DNS in AWS: EC2 and Load Balancers (ELB)

Simply Put: DNS (Domain Name System) is the internet's phone book, translating names (like google.com) into numbers (IP addresses). In AWS, this "phone book" is managed by Route 53. We'll look at how your AWS servers (EC2) look up names and how your traffic balancers (ELB) use names to manage traffic.

---

1. How Your EC2 Server Finds an IP Address (The Internal Flow)

Every time your EC2 instance needs to connect to another service using a name (like database.internal or www.google.com), it asks the AWS built-in resolver.

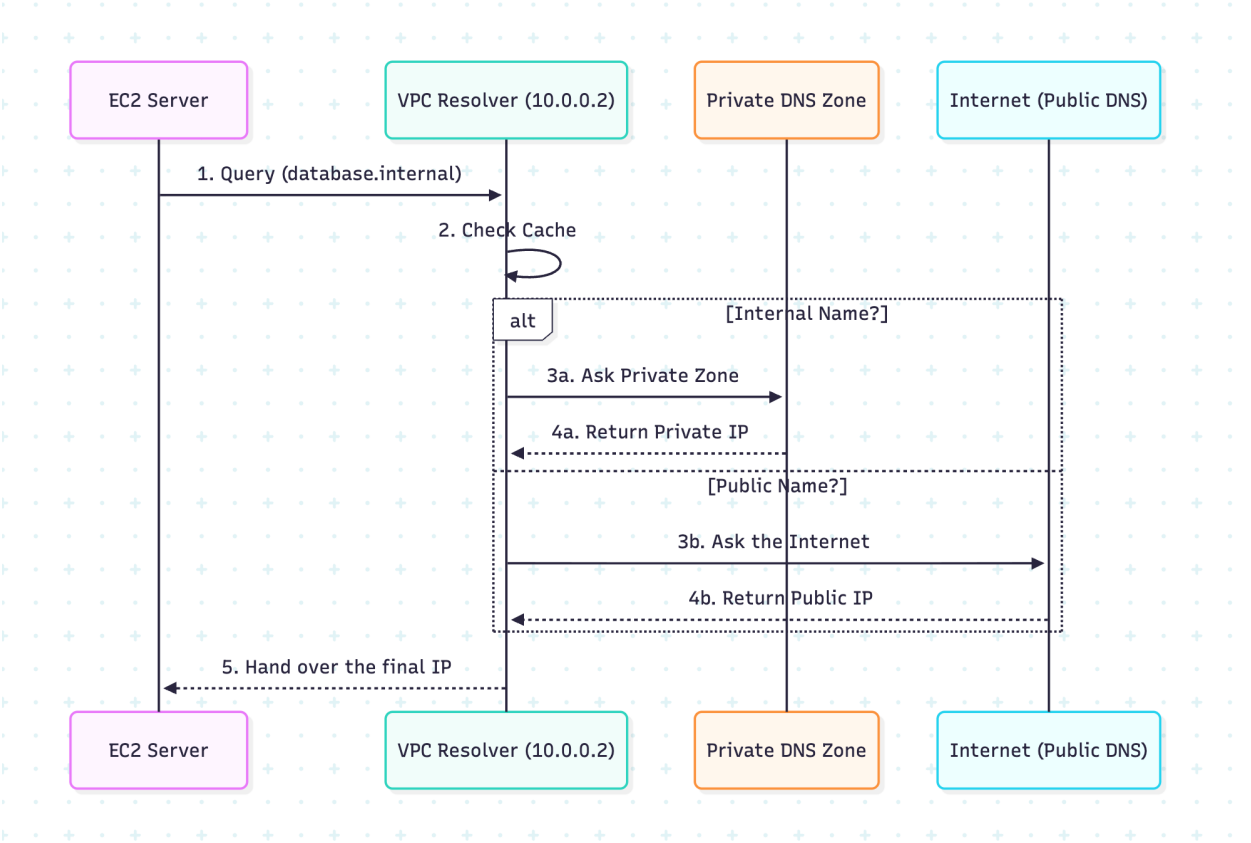
1.1 The AWS DNS "Phone Book" Structure

- VPC Resolver (AmazonProvidedDNS): This is the invisible, highly reliable DNS server that AWS automatically gives to every server in your VPC (your private cloud network).
    - Its Address: It sits at the IP address VPC CIDR+2 (e.g., 10.0.0.2) or 169.254.169.253.
- Route 53 Private Zones: This is your $\mathbf{internal$ phone book. The Resolver uses this to look up private names you create (e.g., api.internal).
- Route 53 Public Zones: This is your public phone book, visible to the internet.

1.2 Step-by-Step Resolution Flow

Imagine your EC2 server is asking, "What is the IP address for database.internal?"

1. EC2 Asks: Your EC2 server sends the query to the VPC Resolver.
2. Resolver Checks (Internal): The Resolver first checks if the name belongs to a service inside your VPC.
    - If it finds a match in a Private Zone, it immediately returns the private IP (10.x.x.x).
3. Resolver Checks (External): If the name is public (google.com), the Resolver forwards the question out to the internet's global DNS servers.
4. IP Returned: The Resolver finally hands the correct IP address back to your EC2 server.



2. How ELBs Use DNS (The Load Balancer Trick)

Load Balancers (ELBs) distribute traffic, but they are constantly changing their own IP addresses as they scale up and down. They can't use a fixed IP (like your phone number). Instead, they are given a DNS Name that dynamically points to their current set of IPs.

ELB Type	**What Does its Name Point To?**	**Key DNS Difference**
ALB (Application)	Multiple, Dynamic IPs.	The IPs change often as the ALB scales. Clients must check for new IPs frequently!
CLB (Classic)	Multiple, Dynamic IPs.	Similar to ALB, IPs change as the balancer scales.
NLB (Network)	Static (Fixed) IP per AZ.	The only type that provides a stable IP address, often used for white-listing or external firewalls.

The Load Balancer Trick: ALB and CLB rely on the fact that you check the phone book often. When the ELB needs to add a new server, it updates the list of IPs under its name. If you check often, you'll see the new list and start sending traffic to the new server immediately.

---

3. TTL, Caching, and Failover Explained

Time-to-Live (TTL)

- TTL: This is the time (in seconds) that your computer or a DNS server is allowed to cache (remember) a resolved IP address.
- ALB/CLB Rule: ALB and CLB use an extremely low TTL (often < 60 seconds). This is critical because it forces clients to re-check the DNS name frequently, preventing them from connecting to an old, decommissioned IP address.
- Route 53 Alias Record: When you point your custom domain (e.g., myapp.com) to an ELB using an Alias record, Route 53 automatically uses the correct, low ELB TTL.

Failover (Disaster Recovery)

The best way to handle failures across regions is using Route 53 Failover Routing:

- Primary/Secondary: You set up a Primary ELB in one region (e.g., US-East) and a Secondary (backup) ELB in another (e.g., US-West).
- Health Check: Route 53 continuously runs a Health Check on the Primary ELB.
- Automatic Switch: If the Health Check fails, Route 53 automatically changes the DNS record to point all traffic to the Secondary ELB.

---

4. Simple Troubleshooting and Mitigation

Problem: Clients stick to old ELB IPs.

Scenario: You scaled down your ALB, but some users are still trying to connect to a retired IP address and getting connection timeouts.

The Cause (The Stale Cache): A caching DNS server (like a corporate server or an old client OS cache) is ignoring the low TTL and holding onto the old, retired IP for too long.

The Solution (Forcing the Check):

1. Enforce Low TTL: For any DNS record pointing to your ELB (especially outside of AWS), set its TTL to the lowest possible value (e.g., 60 seconds).
2. Educate Clients: Ensure your internal clients are configured to honor external TTLs or ask them to flush their local DNS cache (e.g., using ipconfig /flushdns on Windows).
3. Use Retry Logic: Make sure your application code has a fast connection timeout and retry to quickly move past a stale IP.

---

5. Boto3 Code Example (For Engineers)

This code snippet shows how an engineer would use the AWS boto3 library to check the DNS record that points to your ELB inside Route 53.

```python
import boto3

# Configuration variables
HOSTED_ZONE_ID = "Z1A2B3C4D5E6F7" # Example Private or Public Hosted Zone ID
DOMAIN_NAME = "api.internal.example.com." # FQDN of the ELB/Alias record

def query_elb_dns_target(zone_id, domain_name):
    """Retrieves the DNS target (ELB name) of an Alias record in Route 53."""
    try:
        client = boto3.client('route53')
        response = client.list_resource_record_sets(
            HostedZoneId=zone_id,
            StartRecordName=domain_name,
            StartRecordType='A',
            MaxItems='1'
        )

        for record_set in response.get('ResourceRecordSets', []):
            if record_set['Name'] == domain_name and 'AliasTarget' in record_set:
                target_elb_dns = record_set['AliasTarget']['DNSName']
                print(f"✅ Resolved Alias Target for {domain_name}:")
                print(f"   Target ELB DNS Name: {target_elb_dns}")
                return target_elb_dns

        print(f"❌ Alias record not found for {domain_name}")
        return None

    except Exception as e:
        print(f"An error occurred: {e}")
        return None

if __name__ == '__main__':
    query_elb_dns_target(HOSTED_ZONE_ID, DOMAIN_NAME)
```

---

6. Quick Takeaways

- EC2 DNS is Automatic: Your server always uses the VPC Resolver (VPC CIDR+2).
- ALB/CLB IPs Change: They use a name that points to a dynamic list of IPs for scaling.
- NLB IPs are Static: Use NLB when you need a fixed IP address.
- The Golden Rule is TTL: Always honor the low TTL on ELB DNS records to guarantee fast failover and proper traffic distribution.

References

- VPC DNS:
- Route 53 Policies:
- Routing to ELB: