Lab 4

Garrett Gruss 4976-3695

February 21, 2026

1 Summary

This lab covered the creation and configuration of Amazon Virtual Private Clouds (VPCs) and Transit Gateway networking. Exercises 4.1 through 4.9 involved creating two isolated VPCs with custom CIDR blocks (172.16.0.0/16 and 172.17.0.0/16), provisioning subnets within each VPC in separate availability zones, and establishing a Transit Gateway to enable routing between the two otherwise-isolated networks. VPC attachments were created for each subnet and attached to the Transit Gateway, route tables were updated in each VPC to direct cross-VPC traffic through the Transit Gateway, and the resulting routes were verified using the AWS CLI. Exercises 8.1 through 8.4 covered Amazon CloudFront and Route 53, including configuring a CloudFront distribution to serve content from an origin, exploring edge location caching behavior, registering and managing DNS records in Route 53, and applying routing policies to direct traffic based on geographic and latency-based rules.

```
garre@garrettlaptop:~/development/notes$  aws ec2 create-vpc --cidr-block 172.16.0.0/16
{
    "Vpc": {
        "OwnerId": "349577273469",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-0bfb5bd83ff043949",
                "CidrBlock": "172.16.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
        ],
        "IsDefault": false,
        "VpcId": "vpc-08fe8ac778df57a5d",
        "State": "pending",
        "CidrBlock": "172.16.0.0/16",
        "DhcpOptionsId": "dopt-09cf0f9a7617b4b01"
:...skipping...
{
    "Vpc": {
        "OwnerId": "349577273469",
        "InstanceTenancy": "default",
        "Ipv6CidrBlockAssociationSet": [],
        "CidrBlockAssociationSet": [
            {
                "AssociationId": "vpc-cidr-assoc-0bfb5bd83ff043949",
                "CidrBlock": "172.16.0.0/16",
                "CidrBlockState": {
                    "State": "associated"
                }
            }
```

Figure 1: Exercise 4.1 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-vpcs --vpc-ids vpc-08fe8ac778df57a5d
{
    "Vpcs": [
        {
            "OwnerId": "349577273469",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-0bfb5bd83ff043949",
                    "CidrBlock": "172.16.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "VpcId": "vpc-08fe8ac778df57a5d",
            "State": "available",
            "CidrBlock": "172.16.0.0/16",
            "DhcpOptionsId": "dopt-09cf0f9a7617b4b01"
        }
    ]
}
```

Figure 2: Exercise 4.1 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-subnet --vpc-id vpc-0414c3bc8e8d453db --cidr-block 172.16.100.0/24
-availability-zone us-east-1a
{
    "Subnet": {
        "AvailabilityZoneId": "use1-az1",
        "MapCustomerOwnedIpOnLaunch": false,
        "OwnerId": "349577273469",
        "AssignIpv6AddressOnCreation": false,
        "Ipv6CidrBlockAssociationSet": [],
        "SubnetArn": "arn:aws:ec2:us-east-1:349577273469:subnet/subnet-02cdd18e304c943e1",
        "EnableDns64": false,
        "Ipv6Native": false,
        "PrivateDnsNameOptionsOnLaunch": {
            "HostnameType": "ip-name",
            "EnableResourceNameDnsARecord": false,
            "EnableResourceNameDnsAAAARecord": false
        },
        "SubnetId": "subnet-02cdd18e304c943e1",
        "State": "available",
        "VpcId": "vpc-0414c3bc8e8d453db",
        "CidrBlock": "172.16.100.0/24",
        "AvailableIpAddressCount": 251,
        "AvailabilityZone": "us-east-1a",
        "DefaultForAz": false,
        "MapPublicIpOnLaunch": false
    }
}
```

Figure 3: Exercise 4.2 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-subnets --subnet-ids subnet-02cdd18e304c943e1
{
    "Subnets": [
        {
            "AvailabilityZoneId": "use1-az1",
            "MapCustomerOwnedIpOnLaunch": false,
            "OwnerId": "349577273469",
            "AssignIpv6AddressOnCreation": false,
            "Ipv6CidrBlockAssociationSet": [],
            "SubnetArn": "arn:aws:ec2:us-east-1:349577273469:subnet/subnet-02cdd18e304c943e1",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            },
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "SubnetId": "subnet-02cdd18e304c943e1",
            "State": "available",
            "VpcId": "vpc-0414c3bc8e8d453db",
            "CidrBlock": "172.16.100.0/24",
            "AvailableIpAddressCount": 251,
            "AvailabilityZone": "us-east-1a",
            "DefaultForAz": false,
            "MapPublicIpOnLaunch": false
        }
    ]
}
```

Figure 4: Exercise 4.2 (2)

```
garre@garrettlaptop:~/development/notes$  aws ec2 create-network-interface --private-ip-address 172.16.100.99 --subnet-id su
bnet-02cdd18e304c943e1
{
    "NetworkInterface": {
        "AvailabilityZone": "us-east-1a",
        "Description": "",
        "Groups": [
            {
                "GroupId": "sg-0dd5b2c38b191887d",
                "GroupName": "default"
            }
        ],
        "InterfaceType": "interface",
        "Ipv6Addresses": [],
        "MacAddress": "02:81:56:26:26:77",
        "NetworkInterfaceId": "eni-0d7bcb027f6f2cbf7",
        "OwnerId": "349577273469",
        "PrivateIpAddress": "172.16.100.99",
        "PrivateIpAddresses": [
            {
                "Primary": true,
                "PrivateIpAddress": "172.16.100.99"
            }
        ],
        "RequesterId": "AIDAVCZDN4B6XYE24LPOD",
        "RequesterManaged": false,
        "SourceDestCheck": true,
        "Status": "pending",
        "SubnetId": "subnet-02cdd18e304c943e1",
        "TagSet": [],
        "VpcId": "vpc-0414c3bc8e8d453db",
        "Operator": {
            "Managed": false
        },
```

Figure 5: Exercise 4.3 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-network-interfaces --network-interface-ids eni-0d7bcb027f6f2cbf7
    "NetworkInterfaces": [
        {
            "AvailabilityZone": "us-east-1a",
            "Description": "",
            "Groups": [
                {
                    "GroupId": "sg-0dd5b2c38b191887d",
                    "GroupName": "default"
                }
            ],
            "InterfaceType": "interface",
            "Ipv6Addresses": [],
            "MacAddress": "02:81:56:26:26:77",
            "NetworkInterfaceId": "eni-0d7bcb027f6f2cbf7",
            "OwnerId": "349577273469",
            "PublicDnsName": "",
            "PrivateIpAddress": "172.16.100.99",
            "PrivateIpAddresses": [
                {
                    "Primary": true,
                    "PrivateIpAddress": "172.16.100.99"
                }
            ],
            "RequesterId": "AIDAVCZDN4B6XYE24LPOD",
            "RequesterManaged": false,
            "SourceDestCheck": true,
            "Status": "available",
            "SubnetId": "subnet-02cdd18e304c943e1",
            "TagSet": [],
            "VpcId": "vpc-0414c3bc8e8d453db",
            "Operator": {
```

Figure 6: Exercise 4.3 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-internet-gateway
{
    "InternetGateway": {
        "Attachments": [],
        "InternetGatewayId": "igw-01cb78e8bd32aaed9",
        "OwnerId": "349577273469",
        "Tags": []
    }
}
```

Figure 7: Exercise 4.4 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 attach-internet-gateway --internet-gateway-id igw-01cb78e8bd32aaed9 --vpc-i
d vpc-0414c3bc8e8d453db
garre@garrettlaptop:~/development/notes$ aws ec2 describe-route-tables --filters Name=vpc-id,Values=vpc-0414c3bc8e8d453db
{
    "RouteTables": [
        {
            "Associations": [
                {
                    "Main": true,
                    "RouteTableAssociationId": "rtbassoc-0478927546b9f0649",
                    "RouteTableId": "rtb-074e5a02bacb71f1f",
                    "AssociationState": {
                        "State": "associated"
                    }
                }
            ],
            "PropagatingVgws": [],
            "RouteTableId": "rtb-074e5a02bacb71f1f",
            "Routes": [
                {
                    "DestinationCidrBlock": "172.16.0.0/16",
                    "GatewayId": "local",
                    "Origin": "CreateRouteTable",
                    "State": "active"
                }
            ],
            "Tags": [],
            "VpcId": "vpc-0414c3bc8e8d453db",
            "OwnerId": "349577273469"
        }
    ]
}
```

Figure 8: Exercise 4.4 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-route --route-table-id rtb-074e5a02bacb71f1f --destination-cidr-bl
k "0.0.0.0/0" --gateway-id igw-01cb78e8bd32aaed9
{
    "Return": true
}
```

Figure 9: Exercise 4.4 (3)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-security-group --group-name "web-ssh" --description "Web and SSH tra
ffic" --vpc-id vpc-0414c3bc8e8d453db
{
    "GroupId": "sg-0be0c864f3a02e19a",
    "SecurityGroupArn": "arn:aws:ec2:us-east-1:349577273469:security-group/sg-0be0c864f3a02e19a"
}
```

Figure 10: Exercise 4.5 (1)

```
garre@garrettlaptop:~/development/notes$  aws ec2 describe-security-groups --group-id sg-0be0c864f3a02e19a
{
    "SecurityGroups": [
        {
            "GroupId": "sg-0be0c864f3a02e19a",
            "IpPermissionsEgress": [
                {
                    "IpProtocol": "-1",
                    "UserIdGroupPairs": [],
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
                    ],
                    "Ipv6Ranges": [],
                    "PrefixListIds": []
                }
            ],
            "VpcId": "vpc-0414c3bc8e8d453db",
            "SecurityGroupArn": "arn:aws:ec2:us-east-1:349577273469:security-group/sg-0be0c864f3a02e19a",
            "OwnerId": "349577273469",
            "GroupName": "web-ssh",
            "Description": "Web and SSH traffic",
            "IpPermissions": [
                {
                    "IpProtocol": "tcp",
                    "FromPort": 80,
                    "ToPort": 80,
                    "UserIdGroupPairs": [],
                    "IpRanges": [
                        {
                            "CidrIp": "0.0.0.0/0"
                        }
```

Figure 11: Exercise 4.5 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-network-acl --vpc-id vpc-0414c3bc8e8d453db
{
    "NetworkAcl": {
        "Associations": [],
        "Entries": [
            {
                "CidrBlock": "0.0.0.0/0",
                "Egress": true,
                "IcmpTypeCode": {},
                "PortRange": {},
                "Protocol": "-1",
                "RuleAction": "deny",
                "RuleNumber": 32767
            },
            {
                "CidrBlock": "0.0.0.0/0",
                "Egress": false,
                "IcmpTypeCode": {},
                "PortRange": {},
                "Protocol": "-1",
                "RuleAction": "deny",
                "RuleNumber": 32767
            }
        ],
        "IsDefault": false,
        "NetworkAclId": "acl-0670f89aeb5c4d9b3",
        "Tags": [],
        "VpcId": "vpc-0414c3bc8e8d453db",
        "OwnerId": "349577273469"
    },
    "ClientToken": "fcb1ef7e-dfa8-47f1-a02a-bc4441a23b74"
}
```

Figure 12: Exercise 4.6 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-network-acls --network-acl-id acl-0670f89aeb5c4d9b3
{
    "NetworkAcls": [
        {
            "Associations": [],
            "Entries": [
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": true,
                    "Protocol": "-1",
                    "RuleAction": "deny",
                    "RuleNumber": 32767
                },
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": false,
                    "PortRange": {
                        "From": 22,
                        "To": 22
                    },
                    "Protocol": "6",
                    "RuleAction": "allow",
                    "RuleNumber": 70
                },
                {
                    "CidrBlock": "0.0.0.0/0",
                    "Egress": false,
                    "Protocol": "-1",
                    "RuleAction": "deny",
                    "RuleNumber": 32767
                }
            ],
            "IsDefault": false,
            "NetworkAclId": "acl-0670f89aeb5c4d9b3",
```

Figure 13: Exercise 4.6 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 associate-address --allocation-id eipalloc-0d427825a94ad45df --network-inte
rface-id eni-0d7bcb027f6f2cbf7
{
    "AssociationId": "eipassoc-0c09141308d248848"
}
```

Figure 14: Exercise 4.7 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-network-interfaces --network-interface-ids eni-0d7bcb027f6f2cbf7
{
    "NetworkInterfaces": [
        {
            "Association": {
                "AllocationId": "eipalloc-0d427825a94ad45df",
                "AssociationId": "eipassoc-0c09141308d248848",
                "IpOwnerId": "349577273469",
                "PublicDnsName": "",
                "PublicIp": "100.30.207.253"
            },
            "AvailabilityZone": "us-east-1a",
            "Description": "",
            "Groups": [
                {
                    "GroupId": "sg-0dd5b2c38b191887d",
                    "GroupName": "default"
                }
            ],
            "InterfaceType": "interface",
            "Ipv6Addresses": [],
            "MacAddress": "02:81:56:26:26:77",
            "NetworkInterfaceId": "eni-0d7bcb027f6f2cbf7",
            "OwnerId": "349577273469",
            "PublicDnsName": "",
            "PrivateIpAddress": "172.16.100.99",
            "PrivateIpAddresses": [
                {
                    "Association": {
                        "AllocationId": "eipalloc-0d427825a94ad45df",
                        "AssociationId": "eipassoc-0c09141308d248848",
```

Figure 15: Exercise 4.7 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-transit-gateway-vpc-attachment --transit-gateway-id tgw-0acd402fce62
a318c --vpc-id vpc-04f08d3c85b2e0524 --subnet-ids subnet-0234447e5739c40c2
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-077d026864d98558f",
        "TransitGatewayId": "tgw-0acd402fce62a318c",
        "VpcId": "vpc-04f08d3c85b2e0524",
        "VpcOwnerId": "349577273469",
        "State": "pending",
        "SubnetIds": [
            "subnet-0234447e5739c40c2"
        ],
        "CreationTime": "2026-02-21T14:57:52+00:00",
        "Options": {
            "DnsSupport": "enable",
            "SecurityGroupReferencingSupport": "enable",
            "Ipv6Support": "disable",
            "ApplianceModeSupport": "disable"
        }
    }
}
```

Figure 16: Exercise 4.8 (1)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-transit-gateway-vpc-attachment --transit-gateway-id tgw-0acd402fce62
a318c --vpc-id vpc-0414c3bc8e8d453db --subnet-ids subnet-02cdd18e304c943e1
{
    "TransitGatewayVpcAttachment": {
        "TransitGatewayAttachmentId": "tgw-attach-0e31789585627e631",
        "TransitGatewayId": "tgw-0acd402fce62a318c",
        "VpcId": "vpc-0414c3bc8e8d453db",
        "VpcOwnerId": "349577273469",
        "State": "pending",
        "SubnetIds": [
            "subnet-02cdd18e304c943e1"
        ],
        "CreationTime": "2026-02-21T15:01:36+00:00",
        "Options": {
            "DnsSupport": "enable",
            "SecurityGroupReferencingSupport": "enable",
            "Ipv6Support": "disable",
            "ApplianceModeSupport": "disable"
        }
    }
}
```

Figure 17: Exercise 4.8 (2)

```
garre@garrettlaptop:~/development/notes$ aws ec2 search-transit-gateway-routes --transit-gateway-route-table-id tgw-rtb-0a85
014347c9c970d --filters "Name=type,Values=static,propagated"
        {
            "DestinationCidrBlock": "172.16.0.0/16",
            "TransitGatewayRouteTableAnnouncementId": "",
            "TransitGatewayAttachments": [
                {
                    "ResourceId": "vpc-0414c3bc8e8d453db",
                    "TransitGatewayAttachmentId": "tgw-attach-0e31789585627e631",
                    "ResourceType": "vpc"
                }
            ],
            "Type": "propagated",
            "State": "active"
        },
        {
            "DestinationCidrBlock": "172.17.0.0/16",
            "TransitGatewayRouteTableAnnouncementId": "",
            "TransitGatewayAttachments": [
                {
                    "ResourceId": "vpc-04f08d3c85b2e0524",
                    "TransitGatewayAttachmentId": "tgw-attach-077d026864d98558f",
                    "ResourceType": "vpc"
                }
            ],
            "Type": "propagated",
            "State": "active"
        }
    ]
}
```

Figure 18: Exercise 4.8 (3)

```
garre@garrettlaptop:~/development/notes$ aws ec2 describe-route-tables --filters "Name=route.transit-gatewa
y-id,Values=tgw-0acd402fce62a318c"
        {
            "Associations": [
                {
                    "Main": true,
                    "RouteTableAssociationId": "rtbassoc-0478927546b9f0649",
                    "RouteTableId": "rtb-074e5a02bacb71f1f",
                    "AssociationState": {
                        "State": "associated"
                    }
                }
            ],
            "PropagatingVgws": [],
            "RouteTableId": "rtb-074e5a02bacb71f1f",
            "Routes": [
                {
                    "DestinationCidrBlock": "172.16.0.0/16",
                    "GatewayId": "local",
                    "Origin": "CreateRouteTable",
                    "State": "active"
                },
                {
                    "DestinationCidrBlock": "172.17.0.0/16",
                    "TransitGatewayId": "tgw-0acd402fce62a318c",
                    "Origin": "CreateRoute",
                    "State": "active"
                },
                {
                    "DestinationCidrBlock": "0.0.0.0/0",
```

Figure 19: Exercise 4.8 (4)

```
garre@garrettlaptop:~/development/notes$ aws ec2 create-transit-gateway-route --destination-cidr-block 172.
16.100.64/29 --transit-gateway-route-table-id tgw-rtb-0a85014347c9c970d --blackhole
{
    "Route": {
        "DestinationCidrBlock": "172.16.100.64/29",
        "Type": "static",
        "State": "blackhole"
    }
}
```

Figure 20: Exercise 4.9 (1)

Figure 21: Exercise 4.9 (2)

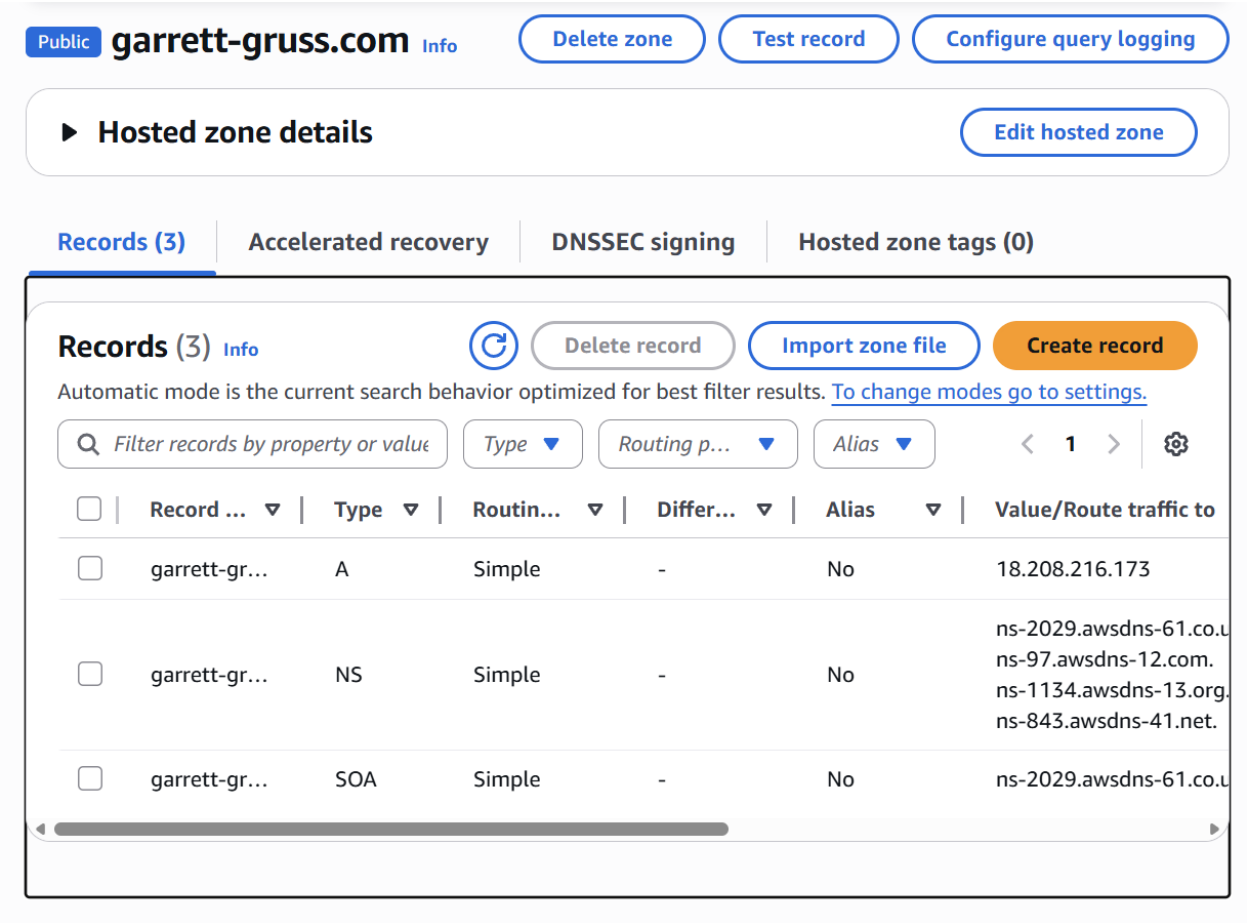

Figure 22: Exercise 8.1

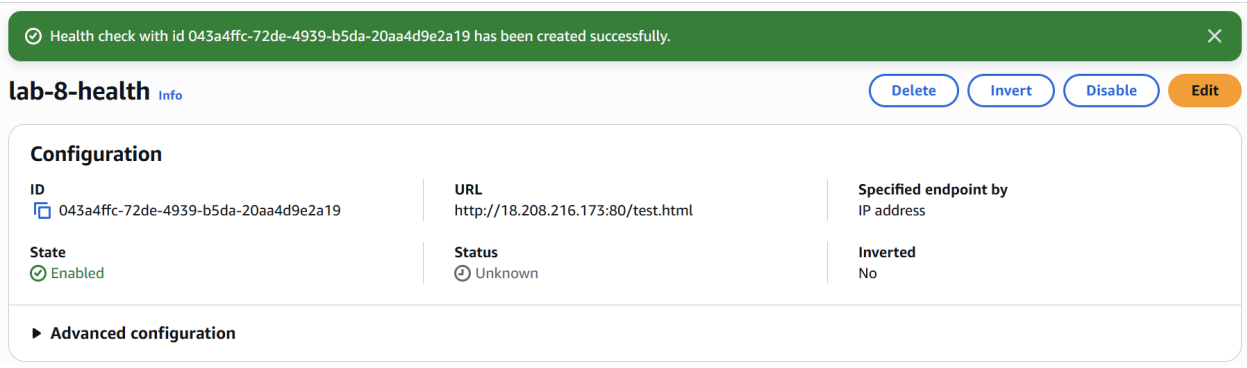

Figure 23: Exercise 8.2

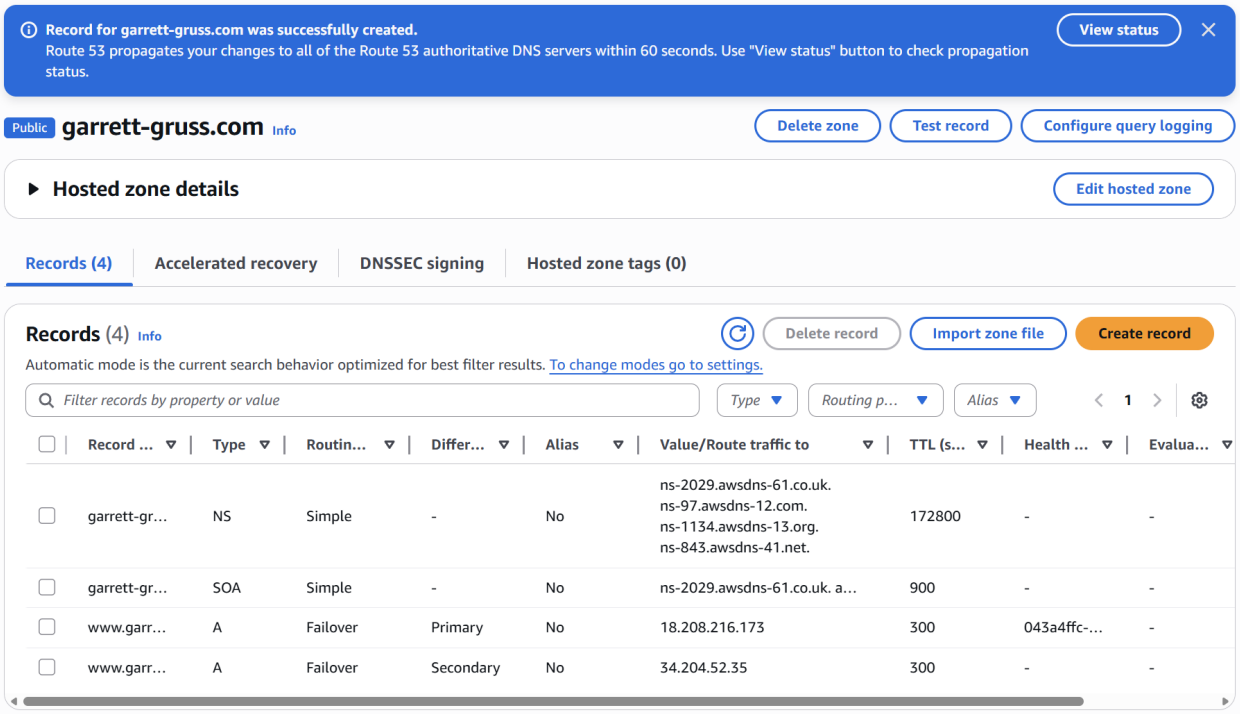

Figure 24: Exercise 8.3

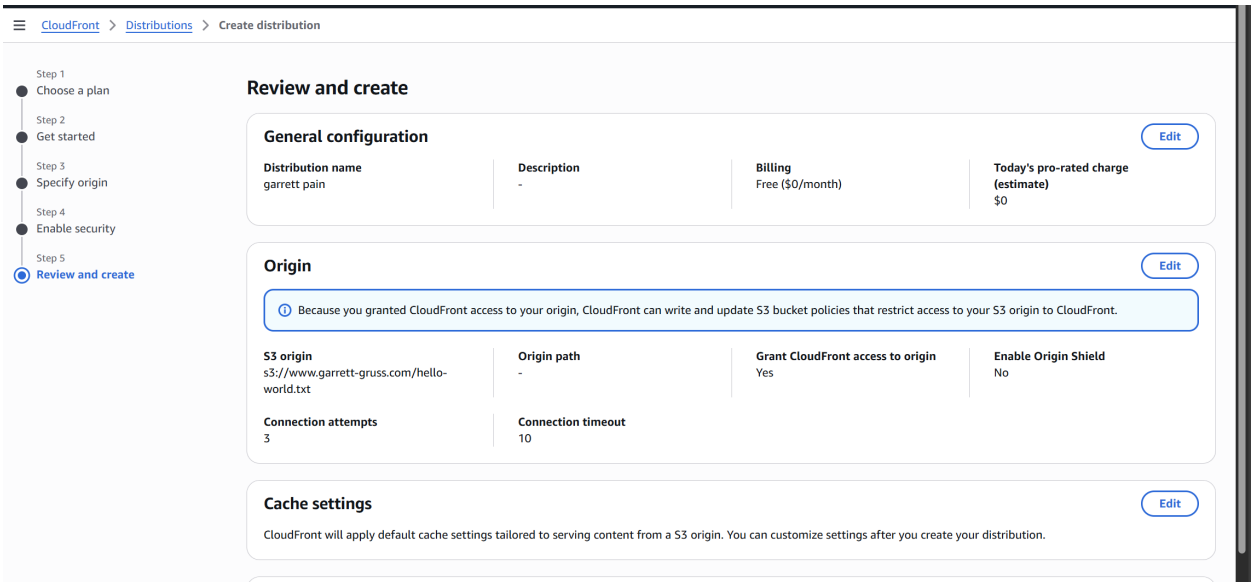

Figure 25: Exercise 8.4 (1)

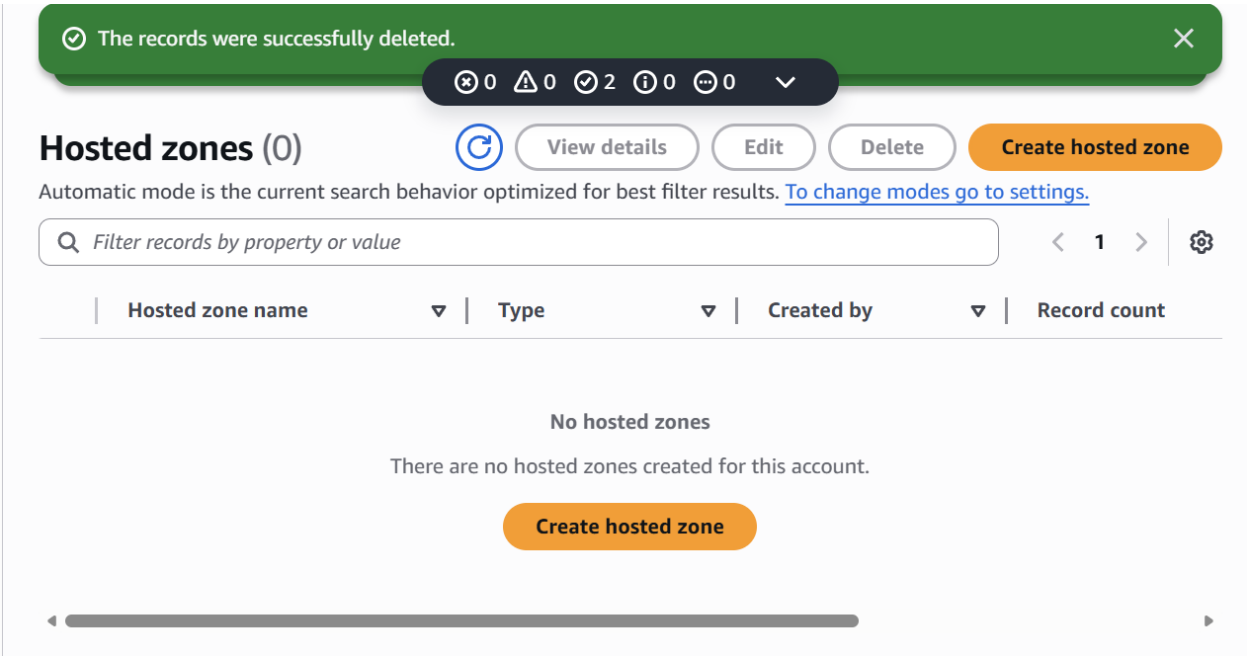

Figure 26: Exercise 8.4 (2)