

課題演習 No.1
(IAM + EC2 + VPC)



課題演習1(IAM + EC2 + VPC)

目的:

ある会社が社内システムをオンプレミス環境からAWS環境への移行を計画している。

機密データが格納されている社内システムなのでインターネットからの直接アクセスはできないようにしたい。

その中で社内システムのメンテナンスも実施したいが、VPNやDirectConnectは準備やコストで負担が大きいため、コストを掛けずにセキュアな方法で社内システムへのアクセスを実現したい。

1:要件

■以下の要件でインフラを構築してください。

1. コストを最小限に抑えるために、冗長構成は不要である。
2. 社内システム用のEC2はインターネットからのアクセスをできないようにしたい。EC2からのインターネットへのアクセスはOSパッチ適用のため、許可したい。
3. 2のEC2にセキュアにアクセスできるサーバーを作成したい。またこの3.で作成されるサーバーは外部からSSH接続できるようにし、サーバーのIPアドレスは固定にしたい。(3のアドレスのみ2のサーバーへアクセス可能にする)
4. 社内システム用のEC2もSSH接続を可能にしたい。セキュリティ強化として、社内システムへのSSH接続時はポート番号を別のものにしたい。
(ポート番号は10022に指定)
5. 運用者用アカウントを作成し、そのIAMユーザーにはEC2の削除(終了)をさせないように権限設定したい。今後もアカウントは増える予定で、追加時の権限設定の手間を軽減させたい。

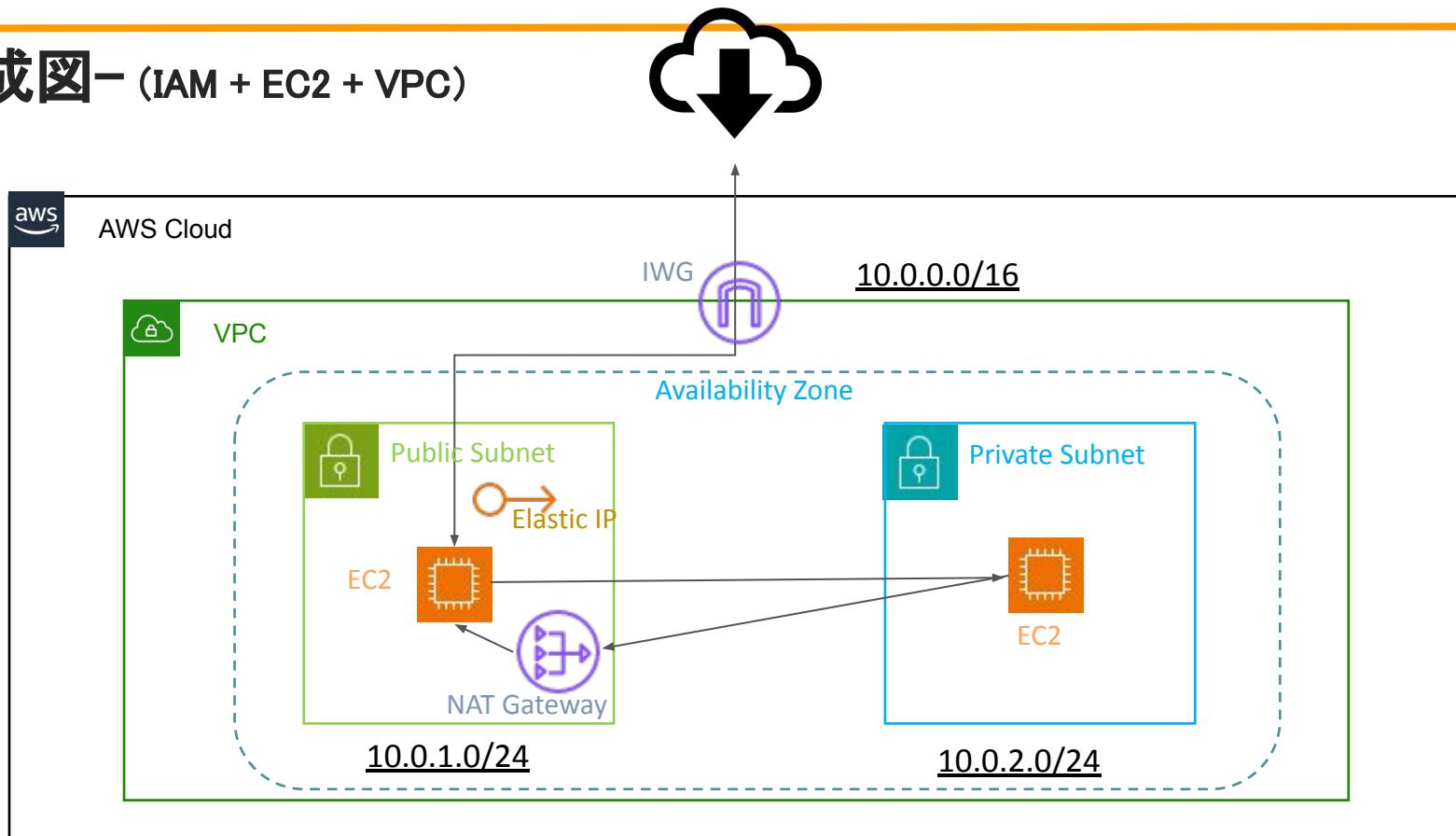
2: 設計

■要件に対しての仕様(解決策)

No	要件	仕様(解決策)
1	コストを最小限に抑えるために、冗長構成も不要である。	シングル構成でコスト最小限に抑える。
2	社内システム用のEC2はインターネットからのアクセスをできないようにしたい。 ソフトのアップデートはできるように設定したい。	プライベートサブネット内にEC2を設定してインターネットからのアクセスをできないようにする。 パブリックサブネット内にNATGatewayを設定し社内システムへのインバウンドを制限。 社内システムから踏み台サーバーを経由してインターネットへSSH接続、ソフトのアップデートは可能。
3	踏み台サーバーへのアクセス時のIPアドレスを固定にしたい。また社内システム用EC2へのSSH接続を可能にする。	踏み台サーバーへElasticIPでIPアドレスを固定。 セキュリティグループの設定をしSSH接続可能。
4	セキュリティ強化の為、社内システムへのSSH接続時はポート番号を別のものにしたい。 (ポート番号は10022に指定)	踏み台サーバーから社内システムへのSSH接続のポート番号は10022に設定。
5	運用者用アカウントを作成し、そのIAMユーザーにはEC2の削除(終了)をさせないように権限設定したい。 今後もアカウントは増える予定で、追加時の権限設定の手間を軽減させたい。	IAMグループを作成してEC2を削除をさせないポリシーをアタッチ、新規ユーザーを作成してIAMグループ内に追加制御する。

2: 設計

-構成図- (IAM + EC2 + VPC)



3: 構築(実装)

① シングルインスタンスでコスト最小化

VPC の設定

作成するリソース [情報](#)
VPC リソースのみ、または VPC と他のネットワークリソースを作成します。

VPC のみ VPC など

名前タグの自動生成 [情報](#)
Name タグの値を入力します。この値は、VPC 内のすべてのリソースの Name タグを自動生成するのに使用されます。

自動生成
test

IPv4 CIDR ブロック [情報](#)
CIDR 表記を使用して VPC の開始 IP とサイズを決定します。

10.0.0.0/16 65,536 IPs

CIDR ブロックサイズは /16 から /28 の間である必要があります。

レビュー



3: 構築(実装)

- ② 社内システム用のEC2はインターネットからのアクセスをできないようプライベートサブネット内に社内システムを構築。
※VPC作成時にVPCなどを選択時に同時に構築

i-0a448b534089a3034 (pri_instace) のインスタンス概要 [情報](#) [接続](#) [インスタンスの状態 ▾](#) [アクション ▾](#)

less than a minute 前に更新済み

インスタンス ID i-0a448b534089a3034	パブリック IPv4 アドレス -	プライベート IPv4 アドレス 10.0.1.239
IPv6 アドレス -	インスタンスの状態 実行中	パブリック IPv4 DNS -
ホスト名のタイプ IP 名: ip-10-0-1-239.ap-northeast-1.compute.internal	プライベート IP DNS 名 (IPv4 のみ) ip-10-0-1-239.ap-northeast-1.compute.internal	Elastic IP アドレス -
プライベートリソースの DNS 名に応答 -	インスタンスタイプ t2.micro	AWS Compute Optimizer の検出結果 レコメンデーションについては、AWS Compute Optimizer にオプトインしてください。
自動的に割り当てられた IP アドレス -	VPC ID vpc-00d08b2e6b14fb682 (test_VPC)	

アベイラビリティゾーン (AZ) の数 [情報](#)
サブネットをプロビジョニングする AZ の数を選択します。可用性を高めるには、少なくとも 2 つの AZ をお勧めします。

[1](#) [2](#) [3](#)

▶ AZ のカスタマイズ

パブリックサブネットの数 [情報](#)
VPC に追加するパブリックサブネットの数。インターネット経由でパブリックにアクセス可能にする必要があるウェブアプリケーションには、パブリックサブネットを使用します。

[0](#) [1](#)

プライベートサブネットの数 [情報](#)
VPC に追加するプライベートサブネットの数。プライベートサブネットを使用して、パブリックアクセスを必要としないバックエンドリソースを保護します。

[0](#) [1](#) [2](#)

▶ サブネット CIDR ブロックをカスタマイズ

3: 構築(実装)

② パブリックインスタンス起動時にキーペアを作成

プライベートインスタンス起動時にも同様のキーペアを選択

キーペアをダウンロードしてローカルPCからインスタンスへログインする際に使用



3: 構築(実装)

- ② 社内システムへのインバウンドのトラフィック制御の為に
NATGatewayをパブリックサブネットへ構築
※VPC作成時にVPCなどを選択時に同時に構築

概要

割り振られた IPv4 アドレス <input checked="" type="checkbox"/> 54.65.201.14	タイプ <input checked="" type="checkbox"/> パブリック IP	割り当て ID <input checked="" type="checkbox"/> eipalloc-0b8f8c283379251be	逆引き DNS レコード -
関連付け ID <input checked="" type="checkbox"/> eipassoc-04bba37291d0e56d9	範囲 <input checked="" type="checkbox"/> VPC	関連付けられたインスタンス ID -	プライベート IP アドレス <input checked="" type="checkbox"/> 10.0.1.125
ネットワークインターフェース ID eni-00c0ad394e16d501f	ネットワークインターフェイスの所有者アカウント ID <input checked="" type="checkbox"/> 584791980123	パブリック DNS -	NAT ゲートウェイ ID nat-05721469a3767a104 (プロジェクト-nat-public1-ap-northeast-1a)
アドレスプール <input checked="" type="checkbox"/> Amazon	ネットワークポーダーグループ <input checked="" type="checkbox"/> ap-northeast-1		

0 1 2 ▶ サブネット CIDR ブロックをカスタマイズ

NAT ゲートウェイ (\$) 情報
NAT ゲートウェイを作成するアベイラビリティゾーン (AZ) の数を選択します。NAT ゲートウェイごとに料金が発生することに注意してください。

なし 1 AZ 内 AZ ごとに 1

VPC エンドポイント 情報
エンドポイントは、VPC から S3 に直接アクセスすることで、NAT ゲートウェイの料金を削減し、セキュリティを向上させるのに役立ちます。デフォルトでは、フルアクセスポリシーが使用されます。このポリシーはいつでもカスタマイズできます。

なし S3 ゲートウェイ

3: 構築(実装)

- ② ソフトウェアのアップデートができるようにIGWをVPCにアタッチ
※VPC作成時にVPCなどを選択時にIGWを同時に構築

igw-04e77e9dc096fde87 / プロジェクト-igw

詳細 情報

インターネットゲートウェイ ID
 igw-04e77e9dc096fde87

状態
 Attached

VPC ID
[vpc-03787ea1890e521f2 | プロジェクト-vpc](#)

3: 構築(実装)

- ③ 踏み台サーバーへElasticIPでIPアドレスを固定。
※VPC作成時にVPCなどを選択時に同時に構築、アタッチ

Elastic IP アドレス: 54.65.156.5

リソースタイプ

Elastic IP アドレスを関連付けるリソースのタイプを選択します。

インスタンス

ネットワークインターフェイス

△ 既に Elastic IP アドレスが関連付けられているインスタンスに Elastic IP アドレスを関連付けると、以
引き続きアカウントに割り振られたままとなります。詳細

プライベート IP アドレスが指定されていない場合、Elastic IP アドレスはプライマリプライベート IP

インスタンス

Q i-082f6f1af4ed83c80

プライベート IP アドレス

Elastic IP アドレスを関連付けるプライベート IP アドレス。

Q 10.0.1.244

再関連付け

i-082f6f1af4ed83c80 (test-pub-instance) のインスタンス概要 情報

 接続  インスタンスの状態 ▾  アクション ▾

less than a minute 前に更新済み

インスタンス ID

Q i-082f6f1af4ed83c80

IPv6 アドレス

ホスト名のタイプ

IP 名: ip-10-0-1-244.ap-northeast-
1.compute.internal

プライベートリソースの DNS 名に応答

-

自動的に割り当てられた IP アドレス

-

パブリック IPv4 アドレス

Q 54.65.156.5 | オープンアドレス

インスタンスの状態

Q 実行中

プライベート IP DNS 名 (IPv4 のみ)

Q ip-10-0-1-244.ap-northeast-1.compute.intern
al

インスタンスタイプ

t2.micro

VPC ID

Q vpc-03787ea1890e521f2 (プロジェクト-
vnc) [?]

プライベート IPv4 アドレス

Q 10.0.1.244

パブリック IPv4 DNS

-

Elastic IP アドレス

Q 54.65.156.5 | [パブリック IP]

AWS Compute Optimizer の検出結果

① レコメンデーションについては、AWS Compute
Optimizer にオプトインしてください。

3: 構築(実装)

- ③ セキュリティグループを設定しSSH接続可能。
- ④ 踏み台サーバーから社内システムへのSSH接続のポート番号は10022に設定。
パブリックインスタンス用のセキュリティグループ

左側の画面（インバウンドルールを編集）:

- セキュリティグループ ID: sgr-0db22099818b62a9e
- プロトコル: SSH
- ポート範囲: TCP 22
- ソース: 133.32.128.50/32

右側の画面（アウトバウンドルールを編集）:

- セキュリティグループ ID: sgr-07e0d3eda8927b884
- プロトコル: すべてのトラフィック
- ポート範囲: すべて
- 送信先: 0.0.0.0/0

警告メッセージ:

⚠️ 完先が 0.0.0.0/0 または ::/0 のルールでは、インスタンスは任意の IPv4 または IPv6 アドレスにトラフィックを送信できます。セキュリティグループのルールは制限を厳しくし、特定の既知の IP アドレスへのトラフィックのみを許可するように設定することをお勧めします。

3: 構築(実装)

- ③ セキュリティグループを設定しSSH接続可能。
- ④ 踏み台サーバーから社内システムへのSSH接続のポート番号は10022に設定。
プライベートインスタンス用のセキュリティグループ

The screenshot shows two side-by-side views of the AWS CloudFront console. Both views are titled 'CloudFront' and show the 'Distribution' tab.

Left View: Shows the 'Create New Distribution' wizard. Step 1: Set Origin. It lists 'Origin' and 'Origin Path'. Step 2: Set Cache Behavior. It lists 'Cache Behavior' and 'Default Cache Behavior'. Step 3: Set Price Class. It lists 'Price Class' and 'Optimize'.

Right View: Shows the 'Distribution' list. It lists three distributions: 'dist-1234567890123456 - test-prj-CF' (Status: Active), 'dist-9876543210987654 - test-prj-CF' (Status: Pending Review), and 'dist-0987654321098765 - test-prj-CF' (Status: Pending Review). The first distribution has a 'Actions' dropdown with options like 'Edit', 'Delete', and 'Edit Origin'.

3: 構築(実装)

- ⑤ IAMグループを作成して運用者アカウントにはEC2を削除をさせないポリシーをアタッチして、制御する。
アカウントの増加を考慮して、追加時の権限設定の手間を軽減させる。



The screenshot shows the 'Policies' section of the AWS IAM console. A new policy named 'DenyEC2Termination' is being created. The JSON editor shows the following policy definition:

```
1 {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Sid": "DenyEC2Termination",  
6             "Effect": "Deny",  
7             "Action": "ec2:TerminateInstances",  
8             "Resource": "*"  
9         }  
10    ]  
11 }
```

4: テスト結果(検証)

- ① ローカルPCにダウンロードしたキーペアをscpコマンドでインスタンスへ転送し、sshコマンドでパブリックインスタンスへのログインを確認

```
C:\Users\edu>scp -i "C:\Users\edu\Downloads\test_key_pair.pem" "C:\Users\edu\Downloa
ds\test_key_pair.pem" ec2-user@54.65.156.5:~/
The authenticity of host '54.65.156.5 (54.65.156.5)' can't be established.
ED25519 key fingerprint is SHA256:0xPt9dNn/LGX0dlklqGsvEmeLiUa/H3SsOS51gbJM7o.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Please type 'yes', 'no' or the fingerprint:
Warning: Permanently added '54.65.156.5' (ED25519) to the list of known hosts.
test_key_pair.pem
    100% 1674   136.2KB/s   00:00

C:\Users\edu>scp -i "C:\Users\edu\Downloads\test_key_pair.pem" "C:\Users\edu\Downloa
ds\test_key_pair.pem" ec2-user@54.65.156.5:~/
test_key_pair.pem
    100% 1674   102.2KB/s   00:00

C:\Users\edu>ssh -i "C:\Users\edu\Downloads\test_key_pair.pem" ec2-user@54.65.156.5
#_
~\_ #####
~~ \#####\ Amazon Linux 2023
~~ \|##|
~~ \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~ V~` _>
~~ /` /
~~ /` /` /
~~ /` /` /` /
/m/` [ec2-user@ip-10-0-1-244 ~]$ chmod 400 ~/test_key_pair.pem
```

4: テスト結果(検証)

- ② 社内システムのセキュリティグループのインバウンドルールをカスタムTCP、ポート番号10022に設定。

```
ec2-user@ip-10-0-1-239:~ ec2-user@ip-10-0-1-239:~ + - ~
#      $OpenBSD: sshd_config,v 1.104 2021/07/02 05:11:21 dtucker Exp $
#
# This is the sshd server system-wide configuration file. See
# sshd_config(5) for more information.
#
# This sshd was compiled with PATH=/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin
#
# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.
#
# To modify the system-wide sshd configuration, create a *.conf file under
# /etc/ssh/sshd_config.d/ which will be automatically included below
Include /etc/ssh/sshd_config.d/*.conf
#
# If you want to change the port on a SELinux system, you have to tell
# SELinux about this change.
# semanage port -a -t ssh_port_t -p tcp #PORTNUMBER
#
#Port 22
Port 10022
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key
"/etc/ssh/sshd_config" 134L, 3865B
```

4: テスト結果(検証)

- ③ プライベートインスタンスのインバウンドのポート番号を10022へ
変更後、パブリックインスタンスから社内システムへログインを確認

```
[ec2-user@ip-10-0-1-244 ~]$ ssh -i test_key_pair.pem ec2-user@10.0.2.15 -p 10022
#_
#_      Amazon Linux 2023
#_      https://aws.amazon.com/linux/amazon-linux-2023
V~' '-->
/
/
/m/
Last login: Wed Feb 12 02:27:43 2025 from 10.0.1.244
[ec2-user@ip-10-0-2-15 ~]$ ping google.com
^C
[ec2-user@ip-10-0-2-15 ~]$ ping google.co.jp
^C
[ec2-user@ip-10-0-2-15 ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=57 time=2.07 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=57 time=1.68 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=57 time=1.78 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=57 time=1.47 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=57 time=1.49 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=57 time=1.20 ms
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/avg/max/mdev = 1.196/1.614/2.073/0.274 ms
[ec2-user@ip-10-0-2-15 ~]$ |
```

4 : テスト結果(検証)

- ④ 踏み台サーバーから社内システムへ、社内システムからインターネットへの通信可能を確認。

```
[ec2-user@ip-10-0-1-244 ~]$ ssh -i test_key_pair.pem ec2-user@10.0.2.15 -p 10022
'      #
`~\_ #####_          Amazon Linux 2023
~~ \_#####\
~~ \##|
~~  \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~   \~' '->
~~     /
~~-.-
~~ /-
`m,'

Last login: Wed Feb 12 02:27:43 2025 from 10.0.1.244
[ec2-user@ip-10-0-2-15 ~]$ ping google.com
^C
[ec2-user@ip-10-0-2-15 ~]$ ping google.co.jp
^C
[ec2-user@ip-10-0-2-15 ~]$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=57 time=2.07 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=57 time=1.68 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=57 time=1.78 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=57 time=1.47 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=57 time=1.49 ms
64 bytes from 8.8.8.8: icmp_seq=6 ttl=57 time=1.20 ms
^C
--- 8.8.8.8 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/avg/max/mdev = 1.196/1.614/2.073/0.274 ms
[ec2-user@ip-10-0-2-15 ~]$
```

4: テスト結果(検証)

⑤ 運用者アカウントをユーザーグループへ追加をしEC2の削除を実行不可の応答を確認。

The screenshot shows the AWS Management Console with the EC2 service selected. A modal dialog box is displayed, indicating that the user 'arnawsiam:584791980123:user/samurai' does not have permission to perform the 'ec2:TerminateInstances' operation on the specified resource. The error message is: 'User: arnawsiam:584791980123:user/samurai is not authorized to perform: ec2:TerminateInstances on resource: arn:aws:ec2:ap-northeast-1:584791980123:instance-/0f3d2c8f7bf0e04 with an explicit deny in an identity-based policy. Encoded authorization failure message: GNxg9ADJldlnKndnN4fwTJEdwTTeUKHc22HOMy8l9C3yxSwZwo-v5oHeeVqTxY-1gLD7Ug7WxwspmLywj23c2j0ck9YR_BkhnoN-0tQypdP2NwxpYugGLVEJM78RtCm3AUUYXa06M50zEx2LqMc8s5U4FBW4McDb8bxDDeOafuxy:9xnlQb_gqf9E5dFWArigOpjV6evhnnP38zYfVsJh7Oc-WdmTE-0S6iKSCsFnMFxHKjrCHDl8382ctLdNywicpG8HpdKePDXJ4FqsCrg5e1wnPtUGMuucqegyFCNeWc3GMVWpdpjKOfrf_PLdoM8E1nGrbl6Uli2Z_4oc-GgatouM_kNh2XSRgDUdfr46KphV-HNcrtW1WeCxvN0RaKK214nf0Ebyst4z1T2xEFaQIPMEKwddt2sl4Gm_NxO_MAPsI4R4H6Ch6ElAEnmkSig5ptLfZ6bizZHl3Im2eTCYVWEP3Qu6sxLbI-Ry036GLDy0cD0zJzqlcfVP5tttg94ireMkYvBqj4DQ32ITlwV57RvAuEcVsGrjwUvEyZ2m1kSkyPulPuOQdox_Qas3ERAigEgRrGo10UkPNRKxgyMPON'.

The screenshot shows the AWS IAM console under the 'Accounts' section. It displays the account ID '5847-9198-0123' and the IAM user 'samurai'. Below this, there are sections for 'Bucket policies' (with a note about saving changes), 'Account' (with a note about saving changes), 'Organizations' (with a note about saving changes), 'Service Quotas' (with a note about saving changes), 'Billing and Cost Management' (with a note about saving changes), and 'Security and Identity' (with a note about saving changes). At the bottom, there are three buttons: 'Multi-session support' (highlighted in orange), 'Role creation' (highlighted in blue), and 'Sign out'.