AWS EC2とCertificate Managerで実現する メンテナンスフリーSSL証明書運用

2024年8月30日

株式会社グローバルネットコア

インフラサービス部　サービス運用課　渡辺 亮

GLOBAL NET CORE

https://www.global-netcore.jp/

自己紹介

■業務：サーバ・ネットワークの運用

■趣味：ゲーム

わりとなんでもやりますが、

最近はFGOとかゼンゼロとか

■保有資格

AWS：SAA、SCS、次はMLS、現在はG検定を学習中

IPA：応用情報、ネスペ、安全確保支援士

SSL証明書の運用、大変じゃないですか？

◆現在の証明書の有効期間は1年＋1ヶ月

◆証明書の更新作業フロー

- ◆鍵ペアの作成
- ◆CSRの作成・認証局への送信
- ◆発行料金の支払い
- ◆証明書の設定・更新

◆弊社でも数百の証明書を管理しています・・・

AWS Certificate Managerの活用

◆AWSサービスで利用する証明書を無償で発行・自動更新が可能

- ACMの証明書ってELBやCloudfrontで使うものでしょ？
  EC2インスタンスには直接適用出来なかったよね？
- 間接的に適用する方法があります！
  - AWS Nitro Enclavesの活用

目次

AWS Nitro Enclavesとは

AWS Certificate Manager for Nitro Enclaves

https://docs.aws.amazon.com/enclaves/latest/user/nitro-enclave-refapp.html

Nitro Enclaves アプリケーション: Nitro Enclaves 用 AWS 証明書マネージャー

PDF | 翻訳

AWS Certificate Manager (ACM) for Nitro Enclaves を使用すると、AWS Nitro Enclaves を使用した Amazon EC2 インスタンスで実行されているウェブアプリケーションおよびウェブサーバーで、パブリックおよびプライベートの SSL/TLS 証明書を使用できます。SSL/TLS 証明書は、ネットワーク通信を保護し、インターネット上のウェブサイトやプライベートネットワーク上のリソースの ID を確立するために使用されます。

これまでは、EC2 インスタンスで Web サーバーを実行する場合、SSL 証明書を作成し、インスタンスにプレーンテキストとして保存していました。ACM for Nitro Enclaves を使用すると、AWS Certificate Manager 証明書をエンクレーブにバインドし、証明書をプレーンテキスト形式で親インスタンスとそのユーザーに公開することなく、それらの証明書を Web サーバーで直接使用できるようになります。

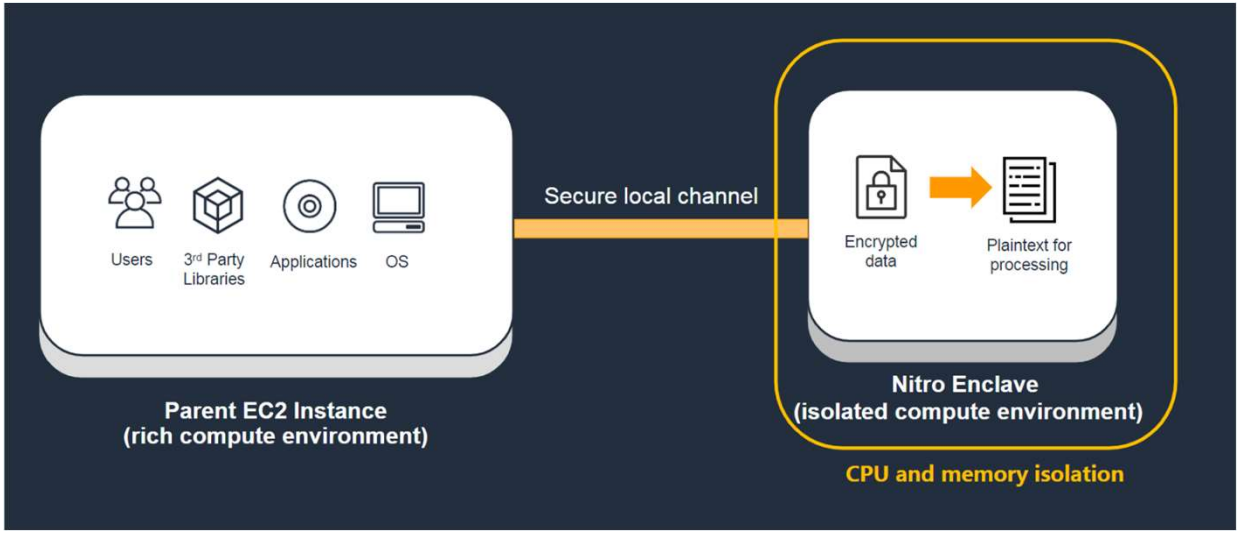
ACM for Nitro Enclaves を使用すると、SSL/TLS 証明書の購入、アップロード、更新といった時間がかかり、エラーが発生しやすい手動のプロセスが不要になります。ACM for Nitro Enclaves は、安全な秘密鍵を作成し、証明書とその秘密鍵をエンクレーブに配布し、証明書の更新を管理します。ACM for Nitro Enclaves を使用すると、証明書の秘密鍵はエンクレーブ内で分離されたままになり、インスタンスとそのユーザーがアクセスできなくなります。

現在、Nitro EnclavesのACMはNGINXサーバーで動作します。 およびApache HTTPサーバー Amazon EC2 インスタンス上で実行して証明書をインストールし、期限切れの証明書をシームレスに置き換えます。追加の Web サーバーのサポートは、今後追加される予定です。

AWS Nitro Enclavesとは

https://aws.amazon.com/jp/ec2/nitro/nitro-enclaves/

- インスタンスのリソースを一部利用する形で稼働する、独立した仮想マシン
- インスタンス側からエンクレーブに直接アクセスしたりSSH接続することはない
- 個人を特定できる情報 (PII) などの最も機密性の高いデータや、データ処理アプリケーションを保護できる



Nitro Enclavesが利用できる環境

https://docs.aws.amazon.com/ja_jp/enclaves/latest/user/nitro-enclave.html#nitro-enclave-reqs

Requirements

Nitro Enclaves has the following requirements:

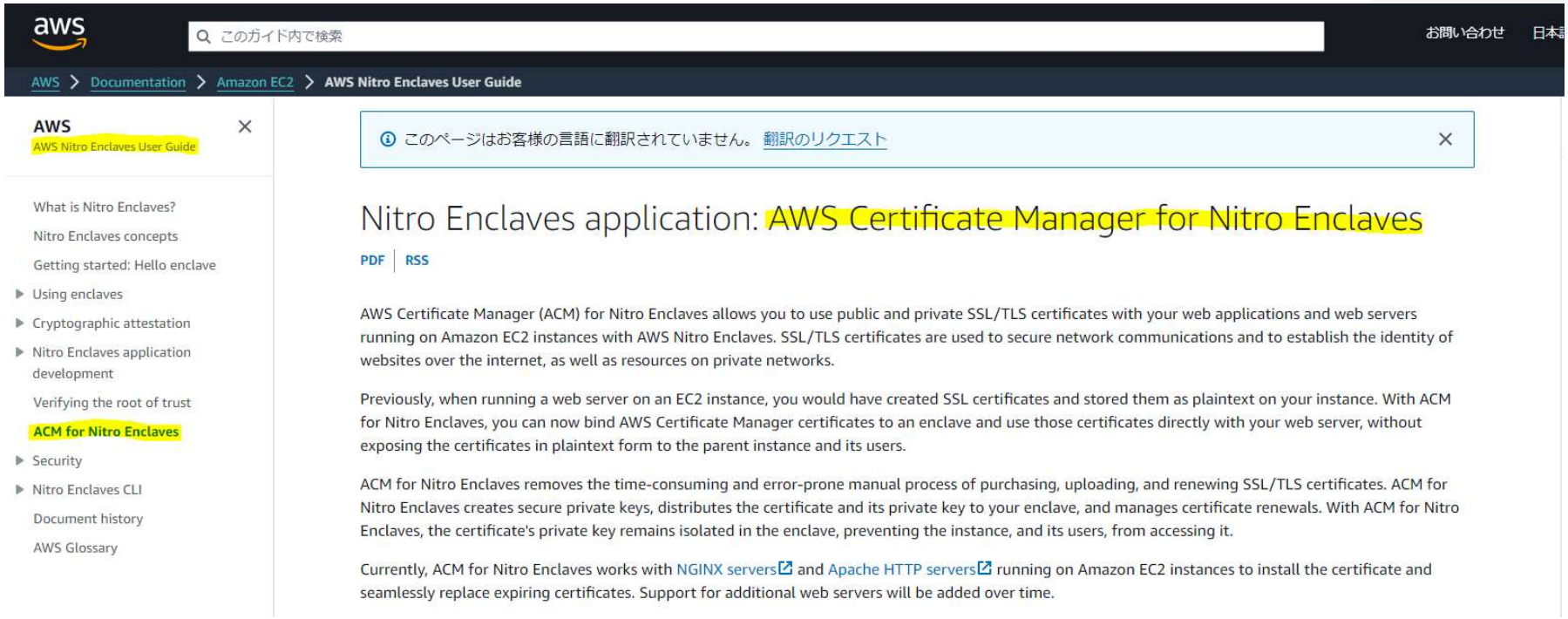
- Parent instance requirements:
  - Virtualized Nitro-based instances
    - Intel or AMD-based instances with at least 4 vCPUs, excluding `C7a`, `C7i`, `G4ad`, `M7a`, `M7i`, `M7i-Flex`, `R7a`, `R7i`, `R7iz`, `T3`, `T3a`, `Trn1`, `Trn1n`, `U-*`, `VT1`
    - AWS Graviton-based instances with at least 2 vCPUs, excluding `A1`, `C7gd`, `C7gn`, `G5g`, `Hpc7g`, `Im4gn`, `Is4gen`, `M7g`, `M7gd`, `R7g`, `R7gd`, `T4g`
  - Linux or Windows (2016 or later) operating system
- Enclave requirements:
  - Linux operating system only

実践

AWS公式ドキュメント

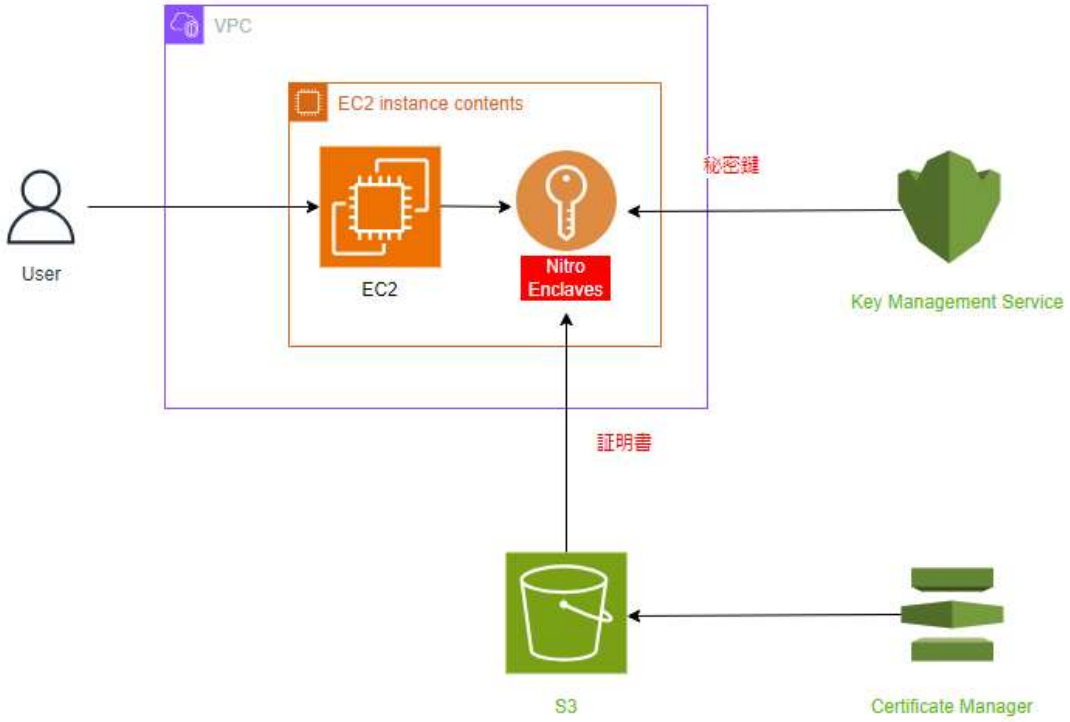
Nitro Encravesの資料の中に、

今回のテーマであるACMと連携する場合の設定方法があります
https://docs.aws.amazon.com/enclaves/latest/user/nitro-enclave-refapp.html



概念図

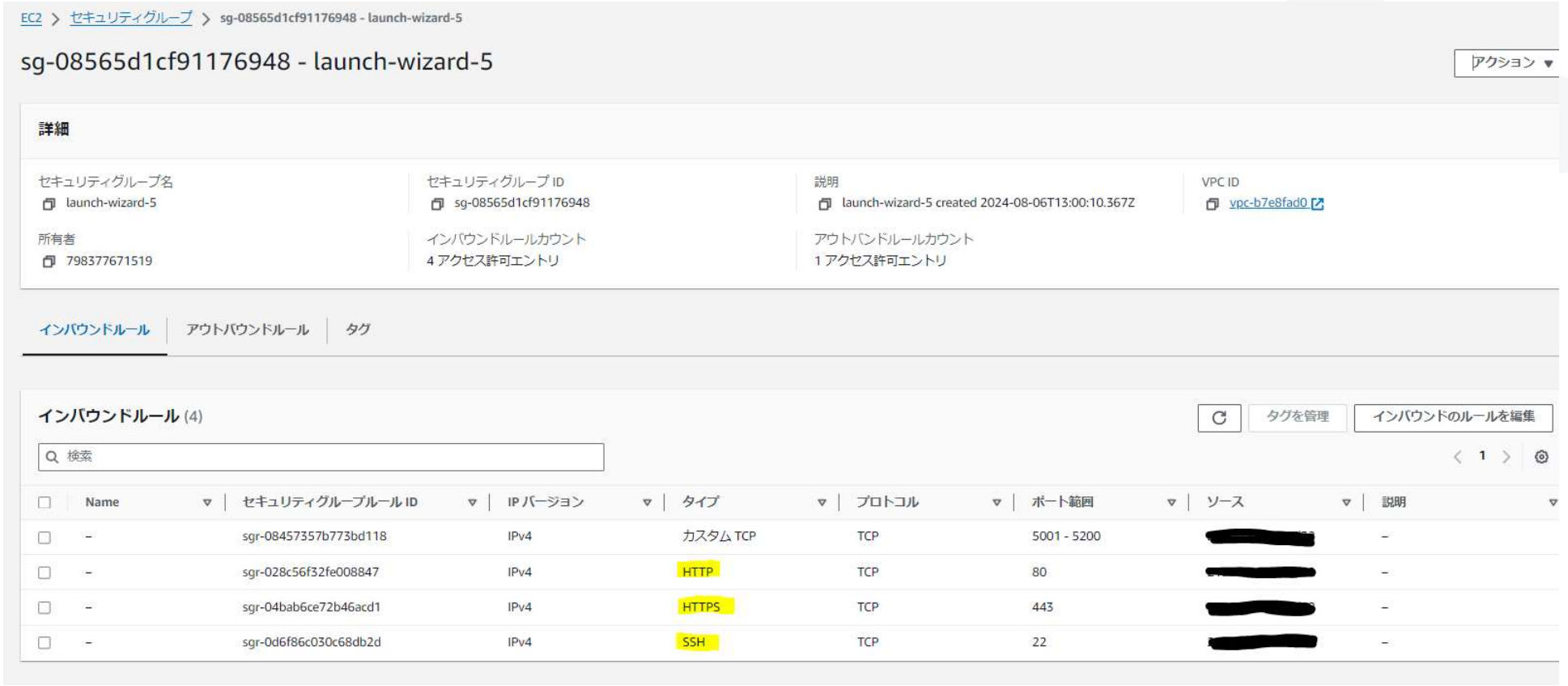
許可設定内容から推測した図になります



インスタンス構築

Nitro Enclavesの設定

セキュリティグループの設定



ACMで証明書の発行

ACMで証明書の発行

aws-nitro-enclaves-cli の有効化

```
[root@a888-r8a ~]# sudo amazon-linux-extras enable aws-nitro-enclaves-cli
 2  httpd_modules                available    [ =1.0  =stable ]
 3  memcached1.5                 available    ¥
       [ =1.5.1  =1.5.16  =1.5.17 ]
 9  R3.4                         available    [ =3.4.3  =stable ]
10  rust1                        available    ¥
       [ =1.22.1  =1.26.0  =1.26.1  =1.27.2  =1.31.0  =1.38.0
         =stable ]
18  libreoffice                  available    ¥
       [ =5.0.6.2_15  =5.3.6.1  =stable ]
19  gimp                         available    [ =2.8.22 ]
20 †docker=latest                enabled      ¥
       [ =17.12.1  =18.03.1  =18.06.1  =18.09.9  =stable ]
21  mate-desktop1.x              available    ¥
       [ =1.19.0  =1.20.0  =stable ]
22  GraphicsMagick1.3            available    ¥
       [ =1.3.29  =1.3.32  =1.3.34  =stable ]
24  epel                         available    [ =7.11  =stable ]
25  testing                      available    [ =1.0  =stable ]
26  ecs                          available    [ =stable ]
27 †corretto8                    available    ¥
       [ =1.8.0_192  =1.8.0_202  =1.8.0_212  =1.8.0_222  =1.8.0_232
         =1.8.0_242  =stable ]
32  lustre2.10                   available    ¥
       [ =2.10.5  =2.10.8  =stable ]
33 †java-openjdk11               available    [ =11  =stable ]
34  lynis                        available    [ =stable ]
36  BCC                          available    [ =0.x  =stable ]
37  mono                         available    [ =5.x  =stable ]
38  nginx1                       available    [ =stable ]
40  mock                         available    [ =stable ]
43  livepatch                    available    [ =stable ]
44 †python3.8                    available    [ =stable ]
45  haproxy2                     available    [ =stable ]
46  collectd                     available    [ =stable ]
47  aws-nitro-enclaves-cli=latest enabled     [ =stable ]
48  R4                           available    [ =stable ]
_   kernel-5.4                   available    [ =stable ]
50  selinux-ng                   available    [ =stable ]
52  tomcat9                      available    [ =stable ]
53  unbound1.13                  available    [ =stable ]
54 †mariadb10.5                  available    [ =stable ]
55  kernel-5.10=latest           enabled      [ =stable ]
56  redis6                       available    [ =stable ]
58 †postgresql12                 available    [ =stable ]
59 †postgresql13                 available    [ =stable ]
60  mock2                        available    [ =stable ]
61  dnsmasq2.85                  available    [ =stable ]
62  kernel-5.15                  available    [ =stable ]
63 †postgresql14                 available    [ =stable ]
64  firefox                      available    [ =stable ]
65  lustre                       available    [ =stable ]
```

```
66 †php8.1                       available    [ =stable ]
67  awscli1                      available    [ =stable ]
68 †php8.2                       available    [ =stable ]
69  dnsmasq                      available    [ =stable ]
70  unbound1.17                  available    [ =stable ]
72  collectd-python3             available    [ =stable ]
† Note on end-of-support. Use 'info' subcommand.

Now you can install:
 # yum clean metadata
 # yum install aws-nitro-enclaves-cli
```

# aws-nitro-enclaves-cli の有効化
sudo amazon-linux-extras enable aws-nitro-enclaves-cli

Apahceとmod_sslのインストール

```
[root@a888-r8a ~]# sudo yum -y install httpd mod_ssl
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                                          | 3.6 kB  00:00:00
amzn2extra-aws-nitro-enclaves-cli                                                                   | 2.9 kB  00:00:00
amzn2extra-docker                                                                                   | 2.9 kB  00:00:00
amzn2extra-kernel-5.10                                                                              | 3.0 kB  00:00:00
(1/2): amzn2extra-aws-nitro-enclaves-cli/2/x86_64/updateinfo                                        |  17 kB  00:00:00
(2/2): amzn2extra-aws-nitro-enclaves-cli/2/x86_64/primary_db                                        | 129 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package httpd.x86_64 0:2.4.62-1.amzn2.0.1 will be installed
--> Processing Dependency: httpd-filesystem = 2.4.62-1.amzn2.0.1 for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.1 for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.1.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.1.x86_64
---> Package mod_ssl.x86_64 1:2.4.62-1.amzn2.0.1 will be installed
--> Processing Dependency: sscg >= 2.2.0 for package: 1:mod_ssl-2.4.62-1.amzn2.0.1.x86_64
```

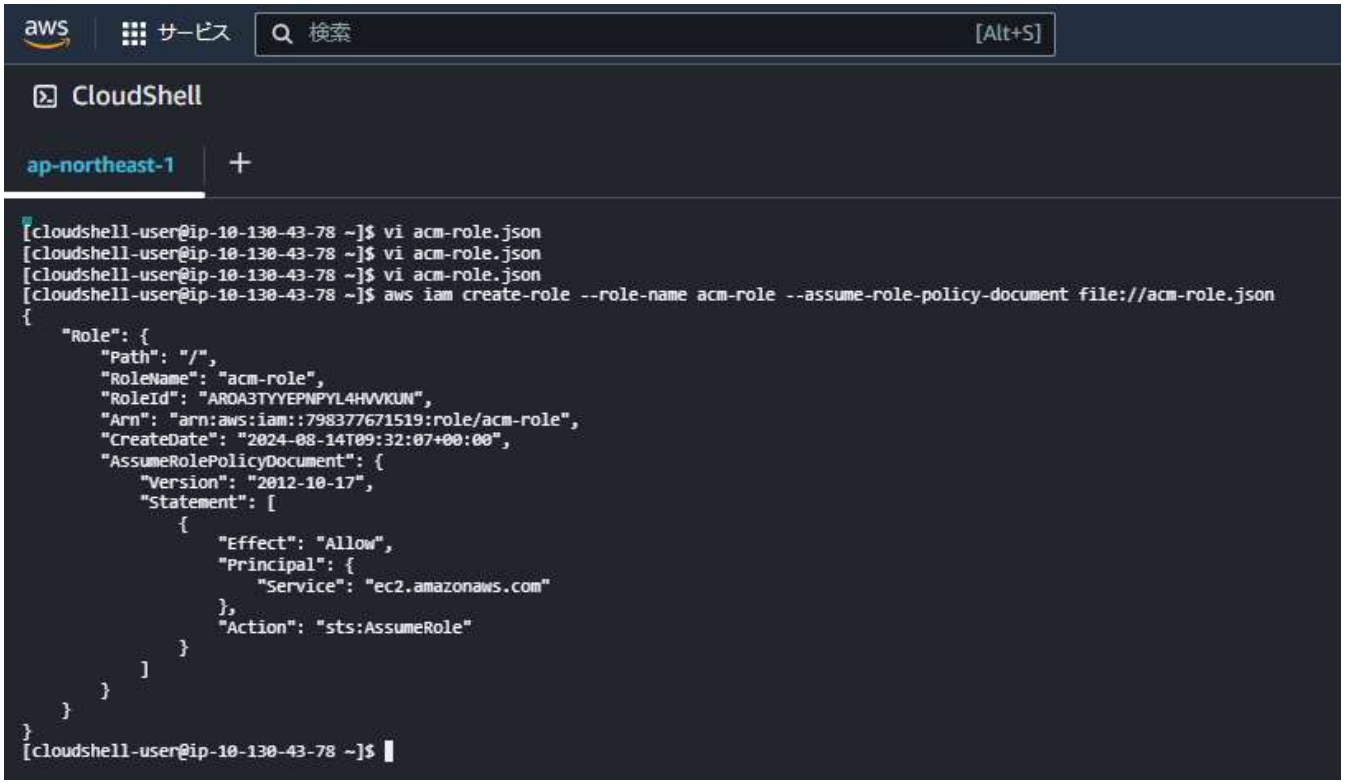
# Apahceとmod_sslのインストール
sudo yum -y install httpd mod_ssl

Nitro Enclaves用のACMをインストール

```
[root@a888-r8a ~]# sudo yum -y install aws-nitro-enclaves-acm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package aws-nitro-enclaves-acm.x86_64 0:1.3.0-2.amzn2 will be installed
--> Processing Dependency: aws-nitro-enclaves-cli for package: aws-nitro-enclaves-acm-1.3.0-2.amzn2.x86_64
--> Processing Dependency: jq for package: aws-nitro-enclaves-acm-1.3.0-2.amzn2.x86_64
--> Processing Dependency: openssl-pkcs11 for package: aws-nitro-enclaves-acm-1.3.0-2.amzn2.x86_64
--> Running transaction check
---> Package aws-nitro-enclaves-cli.x86_64 0:1.3.2-0.amzn2 will be installed
--> Processing Dependency: docker for package: aws-nitro-enclaves-cli-1.3.2-0.amzn2.x86_64
---> Package jq.x86_64 0:1.5-1.amzn2.0.2 will be installed
--> Processing Dependency: libonig.so.2()(64bit) for package: jq-1.5-1.amzn2.0.2.x86_64
---> Package openssl-pkcs11.x86_64 0:0.4.10-3.amzn2.0.1 will be installed
--> Running transaction check
```

# Nitro Enclaves用のACMをインストール
sudo yum -y install aws-nitro-enclaves-acm

IAMロールの準備



IAMロールをACM証明書にアタッチする

```
[cloudshell-user@ip-10-130-43-78 ~]$
[cloudshell-user@ip-10-130-43-78 ~]$ aws ec2 --region ap-northeast-1 associate-enclave-certificate-iam-role --certificate-arn arn:aws:acm:ap-northeast-1:798377671519:certificate/fe9699db-a10e-4aed-8c1a-90a965e8a649 --role-arn arn:aws:iam::798377671519:role/acm-role
{
    "CertificateS3BucketName": "aws-ec2-enclave-certificate-ap-northeast-1-prod",
    "CertificateS3ObjectKey": "arn:aws:iam::798377671519:role/acm-role/arn:aws:acm:ap-northeast-1:798377671519:certificate/fe9699db-a10e-4aed-8c1a-90a965e8a649",
    "EncryptionKmsKeyId": "4d890095-414f-4463-a7c8-6e20382b289f"
}
[cloudshell-user@ip-10-130-43-78 ~]$
```

aws ec2 --region region associate-enclave-certificate-iam-role --certificate-arn certificate_ARN --role-arn role_ARN

IAMロールにSSL証明書と暗号化キーにアクセスするための権限を付与する

```
[cloudshell-user@ip-10-130-43-78 ~]$ cat acm-role-policies.json
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Effect": "Allow",
        "Action": [
        "s3:GetObject"
        ],
        "Resource": ["arn:aws:s3:::aws-ec2-enclave-certificate-ap-northeast-1-prod/*"]
    },
    {
        "Sid": "VisualEditor0",
        "Effect": "Allow",
        "Action": [
            "kms:Decrypt"
        ],
        "Resource": "arn:aws:kms:ap-northeast-1:*:key/4d890095-414f-4463-a7c8-6e20382b289f"
    },
    {
            "Effect": "Allow",
            "Action": "iam:GetRole",
            "Resource": "arn:aws:iam::[redacted]:role/acm-role"
    }
  ]
}
```

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Effect": "Allow",
        "Action": [
        "s3:GetObject"
        ],
        "Resource": ["arn:aws:s3:::CertificateS3BucketName/*"]
    },
    {
        "Sid": "VisualEditor0",
        "Effect": "Allow",
        "Action": [
            "kms:Decrypt"
        ],
        "Resource": "arn:aws:kms:region:*:key/EncryptionKmsKeyId"
    },
    {
            "Effect": "Allow",
            "Action": "iam:GetRole",
            "Resource": "arn:aws:iam::123456789012:role/acm-role"
    }
  ]
}
```

```
[cloudshell-user@ip-10-130-43-78 ~]$ aws iam put-role-policy --role-name acm-role --policy-name acm-role-policy --policy-document file://acm-role-policies.json
[cloudshell-user@ip-10-130-43-78 ~]$
```

IAMロールをEC2インスタンスにアタッチする

```
[cloudshell-user@ip-10-130-43-78 ~]$ aws iam create-instance-profile --instance-profile-name acm-instance-profile
{
    "InstanceProfile": {
        "Path": "/",
        "InstanceProfileName": "acm-instance-profile",
        "InstanceProfileId": "AIPA3TYYEPNP2UEN4UZPP",
        "Arn": "arn:aws:iam::798377671519:instance-profile/acm-instance-profile",
        "CreateDate": "2024-08-14T09:49:00+00:00",
        "Roles": []
    }
}
[cloudshell-user@ip-10-130-43-78 ~]$ aws iam add-role-to-instance-profile --instance-profile-name acm-instance-profile --role-name acm-role
[cloudshell-user@ip-10-130-43-78 ~]$
```

```
[cloudshell-user@ip-10-130-43-78 ~]$ aws ec2 --region ap-northeast-1 associate-iam-instance-profile --instance-id i-0a0f67949ce4c73f4 --iam-instance-profile Name=acm-instance-profile
{
    "IamInstanceProfileAssociation": {
        "AssociationId": "iip-assoc-0e73a75ee7ad79a87",
        "InstanceId": "i-0a0f67949ce4c73f4",
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::798377671519:instance-profile/acm-instance-profile",
            "Id": "AIPA3TYYEPNP2UEN4UZPP"
        },
        "State": "associating"
    }
}
```

aws ec2 --region region associate-iam-instance-profile --instance-id instance_id --iam-instance-profile Name=acm-instance-profile

Nitro Enclavesのセットアップ

```
[root@a888-r8a ~]# sudo mv /etc/nitro_enclaves/acm-httpd.example.yaml /etc/nitro_enclaves/acm.yaml
[root@a888-r8a ~]# cp -p /etc/nitro_enclaves/acm.yaml /home/ec2-user/acm.yaml.20240814
[root@a888-r8a ~]#
```

```
[root@a888-r8a ~]# vi /etc/nitro_enclaves/acm.yaml
[root@a888-r8a ~]# diff -u /home/ec2-user/acm.yaml.20240814 /etc/nitro_enclaves/acm.yaml
--- /home/ec2-user/acm.yaml.20240814    2023-09-07 13:47:26.000000000 +0000
+++ /etc/nitro_enclaves/acm.yaml        2024-08-14 09:58:42.130240523 +0000
@@ -42,7 +42,7 @@
         # Note: this certificate must have been associated with the
         #       IAM role assigned to the instance on which ACM for
         #       Nitro Enclaves is run.
-        certificate_arn: ""
+        certificate_arn: "arn:aws:acm:ap-northeast-1:798377671519:certificate/fe9699db-a10e-4aed-8c1a-90a965e8a649"
     target:
       Conf:
         # Path to the server configuration file to be written by
```

Apacheのセットアップ

```
[root@a888-r8a ~]# cat /etc/httpd/conf.d/httpd-acm.conf
<VirtualHost *:443>
    ServerName a888-r8a.gnc-training.jp
    ServerAlias www.a888-r8a.gnc-training.jp
    SSLEngine on
    SSLProtocol -all +TLSv1.2
    SSLCertificateKeyFile "/etc/pki/tls/private/localhost.key"
    SSLCertificateFile "/etc/pki/tls/certs/localhost.crt"
</VirtualHost>
```

秘密鍵・証明書は
ひとまずローカルで仮設定

サービス起動

```
[root@a888-r8a ~]# systemctl enable --now nitro-enclaves-acm
Created symlink from /etc/systemd/system/multi-user.target.wants/nitro-enclaves-acm.service to /usr/lib/systemd/system/nitro-enclaves-acm.service.
[root@a888-r8a ~]# sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: inactive (dead)
     Docs: man:httpd.service(8)
[root@a888-r8a ~]# sudo systemctl status nitro-enclaves-acm
● nitro-enclaves-acm.service - Nitro Enclaves ACM Agent
   Loaded: loaded (/usr/lib/systemd/system/nitro-enclaves-acm.service; enabled; vendor preset: disabled)
   Active: activating (auto-restart) (Result: exit-code) since Wed 2024-08-14 13:55:34 UTC; 176ms ago
  Process: 1592 ExecStopPost=/usr/bin/rm -r /run/nitro_enclaves/acm (code=exited, status=0/SUCCESS)
  Process: 1586 ExecStart=/usr/bin/p11ne-agent (code=exited, status=1/FAILURE)
  Process: 1584 ExecStartPre=/usr/bin/mkdir -p /run/nitro_enclaves/acm (code=exited, status=0/SUCCESS)
 Main PID: 1586 (code=exited, status=1/FAILURE)

Aug 14 13:55:34 a888-r8a.gnc-training.jp systemd[1]: Unit nitro-enclaves-acm.service entered failed state.
Aug 14 13:55:34 a888-r8a.gnc-training.jp systemd[1]: nitro-enclaves-acm.service failed.
```

```
[root@nitor_enclaves_test4 .ssh]# cat /etc/httpd/conf.d/httpd-acm.conf
<VirtualHost *:443>
ServerName nitor-enclaves-test.gnc-training.jp
SSLEngine on
SSLProtocol -all +TLSv1.2
SSLCertificateKeyFile "pkcs11:model=p11ne-token;manufacturer=Amazon;token=httpd-acm-token;id=%01;object=acm-key;type=private?pin-value=d6ce921a9e2e6bf8edb5d99538e0059e"
SSLCertificateFile "/run/nitro_enclaves/acm/httpd-cert-68747470642d61636d2d746f6b656e.pem"
</VirtualHost>
```

AWSへ問い合わせ ～回答～

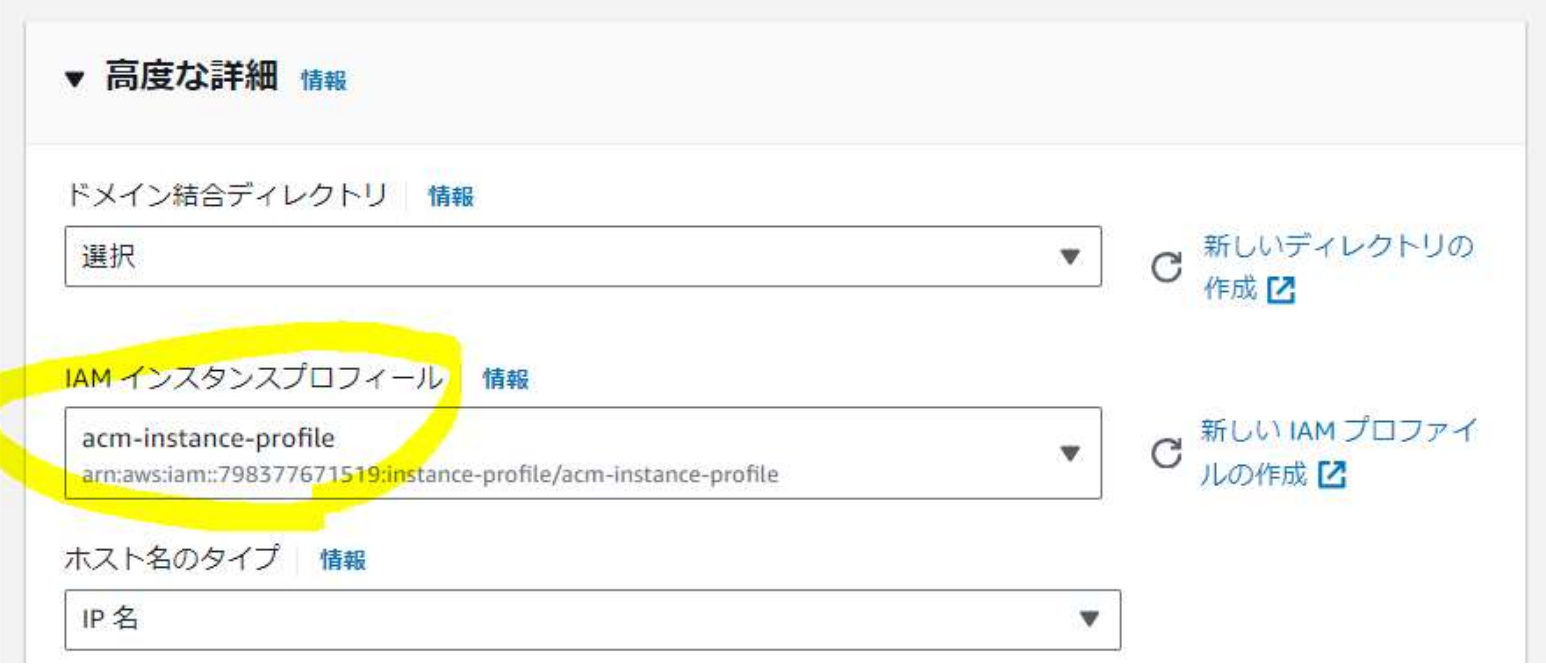
■ 回答
お客様のお問い合わせにつきまして、当方においても同様の事象を確認いたしました。
調査の結果、Apacheとmod_sslの最新バージョン: 2.4.62をnitro-enclaves-acmから使用するとApacheの起動に失敗することが原因となっております。

Apacheとmod_sslのバージョンとして、2.4.59の使用が有効であると確認しております。
そのためには、ドキュメント[1]の、Step 2: Prepare the enclaves-enabled parent instanceのOption 2: Using RPM packagesにて以下の変更を行います。

変更前: sudo yum -y install httpd mod_ssl

変更後: sudo yum -y install httpd-2.4.59 mod_ssl-2.4.59

修正



動作確認

動作確認

Certificate Chain Complete?

All of the correct Intermediate CA Certificates are installed. Your SSL certificate is installed correctly and should be supported in all the major web browsers without problems.

Common Name:
Organization: Starfield Technologies, Inc.
Valid: June 29, 2004 to June 29, 2034
Issuer: Starfield Technologies, Inc.
Serial Number: 0

Common Name: Starfield Services Root Certificate Authority - G2
Organization: Starfield Technologies, Inc.
Valid: September 02, 2009 to June 28, 2034
Issuer: Starfield Technologies, Inc.
Serial Number: A70E4A4C3482B77F

Common Name: Amazon Root CA 1
Organization: Amazon
Valid: May 25, 2015 to December 31, 2037
Issuer: Starfield Services Root Certificate Authority - G2
Serial Number: 067F944A2A27CDF3FAC2AE2B01F908EEB9C4C6

Common Name: Amazon RSA 2048 M02
Organization: Amazon
Valid: August 23, 2022 to August 23, 2030
Issuer: Amazon Root CA 1
Serial Number: 0773124A4BCBD44EC7B53BEAF194842D3A0FA1

Common Name: nitor-enclaves-test.gnc-training.jp
Organization:
Valid: August 15, 2024 to September 14, 2025
Issuer: Amazon RSA 2048 M02
Serial Number: 0AEB6C444618EB9D1BA1CB3159353D99

まとめ

利点

◆証明書を無償で利用できる

◆更新の手間が要らない

◆ワイルドカード証明書等を複数インスタンスに適用している場合などで、効果が大きい

難点

◆インスタンス料金が高い【改善は難しい】
（Nitro Enclavesは無償）

~~◆ACMを使いたいならCloudFrontで良い・・・~~

◆Apache・mod_sslのバージョンに制限がある
【改善される？】

~~◆Let’s Encryptで良い・・・~~

◆ドメイン認証型サーバ証明書しか使えない
【改善は難しい】

発行機関比較

	一般的な認証局	Let's Encrypt	AWS ACM
費用	有償	無償	無償
証明書種類	以下から選択可能 ドメイン認証 企業実在認証 EV認証	ドメイン認証のみ	ドメイン認証のみ
適用・更新方法	基本的に手動	自動更新可能	自動更新可能
証明書有効期限	1年＋1ヶ月	90日	1年＋1ヶ月
検証方法	認証局に依る	HTTP-01チャレンジ DNS-01チャレンジ TLS-ALPN-01チャレンジ	DNS検証 Eメール検証

証明書の有効期間について

◆さくらインターネット様
「何度も短縮し過ぎ？！SSL証明書の有効期間がどんどん短くなる理由とは？」
https://ssl.sakura.ad.jp/column/shortened-ssl/

◆当初は最長5年だったが、3年、2年と短縮されて、2020年には1年+1ヶ月となった

◆証明書のインシデントが発生した際は大量の再発行が発生する
有効期間が短ければ再発行が必要な量が減る可能性がある

今後の証明書管理について

◆認証局の自動更新プラットフォーム整備に期待

GMOグローバルサインの「Atlas」等

https://college.globalsign.com/blog/about_atlas_20210517/

◆認証局のACMEサービスの対応

ACMEクライアントとしては、米Certbot等

ウェブサーバと認証局との間の証明書の相互作用（インストール、更新など）を自動化するための国際標準の通信プロトコル

https://college.globalsign.com/blog/acme_ssl_231120/

確かな未来を、確かな力で。

GLOBAL NET CORE