

ASA IPsec VTI接続アマゾンウェブサービスの設定

内容

[概要](#)

[AWSの設定](#)

[ASA の設定](#)

[検証と最適化](#)

概要

このドキュメントでは、適応型セキュリティアプライアンス(ASA)のIPsec仮想トンネルインターフェイス(VTI)接続を設定する方法について説明します。ASA 9.7.1では、IPsec VTIが導入されています。このリリースでは、IKEv1を使用するsVTI IPv4 over IPv4に制限されています。これは、ASAがAmazon Web Services(AWS)に接続するための設定例です。

注：現在、VTIはシングルコンテキストのルーテッドモードでのみサポートされています。

AWSの設定

ステップ 1：

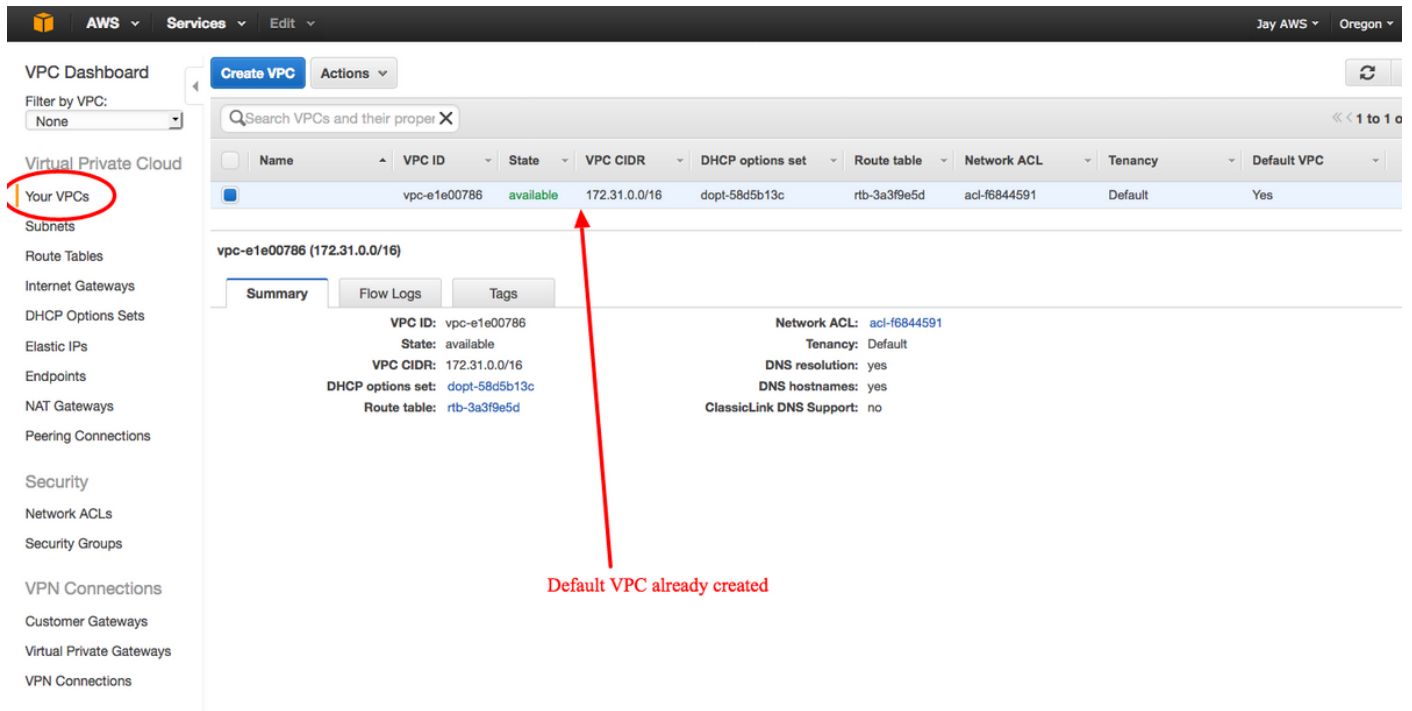
AWSコンソールにログインし、[VPC]パネルに移動します。



VPCダッシュボードに移動します

ステップ 2 :

バーチャルプライベートクラウド(VPC)がすでに作成されていることを確認します。 デフォルトでは、172.31.0.0/16のVPCが作成されます。ここで、仮想マシン(VM)が接続されます。



The screenshot shows the AWS VPC Dashboard. The left sidebar lists various VPC-related services, with 'Your VPCs' circled in red. The main content area displays a table of VPCs with the following columns: Name, VPC ID, State, VPC CIDR, DHCP options set, Route table, Network ACL, Tenancy, and Default VPC. A single VPC is listed with ID 'vpc-e1e00786', State 'available', and CIDR '172.31.0.0/16'. Below the table, the details for this VPC are shown, including its ID, state, CIDR, DHCP options set, route table, network ACL, and tenancy. A red arrow points from the text 'Default VPC already created' to the 'VPC CIDR' field in the table.

Name	VPC ID	State	VPC CIDR	DHCP options set	Route table	Network ACL	Tenancy	Default VPC
	vpc-e1e00786	available	172.31.0.0/16	dopt-58d5b13c	rtb-3a3f9e5d	acl-f6844591	Default	Yes

Default VPC already created

ステップ 3 :

「カスタマーゲートウェイ」を作成します。これは、ASAを表すエンドポイントです。

フィールド	値
名前タグ	これは、ASAを認識するための人間が読み取り可能な名前です。
ルーティング	ダイナミック：これは、ルーティング情報を交換するためにボーダーゲートウェイプロ
iSCSIポータル	これは、ASAの外部インターフェイスのパブリックIPアドレスです。
BGP ASN	ASAで実行されるBGPプロセスの自律システム(AS)番号。組織にパブリックAS番号が

The screenshot shows the AWS Management Console interface for creating a Customer Gateway. A modal dialog box titled "Create Customer Gateway" is open, displaying the following information:

- Name tag:** ASAVTI
- Routing:** Dynamic
- IP address:** 192.0.2.1
- BGP ASN:** 65000

Buttons for "Cancel" and "Yes, Create" are visible at the bottom of the dialog. In the background, a table lists existing Customer Gateways:

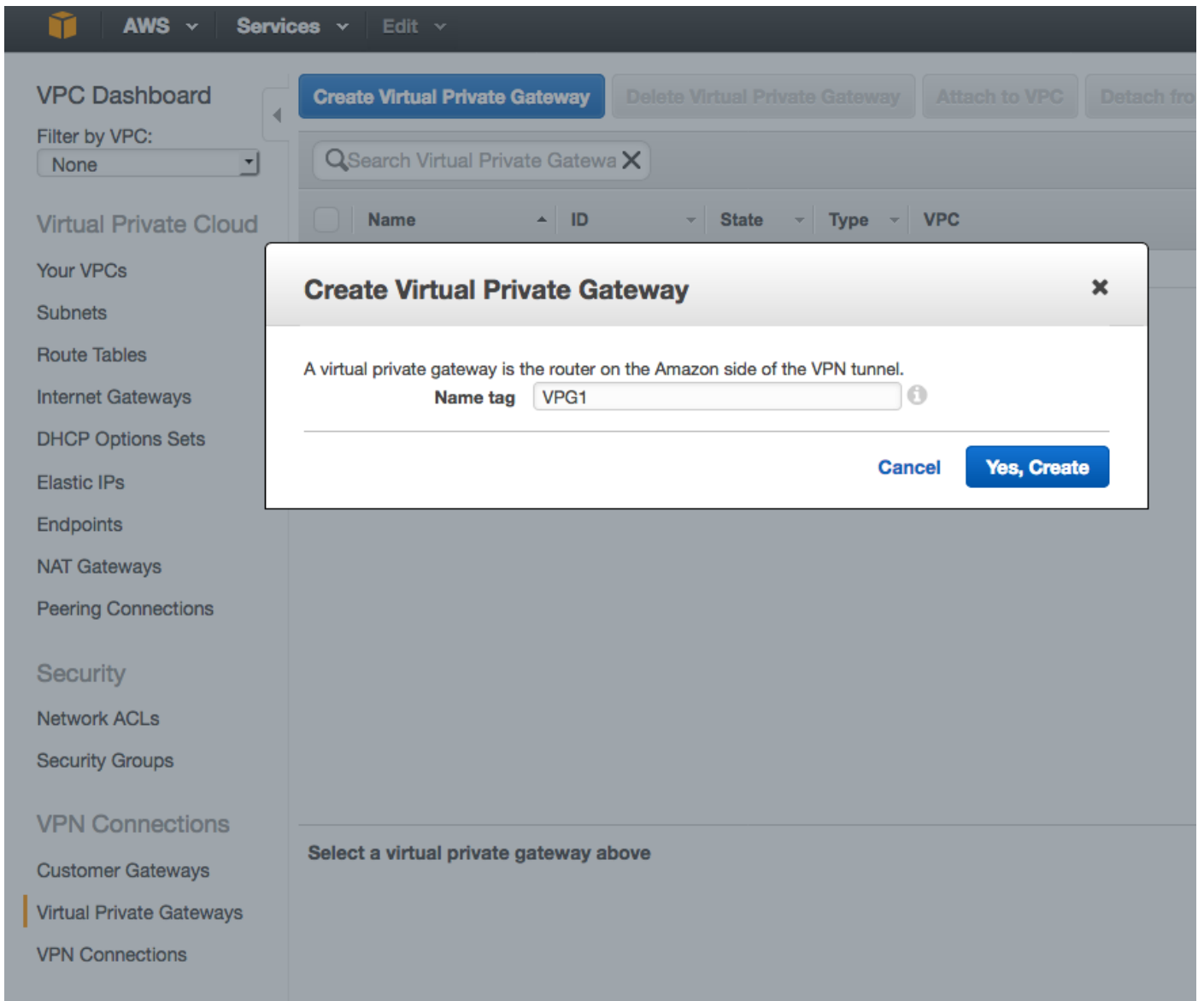
ID	Name	State	Type	IP Address	BGP ASN	VPC
cgw-b778a1a9	(64.100.251.37)	deleted	ipsec.1	64.100.251.37	65000	

ステップ 4 :

仮想プライベートゲートウェイ(VPG)を作成します。これは、IPSecトンネルを終端するAWSでホストされるシミュレートされたルータです。

フィールド値

名前タグ VPGを認識するための人間が読み取り可能な名前。



ステップ 5 :

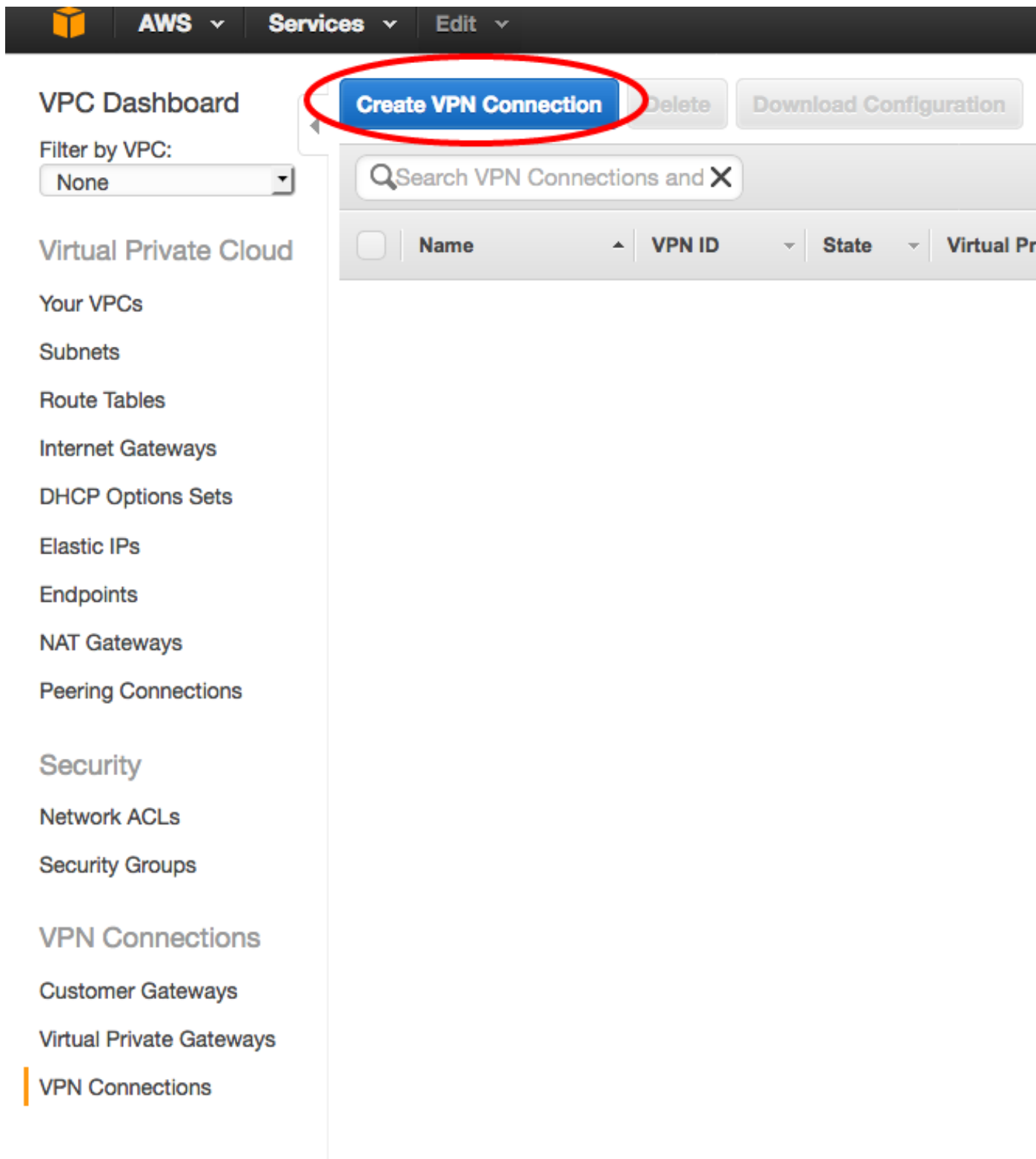
VPGをVPCに接続します。

仮想プライベートゲートウェイを選択し、[Attach to VPC]をクリックし、[VPC]ドロップダウンリストからVPCを選択し、[Yes, Attach]をクリックします。

The screenshot displays the AWS Management Console interface for Virtual Private Gateways. At the top, there are buttons for 'Create Virtual Private Gateway', 'Delete Virtual Private Gateway', 'Attach to VPC', and 'Detach from VPC'. Below these is a search bar and a table of Virtual Private Gateways. The table has columns for Name, ID, State, Type, and VPC. One gateway, 'VPG1' with ID 'vgw-18954d06', is in a 'detached' state and is highlighted with a red circle. A modal dialog box titled 'Attach to VPC' is open, showing a dropdown menu for selecting a VPC. The selected VPC is 'vpc-e1e00786 (172.31.0.0/16)'. The dialog has 'Cancel' and 'Yes, Attach' buttons. A red arrow points from the 'Attach to VPC' button in the top navigation to the 'Yes, Attach' button in the dialog. Below the dialog, the details for the selected gateway are shown: 'vgw-18954d06 | VPG1', with tabs for 'Summary' and 'Tags'. The summary shows: ID: vgw-18954d06 | VPG1, State: detached, Type: ipsec.1, and VPC: (empty).

手順 6 :

VPN接続を作成します。



フィールド

名前タグ

仮想プライベートゲートウェイ

カスタマーゲートウェイ

ルーティングオプション

値

AWSとASA間のVPN接続の人間が読み取り可能なタグ。

作成したばかりのVPGを選択します。

[Existing]ラジオボタンをクリックし、ASAのゲートウェイを選択します。

[Dynamic (requires BGP)]オプションボタンをクリックします。

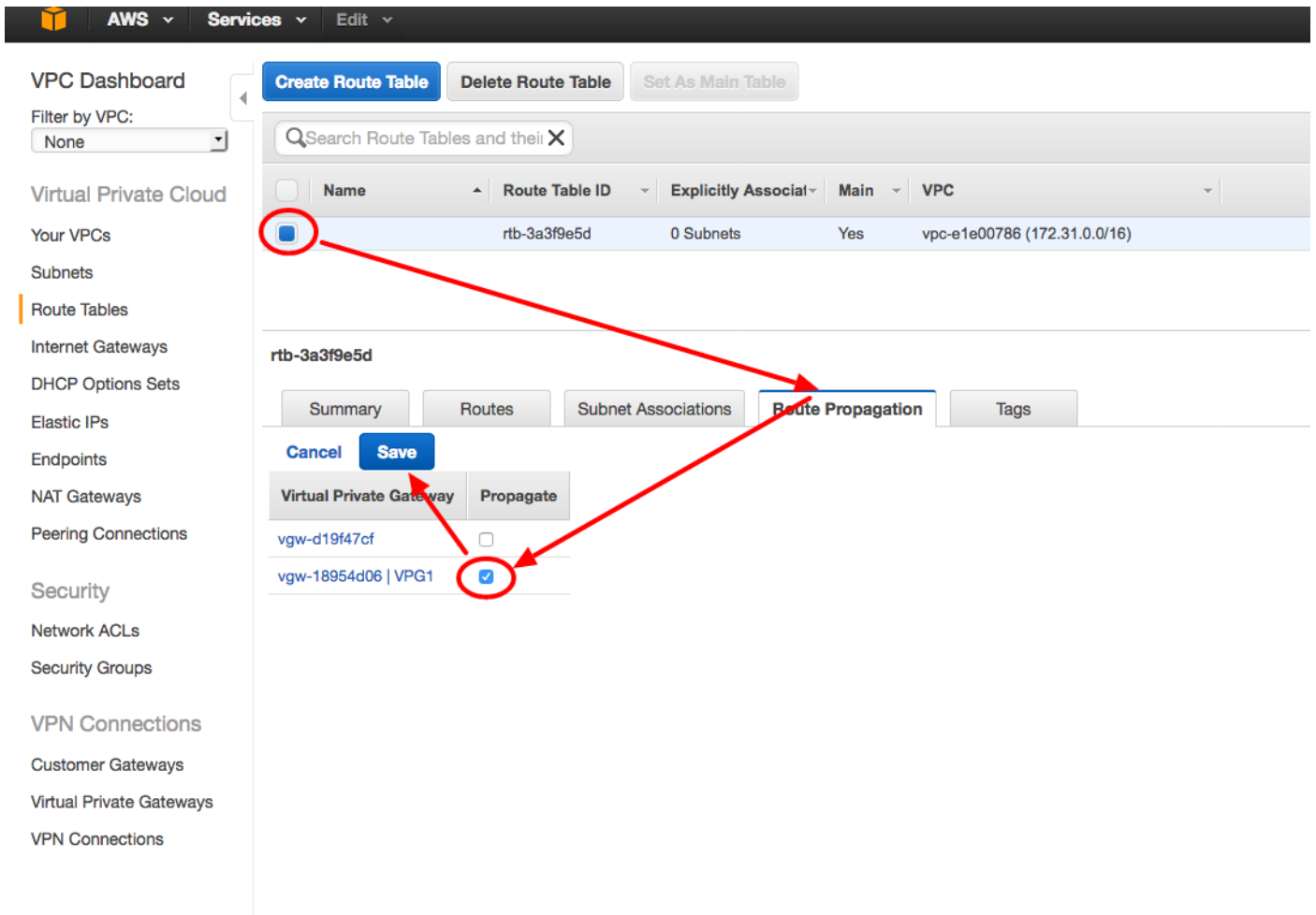
The screenshot shows the AWS Management Console interface for creating a VPN connection. The left sidebar contains navigation options like 'VPC Dashboard', 'Virtual Private Cloud', and 'VPN Connections'. The main area displays a 'Create VPN Connection' dialog box with the following details:

- Name tag:** VPNtoASA
- Virtual Private Gateway:** vgw-18954d06 | VPG1
- Customer Gateway:** Existing (Selected), New. Selected: cgw-837fa69d (64.100.251.37) | ASAVTI
- Routing Options:** Dynamic (requires BGP) (Selected), Static

Buttons at the bottom of the dialog are 'Cancel' and 'Yes, Create'.

手順 7 :

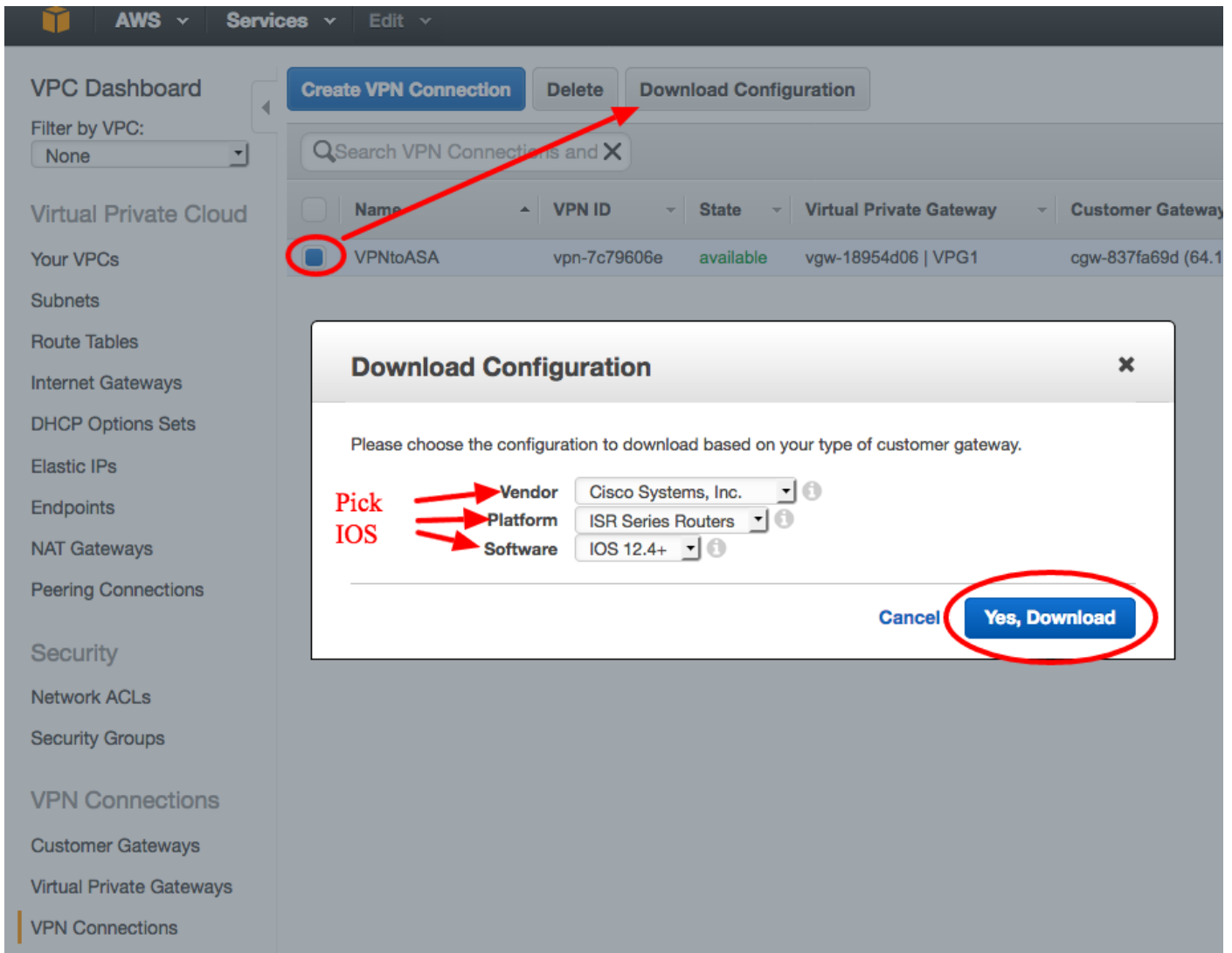
VPGから (BGP経由で) 学習したルートをVPCに伝搬するようにルートテーブルを設定します。



ステップ 8 :

推奨設定をダウンロードします。VTIスタイルの設定である設定を生成するには、次の値を選択します。

フィールド	値
ベンダー	Cisco Systems, Inc.
Platform	ISRシリーズルータ
[ソフトウェア (Software)]	IOS 12.4+



ASA の設定

設定をダウンロードしたら、いくつかの変換が必要になります。

ステップ 1 :

crypto isakmp policy to crypto ikev1 policy。 ポリシー200とポリシー201は同じであるため、必要なポリシーは1つだけです。

推奨設定

```
crypto isakmp policy 200
  encryption aes 128
  authentication pre-share
  group 2
  lifetime 28800
  hash sha
exit
crypto isakmp policy 201
  encryption aes 128
  authentication pre-share
  group 2
```

変更後

```
crypto ikev1 enable outside
crypto ikev1 policy 10
  authentication pre-share
  encryption aes
  hash sha
  group 2
  lifetime 28800
```

```
lifetime 28800
hash sha
exit
```

ステップ 2 :

crypto ipsec transform-set to crypto ipsec ikev1 transform-set。 2つのトランスフォームセットが同じであるため、必要なトランスフォームセットは1つだけです。

推奨設定

```
crypto ipsec transform-set ipsec-prop-vpn-
7c79606e-0 esp-aes 128 esp-sha-hmac
```

```
exit
crypto ipsec transform-set ipsec-prop-vpn-
7c79606e-1 esp-aes 128 esp-sha-hmac
```

```
exit
```

変更後

```
crypto ipsec ikev1 transform-
AWS esp-aes esp-sha-hmac
```

ステップ 3 :

crypto ipsec profile to crypto ipsec profile。 2つのプロファイルが同一であるため、必要なプロファイルは1つだけです。

推奨設定

```
crypto ipsec profile ipsec-vpn-7c79606e-0
set pfs group2
set security-association lifetime seconds
3600
set transform-set ipsec-prop-vpn-7c79606e-0
exit
crypto ipsec profile ipsec-vpn-7c79606e-1
set pfs group2
set security-association lifetime seconds
3600
set transform-set ipsec-prop-vpn-7c79606e-1
exit
```

変更後

```
crypto ipsec profile AWS
set ikev1 transform-set AWS
set pfs group2
set security-association lifet
seconds 3600
```

ステップ 4 :

crypto keyringおよびcrypto isakmp profileは、トンネルごとにtunnel-groupに変換する必要があります。

推奨設定

```
crypto keyring keyring-vpn-7c79606e-0
local-address 64.100.251.37
pre-shared-key address 52.34.205.227 key QZhh90Bjf
exit
!
crypto isakmp profile isakmp-vpn-7c79606e-0
local-address 64.100.251.37
match identity address 52.34.205.227
keyring keyring-vpn-7c79606e-0
exit
```

変更後

```
tunnel-group
52.34.205.227 type
ipsec-l2l
tunnel-group
52.34.205.227 ipsec-
attributes
ikev1 pre-shared-ke
QZhh90Bjf
isakmp keepalive
threshold 10 retry 1
```

```

!
crypto keyring keyring-vpn-7c79606e-1
  local-address 64.100.251.37
  pre-shared-key address 52.37.194.219 key JjxCWy4Ae
  exit
!
crypto isakmp profile isakmp-vpn-7c79606e-1
  local-address 64.100.251.37
  match identity address 52.37.194.219
  keyring keyring-vpn-7c79606e-1
  exit
tunnel-group
52.37.194.219 type
ipsec-l2l
tunnel-group
52.37.194.219 ipsec-
attributes
ikev1JjxCWy4Ae
isakmp keepalive
threshold 10 retry 1

```

ステップ 5 :

トンネル設定はほぼ同じです。ASAでは、`ip tcp adjust-mss`コマンドまたは`ip virtual-reassembly`コマンドはサポートされていません。

推奨設定

```

interface Tunnel1
  ip address 169.254.13.190 255.255.255.252
  ip virtual-reassembly
  tunnel source 64.100.251.37
  tunnel destination 52.34.205.227
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile ipsec-vpn-
7c79606e-0
  ip tcp adjust-mss 1387
  no shutdown
  exit
!
interface Tunnel2
  ip address 169.254.12.86 255.255.255.252
  ip virtual-reassembly
  tunnel source 64.100.251.37
  tunnel destination 52.37.194.219
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile ipsec-vpn-
7c79606e-1
  ip tcp adjust-mss 1387
  no shutdown
  exit

```

変更後

```

interface Tunnel1
  nameif AWS1
  ip address 169.254.13.190
255.255.255.252
  tunnel source interface outside
  tunnel destination 52.34.205.2
  tunnel mode ipsec ipv4
  tunnel protection ipsec profil
AWS
!
interface Tunnel2
  nameif AWS2
  ip address 169.254.12.86
255.255.255.252
  tunnel source interface outside
  tunnel destination 52.37.194.2
  tunnel mode ipsec ipv4
  tunnel protection ipsec profil
AWS

```

手順 6 :

この例では、ASAは内部サブネット(192.168.1.0/24)のみをアドバタイズし、AWS(172.31.0.0/16)内のサブネットを受信します。

推奨設定

```

router bgp 65000
  neighbor 169.254.13.189 remote-as 7224
  neighbor 169.254.13.189 activate
  neighbor 169.254.13.189 timers 10 30 30
  address-family ipv4 unicast
  neighbor 169.254.13.189 remote-as 7224

```

変更後

```

router bgp 65000
  bgp log-neighbor-changes
  timers bgp 10 30 0
  address-family ipv4 unicast
  neighbor 169.254.12.85
remote-as 7224

```

```

neighbor 169.254.13.189 timers 10 30 30
neighbor 169.254.13.189 default-originate
neighbor 169.254.13.189 activate
neighbor 169.254.13.189 soft-reconfiguration
inbound
  network 0.0.0.0
  exit
exit
router bgp 65000
  neighbor 169.254.12.85 remote-as 7224
  neighbor 169.254.12.85 activate
  neighbor 169.254.12.85 timers 10 30 30
  address-family ipv4 unicast
    neighbor 169.254.12.85 remote-as 7224
    neighbor 169.254.12.85 timers 10 30 30
    neighbor 169.254.12.85 default-originate
    neighbor 169.254.12.85 activate
    neighbor 169.254.12.85 soft-reconfiguration
  inbound
    network 0.0.0.0
    exit
  exit
neighbor 169.254.12.85
activate
neighbor 169.254.13.189
remote-as 7224
neighbor 169.254.13.189
activate
network 192.168.1.0
no auto-summary
no synchronization
exit-address-family

```

検証と最適化

ステップ 1:

ASAがAWSの2つのエンドポイントとのIKEv1セキュリティアソシエーションを確立したことを確認します。SAの状態はMM_ACTIVEである必要があります。

```
ASA# show crypto ikev1 sa
```

```
IKEv1 SAs:
```

```

Active SA: 2
Rekey SA: 0 (A tunnel will report 1 Active and 1 Rekey SA during rekey)
Total IKE SA: 2

```

```

1  IKE Peer: 52.37.194.219
   Type    : L2L           Role    : initiator
   Rekey   : no           State   : MM_ACTIVE
2  IKE Peer: 52.34.205.227
   Type    : L2L           Role    : initiator
   Rekey   : no           State   : MM_ACTIVE

```

```
ASA#
```

ステップ 2:

IPsec SAがASAにインストールされていることを確認します。各ピアに対してインバウンドおよびアウトバウンドSPIがインストールされている必要があります、一部のエンキャプおよびデカンプカウンタが増加している必要があります。

```
ASA# show crypto ipsec sa
```

interface: AWS1

Crypto map tag: __vti-crypto-map-5-0-1, seq num: 65280, local addr: 64.100.251.37

access-list __vti-def-acl-0 extended permit ip any any
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
current_peer: 52.34.205.227

#pkts encaps: 2234, #pkts encrypt: 2234, #pkts digest: 2234
#pkts decaps: 1234, #pkts decrypt: 1234, #pkts verify: 1234
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 2234, #pkts comp failed: 0, #pkts decomp failed: 0
#pre-frag successes: 0, #pre-frag failures: 0, #fragments created: 0
#PMTUs sent: 0, #PMTUs rcvd: 0, #decapsulated frgs needing reassembly: 0
#TFC rcvd: 0, #TFC sent: 0
#Valid ICMP Errors rcvd: 0, #Invalid ICMP Errors rcvd: 0
#send errors: 0, #recv errors: 0

local crypto endpt.: 64.100.251.37/4500, remote crypto endpt.: 52.34.205.227/4500
path mtu 1500, ipsec overhead 82(52), media mtu 1500
PMTU time remaining (sec): 0, DF policy: copy-df
ICMP error validation: disabled, TFC packets: disabled
current outbound spi: 874FCCF3
current inbound spi : 5E653906

inbound esp sas:

spi: 0x5E653906 (1583692038)
transform: esp-aes esp-sha-hmac no compression
in use settings ={L2L, Tunnel, NAT-T-Encaps, PFS Group 2, IKEv1, VTI, }
slot: 0, conn_id: 73728, crypto-map: __vti-crypto-map-5-0-1
sa timing: remaining key lifetime (kB/sec): (4373986/2384)
IV size: 16 bytes
replay detection support: Y
Anti replay bitmap:
0xFFFFFFFF 0xFFFFFFFF

outbound esp sas:

spi: 0x874FCCF3 (2270153971)
transform: esp-aes esp-sha-hmac no compression
in use settings ={L2L, Tunnel, NAT-T-Encaps, PFS Group 2, IKEv1, VTI, }
slot: 0, conn_id: 73728, crypto-map: __vti-crypto-map-5-0-1
sa timing: remaining key lifetime (kB/sec): (4373986/2384)
IV size: 16 bytes
replay detection support: Y
Anti replay bitmap:
0x00000000 0x00000001

interface: AWS2

Crypto map tag: __vti-crypto-map-6-0-2, seq num: 65280, local addr: 64.100.251.37

access-list __vti-def-acl-0 extended permit ip any any
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
current_peer: 52.37.194.219

#pkts encaps: 1230, #pkts encrypt: 1230, #pkts digest: 1230
#pkts decaps: 1230, #pkts decrypt: 1230, #pkts verify: 1230
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 1230, #pkts comp failed: 0, #pkts decomp failed: 0
#pre-frag successes: 0, #pre-frag failures: 0, #fragments created: 0
#PMTUs sent: 0, #PMTUs rcvd: 0, #decapsulated frgs needing reassembly: 0
#TFC rcvd: 0, #TFC sent: 0
#Valid ICMP Errors rcvd: 0, #Invalid ICMP Errors rcvd: 0

```
#send errors: 0, #recv errors: 0
```

```
local crypto endpt.: 64.100.251.37/4500, remote crypto endpt.: 52.37.194.219/4500
path mtu 1500, ipsec overhead 82(52), media mtu 1500
PMTU time remaining (sec): 0, DF policy: copy-df
ICMP error validation: disabled, TFC packets: disabled
current outbound spi: DC5E3CA8
current inbound spi : CB6647F6
```

```
inbound esp sas:
```

```
spi: 0xCB6647F6 (3412477942)
transform: esp-aes esp-sha-hmac no compression
in use settings ={L2L, Tunnel, NAT-T-Encaps, PFS Group 2, IKEv1, VTI, }
slot: 0, conn_id: 77824, crypto-map: __vti-crypto-map-6-0-2
sa timing: remaining key lifetime (kB/sec): (4373971/1044)
IV size: 16 bytes
replay detection support: Y
Anti replay bitmap:
0xFFFFFFFF 0xFFFFFFFF
```

```
outbound esp sas:
```

```
spi: 0xDC5E3CA8 (3697163432)
transform: esp-aes esp-sha-hmac no compression
in use settings ={L2L, Tunnel, NAT-T-Encaps, PFS Group 2, IKEv1, VTI, }
slot: 0, conn_id: 77824, crypto-map: __vti-crypto-map-6-0-2
sa timing: remaining key lifetime (kB/sec): (4373971/1044)
IV size: 16 bytes
replay detection support: Y
Anti replay bitmap:
0x00000000 0x00000001
```

ステップ 3 :

ASAで、BGP接続がAWSと確立されていることを確認します。 AWSがASAに対して172.31.0.0/16サブネットをアドバタイズするため、State/PfxRcdカウンタは1である必要があります。

```
ASA# show bgp summary
```

```
BGP router identifier 192.168.1.55, local AS number 65000
BGP table version is 5, main routing table version 5
2 network entries using 400 bytes of memory
3 path entries using 240 bytes of memory
3/2 BGP path/bestpath attribute entries using 624 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 1288 total bytes of memory
BGP activity 3/1 prefixes, 4/1 paths, scan interval 60 secs
```

Neighbor	V	AS	MsgRcvd	MsgSent	TblVer	InQ	OutQ	Up/Down	State/PfxRcd
169.254.12.85	4	7224	1332	1161	5	0	0	03:41:31	1
169.254.13.189	4	7224	1335	1164	5	0	0	03:42:02	1

ステップ 4 :

ASAで、172.31.0.0/16へのルートがトンネルインターフェイス経由で学習されたことを確認します。 この出力は、ピア169.254.12.85と169.254.13.189から172.31.0.0へのパスが2つあることを示しています。 Tunnel 2(AWS2)から169.254.13.189へのパスは、メトリックが低いため優先されます。

```
ASA# show bgp
```

```
BGP table version is 5, local router ID is 192.168.1.55
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network	Next Hop	Metric	LocPrf	Weight	Path
* 172.31.0.0	169.254.12.85	200		0	7224 i
*>	169.254.13.189	100		0	7224 i
*> 192.168.1.0	0.0.0.0	0		32768	i

```
ASA# show route
```

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, V - VPN
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, + - replicated route
Gateway of last resort is 64.100.251.33 to network 0.0.0.0
```

```
S* 0.0.0.0 0.0.0.0 [1/0] via 64.100.251.33, outside
C 64.100.251.32 255.255.255.224 is directly connected, outside
L 64.100.251.37 255.255.255.255 is directly connected, outside
C 169.254.12.84 255.255.255.252 is directly connected, AWS2
L 169.254.12.86 255.255.255.255 is directly connected, AWS2
C 169.254.13.188 255.255.255.252 is directly connected, AWS1
L 169.254.13.190 255.255.255.255 is directly connected, AWS1
B 172.31.0.0 255.255.0.0 [20/100] via 169.254.13.189, 03:52:55
C 192.168.1.0 255.255.255.0 is directly connected, inside
L 192.168.1.55 255.255.255.255 is directly connected, inside
```

ステップ 5 :

AWSから戻るトラフィックが対称パスに従うことを確認するには、優先パスに一致するようにルートマップを設定し、アドバタイズされたルートを変更するようにBGPを調整します。

```
route-map toAWS1 permit 10
  set metric 100
  exit
!
route-map toAWS2 permit 10
  set metric 200
  exit
!
router bgp 65000
  address-family ipv4 unicast
    neighbor 169.254.12.85 route-map toAWS2 out
    neighbor 169.254.13.189 route-map toAWS1 out
```

手順 6 :

ASAで、192.168.1.0/24がAWSにアドバタイズされていることを確認します。

```
ASA# show bgp neighbors 169.254.12.85 advertised-routes
```

```

BGP table version is 5, local router ID is 192.168.1.55
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

```

```

      Network      Next Hop      Metric LocPrf Weight Path
*> 172.31.0.0      169.254.13.189      100          0 7224 i
*> 192.168.1.0      0.0.0.0              0          32768 i

```

Total number of prefixes 2

ASA# **show bgp neighbors 169.254.13.189 advertised-routes**

```

BGP table version is 5, local router ID is 192.168.1.55
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
               r RIB-failure, S Stale, m multipath
Origin codes: i - IGP, e - EGP, ? - incomplete

```

```

      Network      Next Hop      Metric LocPrf Weight Path
*> 192.168.1.0      0.0.0.0              0          32768 i

```

Total number of prefixes 1

手順 7 :

AWSで、VPN接続のトンネルがUPであり、ルートがピアから学習されていることを確認します。また、ルートがルーティングテーブルに伝播されていることも確認します。

The screenshot shows the AWS Management Console interface for a VPN connection named 'VPNtoASA'. The 'Tunnel Details' tab is selected, displaying a table with the following data:

VPN Tunnel	IP Address	Status	Status Last Changed	Details
Tunnel 1	52.34.205.227	UP	2016-10-18 14:23 UTC	1 BGP ROUTES
Tunnel 2	52.37.194.219	UP	2016-10-18 14:23 UTC	1 BGP ROUTES



VPC Dashboard

Filter by VPC:

None

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

NAT Gateways

Peering Connections

Security

Network ACLs

Security Groups

VPN Connections

Customer Gateways

Virtual Private Gateways

VPN Connections

Create Route Table

Delete Route Table

Set As Main Table

Search Route Tables and their

<input type="checkbox"/>	Name	Route Table ID	Explicitly Associat	Main	VPC
<input checked="" type="checkbox"/>		rtb-3a3f9e5d	0 Subnets	Yes	vpc-e1e00786 (172.31.0.0/16)

rtb-3a3f9e5d

Summary

Routes

Subnet Associations

Route Propagation

Tags

Edit

Destination	Target	Status	Propagated
172.31.0.0/16	local	Active	No
0.0.0.0/0	igw-e5ad1481	Active	No
192.168.1.0/24	vgw-18954d06	Active	Yes