

# Microsoft Azure を利用した、自作トークン作成マニュアル

## ▶ 目次

### ≫ ステップ 1（トークン作成に必要な基本情報を控える）

1. 自分の IP アドレスを調べる。
2. Metamask をインストールする。
3. Metamask からニーモニック 12 単語を取得する。
4. ニーモニックを利用してイーサリアムアドレスを取得する。
5. イーサリアムノード管理パスワードを作成する。
6. トリュッフル インスタンス (Truffle instance) を作成する。

### ≫ ステップ 2（Microsoft Azure アカウントを準備する）

1. Microsoft Live アカウントを取得する。
2. 本人認証を行う。
3. クレジットカードを登録する（入力が必要ですが、継続利用しない限り課金されません。）

### ≫ ステップ 3（Ethereum Proof-of-Authority Consortium をインストールする）

1. ethereum と検索して MARKETPLACE から Ethereum Proof-of-Authority Consortium（あるいは Ethereum on Azure）を選びます。
2. 基本情報や地域などを入力します。
3. 仮想 PC のタイプを選択します。
4. イーサリアムノードを生成します。

### ≫ ステップ 4（仮想マシンから自作トークンを発行する）

1. ポータルから truffle を検索して作成を行います。
2. 仮想マシンの基本情報を入力します。
3. 仮想マシンを起動して自作トークン情報（トークン名、発行量、サーバーホスト等）を入力します。
4. 自作トークンの送信テストを行います。

〈このマニュアルの執筆者〉

高 永宇 (コウ・ヨンウ)

ロケットスタッフ株式会社 CTO。ACAX 株式会社 CTO。韓国出身。大学時代からプログラミングを学び、その才能を認められ特例で徴兵制度を免除される。その後、韓国のコンシューマーゲーム開発のパイオニア Sonnori 社に入社し、日本向けのコンシューマーゲームに携わる。2010 年には Fuzz Tech 社を設立し、アプリケーション開発や開発ツールなどを提供する。2014 年にはロケットスタッフ株式会社の韓国法人である JungleSky 株式会社を、2018 年にはブロックチェーン広告配信事業を行う ACAX 株式会社を共同創業。

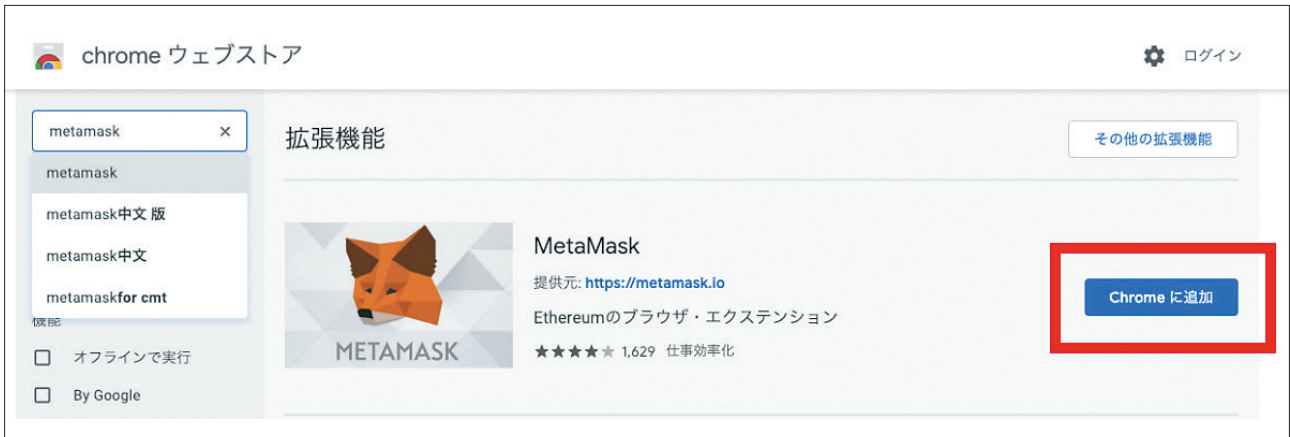
## ▶ ステップ1 (トークン作成に必要な基本情報まとめ)

1. 自分の IP アドレスを調べる。

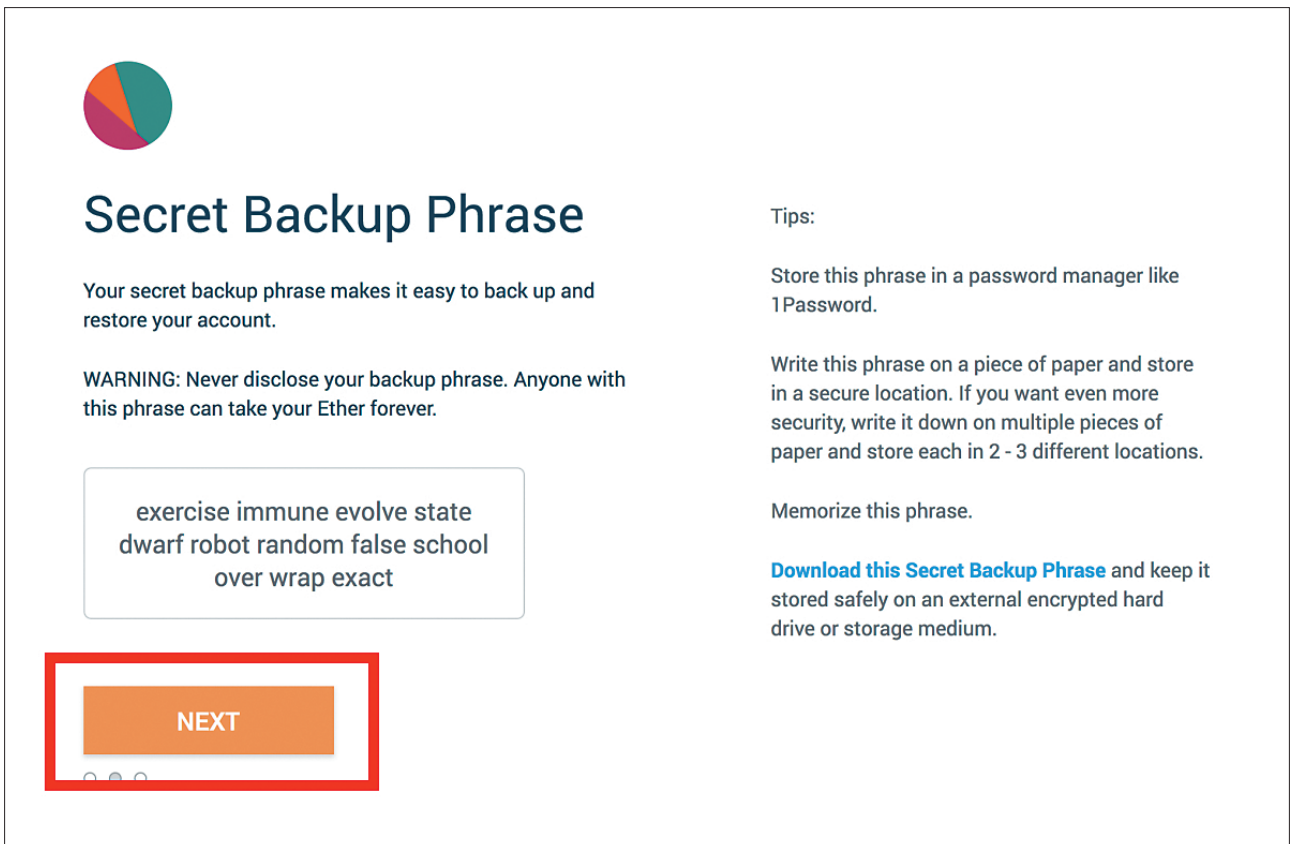
<http://www.myipaddress.com/show-my-ip-address/>

メモ：

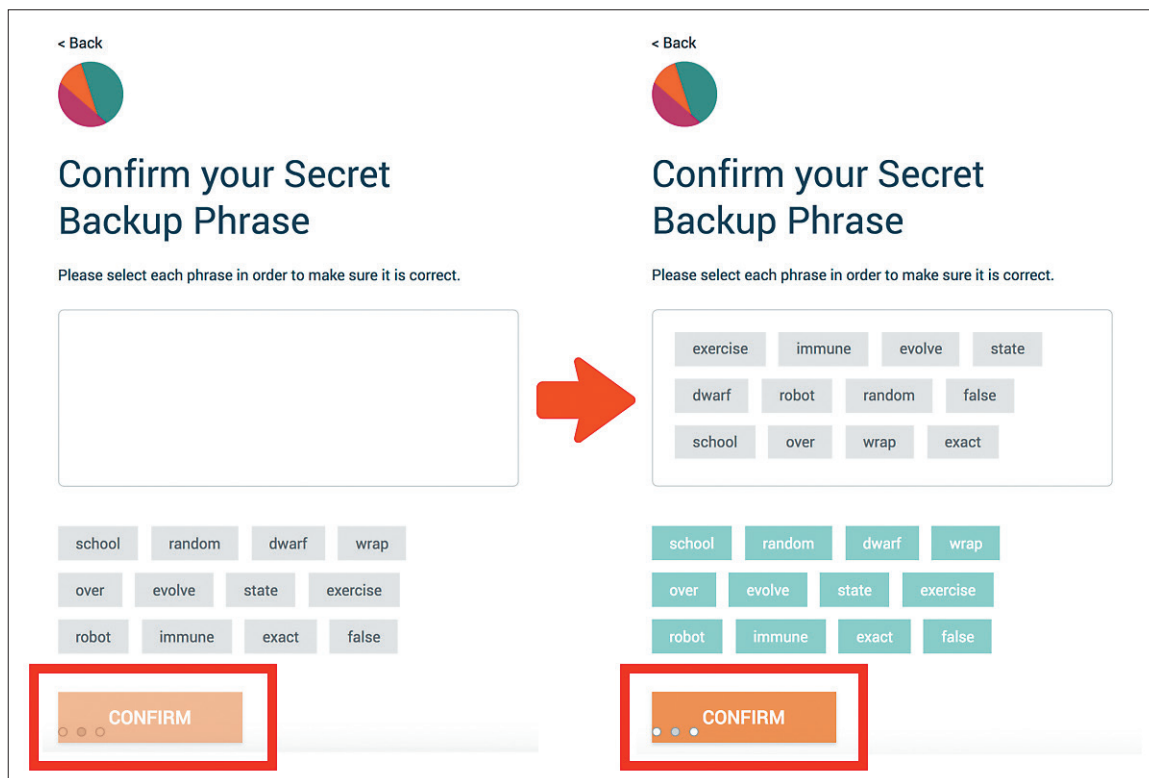
2. Chrome を起動して、Metamask (<https://metamask.io>) をインストールする。



3. パスワードなどを設定して利用規約に同意すると、以下のようにニーモニックバックアップ画面を見ることができます。12 単語をコピーします。

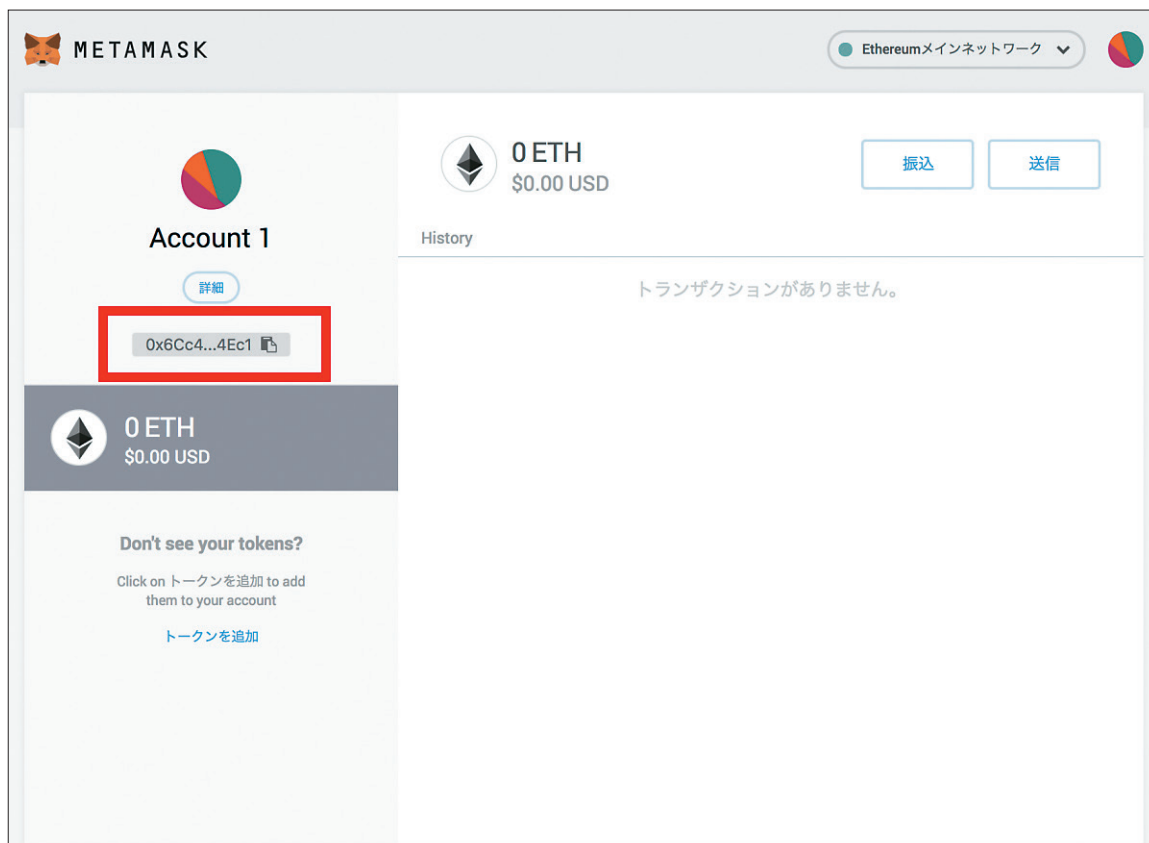


4. 下に表示される単語を順番に選択して確認 (CONFIRM) ボタンを選択します。



5. 自分のイーサリアムアドレスをコピーして控えます。

メモ：



6. イーサリアムノード管理パスワードを作成する。

メモ：

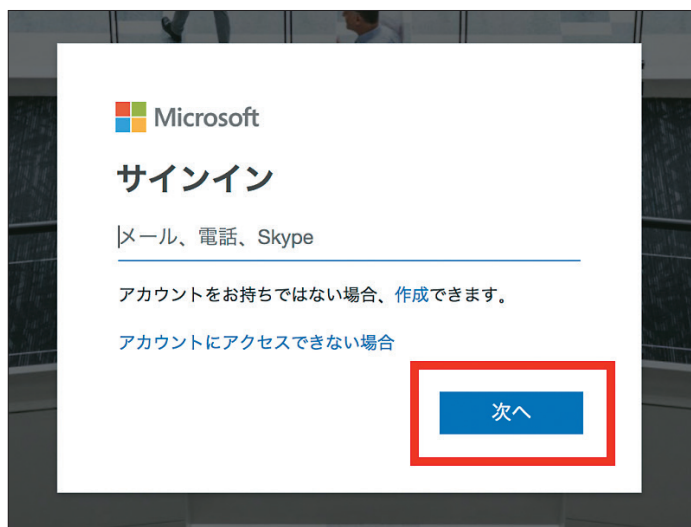
7. トリュッフル インスタンス (Truffle instance) パスワードを作成する。

メモ：

## ▶ ステップ2 (Microsoft Azure アカウントを準備する)

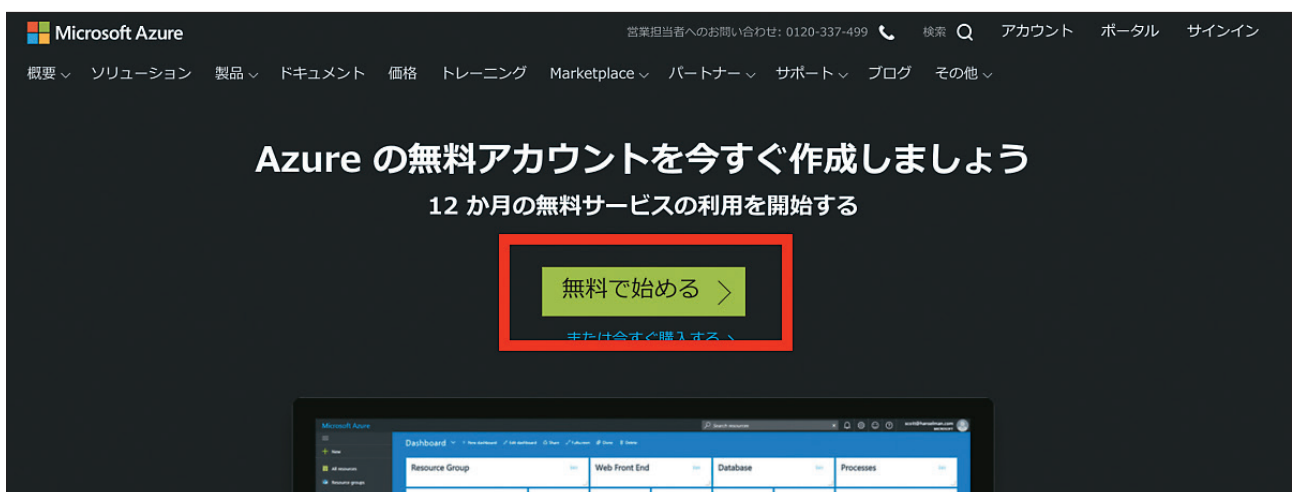
1. Microsoft Live アカウントを取得する。

<https://account.microsoft.com/account?lang=ja-JP>



2. Microsoft Azure へアクセスし「無料で始める」を選択して、Microsoft Live アカウントで「サインイン」します。

<https://azure.microsoft.com/>



3. 基本情報を入力します。

Microsoft Azure Sign out

## Azure free account sign up

Start with a ¥ 22,500 credit for 30 days, and keep going for free

### 1 About you ^

Country/Region ⓘ  
Japan

First name

Last name

Email address ⓘ

Phone  
Example: 090 XXXX XXXX

First name pronunciation

Last name pronunciation

By proceeding you acknowledge the [privacy statement](#) and [subscription agreement](#)

**Next**

4. クレジットカードを登録する（入力が必要ですが、継続利用しない限り課金されません。）

### 2 Identity verification by card ^

We ask for your credit card number to verify your identity and to keep out spam and bots.  
**You won't be charged unless you upgrade.**

Card number

Expiration date CVV ⓘ  
Month  Year

Name on card

Postal Code

State

City

Address line 1

Address line 2

**Next**

5. 利用規約を確認してチェックします。

### 3 Agreement ^

I agree to the [subscription agreement](#), [offer details](#), and [privacy statement](#)

I would like information, tips, and offers from Microsoft or selected partners about Azure, including Azure Newsletter, Pricing updates, and other Microsoft products and services.

**Sign up**

6. ポータルに移動します。

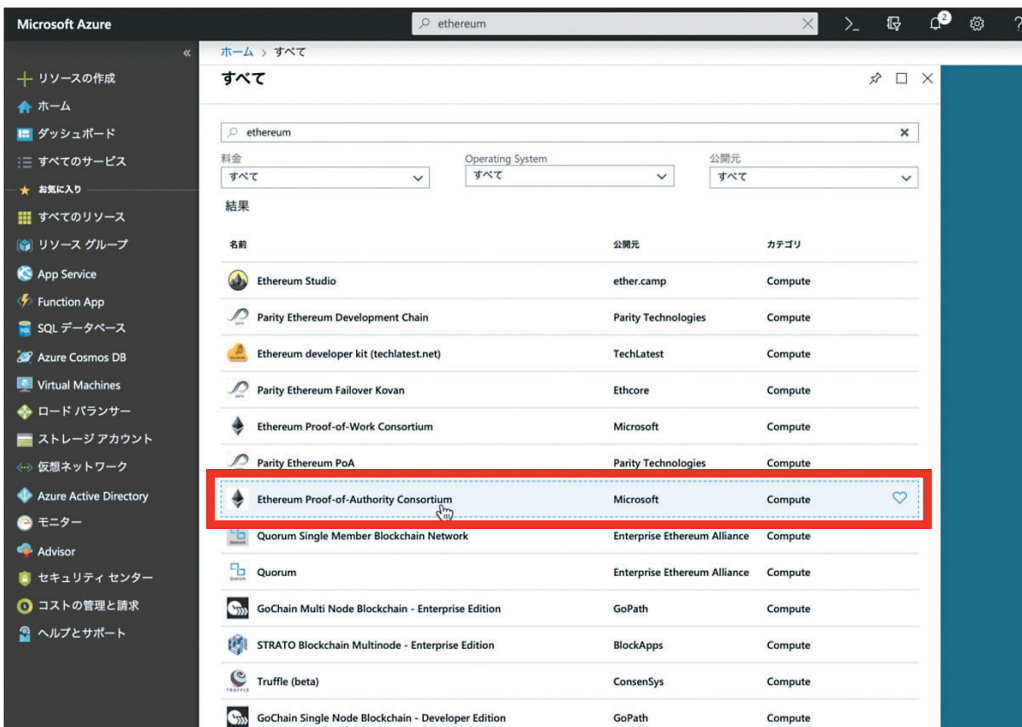
The screenshot shows the Microsoft Azure website's landing page. At the top, there is a navigation bar with the Microsoft Azure logo and various links like '概要', 'ソリューション', '製品', 'ドキュメント', '価格', 'トレーニング', 'Marketplace', 'パートナー', 'サポート', 'ブログ', and 'その他'. Below the navigation bar, a blue banner contains the text 'Azureのご利用を開始できます' and a green button labeled 'ポータルに移動' with a right-pointing arrow. This button is highlighted with a red rectangular box. Below the banner, there are two main content areas. The left area is titled 'デモに参加して Azure の実際の動作を確認する' and lists several topics: '仮想マシンの構築', 'Web アプリの作成', 'SQL データベースをデプロイします。', and 'DevOps ダッシュボードのカスタマイズ'. Below this list is an illustration of a laptop with a play button on the screen and speech bubbles. The right area is titled 'ライブ デモに登録するか、オンデマンドで見る' and contains several sections: '時刻を選択するか、オンデマンドで見る' with radio buttons for 'オンデマンドで見る - 日本語', '英語版を見る' with radio buttons for 'オンデマンド - 英語', '2019/1/11 2:00 GMT+9 (タイムゾーンを変更)', '2019/1/18 2:00 GMT+9', and '2019/1/25 2:00 GMT+9'. Below this is a section 'サインインして登録を完了する' with a Microsoft logo and a text prompt: 'または、デモに登録するには、お客様の情報をご入力ください。' followed by a text input field labeled '姓'. At the bottom right, there is a green button labeled 'Microsoft にメッセージ'.

## ▶ ステップ3 (Ethereum Proof-of-Authority Consortium をインストールする)

1. ethereum と検索して MARKETPLACE から Ethereum Proof-of-Authority Consortium (あるいは Ethereum on Azure) を検索します



2. Ethereum Proof-of-Authority Consortium (あるいは Ethereum on Azure)' を選択します。



### 3. 「作成」を選択します。

**Ethereum Proof-of-Authority Consortium** 🔗 □ ×

Microsoft

We are excited to support Blockchain on Microsoft Azure with a set of solution templates that deploy and configure your choice of blockchain network with minimal Azure and blockchain knowledge.

With a handful of user inputs and a simple single-click deployment through the Azure portal, you can provision a fully configured blockchain network topology in minutes, using Microsoft Azure compute, networking, and storage services across the globe. Rather than spending hours building out and configuring the infrastructure, we have automated these time-consuming pieces to allow you to focus on building out your scenarios and applications. You are only charged for the underlying infrastructure resources consumed, such as compute, storage, and networking. There are no incremental charges for the solution itself.

This solution deploys a flexible Ethereum Proof-of-Authority (PoA) network, consisting of a set of highly-available Parity nodes with which an application or user can interact to submit transactions. It can take between 5-45 minutes to provision, depending on the size and complexity of the network. Once complete, through the Governance DApp, you can vote to add additional authorities on the network. This deployment can be configured to connect to an existing network.

♡ 後で使用するために保存

公開元	Microsoft
役に立つリンク	<a href="#">Microsoft Azure Blockchain</a> <a href="#">Pricing Details</a> <a href="#">Support Forum</a> <a href="#">Technical Walkthroughs</a>
サポート	<a href="https://social.msdn.microsoft.com/Forums/en-US/home?forum=azureblockchain">https://social.msdn.microsoft.com/Forums/en-US/home?forum=azureblockchain</a>

**作成**

### 4. 基本情報を入力します。ステップ1で作成した「イーサリアムノード管理パスワード」を記入します。

**Ethereum Proof-of-Authorit...** □ ×

**基本**

\* Create a new network or join existing network? 🔔  
**Create new** **Join existing**

Email Address 🔔  
kou@rocketstaff.com ✓

\* VM user name 🔔  
tokeneconomy ✓

\* Authentication type 🔔  
**パスワード** **SSH 公開キー**

\* Password 🔔  
..... ✓

\* Confirm password  
..... ✓

サブスクリプション  
Free Trial

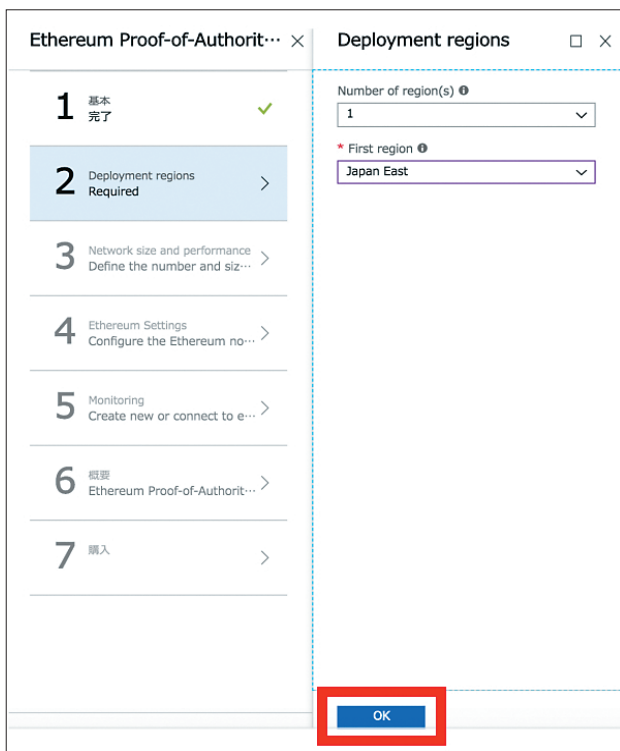
\* リソース グループ 🔔  
(新規) tokeneconomygroup ✓  
新規作成

\* 場所  
東日本

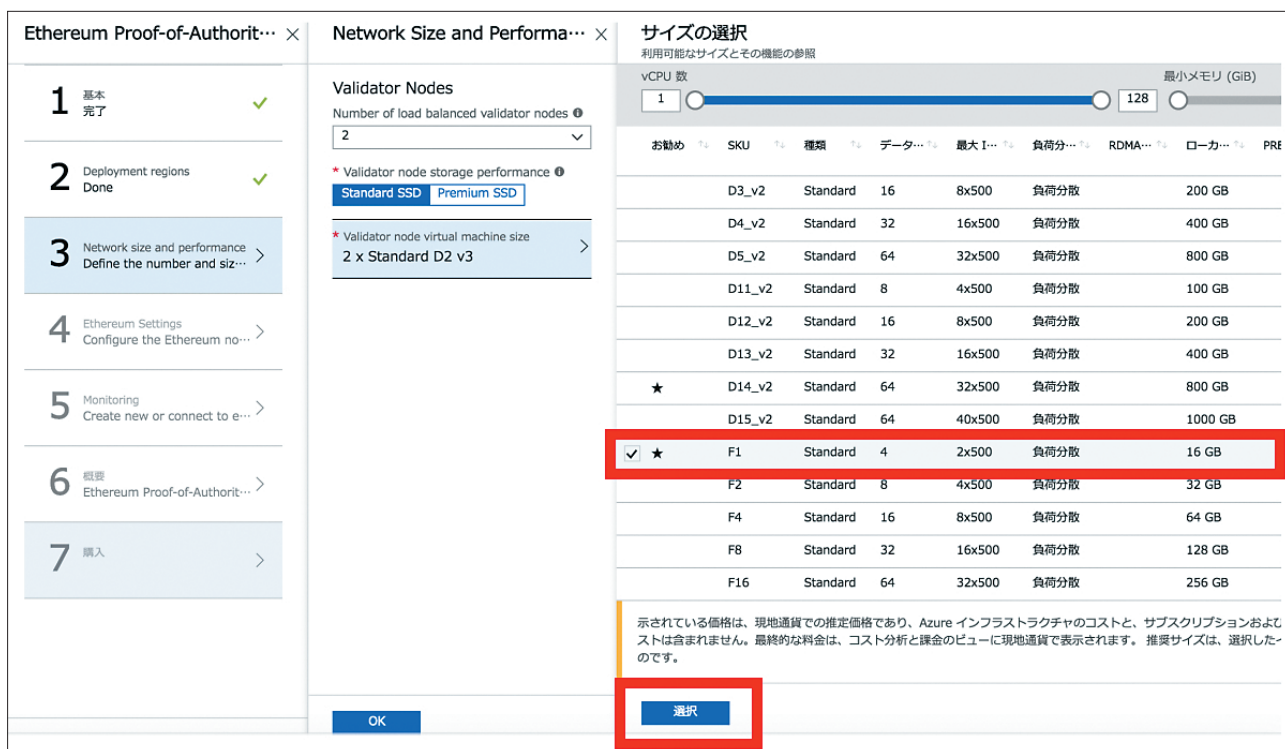
**OK**



5. 地域情報を選択します。



6. Standard F1 を選択します。



7. Consortium Member Id と Network Id を以下のように記入してステップ1で作成した「自分のイーサリアムアドレス」をコピー・アンド・ペーストします。

Ethereum Proof-of-Authorit... × Ethereum Settings □ ×

1 基本 完了 ✓

2 Deployment regions Done ✓

3 Network size and performance Done ✓

4 Ethereum Settings Configure the Ethereum no... >

5 Monitoring Create new or connect to e... >

6 概要 Ethereum Proof-of-Authorit... >

7 購入 >

\* Consortium Member Id ⓘ  
1 ✓

\* Network ID ⓘ  
10101010

\* Admin Ethereum Address ⓘ  
0x6Cc455C122dA22ef3206C0c3b7a1... ✓

\* Advanced Options ⓘ  
Enable Disable

OK

8. Monitoring は「Disable」を選択して「OK」を選択します。

Ethereum Proof-of-Authorit... × Monitoring □ ×

1 基本 完了 ✓

2 Deployment regions Done ✓

3 Network size and performance Done ✓

4 Ethereum Settings Done ✓

5 Monitoring Create new or connect to e... >

6 概要 Ethereum Proof-of-Authorit... >

7 購入 >

\* Monit...  
Enable Disable

OK

9. リソースプロバイダーが登録されています。(最大 10 分程度時間がかかります。)

10. 利用規約を同意し作成するとイーサリアムノードが生成されます。

※メールでも告知されます。

11. 再びリソースグループにアクセスして「token-economy」を選択してリストの一番下にある「microsoft-azure-blockchain.azure-blockchain-ether-\*\*\*\*\*」を選択します。

The screenshot shows the Microsoft Azure portal interface. The breadcrumb navigation at the top reads: ホーム > リソースグループ > token-economy - デプロイ. The left-hand navigation pane has the 'デプロイ' (Deploy) option highlighted with a red box. The main content area displays a table of deployment tasks. The task 'microsoft-azure-blockchain.azure' is highlighted with a red box at the bottom of the list.

デプロイ名	状態	最終更新日時	時間
vm-dep-reg1-config	成功	2019/1/11 2:13:51	6分 22秒
vm-dep-reg1-0-keyVaultAccess	成功	2019/1/11 2:07:15	17秒
vm-dep-reg1-1-keyVaultAccess	成功	2019/1/11 2:07:07	14秒
vm-dep-reg1-publicIP	成功	2019/1/11 2:04:27	19秒
vnet-gateway-deploy	成功	2019/1/11 2:04:07	6秒
vm-dep-reg1	成功	2019/1/11 2:07:21	3分 21秒
network-resources-lb0	成功	2019/1/11 2:03:44	13秒
network-resources-deploy	成功	2019/1/11 2:03:51	47秒
keyVaultDeploy	成功	2019/1/11 2:03:35	31秒
deployOMS	成功	2019/1/11 2:03:08	5秒
storageDeploy	成功	2019/1/11 2:03:45	43秒
pid-a5842b51-d2c6-486a-82e7-	成功	2019/1/11 2:03:06	4秒
<b>microsoft-azure-blockchain.azure</b>	<b>成功</b>	<b>2019/1/11 2:14:06</b>	<b>11分 10秒</b>

12. 「出力」を選択します。

The screenshot shows the Microsoft Azure portal interface for the deployment details of 'microsoft-azure-blockchain.azure'. The breadcrumb navigation at the top reads: ホーム > リソースグループ > token-economy - デプロイ > microsoft-azure-blockchain.azure-blockchain-ether-20190111020254 - 概要. The left-hand navigation pane has the '出力' (Output) option highlighted with a red box. The main content area displays a table of deployment tasks, with the task 'microsoft-azure-blockchain.azure' highlighted in blue. The right-hand pane shows the deployment summary, including the deployment name, start time, duration, and related ID.

最終更新日時	時間	関連イベント
2019/1/11 2:13:51	6分 22秒	関連イベント
2019/1/11 2:07:15	17秒	関連イベント
2019/1/11 2:07:07	14秒	関連イベント
2019/1/11 2:04:27	19秒	関連イベント
2019/1/11 2:04:07	6秒	関連イベント
2019/1/11 2:07:21	3分 21秒	関連イベント
2019/1/11 2:03:44	13秒	関連イベント
2019/1/11 2:03:51	47秒	関連イベント
2019/1/11 2:03:35	31秒	関連イベント
2019/1/11 2:03:08	5秒	関連イベント
2019/1/11 2:03:45	43秒	関連イベント
2019/1/11 2:03:06	4秒	関連イベント
2019/1/11 2:14:06	11分 10秒	関連イベント

### 13. 「ETHEREUM\_RPC\_ENDPOINT」のアドレスを控えます。(トークン生成の際に利用します。)

The screenshot shows the Azure portal interface for a deployment. The breadcrumb navigation is: ホーム > リソースグループ > token-economy - デプロイ > microsoft-azure-blockchain.azure-blockchain-ether-20190111020254 - 出力. The main content area displays a list of output variables for the deployment 'microsoft-azure-blockchain.azure-blockchain-ether-20190111020254 - 出力'. The 'ETHEREUM\_RPC\_ENDPOINT' variable is highlighted with a red box, showing the value 'http://ethdqlqbc-dns-reg1.japaneast.cloudapp.azure.com:8540'. Other variables include ADMIN\_SITE, OMS\_PORTAL\_URL, ETHEREUM\_WEBSOCKET\_ENDPOINT, SSH\_TO\_FIRST\_VL\_NODE\_REGION1, CONSORTIUM\_DATA\_URL, CONSORTIUM\_MEMBER\_GATEWAY\_ID\_REGION1, and REGIONONEVNET.

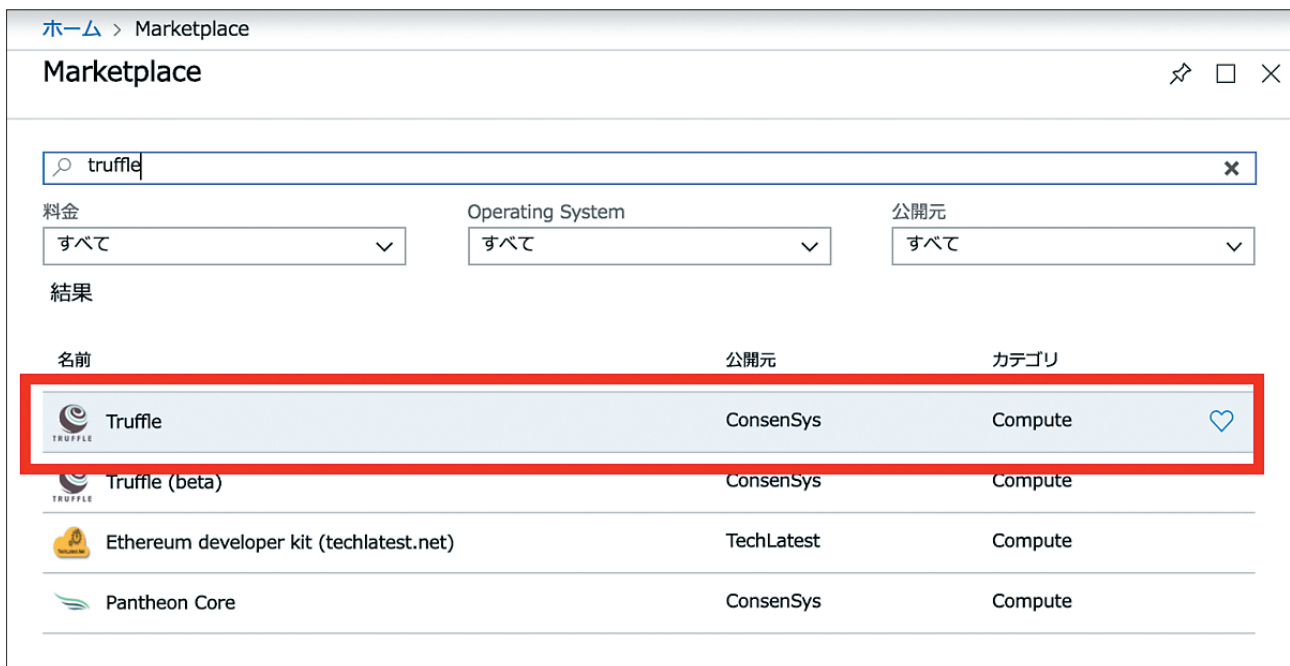
Variable Name	Value
ADMIN_SITE	http://ethdqlqbc-dns-reg1.japaneast.cloudapp.azure.com
OMS_PORTAL_URL	OMS Portal not deployed
<b>ETHEREUM_RPC_ENDPOINT</b>	<b>http://ethdqlqbc-dns-reg1.japaneast.cloudapp.azure.com:8540</b>
ETHEREUM_WEBSOCKET_ENDPOINT	ws://ethdqlqbc-dns-reg1-0.japaneast.cloudapp.azure.com:8547
SSH_TO_FIRST_VL_NODE_REGION1	ssh -p 4000 tokeneconomy@ethdqlqbc-dns-reg1.japaneast.cloudapp.azure.com
CONSORTIUM_DATA_URL	http://ethdqlqbc-dns-reg1.japaneast.cloudapp.azure.com:3001
CONSORTIUM_MEMBER_GATEWAY_ID_REGION1	NA
REGIONONEVNET	ethdqlqbc-vnet-reg1

## ▶ ステップ4（仮想マシンから自作トークンを発行します。）

1. ポータルから truffle を検索して作成を行います。



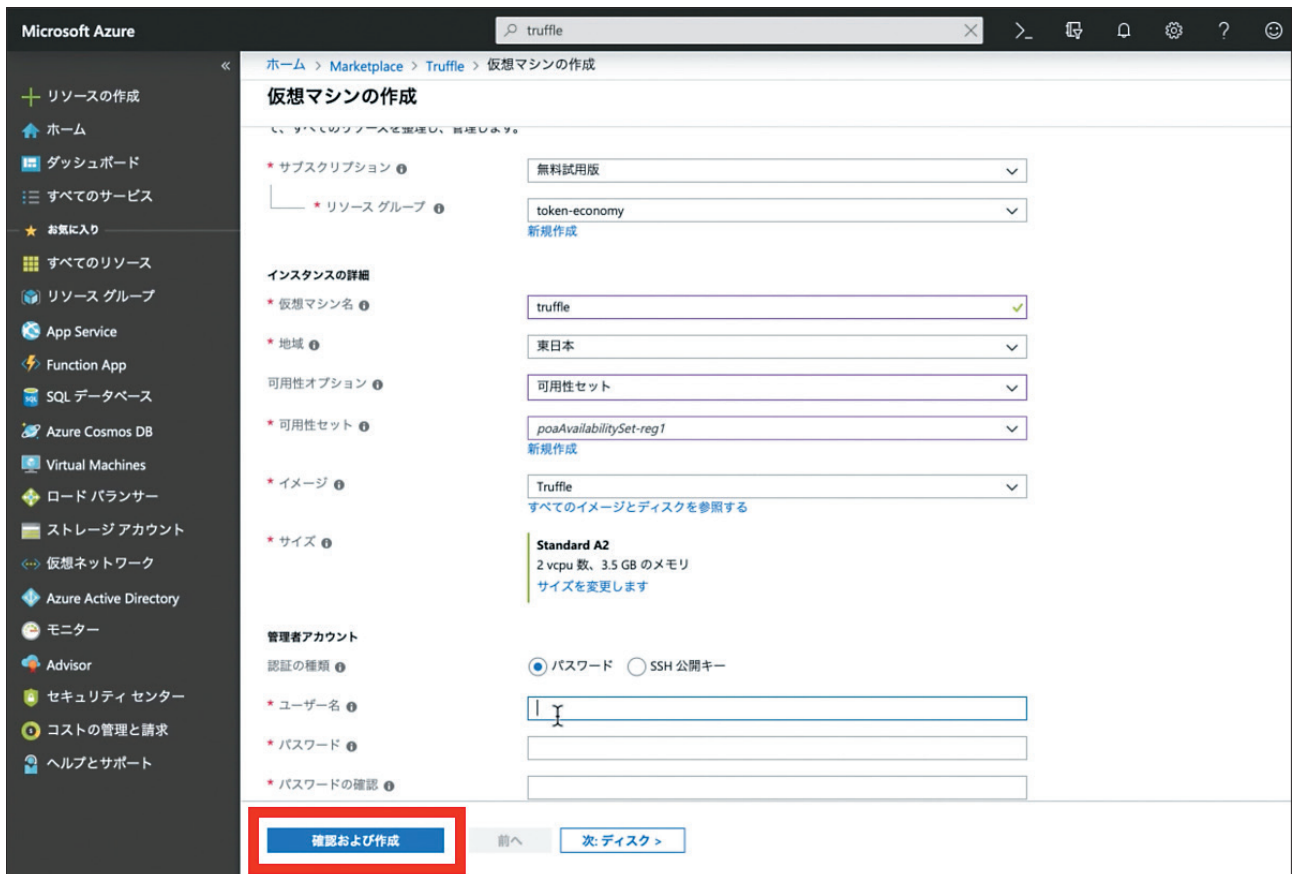
2. truffle を選択します。



3. 「作成」を選択します。



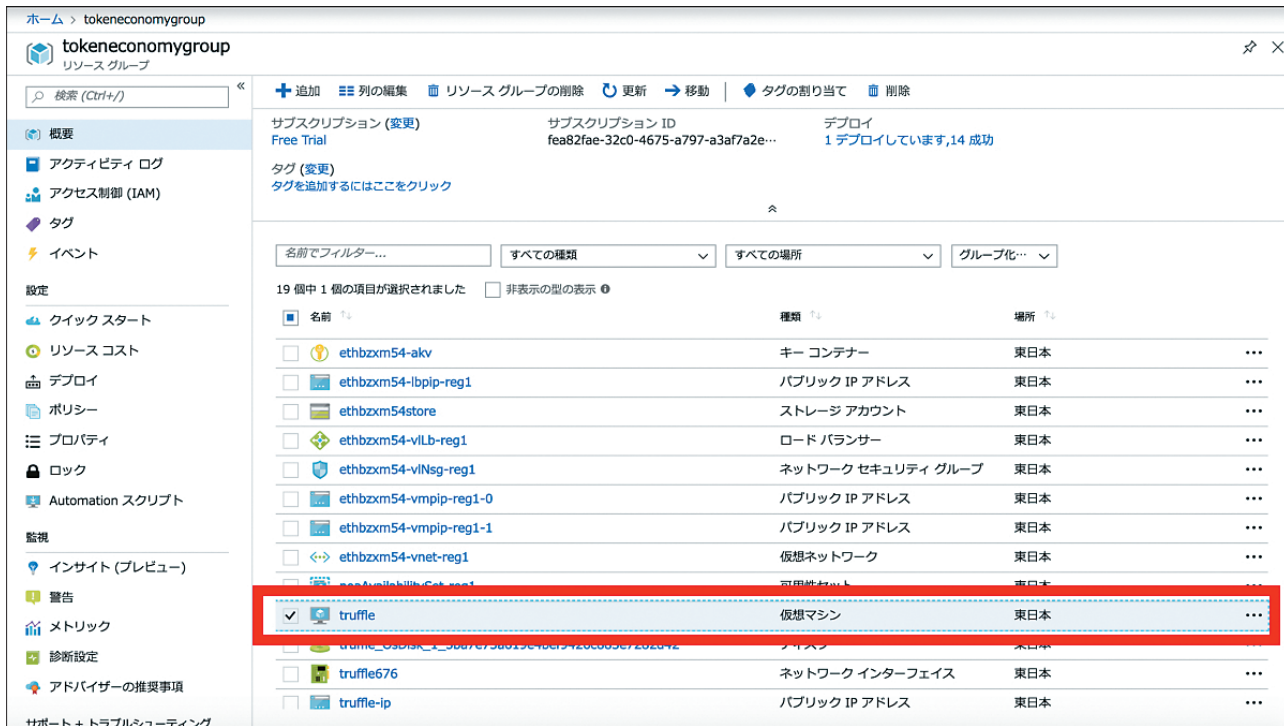
4. 以下のように記入します。ユーザー名は「truffle」と記入し、パスワードはステップ1で作成した「トリュッフル インスタンス (Truffle instance) パスワード」を入力して「確認および作成」ボタンを選択します。(およそ5分ほど確認時間が発生します。)



5. 確認が終わったら「作成」ボタンを選択します。

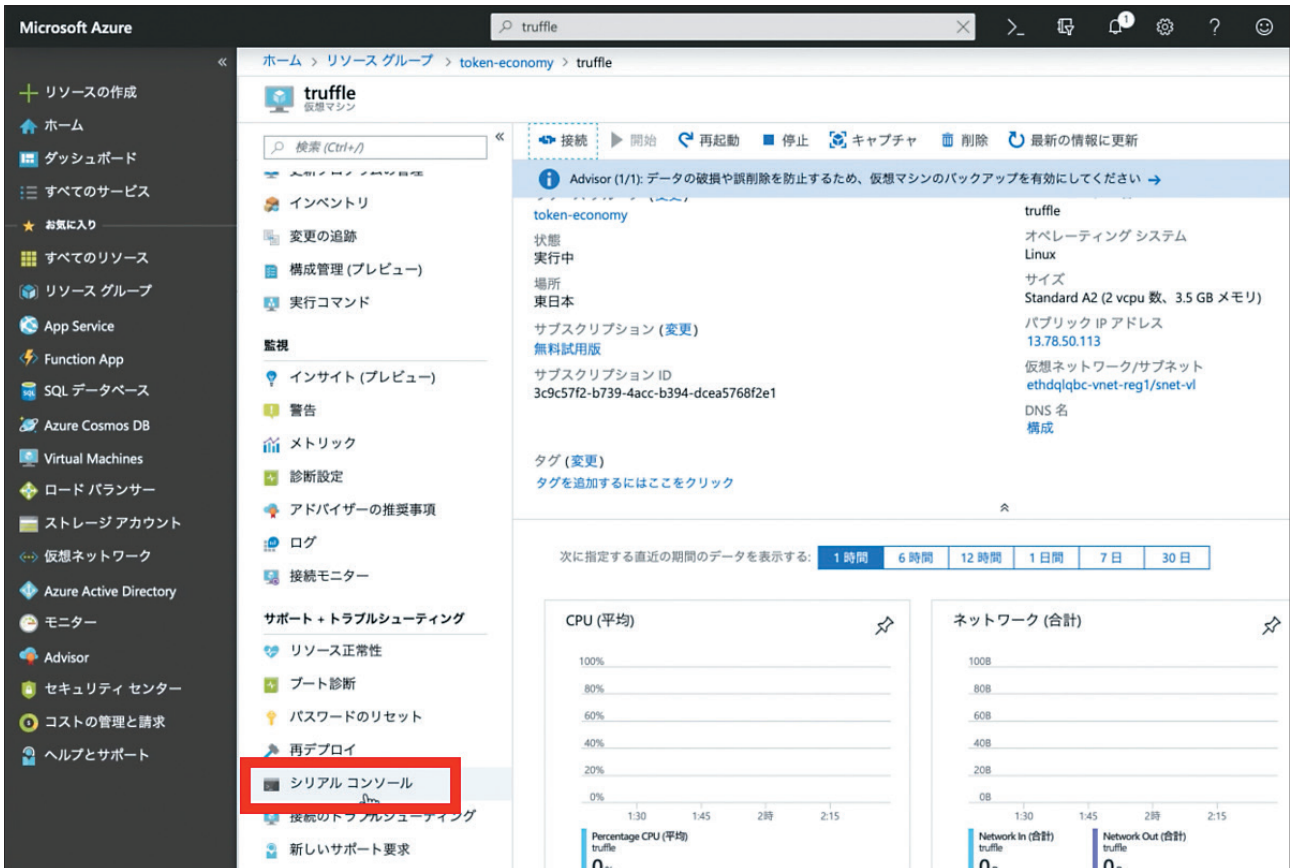


6. Truffle マシンがリストに表示されます。





7. 「シリアルコンソール」を選択して仮想マシンにアクセスします。



8. 以下のような画面が表示されたら「エンターキー」を一度押した後、ID とパスワードを入力して再度「エンターキー」を押します。

```
Ubuntu 16.04.5 LTS truffle ttyS0

truffle login: truffle
Password:
Last login: Thu Jan 10 17:25:03 UTC 2019 on ttyS0
Welcome to Ubuntu 16.04.5 LTS (GNU/Linux 4.15.0-1036-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

13 packages can be updated.
0 updates are security updates.

New release '18.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

truffle@truffle:~$
```

9. 以下のコマンドを入力します。

```
truffle@truffle:~$ git clone https://github.com/fuzzface/tokeneconomy.git
Cloning into 'tokeneconomy'...
Username for 'https://github.com': fuzzface
Password for 'https://fuzzface@github.com':
remote: Enumerating objects: 20, done.
remote: Counting objects: 100% (20/20), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 20 (delta 3), reused 15 (delta 2), pack-reused 0
Unpacking objects: 100% (20/20), done.
Checking connectivity... done.
truffle@truffle:~$ cd tokeneconomy/
truffle@truffle:~/tokeneconomy$ npm install

> websocket@1.0.28 install /home/truffle/tokeneconomy/node_modules/websocket
> (node-gyp rebuild 2> builderror.log) || (exit 0)

npm WARN token@1.0.0 No description
npm WARN token@1.0.0 No repository field.

added 18 packages from 24 contributors and audited 18 packages in 10.408s
found 0 vulnerabilities

truffle@truffle:~/tokeneconomy$
```

10. 以下のコマンドを入力します。

- ・小文字のアルファベットの単語 12 個を入力してください：ステップ 1 で作成した「ニーモニック 12 単語」を入力します。
- ・server host：ステップ 3 の No.13 で控えた「ETHEREUM\_RPC\_ENDPOINT」を入力します。
- ・network id：101010
- ・Token Name：アルファベットで自作トークン名を記入します。
- ・Token Symbol：アルファベット 3 文字で記入します。(例：ビットコイン：BTC)
- ・Total Supply：総発行数を決めます

```
truffle@truffle:~/tokeneconomy$ npm run setup

> token@1.0.0 setup /home/truffle/tokeneconomy
> node setup.js

小文字のアルファベットの単語 12 個を入力してください：
server host: http://:8540
network id: 10101010
Token Name: My Token
Token Symbol: TKN
Total Supply: 10000
done
truffle@truffle:~/tokeneconomy$
```

11. 以下のコマンドを入力して「エンターキー」を押します。

```
truffle@truffle:~/tokeneconomy$ npm run clean
> token@1.0.0 clean /home/truffle/tokeneconomy
> rm -rf ~/.config && sudo truffle compile && sudo chown -R $USER:$(id -gn $USER) ~/.config && sudo chown -R $USER:$(id -gn $USER) ./

2019/01/10 17:26:59.807373 INFO ExtHandler After updating cgroup telemetry, tracking [Agents+Extensions] [walinuxagent.service]
Compiling ./contracts/Migrations.sol...
Compiling ./contracts/Token.sol...
Compiling openzeppelin-solidity/contracts/access/Roles.sol...
Compiling openzeppelin-solidity/contracts/access/roles/MinterRole.sol...
Compiling openzeppelin-solidity/contracts/math/SafeMath.sol...
Compiling openzeppelin-solidity/contracts/token/ERC20/ERC20.sol...
Compiling openzeppelin-solidity/contracts/token/ERC20/ERC20Burnable.sol...
Compiling openzeppelin-solidity/contracts/token/ERC20/ERC20Detailed.sol...
Compiling openzeppelin-solidity/contracts/token/ERC20/ERC20Mintable.sol...
Compiling openzeppelin-solidity/contracts/token/ERC20/IERC20.sol...
Writing artifacts to ./build/contracts

truffle@truffle:~/tokeneconomy$ truffle migrate --network azure
```

12. 正常に完了された場合、以下のように表示されます。赤い部分が「スマートコントラクトアドレス」になりますので控えてください。そして、「truffle console --network azure」コマンドを利用して実際にトークンを利用してみます。

```
Deploying 'Token'
-----
> transaction hash: 0xee308100197e5653b18472e11e9ac042a9095f88ece006a0af32bd7972b70ea3
> Blocks: 0
> contract address: 0x45120b65dEB4a26781757036B10ce6BE10Af0d24
> account: 0x3Cacf5e38fE07300E4DEad12a22eCbfd32099CE
> balance: 0
> gas used: 2080399
> gas price: 0 gwei
> value sent: 0 ETH
> total cost: 0 ETH

> Saving migration to chain.
> Saving artifacts
-----
> Total cost: 0 ETH

Summary
=====
> Total deployments: 2
> Final cost: 0 ETH

truffle@truffle:~/tokeneconomy$ truffle console --network azure
```

13. 以下のコマンドでテストを行います。

1. トークンのインスタンスを取得します。  
let tokenInstance= await Token.at('[[ トークンのアドレス ]]')
2. トークンを特定のアドレスに送信します。  
tokenInstance.transfer('[[ 送信受信アドレス ]]', web3.utils.toWei('[[ 転送トークンの数 ]]', 'ether'))
3. 特定のアドレスにトークンを生成します。この際にトークンの合計発行量が増えます。  
tokenInstance.mint('[[ トークンをもらうアドレス ]]', web3.utils.toWei('[[ 発行するトークンの数 ]]', 'ether'))

4. トークンの合計発行量を確認します。

```
tokenInstance.totalSupply().then(result=>{console.log(web3.utils.fromWei(result,'ether').toString())})
```

5. 特定のアドレスのトークン数を確認します。

```
tokenInstance.balanceOf('[ アドレス ]').then(result=>{console.log(web3.utils.fromWei(result,'ether').toString())})
```

以上