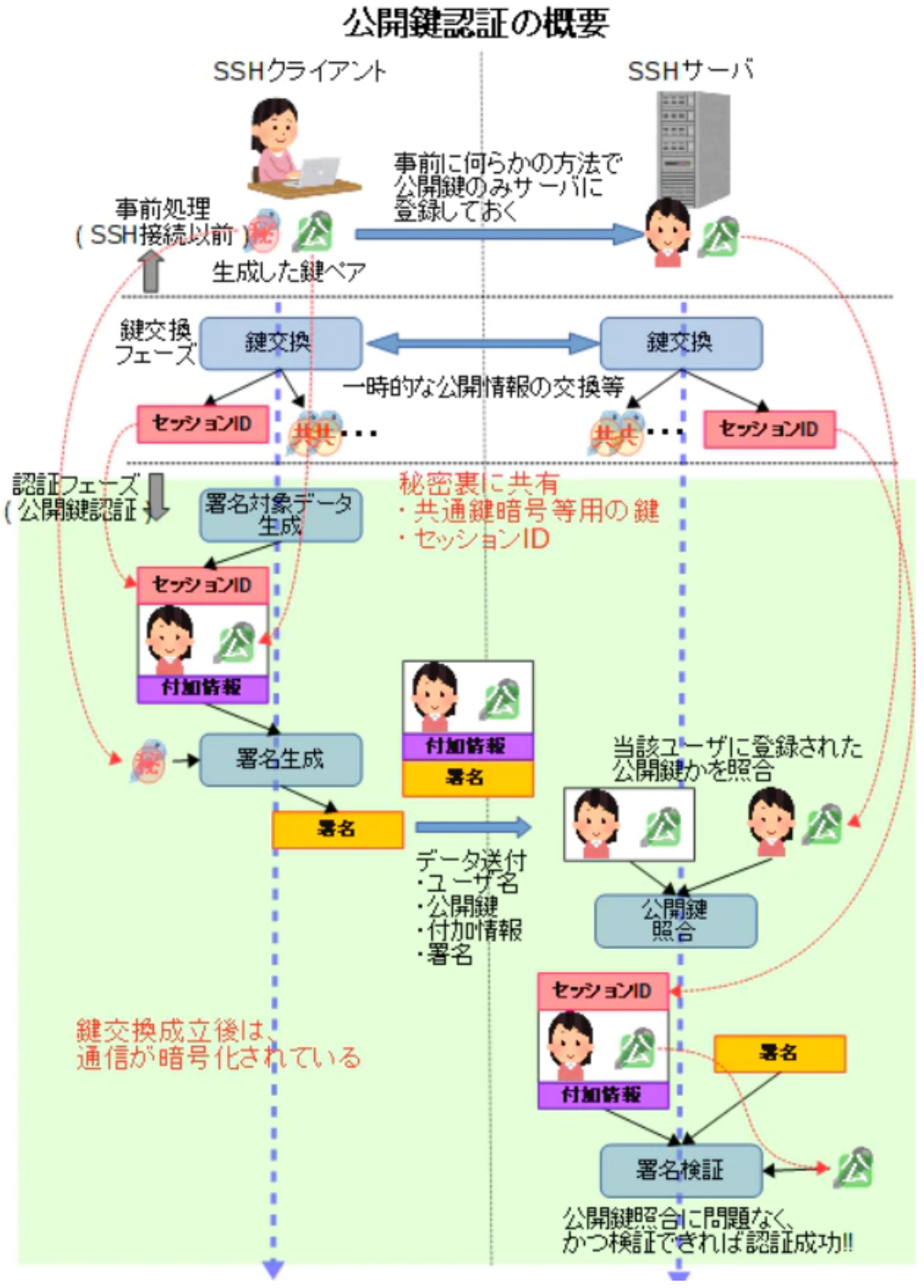
1.1 接続先となるIPアドレスを確認する

- 前半でセットしたパブリックIPアドレスをEC2のメニュー、インスタンスから確認



1.2 SSH

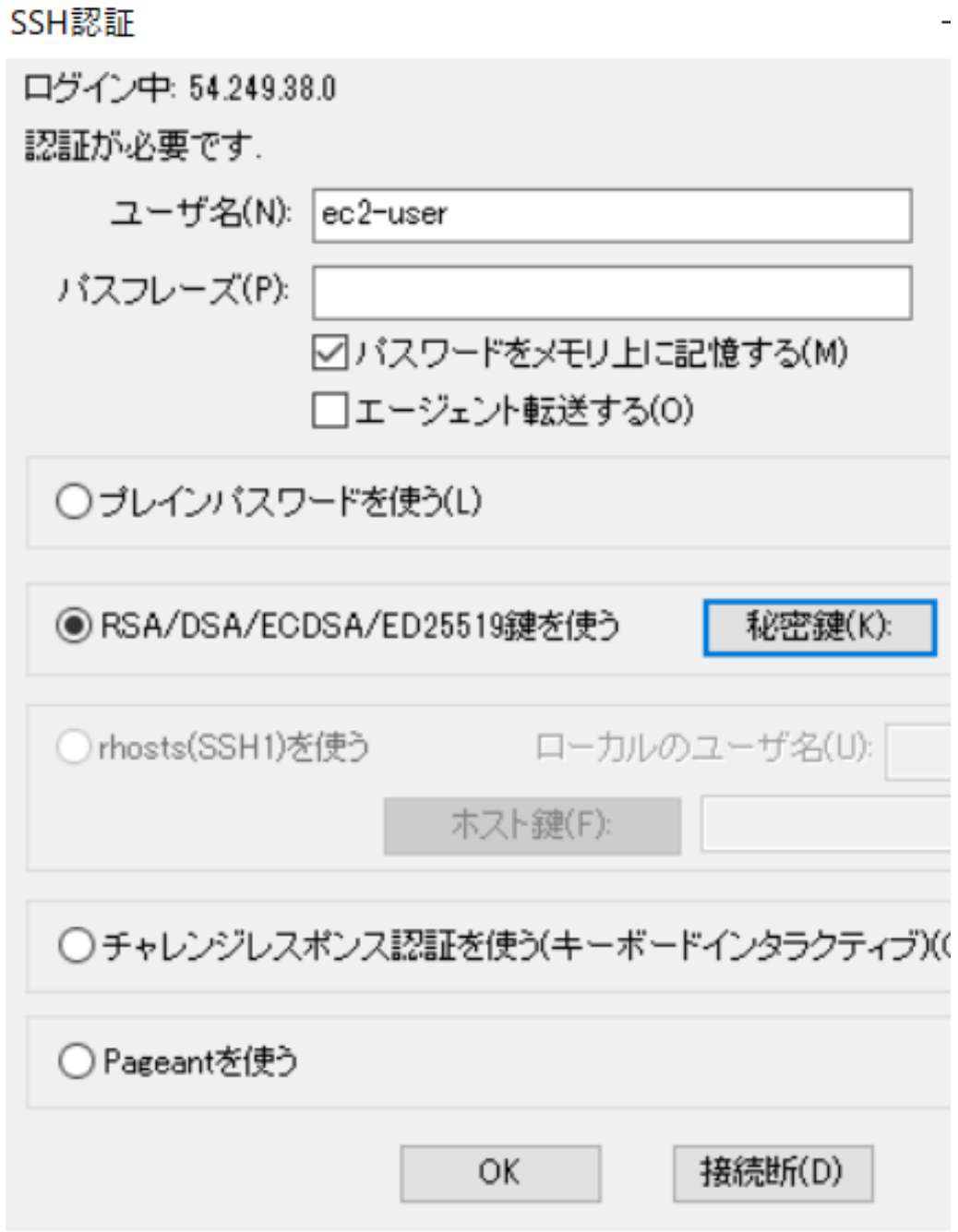
- SecureShellの略称でSSH
- 前回作成した秘密鍵とサーバー(EC2インスタンス)にある公開鍵を使用
- SSHの通信は、セッション鍵、共通鍵暗号による通信暗号化、RSAを利用した公開鍵暗号、の組み合わせといったハイブリッド暗号



1.3 SSHで接続する

- Tera Termの場合は確認したパブリックIPアドレスに接続
- Puttyで接続する人はリンクを参照(Puttygenでプライベートキーの形式を変更)

- Amazon Linux,Linux2の場合、ユーザー名には「ec2-user」を入力
- 「RSA/DSA/ECDSA/ED25519鍵を使う」を選択し、キーペアファイルを選択し接続
- パスフレーズは空欄のまま



1.4 rootユーザーで操作

- 普通のLinuxサーバー同様に扱える
- 「sudo -i」コマンドでrootユーザーに変更できる

```
       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-0-193 ~]$
```

```
[ec2-user@ip-10-0-0-193 ~]$ sudo yum update
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                               | 2.4 kB     00:00
amzn2extra-docker                                        | 1.8 kB     00:00
No packages marked for update
[ec2-user@ip-10-0-0-193 ~]$
```

2 ENIの状態確認・パブリックIPアドレス取得

2.1 IPアドレスを確認する

- ifconfigコマンドを打つ
- eth0というネットワークインターフェイスがAWSにおけるENI
- パブリックIPアドレスの情報はない

```
[root@ip-10-0-0-193 etc]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9001
        inet 10.0.0.193  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::47b:fcff:fef2:84fc  prefixlen 64  scopeid 0x20<link>
        ether 06:7b:fc:f2:84:fc  txqueuelen 1000  (Ethernet)
        RX packets 220763  bytes 100537841 (95.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 184418  bytes 27983559 (26.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 8  bytes 648 (648.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8  bytes 648 (648.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@ip-10-0-0-193 etc]#
```

2.2 DNSサーバーの設定を確認する

- resolv.confにDHCPサーバーから割り当てられたDNSサーバーの構成値が記入されている
- catコマンドで確認
- nameserver10.0.0.2がサブネット上に構成されたDNSサーバー

```
[root@ip-10-0-0-193 ~]# cat /etc/resolv.conf
; generated by /usr/sbin/dhclient-script
search ap-northeast-1.compute.internal
options timeout:2 attempts:5
nameserver 10.0.0.2
```

- digコマンドでドメイン名からipアドレスを調べることが可能

```
[ec2-user@ip-10-0-0-193 ~]$ dig kbmk.wiki.fc2.com

; <<>> DiG 9.11.4-P2-RedHat-9.11.4-9.P2.amzn2.0.2 <<>> kbmk.wiki.fc2.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32856
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;kbmk.wiki.fc2.com.             IN      A

;; ANSWER SECTION:
kbmk.wiki.fc2.com.      60      IN      A       104.244.99.14

;; Query time: 4 msec
;; SERVER: 10.0.0.2#53(10.0.0.2)
;; WHEN: Sat Apr 18 18:06:05 JST 2020
;; MSG SIZE  rcvd: 62
```

2.3 EC2インスタンスからインターネットに到達可能か確認

- pingコマンドで到達を確認
- curlコマンドでHTTP通信ができるか確認

```
[ec2-user@ip-10-0-0-193 ~]$ ping kbmk.wiki.fc2.com
PING kbmk.wiki.fc2.com (104.244.99.14) 56(84) bytes of data.
64 bytes from 104.244.99.14 (104.244.99.14): icmp_seq=1 ttl=39 time=120 ms
64 bytes from 104.244.99.14 (104.244.99.14): icmp_seq=2 ttl=39 time=117 ms
64 bytes from 104.244.99.14 (104.244.99.14): icmp_seq=3 ttl=39 time=117 ms
64 bytes from 104.244.99.14 (104.244.99.14): icmp_seq=4 ttl=39 time=117 ms
^C
--- kbmk.wiki.fc2.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 117.109/117.942/120.084/1.291 ms
```

```
[root@ip-10-0-0-193 ~]# curl google.com
<HTML><HEAD><meta http-equiv="content-type" content="text/html;charset=utf-8">
<TITLE>301 Moved</TITLE></HEAD><BODY>
<H1>301 Moved</H1>
The document has moved
<A HREF="http://www.google.com/">here</A>.
</BODY></HTML>
[root@ip-10-0-0-193 ~]#
```

2.4 メタデータを配信するHTTPサーバー

- curlでメタデータの取得を試す 169.254.169.254/latest/meta-data/
- メタデータの中の「public-ipv4」を取得

```
[root@ip-10-0-0-193 ~]# curl 169.254.169.254/latest/meta-data/public-ipv4
54.249.38.0[root@ip-10-0-0-193 ~]#
```

```
[root@ip-10-0-0-193 ~]# curl 169.254.169.254/latest/meta-data/
ami-id
ami-launch-index
ami-manifest-path
block-device-mapping/
events/
hibernation/
hostname
identity-credentials/
instance-action
instance-id
instance-type
local-hostname
local-ipv4
mac
metrics/
network/
placement/
profile
public-hostname
public-ipv4
public-keys/
reservation-id
security-groups
```

3. リンク集

SSH(Wikipedia)
https://ja.wikipedia.org/wiki/Secure_Shell

秘密鍵と公開鍵やその利用例(SSH暗号化通信)
https://milestone-of-se.nesuke.com/sv-advanced/digicert/public-private-key/

SSHの公開鍵認証
https://qiita.com/angel_p_57/items/2e3f3f8661de32a0d432

Amazon EC2のキーペア
https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/ec2-key-pairs.html

おまけ Puttyでの接続方法

https://docs.aws.amazon.com/ja_jp/AWSEC2/latest/UserGuide/putty.html

- puttygenを起動し「ロード」からプライベートキーを読み込む
- キータイプがRSAか確認し、「Save private key (プライベートキーの保存)」
- puttyを起動し、セッションに「user_name@public_ip」の形式で入力
- Port22番、接続タイプSSHを確認
- 「Connection」、「SSH」 の順に展開し、「Auth」 を選択
- browseから先ほど作成したファイルを読み込む
- アラートが出るので「はい」を押すと接続完了