Centralized DNS management of hybrid cloud with Amazon Route 53 and AWS Transit Gateway

A successful hybrid networking strategy goes beyond private network connectivity. It often requires dealing with independent internal zones both in Amazon Virtual Private Cloud (Amazon VPC) and on-premises. Such a strategy needs Domain Name System (DNS) naming that spans the entire network. Typically, this is managed by providing name resolution services in the same place that the resources and workloads are located. For example, this happens in places such as on-premises DNS infrastructure that answers queries for on-premises resources and Amazon Route 53 Private DNS Resolver that answers queries for resources in the AWS Cloud. To get the unified view of DNS across this hybrid network that your applications require, you might need bi-directional query forwarding. This gives you the ability to query on-premises internal zones from within your Amazon VPCs, as well as to query Route 53 Resolver private DNS from on-premises data centers.

AWS released Amazon [Route 53 Resolver](https://aws.amazon.com/blogs/aws/new-amazon-route-53-resolver-for-hybrid-clouds/) for hybrid cloud in November 2018. This makes migration to cloud and hybrid architectures easier by solving many DNS challenges. These challenges might not be evident when you’re first planning your journey to the cloud, but they frequently appear in the middle of a migration, and they can slow your developers down.

Many of you have your infrastructure spread over multiple accounts and multiple Amazon VPCs where each account’s VPC has several domains and private hosted zones. Integrating DNS across accounts and on-premises DNS servers can get complex. You’ll need to have simple management of recursive DNS behavior across the accounts of multiple VPCs by means of a unified set of rules and a single Route 53 Resolver. You can achieve this today by centralizing DNS management for all VPCs and on-premises domains.

This blog, will go over best practices and trade-offs to consider while designing centrally managed DNS. From a high-level, we will describe the recommended approach for managing DNS across multiple inter-connected Amazon VPCs.

### Single view of DNS for on-premises networks and Amazon VPCs

In this situation, you have an existing infrastructure of on-premises DNS resolvers that are authoritative for resolving on-premises domain names. You also have multiple VPCs across multiple accounts in AWS with domains and private hosted zones associated with these VPCs. These Amazon VPCs are inter-connected using hub and spoke topology via the Transit Gateway. To simplify DNS across this hybrid environment, you can implement a centrally managed DNS which can resolve DNS names between your on-premises data center and VPCs.

Route 53 Resolver endpoints and forwarding rules are designed primarily to offer integrated DNS for hybrid networks where real-time forwarding of DNS queries is often unavoidable. These new features allow Route 53 Resolver and your DNS resolvers on premises to each resolve domains hosted by the other by forwarding queries to each other in real time.

### Sharing PrivateLink endpoints between Amazon VPCs

In this setup, you are using [AWS PrivateLink](https://aws.amazon.com/privatelink/) to securely access AWS services over private connectivity from VPCs and on-premises servers. You need [interface endpoints](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-endpoints.html) that serve as entry points for the traffic destined to AWS services powered by AWS PrivateLink. You can configure interface endpoints in each VPC to access AWS services, however, you will be billed for each hour that your endpoint remains provisioned in each Availability Zone. To reduce costs and simplify administrative overhead of managing interface endpoints in each VPC you can launch PrivateLink endpoints in a shared services VPC and share it with other VPCs. We’ll go over the details of how-to best design DNS resolution across VPCs for centralized PrivateLink.

With the introduction of the Route 53 Resolver service, you can achieve the following design goals:

1. You can use a fully managed service, where AWS does the undifferentiated heavy lifting of availability and reliability.
2. The service preserves Availability Zone isolation and Availability Zone local answers.
3. Inbound and outbound endpoints support 10000 queries per second per IP address in an endpoint. You only have to query the endpoints for forwarded traffic.
4. The service provides [query volume](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/monitoring-resolver-with-cloudwatch.html) metrics in Amazon CloudWatch which allows you to set alarms.

Now, let’s go through some of the best practices for Route 53 Resolver:

1. We recommend that regular Amazon EC2 instances should use .2 for DNS resolution, which gives each EC2 instance a maximum of 1024 [packets per second](https://docs.aws.amazon.com/vpc/latest/userguide/vpc-dns.html#vpc-dns-limits) (pps) per network interface and preserves Availability Zone isolation.
2. You should ensure that the domain name servers in your DHCP option sets point to AmazonProvidedDNS rather than pointing at resolver endpoints. Use forwarding rules to ensure AmazonProvidedDNS has the view of DNS you need.
3. Resolver endpoints are designed to integrate Route 53 Resolver with on-premises DNS.
4. DNS resolution is critical to almost every application, so we always recommend designing for reliability. Query forwarding adds additional complexity to real-time DNS resolution, so the best practice is to use it only when necessary.
5. Although it is possible to use forwarding rules to resolve private hosted zones in other VPCs, we do not recommend that. The most reliable, performant and low-cost approach is to share and associate private hosted zones directly to all VPCs that need them.

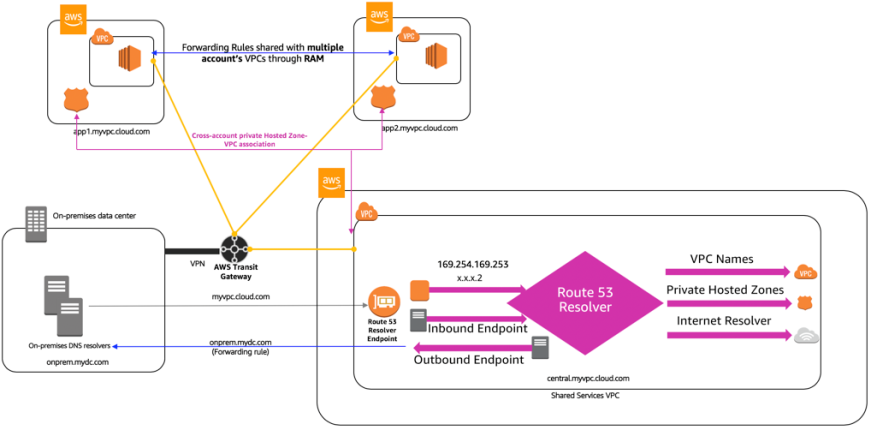
Let’s dive deep into some design setup and configuration examples for centrally managed DNS using Route 53 Resolver.

## Single view of DNS for on-premises networks and Amazon VPCs

In this setup you want to query Route 53 private hosted zone resolution across multiple accounts, and VPC’s from your resources on-premises. In this design setup you will use a shared services VPC to accomplish this. At the same time, you also want to conditionally forward queries for on-premises domains from the VPCs to the on-premises DNS resolver. These VPCs are inter-connected using a hub and spoke topology. Each of the spoke VPCs belongs to a different account, and they are managed by their respective accounts.

When a Route 53 [private hosted zone](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zones-private.html) needs to be resolved in multiple VPCs and AWS accounts as described earlier, the most reliable pattern is to share the private hosted zone between accounts and associate it to each VPC that needs it. Although it’s possible to use Route 53 Resolver forwarding to solve this use case, this introduces additional costs, possible inter-Availability Zone dependencies, and complexity, which directly associating zones avoids.

The following diagram shows the architecture for this setup.



To create this setup, follow these high-level steps.

1. Establish network IP connectivity.

* Establish network connectivity from spoke VPCs to the shared services VPC by using [AWS Transit Gateway](https://docs.aws.amazon.com/vpc/latest/tgw/what-is-transit-gateway.html).
* Establish connectivity to on-premises servers by using [AWS Direct Connect](https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html) or [AWS Site-to-Site VPN](https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html).

1. Configure Route 53 Resolver endpoints.

* Create an [inbound](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-forwarding-inbound-queries.html) endpoint. We recommend that you specify IP addresses in at least two Availability Zones for high availability in the shared services VPC.
* Create an [outbound](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-forwarding-outbound-queries.html) endpoint. We recommend that you specify IP addresses in at least two Availability Zones for high availability in the shared services VPC.

1. Create private hosted zones.

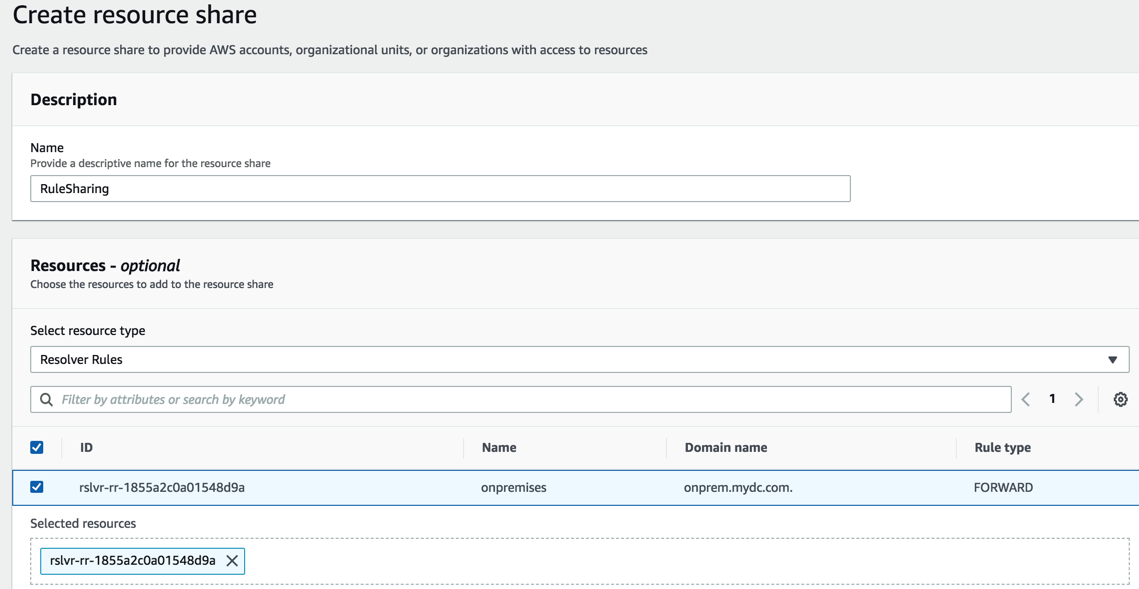
* Create Route 53 [private hosted zones](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zones-private.html) in the shared services VPC and associate them. In addition, complete the [cross-account private hosted zone-VPC association](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zone-private-associate-vpcs-different-accounts.html) of the spoke VPCs because the spoke VPCs are in different accounts. All VPC’s will need to associate their private hosted zones to all other VPC’s if required to.

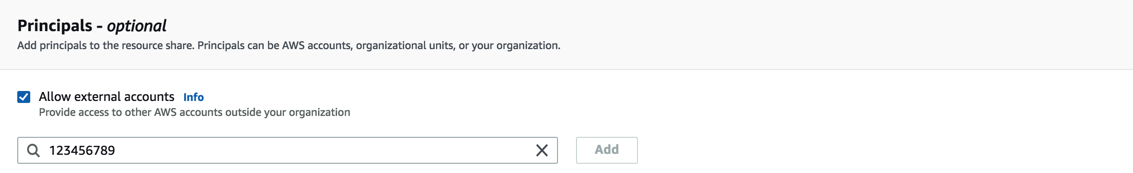
1. Create forwarding rules.

* Create [conditional forwarding rules](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-rules-managing.html#resolver-rules-managing-creating-rules) that specify the on-premises domain names of the DNS queries that you want to forward to on-premises DNS resolvers.  
  Example: A rule with domain name “onprem.mydc.com” will cause all queries for onprem.mydc.com and subdomains to be forwarded.
* When you configure the rule for outbound traffic, set **VPCs that use this rule** to be the shared service.
* If the spoke VPCs are in the same account as the shared services VPC, then you can choose multiple VPCs as **VPCs that use this rule** at the time you create the outbound rule or you can associate VPCs to the rule after creation.

1. Sharing forwarding rules with other AWS accounts and using shared rules.

* In this setup the spoke VPCs are across different accounts, you can share the forwarding rules with the VPCs in other AWS accounts using [shared rules](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-rules-managing.html#resolver-rules-managing-sharing) through [AWS Resource Access Manager](https://docs.aws.amazon.com/ram/latest/userguide/what-is.html) (AWS RAM). The following screenshot shows forwarding resolver rules shared with VPCs for multiple accounts using AWS RAM:





## Sharing PrivateLink endpoints between VPCs

The following setup is often complementary to the single view of DNS described earlier. In this setup, you want to centralize PrivateLink for securely accessing AWS services in the shared services VPC. You want to connect to PrivateLink endpoints in the shared services VPC from other spoke VPCs and on-premises servers and therefore need DNS resolution to work throughout.

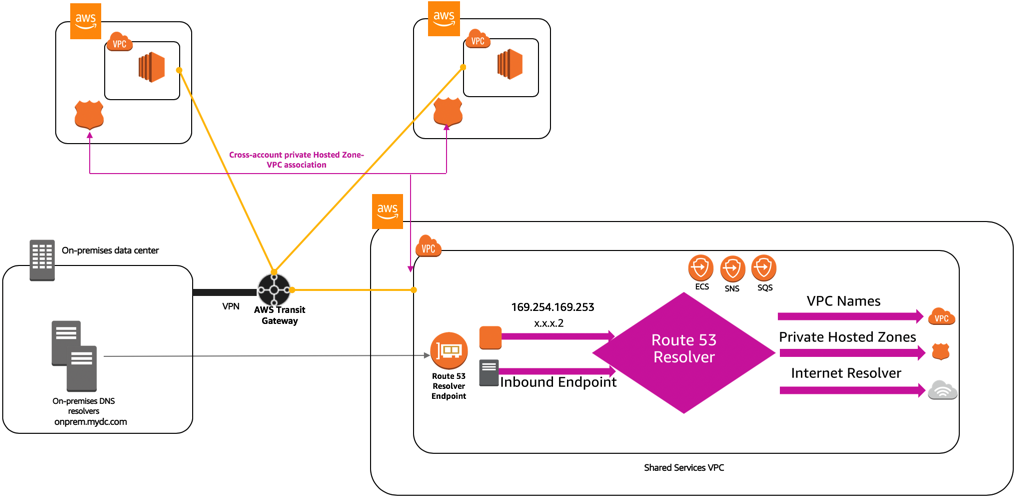
All of the spoke VPCs belong to different accounts and are managed by their respective account owners.

### Interface endpoint private DNS

At creation of an interface endpoint, you receive an endpoint-specific regional DNS hostname that includes a unique endpoint identifier, service identifier, the Region, and vpce.amazonaws.com in its name. For example, vpce-12345678-abcdefg.sqs.us-east-1.vpce.amazonaws.com. This name resolves to the IP addresses of the endpoint in the shared services VPC. The default DNS hostname for this service is sqs.us-east-1.amazonaws.com. Resolution of this DNS hostname results in a public IP address.

Enabling [Private DNS](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-interface.html#vpce-private-dns) for the interface endpoint creates a private hosted zone so that the default name (sqs.us-east-1.amazonaws.com) resolves to the endpoint IP addresses. However, today this only works in the VPC where the endpoint resides. Other VPCs continue to resolve the public IP address. The following steps describe how to ensure that all VPCs and on-premises servers resolve the private endpoint IP addresses.

The following diagram shows the architecture for this setup where we have illustrated PrivateLink endpoints to access some of the AWS services like [Amazon ECS](https://docs.aws.amazon.com/AmazonECS/latest/developerguide/vpc-endpoints.html), [Amazon SNS](https://docs.aws.amazon.com/sns/latest/dg/sns-publishing-to-topics-from-vpc.html) and [Amazon SQS](https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-vpc-endpoints.html). This can be extended to other AWS services that are supported by PrivateLink.



To create this setup, follow these steps.

1. Establish Network IP Connectivity.

* Establish network connectivity from spoke VPCs to the shared services VPC by using [AWS Transit Gateway](https://docs.aws.amazon.com/vpc/latest/tgw/what-is-transit-gateway.html).
* Establish connectivity to on-premises servers by using [AWS Direct Connect](https://docs.aws.amazon.com/directconnect/latest/UserGuide/Welcome.html) or [AWS Site-to-Site VPN](https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html).

1. Configure Route 53 Resolver endpoints.

* Create an [inbound](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resolver-forwarding-inbound-queries.html) endpoint. We recommend that you specify IP addresses in at least two Availability Zones for high availability in the shared services VPC.

1. Create a PrivateLink endpoint.

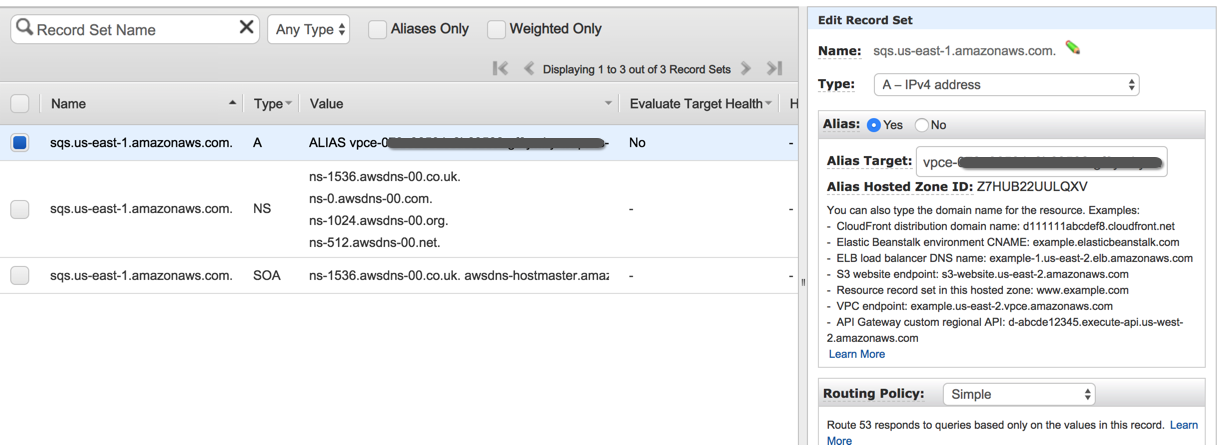
* Create an [interface endpoint](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-interface.html#create-interface-endpoint) for AWS service and do NOT enable **Enable Private DNS Name**.

1. Create private hosted zones.

* If you want to access the interface endpoint sqs.us-east-1.amazonaws.com in the shared services VPC from spoke VPCs and on-premises servers, do the following:

a. Create a Route 53 [private hosted zone](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/hosted-zones-private.html) called sqs.us-east-1.amazonaws.com associated to the shared services VPC.

b. Create an [alias record](https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/resource-record-sets-choosing-alias-non-alias.html) called sqs.us-east-1.amazonaws.com that targets the endpoint-specific [regional DNS hostname](https://docs.aws.amazon.com/vpc/latest/userguide/vpce-interface.html#access-service-though-endpoint) (vpce-12345678-abcdefg.sqs.us-east-1.vpce.amazonaws.com).



* Complete the cross-account private hosted zone-VPC association to the spoke VPCs.

### Important considerations

The above setup provides a solution in centralizing DNS management for PrivateLink, but does not necessarily solve problems for AWS services like Amazon Elastic File System (Amazon EFS), which relies on Availability Zone-specific answers and doesn’t currently offer the ability to create an ALIAS record.

Although you can create forwarding rules for Amazon EFS domain names across accounts through outbound endpoints, clients will not receive answers optimized for their Availability Zone. This might result in cross-Availability Zone networking charges, higher operation latencies, and less durability. We recommend only forwarding Availability Zone-specific EFS domain names using this technique, taking special care with each client to mount EFS using the DNS entry specific to their Availability Zone.

If you are mounting across accounts, be aware that the Availability Zone name (e.g., us-east-1a) might not be the same from one account to another. You might need to map Availability Zone names to Availability Zone IDs. This is a unique and consistent identifier for an Availability Zone that ensures that the correct connections are made. For example, use1-az1 is an Availability Zone ID for the us-east-1 Region, and it has the same location in every AWS account. Route 53 Resolver endpoints are specifically designed for the hybrid DNS setup, and they are not built for the Amazon EFS like use case.