# AWS 同 AZ 部署 FortiGate HA

| | |
|---|---|
| 版本 | V1.0 |
| 时间 | 2021 年 9 月 |
| 作者 | 王祥 |
| 状态 | |
| 反馈 | support_cn@fortinet.com |

# 目录

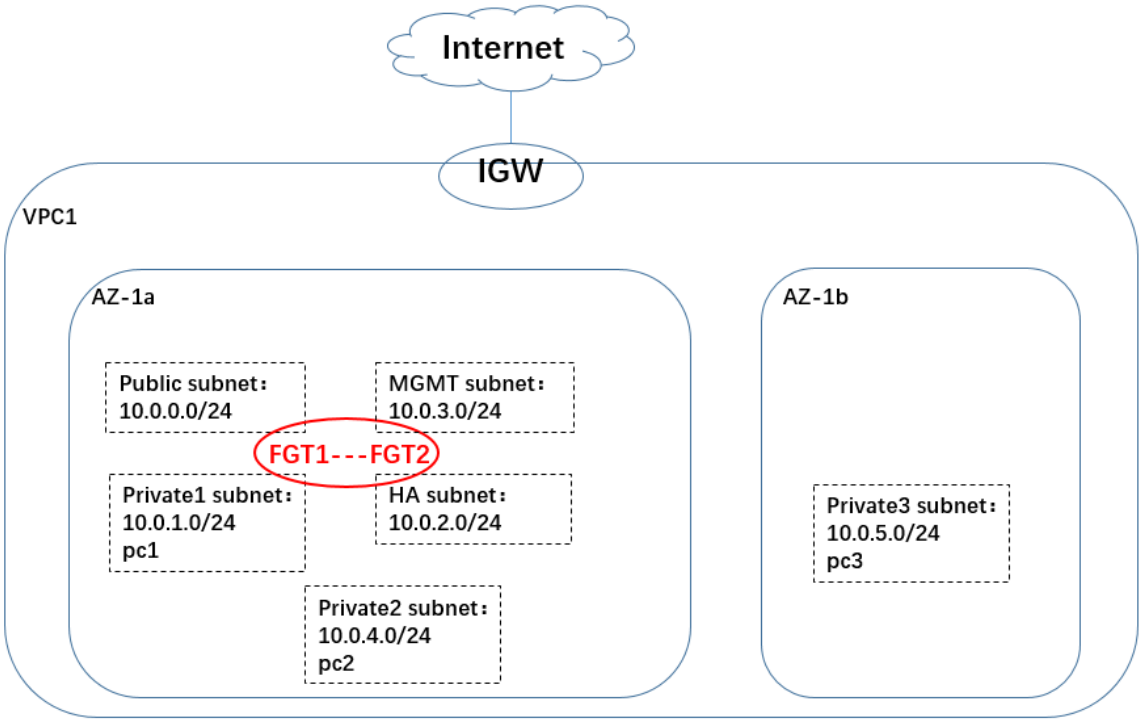
# 1. 介绍

本文档介绍如何在 AWS 同 AZ 部署 FortiGate HA，以提供统一的威胁管理安全解决方案，保护您在 AWS 中的工作负载。

# 2. 网络拓扑

FGT1 和 FGT2 部署在同一 AZ 中，FGT1 和 FGT2 的实例分别需要 4 块网卡。AWS 每种实例类型的最大接口数及每个网络接口的 IP 地址数的查询链接如下：

https://docs.aws.amazon.com/zh_cn/AWSEC2/latest/UserGuide/using-eni.html



# 3. 地址规划

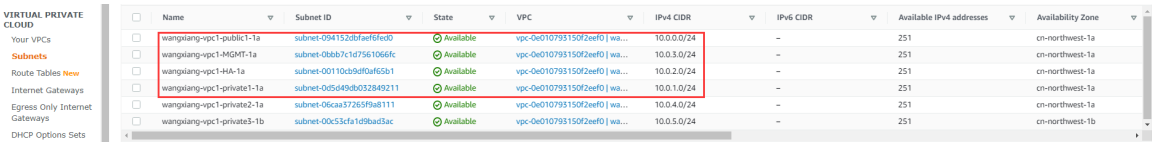
| **FGT1 地址规划（AZ-1a）** | | |
|---|---|---|
| Port | AWS primary address | AWS secondary address |
| Port1 | 10.0.0.11 | 10.0.0.13 |
| Port2 | 10.0.1.11 | 10.0.1.13 |
| Port3 | 10.0.2.11 | NA |
| Port4 | 10.0.3.11 | NA |
| **FGT2 地址规划（AZ-1a）** | | |
| Port | AWS primary address | AWS secondary address |
| Port1 | 10.0.0.12 | NA |
| Port2 | 10.0.1.12 | NA |
| Port3 | 10.0.2.12 | NA |
| Port4 | 10.0.3.12 | NA |
| **测试 PC** | | |
| PC 名称 | 测试地址 | AZ |
| PC1 | 10.0.1.20 | AZ-1a |
| PC2 | 10.0.4.20 | AZ-1a |
| PC3 | 10.0.5.20 | AZ-1b |

# 4. 配置步骤

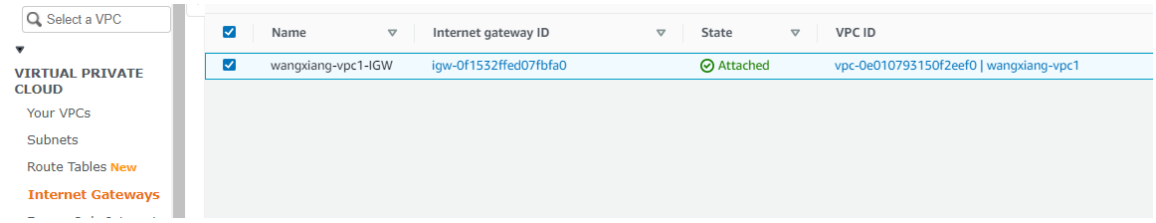
## 4.1. 创建 VPC、子网和 IGW

创建 VPC。


创建子网。

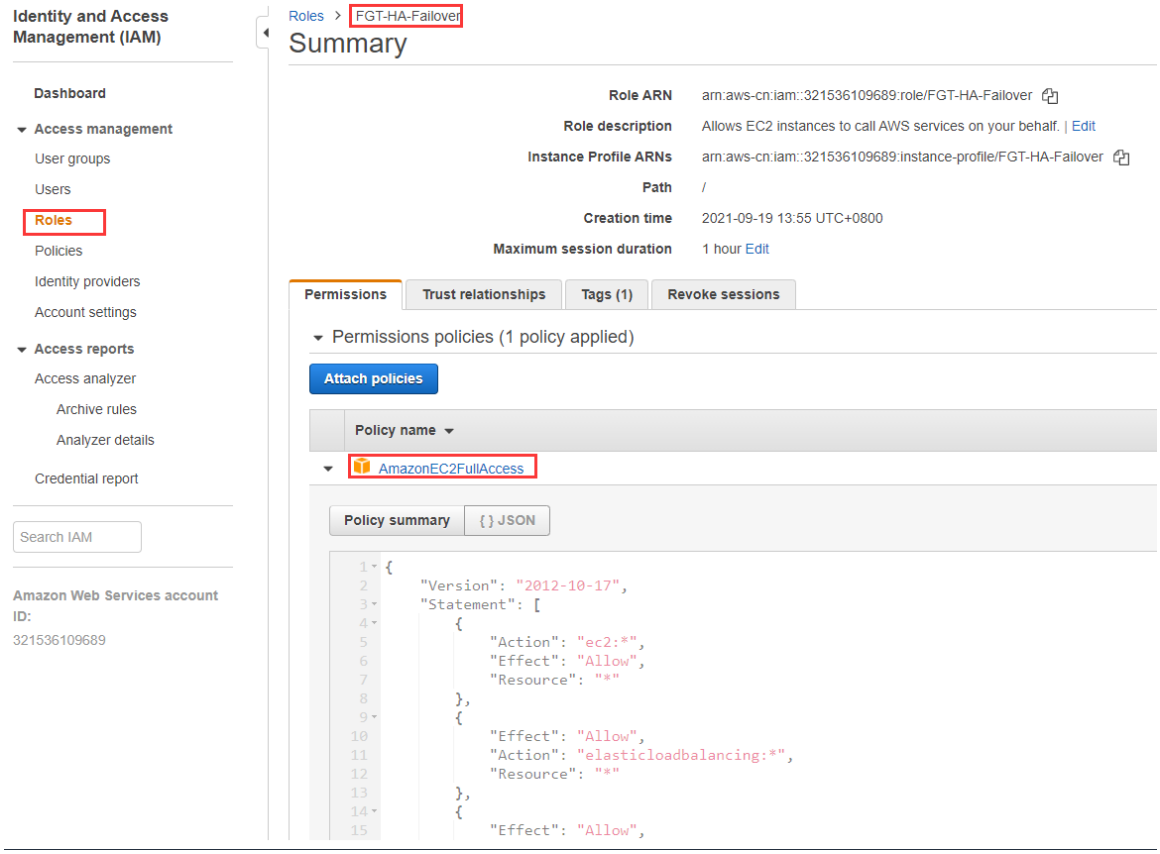



创建 IGW。



## 4.2. 创建 IAM 角色

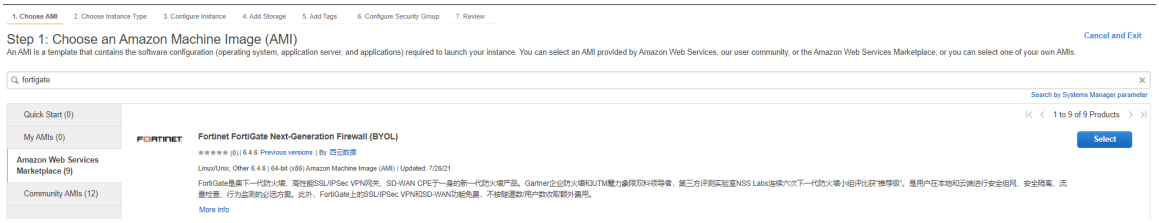
EC2 实例赋予角色后，才能根据权限使用 API 操作和资源。



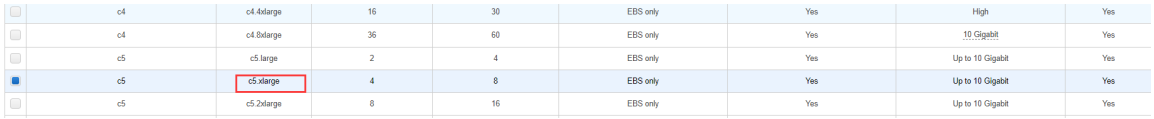
## 4.3. 创建 FortiGate 实例

部署实例时请提前准备好两台 FGT 的 license。

创建 FGT1，选择 FortiGate 镜像。

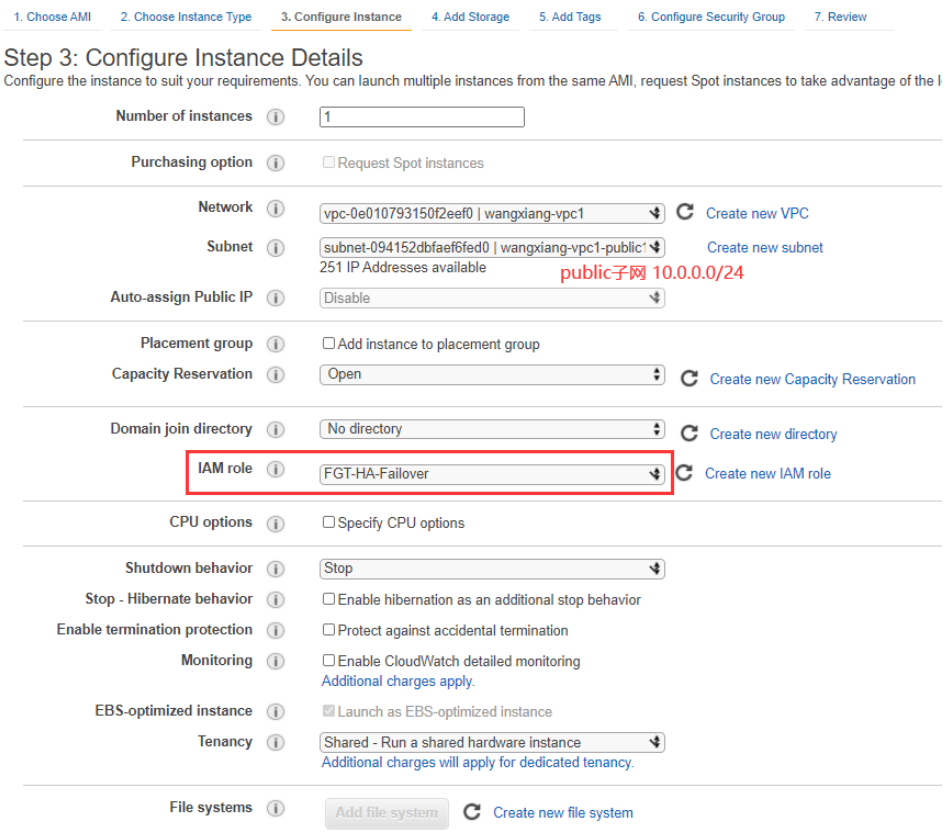



实例类型请选择**计算优化型**，这里使用 c5.xlarge。

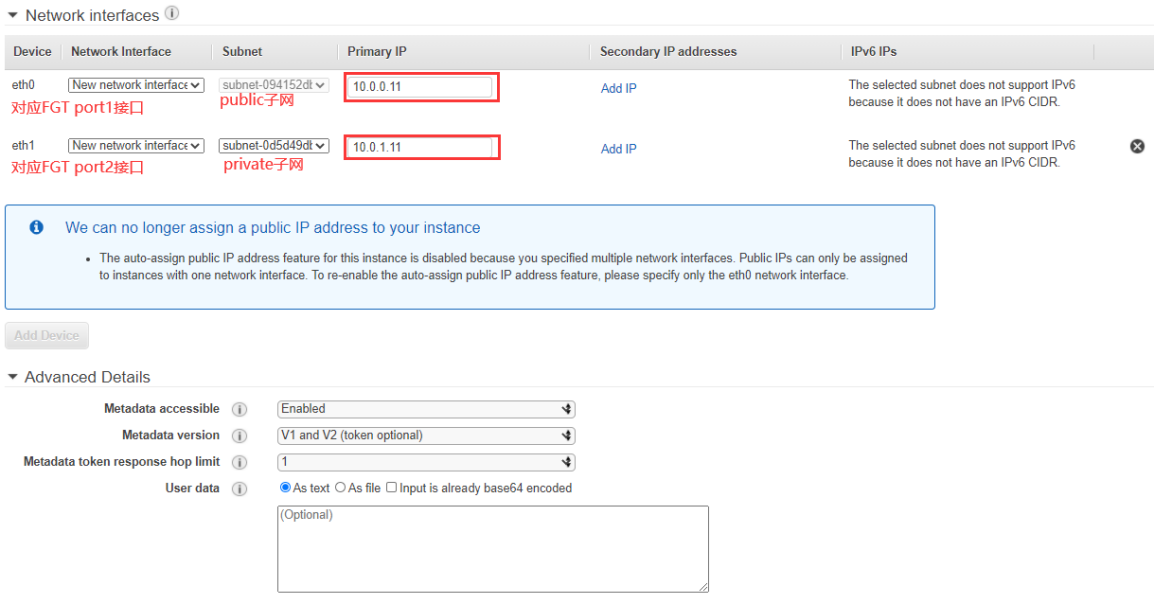


| | | | | | | | |
|---|---|---|---|---|---|---|---|
| c4 | c4.4xlarge | 16 | 30 | EBS only | Yes | High | Yes |
| c4 | c4.8xlarge | 36 | 60 | EBS only | Yes | 10 Gigabit | Yes |
| c5 | c5.large | 2 | 4 | EBS only | Yes | Up to 10 Gigabit | Yes |
| c5 | c5.xlarge | 4 | 8 | EBS only | Yes | Up to 10 Gigabit | Yes |
| c5 | c5.2xlarge | 8 | 16 | EBS only | Yes | Up to 10 Gigabit | Yes |

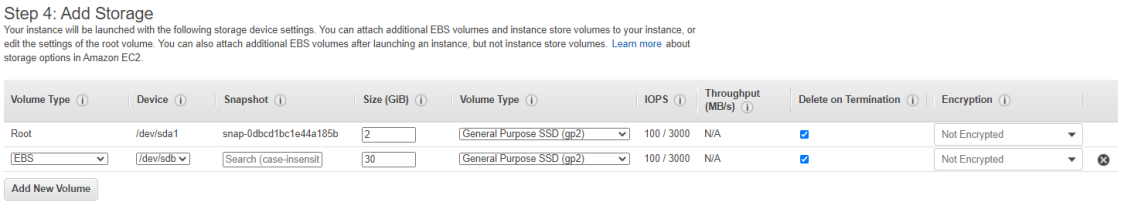
部署实例时关联 IAM 角色。



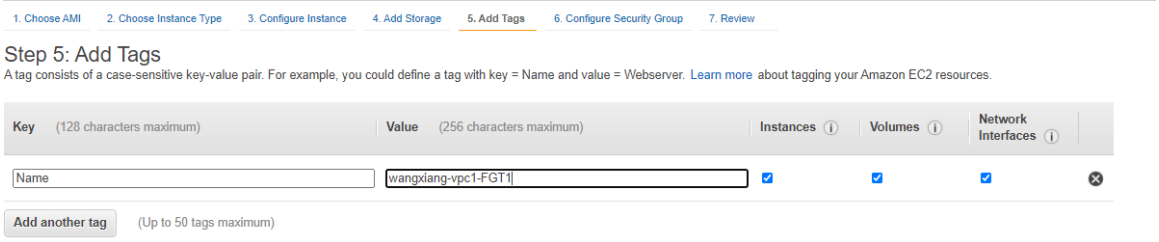
GUI 创建实例时，最多只能创建两块网卡，另外两块需要单独创建。



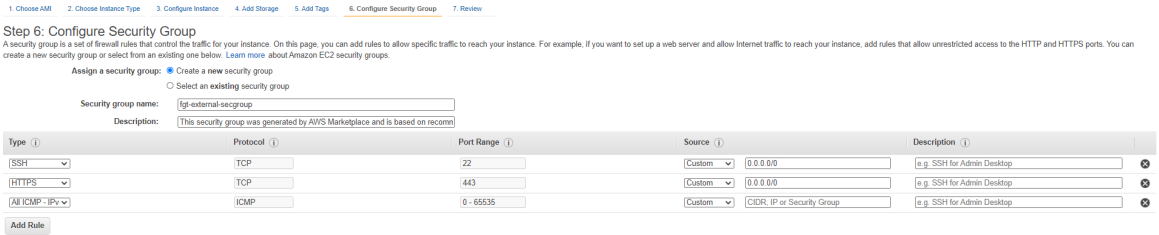

添加存储，第二块磁盘用于记录日志，如果 FGT 需要开启流量日志，建议发送到 FAZ 或者 syslog 服务器。



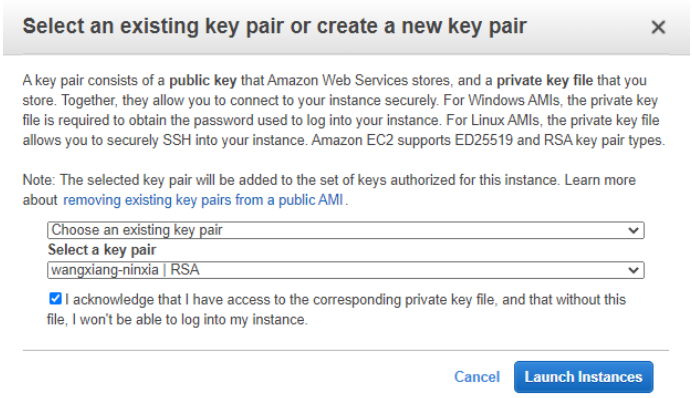

添加标签用于标识一个实例。



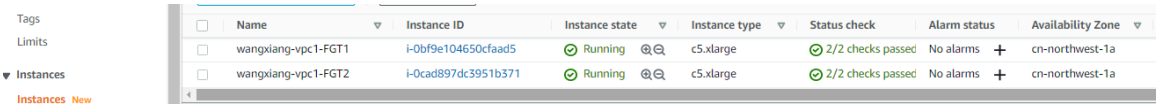

创建安全组 fgt-external-secgroup，对于管理 FortiGate 需要放行 22、443 端口，ICMP，其他端口根据业务需求放行。



创建 key，或者选择自己已经存在的 key，并启动实例。

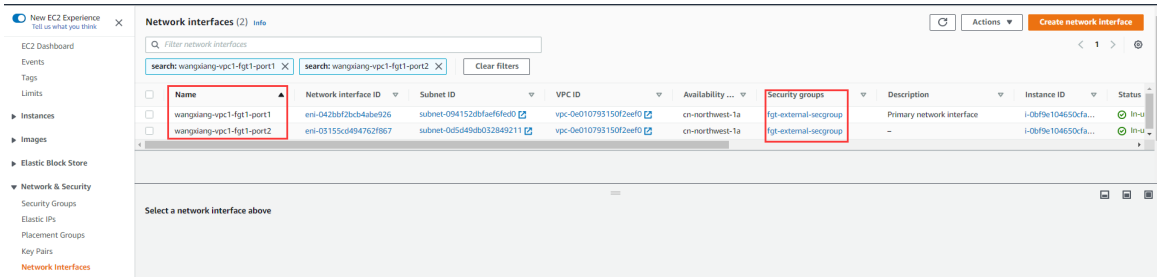



创建完成。FGT2 同理。



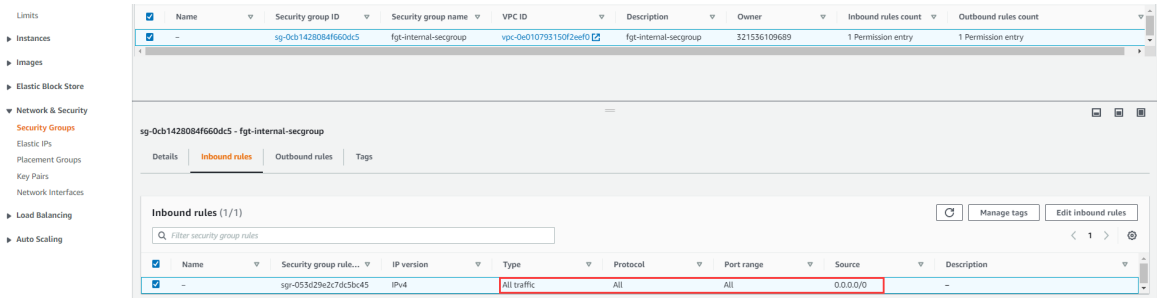
## 4. 4. 安全组

点击网络接口可以查看到 FGT 实例创建的接口，建议给每个接口命名以便查询。FortiGate 实例给其两个接口使用的安全组都是刚新建的 fgt-external-secgroup。

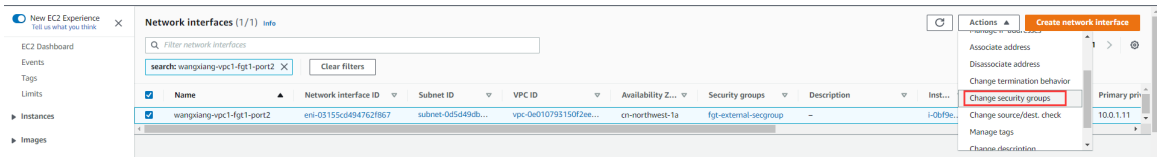


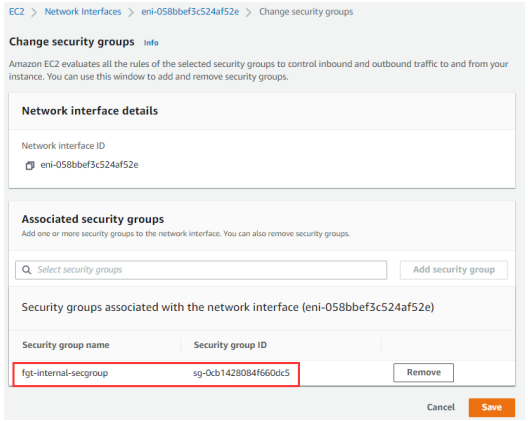

AWS 安全组是基于接口的，对于 FortiGate port2 而言，port2 对应的是由内向外的数据，因此 port2 的安全组要全放通。

新建安全组 fgt-internal-secgroup。

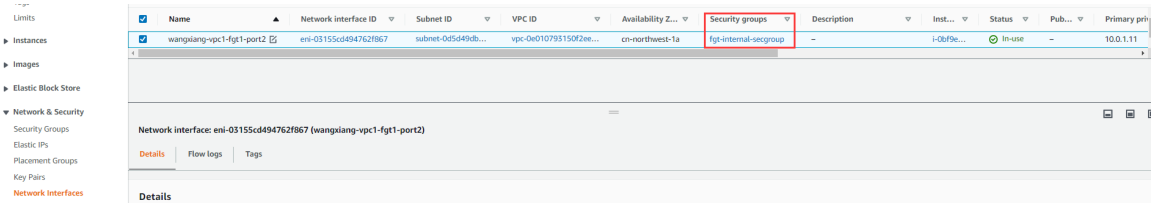



修改 port2 接口的安全组为 fgt-internal-secgroup。

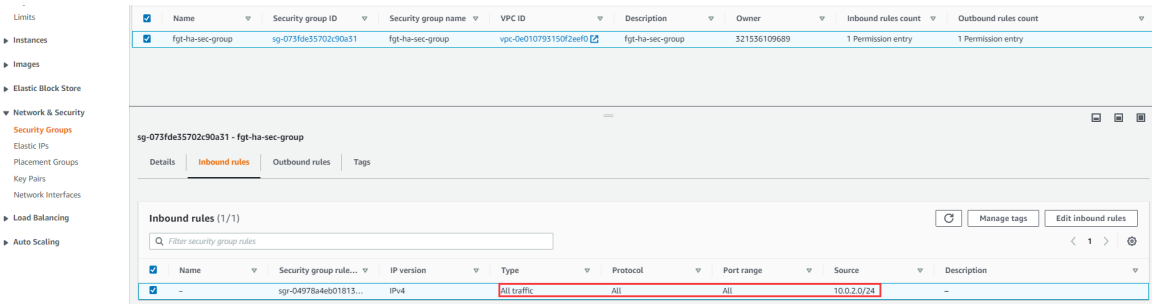


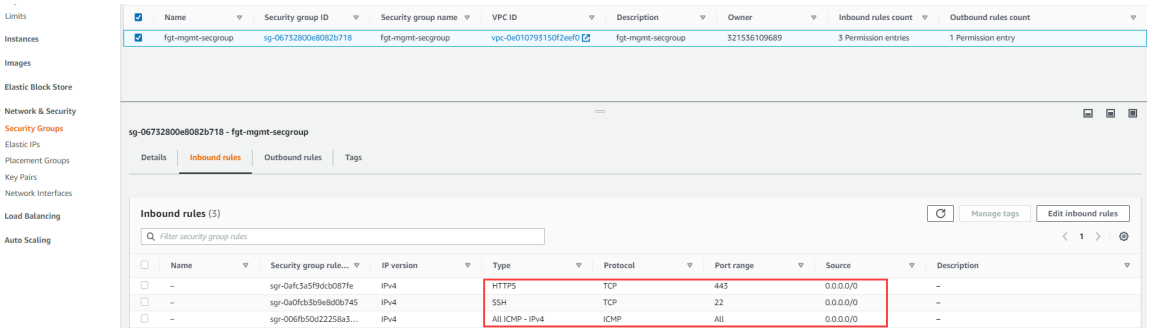
修改完成。




FortiGate 还需要新建两个端口，一个 HA 接口 port3，新建对应的安全组 fgt-ha-secgroup，放通 port3 接口网段（10.0.2.0/24）的所有流量。。



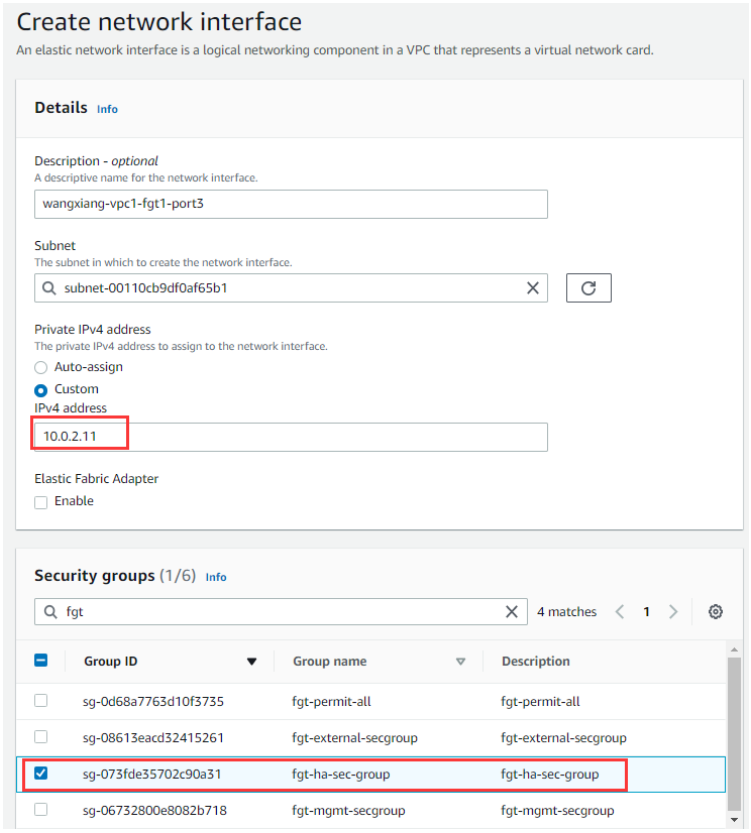

一个MGMT管理口port4，新建对应的安全组fgt-mgmt-secgroup，放通SSH，HTTPS和 ICMP。



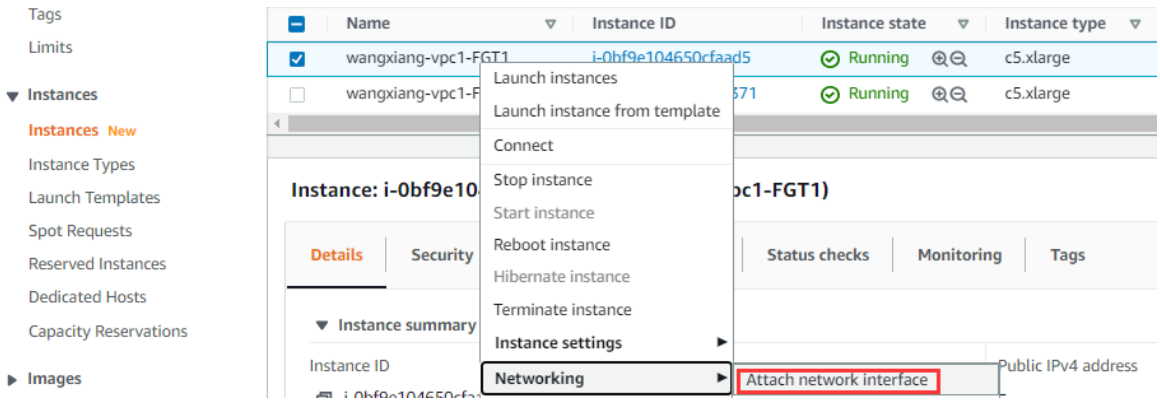

备注：FortiGate 本身就有防火墙策略对流量进行控制，如果嫌安全组控制太麻烦，可以将 FortiGate 的所有端口都应用**允许所有**的安全组。

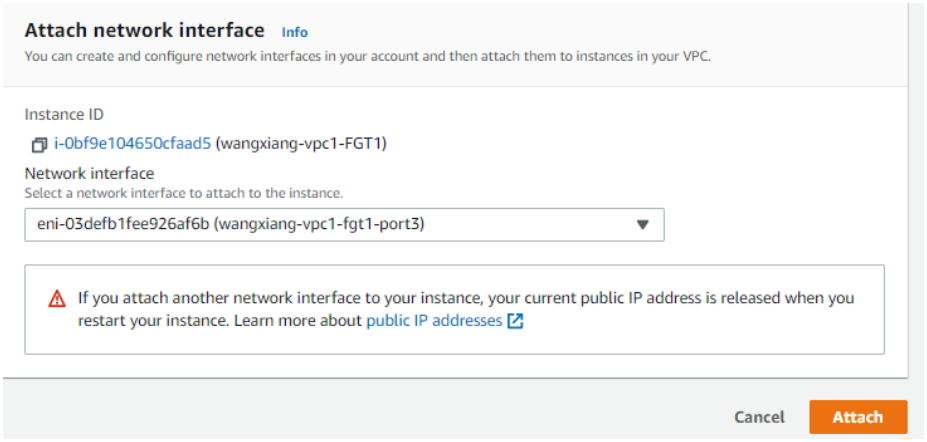
## 4.5. 创建网卡

选择 Services→ Network & Security→ Network interfaces，点击 Create Network interface，创建 FGT1 的 port3 接口，安全组选择 fgt-ha-secgroup，如果是 port4，则选择 fgt-mgmt-secgroup。

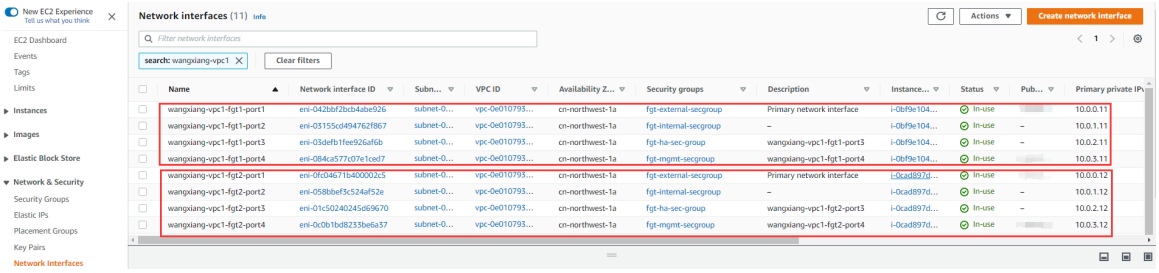



关联网卡到 FGT 实例。请先关联 port3，port3 关联成功后，再关联 port4。



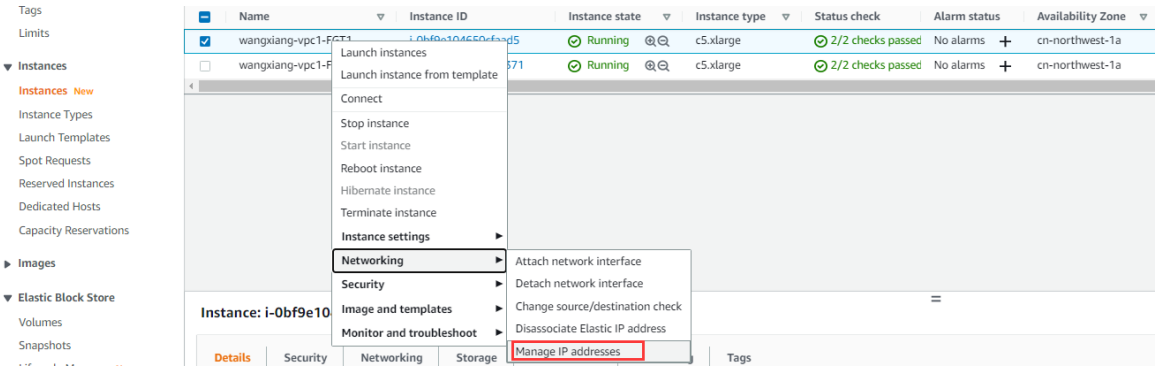

Attach network interface Info
You can create and configure network interfaces in your account and then attach them to instances in your VPC.
Instance ID
i-0bf9e104650cfaad5 (wangxiang-vpc1-FGT1)
Network interface
Select a network interface to attach to the instance.
eni-03defb1fee926af6b (wangxiang-vpc1-fgt1-port3)
If you attach another network interface to your instance, your current public IP address is released when you restart your instance. Learn more about public IP addresses
Cancel
Attach

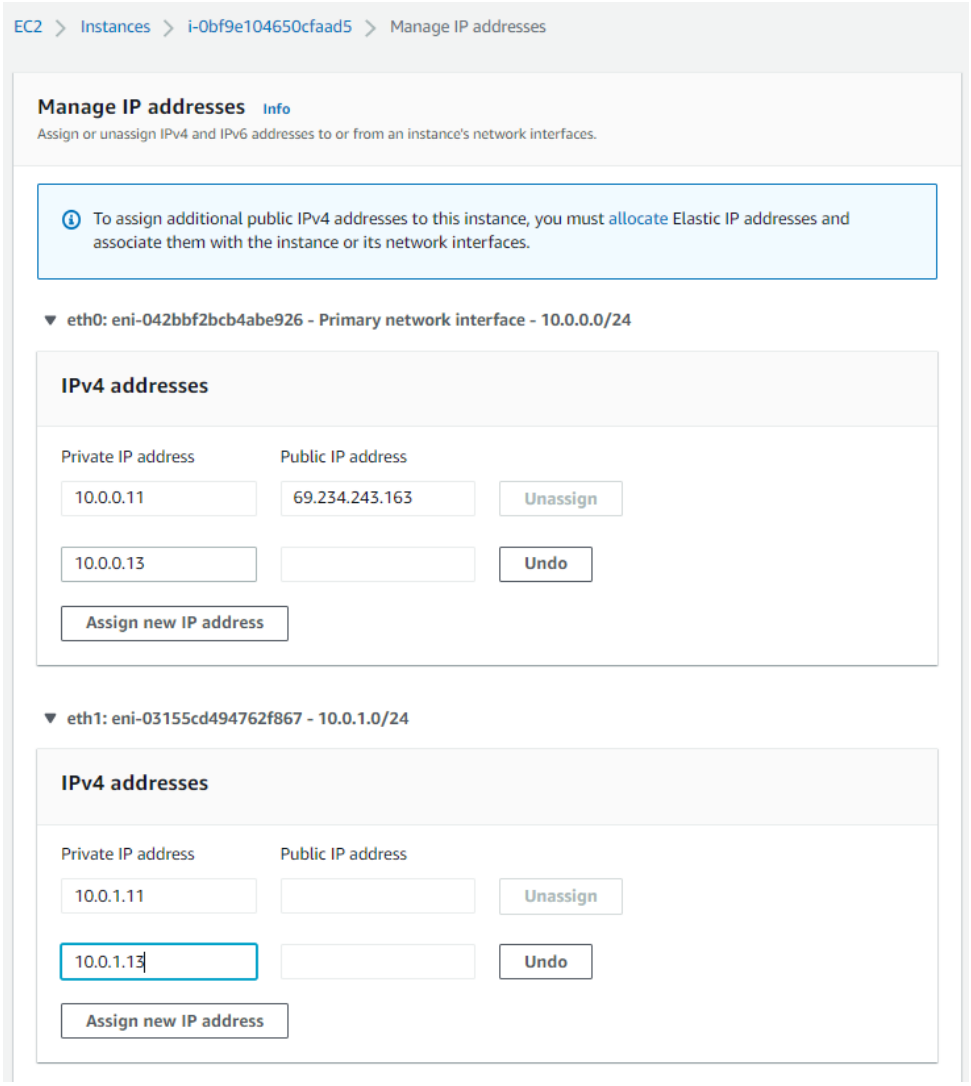
关联完成。


Network interfaces (11) Info
Actions
Create network interface
Clear filters
wangxiang-vpc1-fgt1-port1
wangxiang-vpc1-fgt1-port2
wangxiang-vpc1-fgt1-port3
wangxiang-vpc1-fgt1-port4
wangxiang-vpc1-fgt2-port1
wangxiang-vpc1-fgt2-port2
wangxiang-vpc1-fgt2-port3
wangxiang-vpc1-fgt2-port4

## 4. 6. 实例第二地址

给 FGT1 NIC1 分配第二地址 10.0.0.13，NIC2 分配第二地址 10.0.1.13，这两个地址是用于 FGT HA Cluster 的地址。



## 4.7. 弹性 IP

创建 5 个弹性 IP，最终使用 3 个，另外两个是临时的，HA 形成后可以释放。

**最终使用的 3 个弹性 IP：**

一个弹性 IP 关联 FGT1 实例 NIC4 10.0.3.11，即 FGT1 HA 的独立管理口 port4。

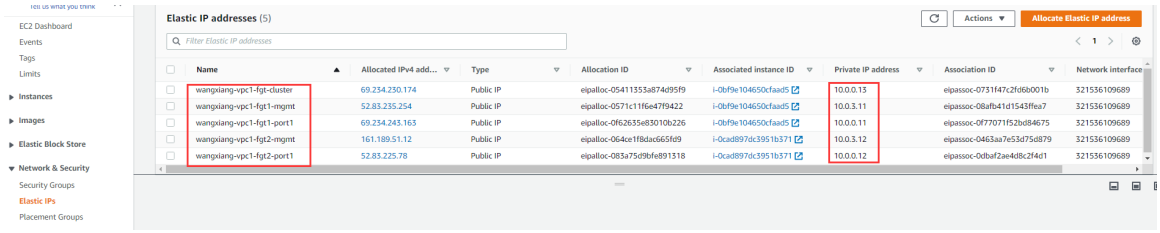
一个弹性 IP 关联 FGT2 实例 NIC4 10.0.3.12，即 FGT2 HA 的独立管理口 port4。

一个弹性 IP 关联 FGT HA 的 Master 实例（当前 FGT1）NIC1 的第二地址 10.0.0.13。

**临时配置 FGT 的 2 个弹性 IP:**

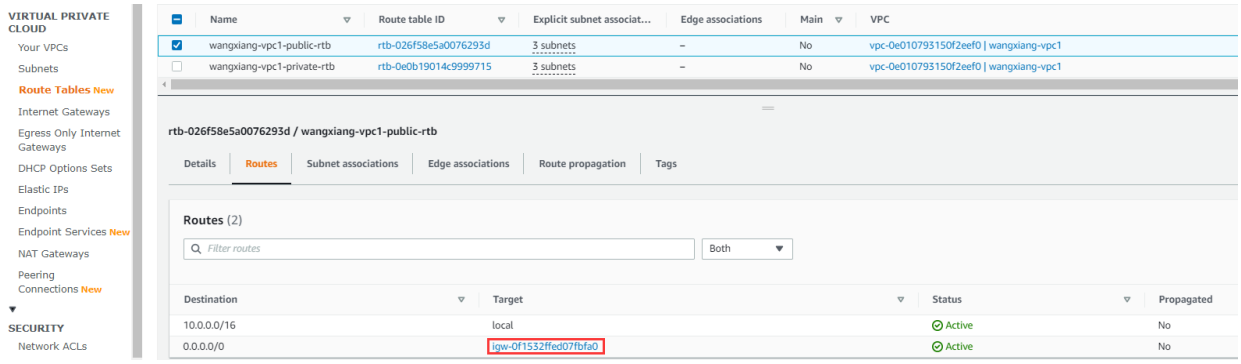
一个弹性 IP 关联 FGT1 实例 NIC1 10.0.0.11，即 FGT1 port1 接口。

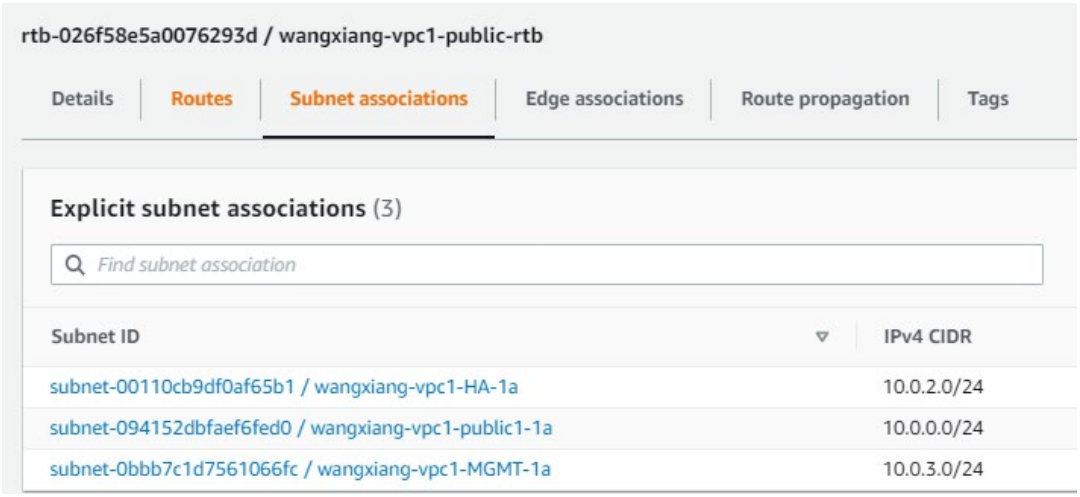
一个弹性 IP 关联 FGT2 实例 NIC1 10.0.0.12，即 FGT2 port1 接口。



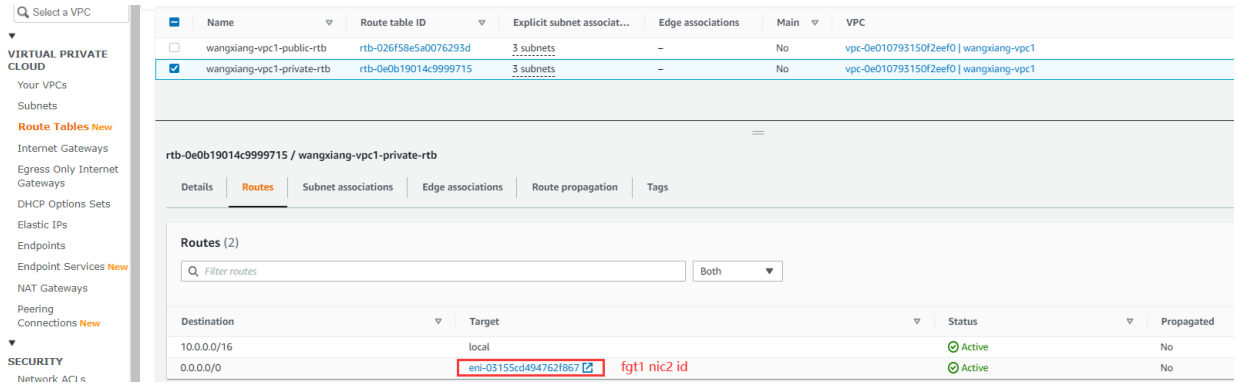
## 4.8. 配置 VPC 路由表

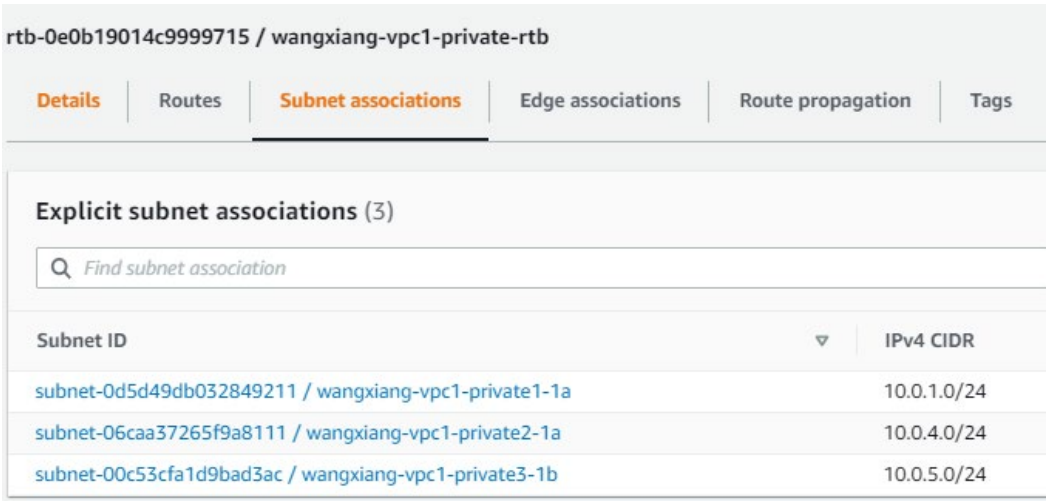
Public 路由表默认路由指向 IGW，关联 public-1a（10.0.0.0/24），ha（10.0.2.0/24），mgmt（10.0.3.0/24）三个子网。



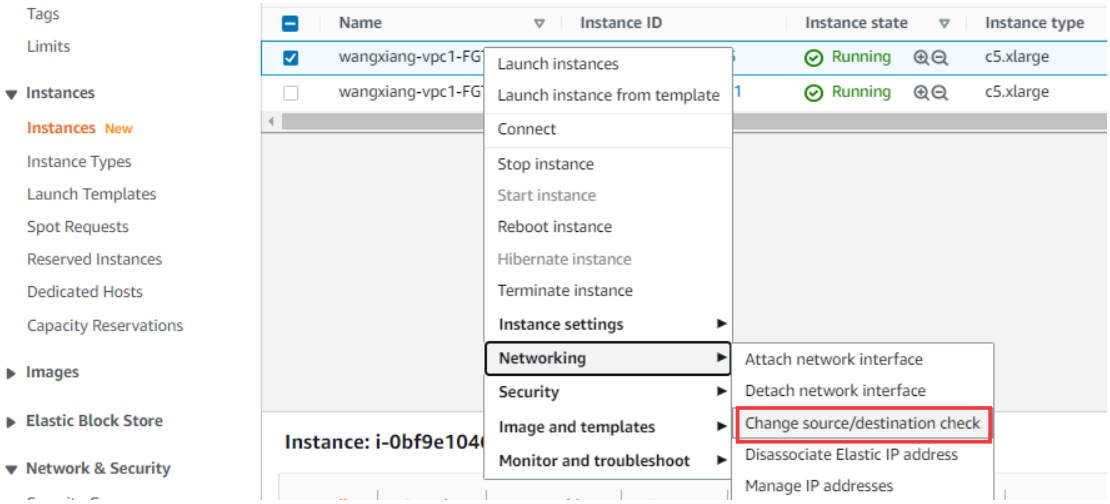
Private 路由表默认路由指向 FGT HA Master（当前 FGT1）实例的 NIC2 的接口 ID，关联 private 子网（10.0.1.0/24，10.0.4.0/24，10.0.5.0/24）

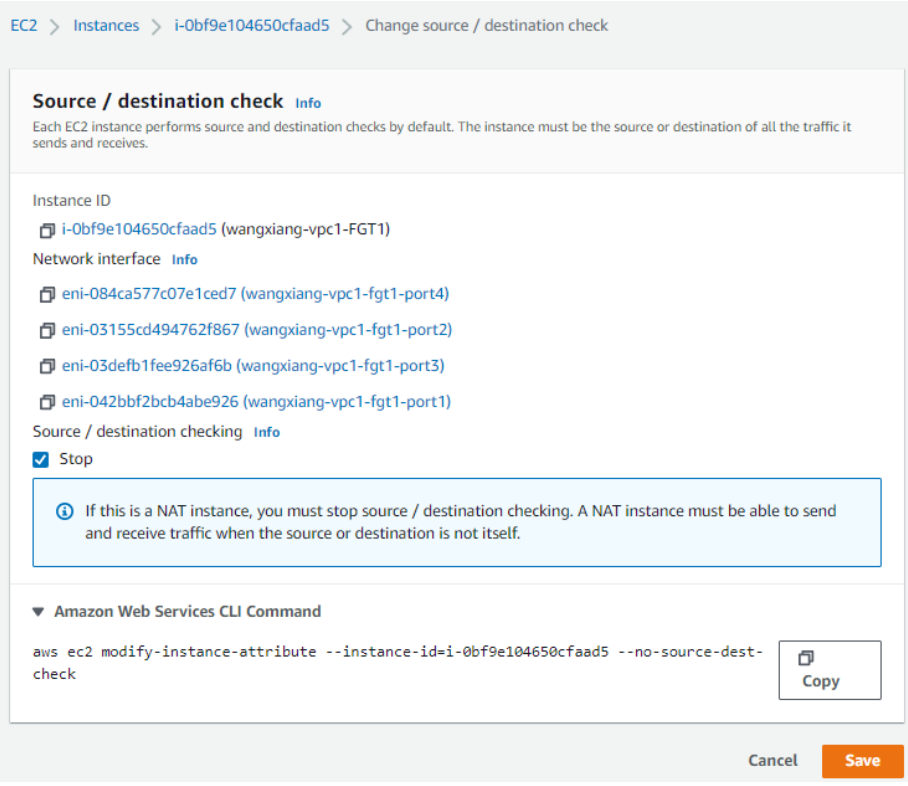


## 4. 9. 禁用源/目标检查

每个 EC2 实例都会默认执行源/目标检查。这意味着实例必须为其发送或接收的数据流的源头或目标。 但是，NAT 实例必须能够在源或目标并非其本身时发送和接收数据流。因此，FGT 实例必须禁用源/目标检查。

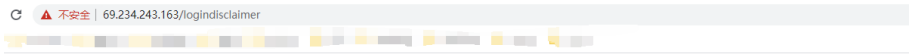


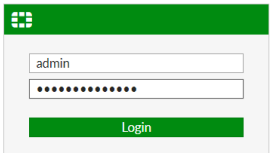



## 4.10. 访问 FortiGate

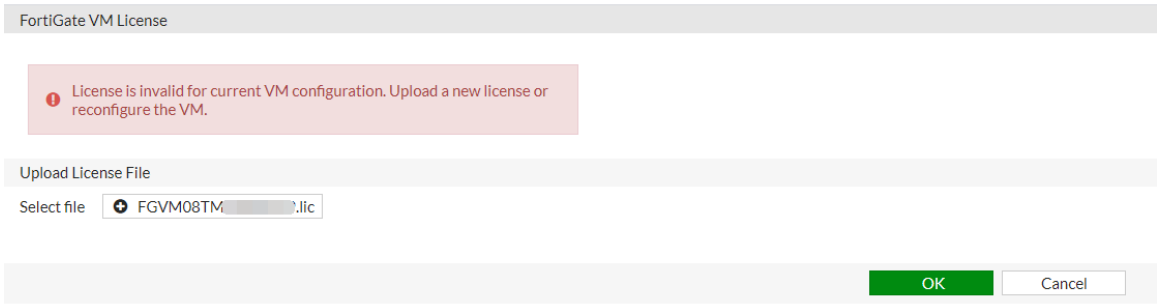
**Https 访问 FortiGate：**

使用 https://69.234.243.163（弹性 IP）访问 FortiGate，账号是 admin，密码默认是实例 ID。首次登录后，请按照提示修改密码。

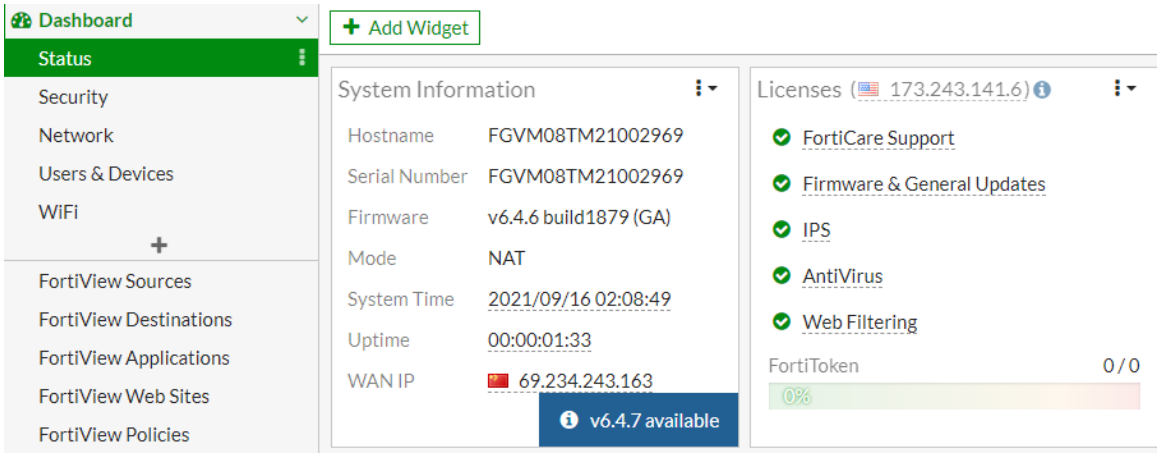



登录后，请先上传购买好的 license，导入 license 会重启 FortiGate。

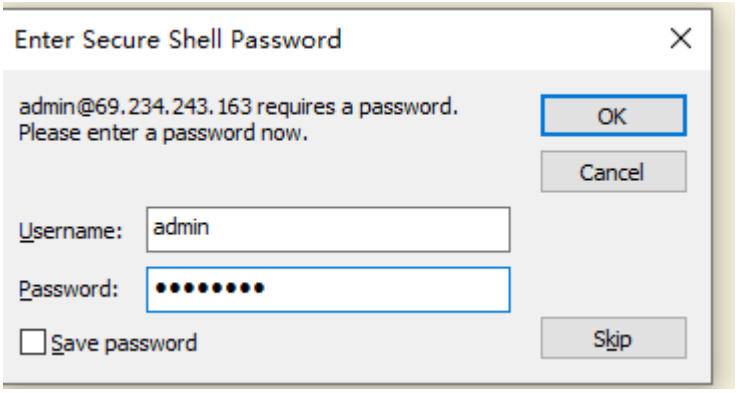



FortiGate 登录成功。

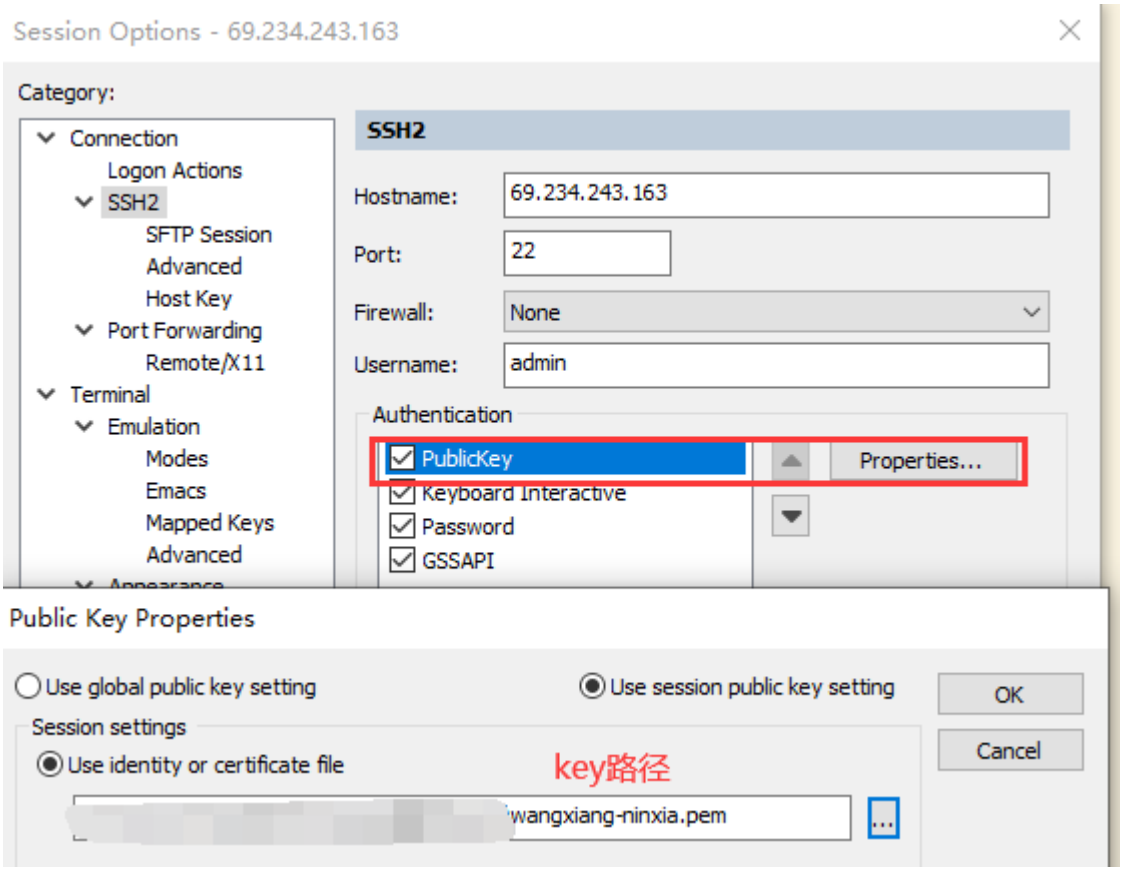


**SSH 访问 FortiGate 有两种方式，一下是 CRT 软件的访问截图：**

一种是通过账号密码的方式：




一种是通过 key 的方式：



## 4.11. 配置 FortiGate

FGT1 基本配置，先配置路由，再修改地址，否则 FGT1 将无法访问。

```
config router static
    edit 1
        set gateway 10.0.0.1
        set device "port1"
    next
    edit 2
        set dst 10.0.0.0 255.255.0.0
        set gateway 10.0.1.1
        set device "port2"
    next
end
config system interface
    edit "port1"
        set mode static
        set ip 10.0.0.11 255.255.255.0
        set allowaccess ping https ssh
    next
    edit "port2"
        set mode static
        set ip 10.0.1.11 255.255.255.0
        set allowaccess ping
    next
    edit "port3"
        set mode static
        set ip 10.0.2.11 255.255.255.0
        set allowaccess ping
    next
    edit "port4"
        set mode static
        set ip 10.0.3.11 255.255.255.0
        set allowaccess ping https ssh
    next
end
config system global
    set admintimeout 50
    set hostname "FGT1"
    set timezone 55
```

```
end
```

FGT2 基本配置，先配置路由，再修改地址，否则 FGT2 将无法访问。

```
config router static
    edit 1
        set gateway 10.0.0.1
        set device "port1"
    next
    edit 2
        set dst 10.0.0.0 255.255.0.0
        set gateway 10.0.1.1
        set device "port2"
    next
end
config system interface
    edit "port1"
        set mode static
        set ip 10.0.0.12 255.255.255.0
        set allowaccess ping https ssh
    next
    edit "port2"
        set mode static
        set ip 10.0.1.12 255.255.255.0
        set allowaccess ping
    next
    edit "port3"
        set mode static
        set ip 10.0.2.12 255.255.255.0
        set allowaccess ping
    next
    edit "port4"
        set mode static
        set ip 10.0.3.12 255.255.255.0
        set allowaccess ping https ssh
    next
end
config system global
    set admintimeout 50
    set hostname "FGT2"
    set timezone 55
end
```

在配置 HA 之前，先测试 FGT1 和 FGT2 port3 之前互 ping 是否能通。

```
FGT1 # execute ping-options source 10.0.2.11

FGT1 # execute ping 10.0.2.12
PING 10.0.2.12 (10.0.2.12): 56 data bytes
64 bytes from 10.0.2.12: icmp_seq=0 ttl=255 time=0.2 ms
64 bytes from 10.0.2.12: icmp_seq=1 ttl=255 time=0.1 ms
64 bytes from 10.0.2.12: icmp_seq=2 ttl=255 time=0.1 ms
64 bytes from 10.0.2.12: icmp_seq=3 ttl=255 time=0.1 ms
64 bytes from 10.0.2.12: icmp_seq=4 ttl=255 time=0.1 ms

--- 10.0.2.12 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.1/0.1/0.2 ms
```

## FGT1 HA 配置：

```
config system ha
    set group-name "FGTHA"
    set mode a-p
    set password fortinet
    set hbdev "port3" 50
    set session-pickup enable
    set session-pickup-connectionless enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "port4"
            set gateway 10.0.3.1
        next
    end
    set override disable
    set priority 200
    set unicast-hb enable
    set unicast-hb-peerip 10.0.2.12
end
```

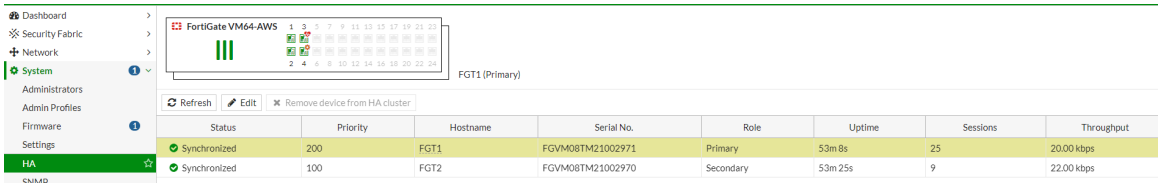
## FGT2 HA 配置：

```
config system ha
    set group-name "FGTHA"
    set mode a-p
    set password fortinet
    set hbdev "port3" 50
    set session-pickup enable
    set session-pickup-connectionless enable
    set ha-mgmt-status enable
    config ha-mgmt-interfaces
        edit 1
            set interface "port4"
```

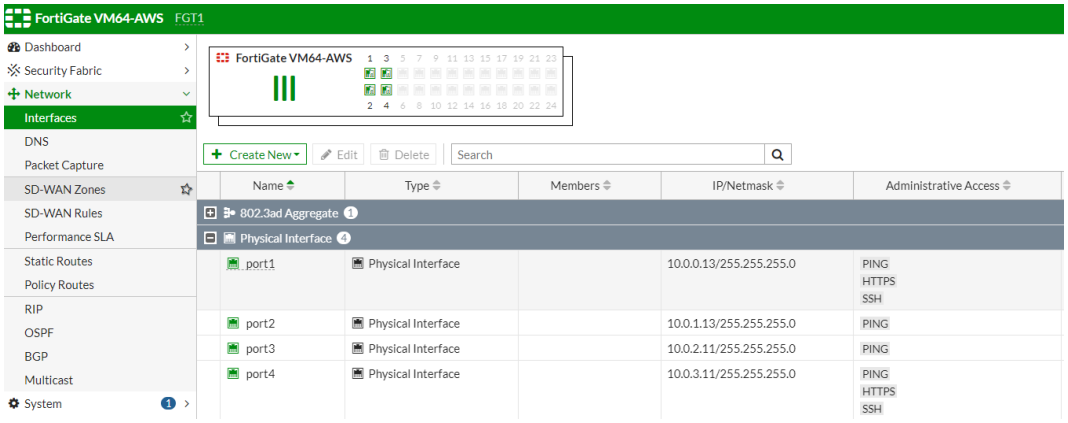
```
            set gateway 10.0.3.1
        next
    end
    set override disable
    set priority 100
    set unicast-hb enable
    set unicast-hb-peerip 10.0.2.11
end
```

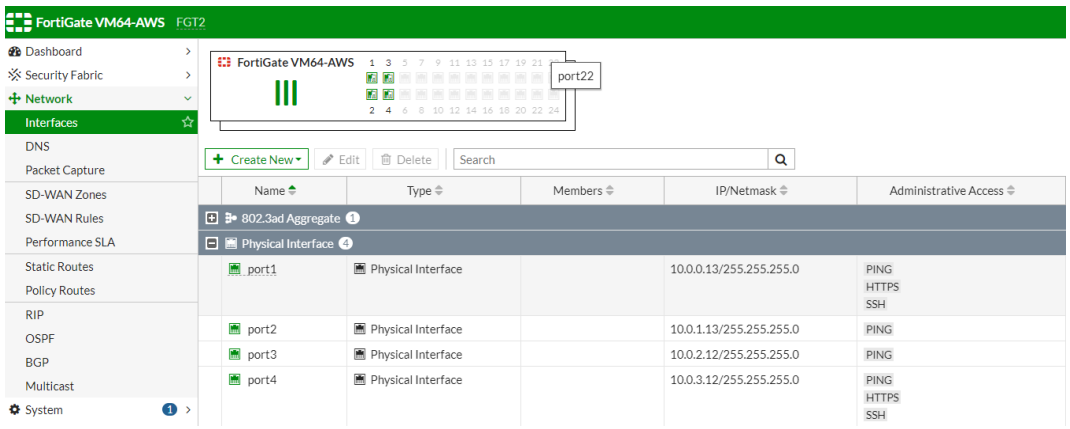
配置完成后，即可使用 FGT 实例 NIC4 关联的弹性 IP 进行访问。

注意：如果删除 HA 的配置，port3 和 port4 的接口地址也会移除。




将 FGT1 port1 接口修改为 10.0.0.13，port2 接口 ip 修改为 10.0.1.13，会同步给 FGT2（Slave）。这样 FGT1（Master）可以通过该地址访问互联网进行特征库更新。此时可以删除临时的 2 个弹性 IP。



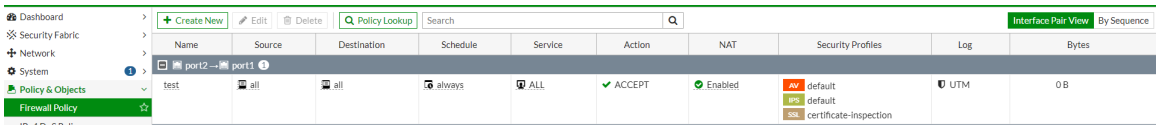



## 4.12. FortiGate 配置源 NAT

FGT1 的路由表如下：

```
FGT1 # get router info routing-table  all
Codes: K - kernel, C - connected, S - static, R - RIP, B - BGP
       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
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default

Routing table for VRF=0
S*       0.0.0.0/0 [10/0] via 10.0.0.1, port1
S        10.0.0.0/16 [10/0] via 10.0.1.1, port2
C        10.0.0.0/24 is directly connected, port1
C        10.0.1.0/24 is directly connected, port2
```

配置防火墙策略，会自动同步给 FGT2：

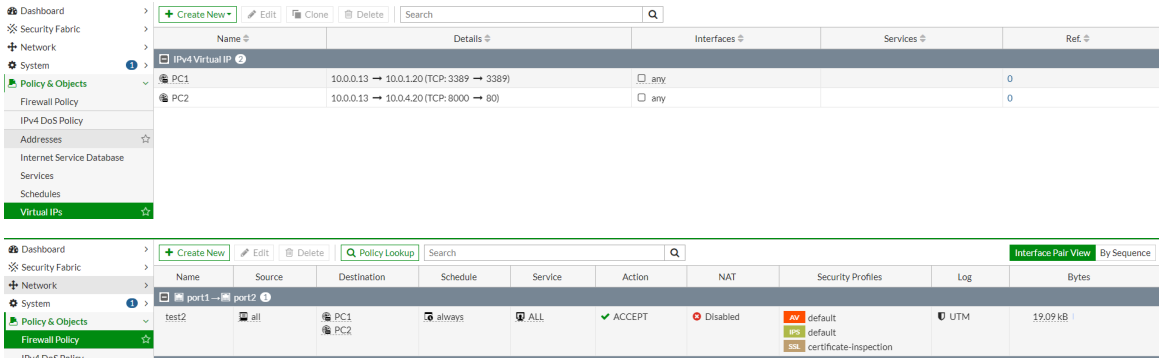



## 4.13. FortiGate 配置目的 NAT

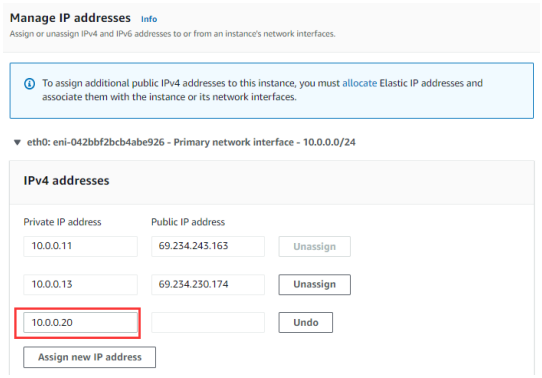
可以使用 port1 接口的地址做目的 NAT，也可以分配一个单独的 IP 来做目的 NAT，

**用 port1 接口的地址做目的 NAT：**

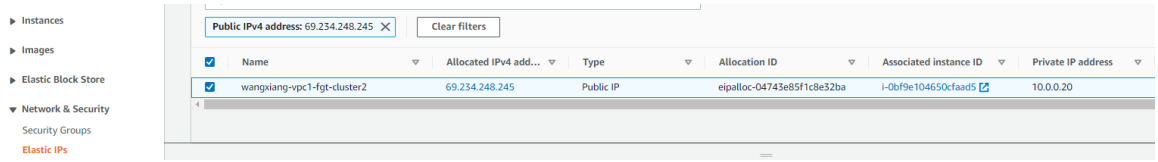
FGT1 配置 VIP 和策略，会同步给 FGT2。



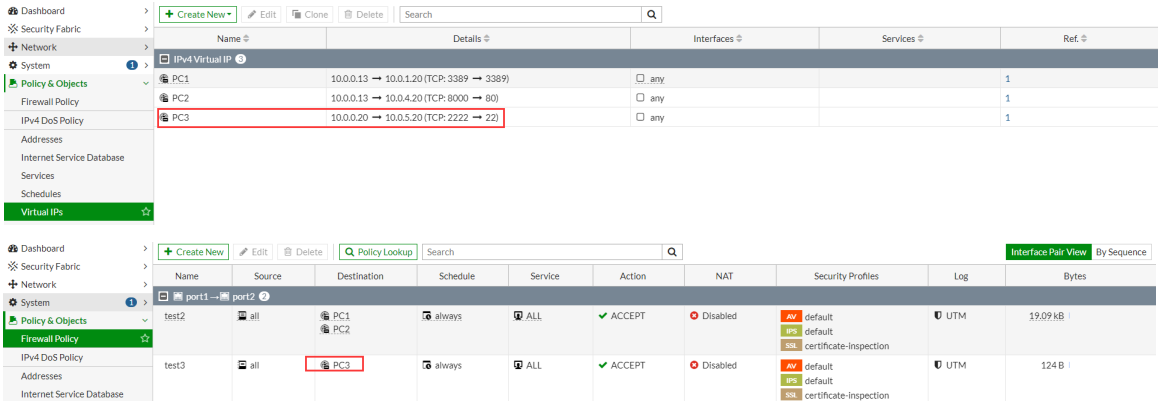
**分配单独的 IP 来做目的 NAT：**



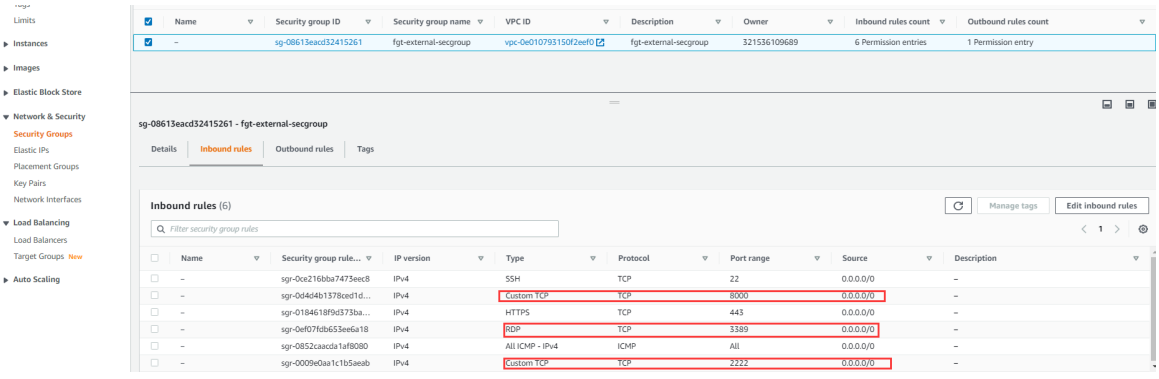

10.0.0.20 是没有关联弹性 IP 的，因此给此 IP 关联弹性 IP 才可以连接 Internet。分配弹性 IP 并关联 10.0.0.20。




FGT1 配置 VIP 和策略，会同步给 FGT2。

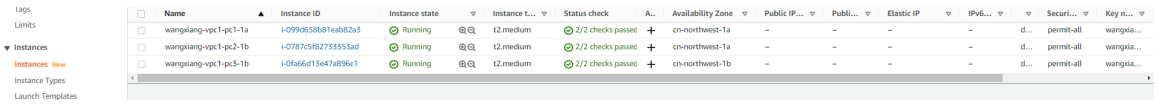



FGT 实例 NIC1，即 port1 安全组 fgt-external-secgroup 放通 3389，8000 和 2222 这 3 个对外的端口。



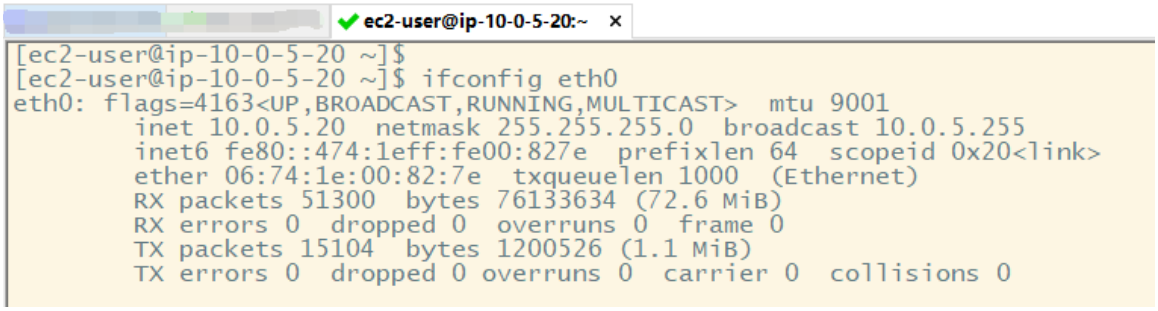
# 5. 业务测试

测试的 PC 如下：

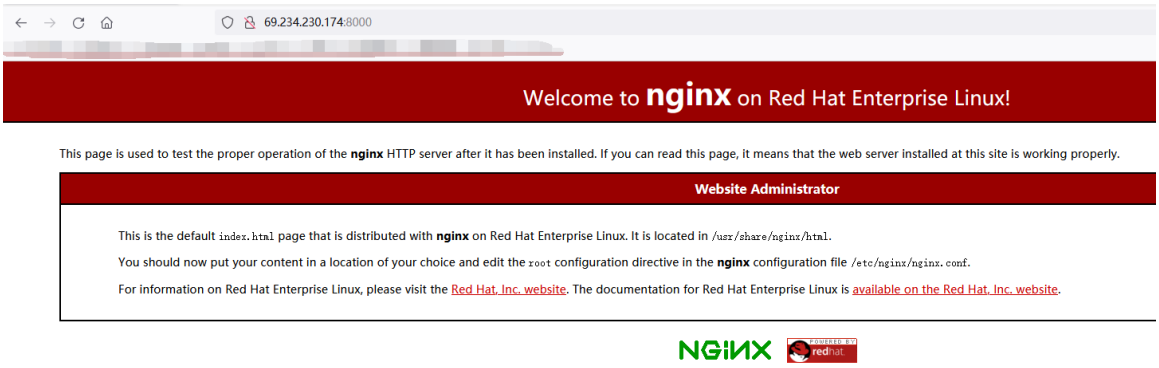



**目的 NAT 测试：**

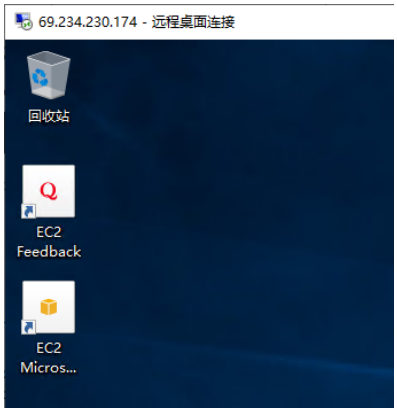
ssh 69.234.248.245 2222 访问正常。




http://69.234.230.174:8000 访问正常。




远程桌面访问正常。



**源 NAT 测试：**

ping 114.114.114.114 正常

```
[ec2-user@ip-10-0-5-20 ~]$ ping 114.114.114.114
PING 114.114.114.114 (114.114.114.114) 56(84) bytes of data.
64 bytes from 114.114.114.114: icmp_seq=1 ttl=74 time=40.3 ms
64 bytes from 114.114.114.114: icmp_seq=2 ttl=77 time=40.2 ms
64 bytes from 114.114.114.114: icmp_seq=3 ttl=58 time=40.1 ms
64 bytes from 114.114.114.114: icmp_seq=4 ttl=72 time=40.2 ms
^C
--- 114.114.114.114 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 7ms
```

使用 10.0.5.20 作为跳板登录到 10.0.4.20，访问 hao123 正常。

sudo ssh -i "wangxiang-ninxia.pem" ec2-user@10.0.4.20

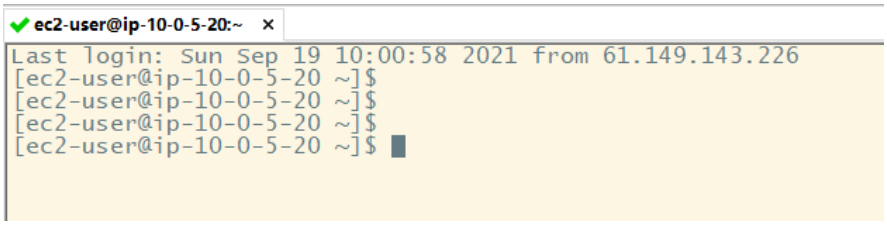
```
[ec2-user@ip-10-0-5-20 ~]$ sudo ssh -i "wangxiang-ninxia.pem" ec2-user@10.0.4.20
Last login: Sun Sep 19 10:09:20 2021 from 10.0.5.20
[ec2-user@ip-10-0-4-20 ~]$ curl www.hao123.com
<!DOCTYPE html><html><head><noscript><meta http-equiv="refresh" content="0; URL='/?__
"_blank"><meta charset="utf-8"/><meta http-equiv=X-UA-Compatible content="IE=edge,chr
dns-prefetch" href="//s1.hao123img.com" /><link rel="dns-prefetch" href="//s0.hao123i
refetch" href="//img1.hao123.com" /><meta name="keywords" content="上网导航,网址大全,
活动"/><meta name="description"content="hao123是汇集全网优质网址及资源的中文上网导航。
上网，从hao123开始。"/><script>window.HAO=window.HAO||{};window.HAO.https = false;win
></style><script>(function (win) {var HAO = win.HAO = win.HAO || {};HAO.domainMap = {
com","https:\/\/dgss1.bdstatic.com\/5eN1dDebRNRTm2_p8IuM_a":"http:\/\/s1.hao123img.co
3img.com","https:\/\/dgss1.bdstatic.com\/5bVXsj_p_tVS5dKfpU_Y_D3":"http:\/\/sc1.hao12
nOO.baidu-img.cn","https:\/\/graph.baidu.com":"http:\/\/himg.baidu.com"};HAO.https =
httpsMon=[],n,httpsTrans=function(t){try{if(!t)return t;var e=t.replace(/(^\s*)|(\s*$
```

10.0.1.20 访问 baidu 正常。

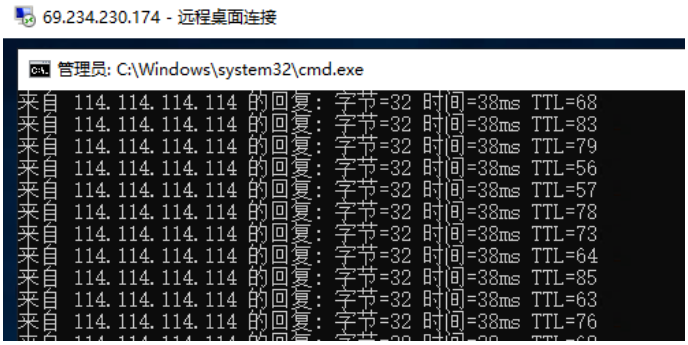
# 6. HA 切换测试

**HA 切换测试目的：**FGT 支持跨 AZ 的 HA，查看 HA 切换时，外网的 ssh 连接，RDP 连接是否会断开，内网 ping 会丢几个包。

从外网 ssh 到 pc3。

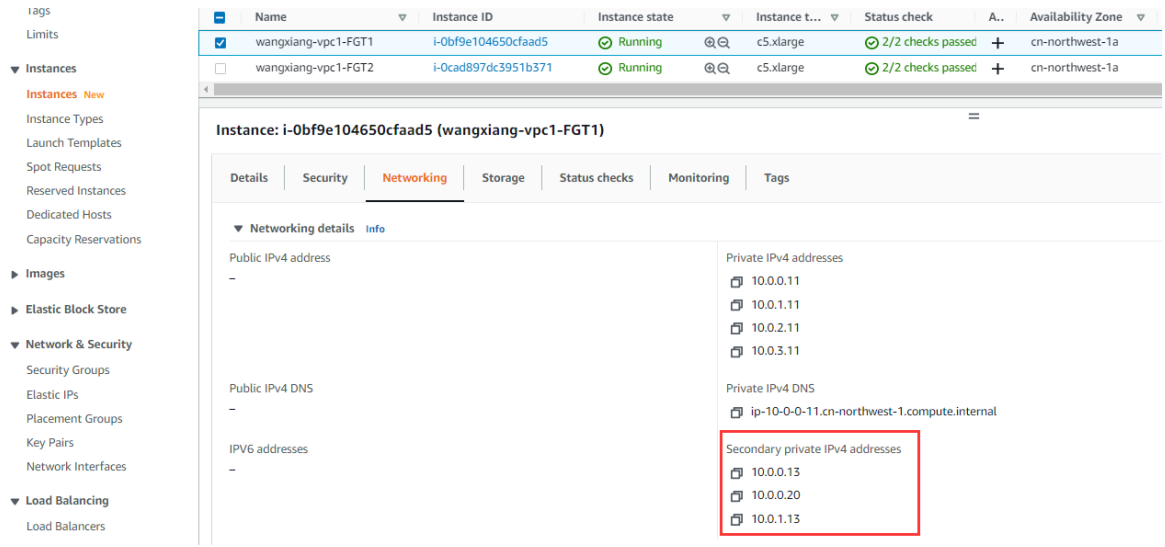



从外网 RDP 连接到 pc1，从 pc1 ping 114。

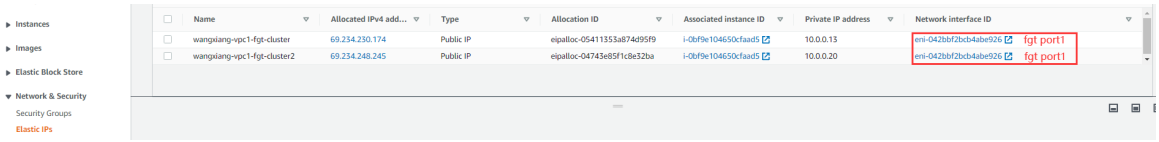



**HA 切换前，FGT 是主，FGT2 是备：**

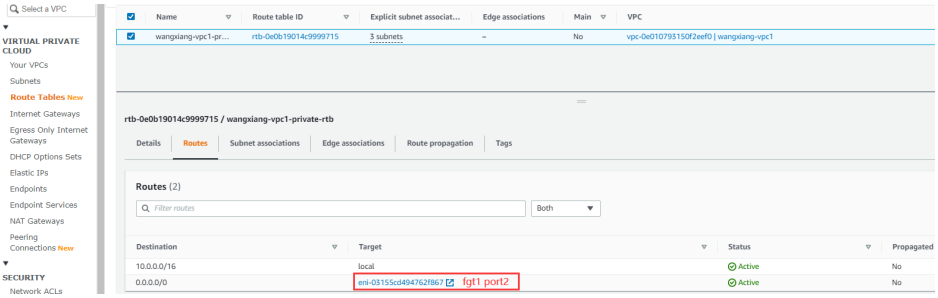
FGT1 关联了 Secondary IP。



业务口的弹性 IP 关联到 FGT1 NIC1 网卡，即 port1。




Private 路由表默认路由指向 FGT1 NIC2 网卡，即 port2




会话同步正常：

RDP 会话：

```
FGT1 # diagnose sys session filter dport 3389

FGT1 # diagnose sys session list

session info: proto=6 proto_state=11 duration=89 expire=3599 timeout=3600 flags=0000000
0 socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty synced
statistic(bytes/packets/allow_err): org=248417/2315/1 reply=768527/2262/1 tuples=3
tx speed(Bps/kbps): 2764/22 rx speed(Bps/kbps): 8551/68
orgin->sink: org pre->post, reply pre->post dev=3->4/4->3 gwy=10.0.1.20/10.0.0.1
hook=pre dir=org act=dnat 61.149.143.226:33419->10.0.0.13:3389(10.0.1.20:3389)
hook=post dir=reply act=snat 10.0.1.20:3389->61.149.143.226:33419(10.0.0.13:3389)
hook=post dir=org act=noop 61.149.143.226:33419->10.0.1.20:3389(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=0000031c tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041008
total session 1
```

```
FGT2 # diagnose sys session filter dport 3389

FGT2 # diagnose sys session list

session info: proto=6 proto_state=11 duration=95 expire=3504 timeout=3600 flags=00000000 s
ocktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=dirty may_dirty ndr syn_ses
statistic(bytes/packets/allow_err): org=0/0/0 reply=0/0/0 tuples=3
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=3->4/4->3 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=dnat 61.149.143.226:33419->10.0.0.13:3389(10.0.1.20:3389)
hook=post dir=reply act=snat 10.0.1.20:3389->61.149.143.226:33419(10.0.0.13:3389)
hook=post dir=org act=noop 61.149.143.226:33419->10.0.1.20:3389(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=2 auth_info=0 chk_client_info=0 vd=0
serial=0000031c tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041000
total session 1
```

SSH 会话：

```
FGT1 # diagnose sys session filter dport 2222

FGT1 # diagnose sys session list

session info: proto=6 proto_state=11 duration=77 expire=3580 timeout=3600 flags=0000000
0 socktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty ndr synced
statistic(bytes/packets/allow_err): org=3906/22/1 reply=4937/20/1 tuples=3
tx speed(Bps/kbps): 50/0 rx speed(Bps/kbps): 63/0
orgin->sink: org pre->post, reply pre->post dev=1->4/4->3 gwy=10.0.1.1/10.0.0.1
hook=pre dir=org act=dnat 61.149.143.226:33463->10.0.0.20:2222(10.0.5.20:22)
hook=post dir=reply act=snat 10.0.5.20:22->61.149.143.226:33463(10.0.0.20:2222)
hook=post dir=org act=noop 61.149.143.226:33463->10.0.5.20:22(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=3 auth_info=0 chk_client_info=0 vd=0
serial=00000335 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041008
total session 1
```

```
FGT2 # diagnose sys session filter dport 2222

FGT2 # diagnose sys session list

session info: proto=6 proto_state=11 duration=87 expire=3512 timeout=3600 flags=00000000 s
ocktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=dirty may_dirty ndr syn_ses
statistic(bytes/packets/allow_err): org=0/0/0 reply=0/0/0 tuples=3
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=3->4/4->3 gwy=0.0.0.0/0.0.0.0
hook=pre dir=org act=dnat 61.149.143.226:33463->10.0.0.20:2222(10.0.5.20:22)
hook=post dir=reply act=snat 10.0.5.20:22->61.149.143.226:33463(10.0.0.20:2222)
hook=post dir=org act=noop 61.149.143.226:33463->10.0.5.20:22(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=3 auth_info=0 chk_client_info=0 vd=0
serial=00000335 tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041000
total session 1
```

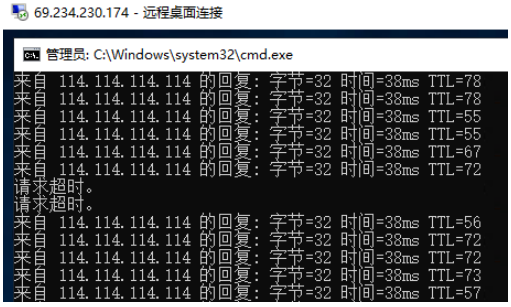
Ping 会话：

```
FGT1 # diagnose sys session filter proto 1

FGT1 #
FGT1 # diagnose sys session list

session info: proto=1 proto_state=00 duration=776 expire=59 timeout=0 flags=00000000 so
cktype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=may_dirty ndr synced syn_ses
statistic(bytes/packets/allow_err): org=43620/727/1 reply=43620/727/1 tuples=3
tx speed(Bps/kbps): 56/0 rx speed(Bps/kbps): 56/0
orgin->sink: org pre->post, reply pre->post dev=4->3/3->4 gwy=10.0.0.1/10.0.1.20
hook=post dir=org act=snat 10.0.1.20:1->114.114.114.114:8(10.0.0.13:60417)
hook=pre dir=reply act=dnat 114.114.114.114:60417->10.0.0.13:0(10.0.1.20:1)
hook=post dir=reply act=noop 114.114.114.114:1->10.0.1.20:0(0.0.0.0:0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=0000101b tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041008
total session 1

FGT1 #
```

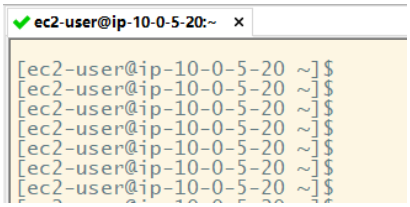
```
FGT2 # diagnose sys session filter proto 1

FGT2 #
FGT2 # diagnose sys session list

session info: proto=1 proto_state=00 duration=778 expire=40 timeout=0 flags=00000000 sockt
ype=0 sockport=0 av_idx=0 use=4
origin-shaper=
reply-shaper=
per_ip_shaper=
class_id=0 ha_id=0 policy_dir=0 tunnel=/ vlan_cos=0/255
state=dirty may_dirty ndr syn_ses
statistic(bytes/packets/allow_err): org=0/0/0 reply=0/0/0 tuples=3
tx speed(Bps/kbps): 0/0 rx speed(Bps/kbps): 0/0
orgin->sink: org pre->post, reply pre->post dev=4->3/3->4 gwy=0.0.0.0/0.0.0.0
hook=post dir=org act=snat 10.0.1.20:1->114.114.114.114:8(10.0.0.13:60417)
hook=pre dir=reply act=dnat 114.114.114.114:60417->10.0.0.13:0(10.0.1.20:1)
hook=post dir=reply act=noop 114.114.114.114:1->10.0.1.20:0(0.0.0.0:0)
misc=0 policy_id=1 auth_info=0 chk_client_info=0 vd=0
serial=0000101b tos=ff/ff app_list=0 app=0 url_cat=0
sdwan_mbr_seq=0 sdwan_service_id=0
rpdb_link_id=00000000 rpdb_svc_id=0 ngfwid=n/a
npu_state=0x041000
total session 1

FGT2 #
```

**重启 FGT2，HA 切换后，FGT2 是主，FGT1 是备：**

RDP 没有断开，ping 丢两个包。



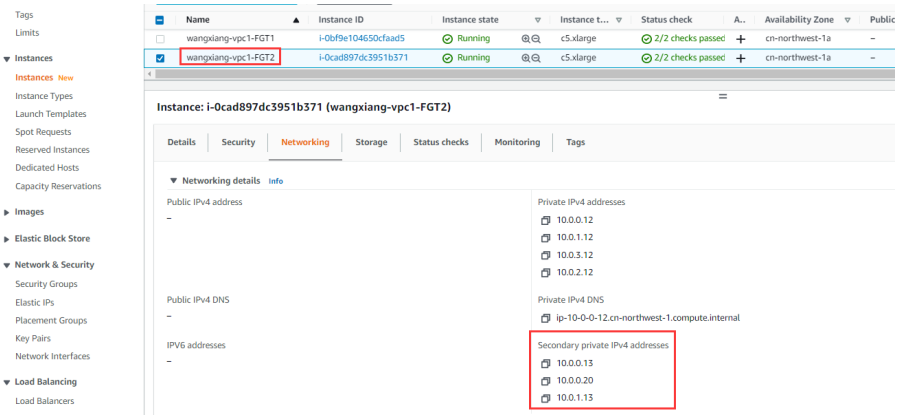

SSH 没有断开。



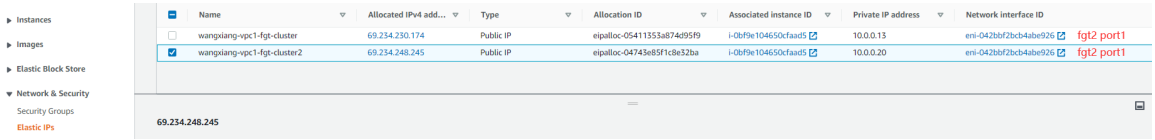

从 FGT2 debug 可以看出，除了 FGT 本身的 HA 切换以外，还需要移动 secondary ip，弹性 ip 到 FGT2 实例，更新 AWS private 路由表的默认路由指向 FGT2 NIC2。

```
FGT2 # diagnose debug application awsd -1
Debug messages will be on for 30 minutes.

FGT2 # diagnose debug  enable

FGT2 #
FGT2 #
FGT2 # HA event
HA state: primary
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port2 ip 10.0.1.13
send_vip_arp: vd root primary 1 intf fortilink ip 169.254.1.1
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.20
awsd get instance id i-0cad897dc3951b371
awsd get iam role FGT-HA-Failover
awsd get region cn-northwest-1
awsd get vpc id vpc-0e010793150f2eef0
awsd doing ha failover for vdom root
awsd moving secondary ip for port1
awsd moving secip 10.0.0.13 from eni-042bbf2bcb4abe926 to eni-0fc04671b400002c5
awsd moving secip 10.0.0.20 from eni-042bbf2bcb4abe926 to eni-0fc04671b400002c5
awsd move secondary ip successfully
awsd associate elastic ip allocation eipalloc-05411353a874d95f9 to 10.0.0.13 of eni eni-0fc04671b400002c5
awsd associate elastic ip successfully
awsd associate elastic ip allocation eipalloc-04743e85f1c8e32ba to 10.0.0.20 of eni eni-0fc04671b400002c5
awsd associate elastic ip successfully
awsd moving secondary ip for port2
awsd moving secip 10.0.1.13 from eni-03155cd494762f867 to eni-058bbef3c524af52e
awsd move secondary ip successfully
awsd update route table rtb-0e0b19014c9999715, replace route of dst 0.0.0.0/0 to eni-058bbef3c524af52e
awsd update route successfully
HA event
HA state: primary
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port2 ip 10.0.1.13
send_vip_arp: vd root primary 1 intf fortilink ip 169.254.1.1
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.13
send_vip_arp: vd root primary 1 intf port1 ip 10.0.0.20
awsd get instance id i-0cad897dc3951b371
awsd get iam role FGT-HA-Failover
awsd get region cn-northwest-1
awsd get vpc id vpc-0e010793150f2eef0
awsd doing ha failover for vdom root
```

FGT1 的实例 Secondary ip 已经移动到 FGT2 实例。




弹性 IP 重新在 FGT2 实例绑定。




Private 路由表默认路由指向 FGT2 NIC2。