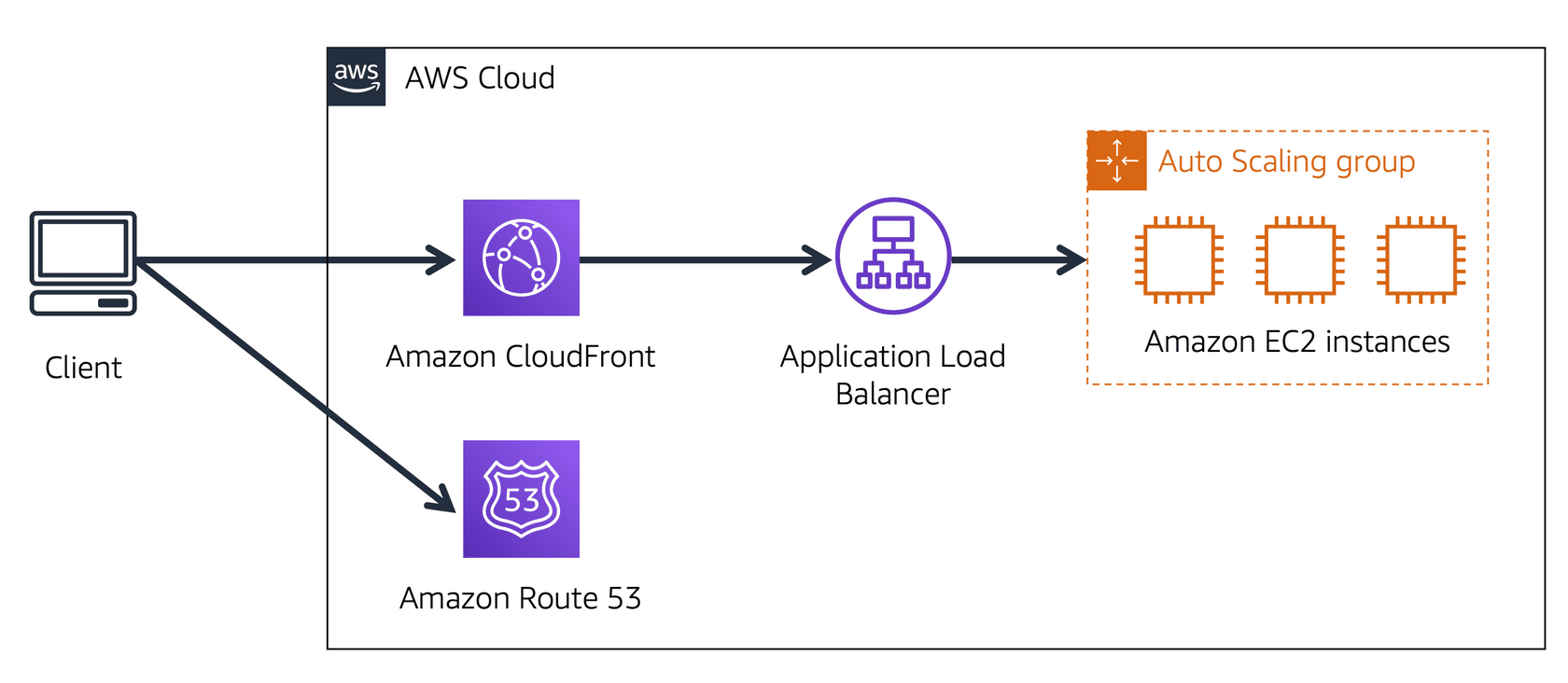
**Amazon Route 53**

[**Amazon Route 53**](https://aws.amazon.com/route53) is a DNS web service. It gives developers and businesses a reliable way to route end users to internet applications hosted in AWS.

Amazon Route 53 connects user requests to infrastructure running in AWS (such as Amazon EC2 instances and load balancers). It can route users to infrastructure outside of AWS.

Another feature of Route 53 is the ability to manage the DNS records for domain names. You can register new domain names directly in Route 53. You can also transfer DNS records for existing domain names managed by other domain registrars. This enables you to manage all of your domain names within a single location.

**Example: How Amazon Route 53 and Amazon CloudFront deliver content**



Suppose that AnyCompany’s application is running on several Amazon EC2 instances. These instances are in an Auto Scaling group that attaches to an Application Load Balancer.

* 1

1

A customer requests data from the application by going to AnyCompany’s website.

* 2

2

Amazon Route 53 uses DNS resolution to identify AnyCompany.com’s corresponding IP address, 192.0.2.0. This information is sent back to the customer.

* 3

3

The customer’s request is sent to the nearest edge location through Amazon CloudFront.

* 4

4

Amazon CloudFront connects to the Application Load Balancer, which sends the incoming packet to an Amazon EC2 instance.

**The AWS Well-Architected Framework**

The [**AWS Well-Architected Framework**](https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf) helps you understand how to design and operate reliable, secure, efficient, and cost-effective systems in the AWS Cloud. It provides a way for you to consistently measure your architecture against best practices and design principles and identify areas for improvement.

The Well-Architected Framework is based on five pillars:

* Operational excellence
* Security
* Reliability
* Performance efficiency
* Cost optimization

**Operational excellence**

–

**Operational excellence** is the ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures.

Design principles for operational excellence in the cloud include performing operations as code, annotating documentation, anticipating failure, and frequently making small, reversible changes.

**Security**

–

The **Security** pillar is the ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies.

When considering the security of your architecture, apply these best practices:

* Automate security best practices when possible.
* Apply security at all layers.
* Protect data in transit and at rest.

**Reliability**

**Reliability** is the ability of a system to do the following:

* Recover from infrastructure or service disruptions
* Dynamically acquire computing resources to meet demand
* Mitigate disruptions such as misconfigurations or transient network issues

Reliability includes testing recovery procedures, scaling horizontally to increase aggregate system availability, and automatically recovering from failure.

**Performance efficiency**

–

**Performance efficiency** is the ability to use computing resources efficiently to meet system requirements and to maintain that efficiency as demand changes and technologies evolve.

Evaluating the performance efficiency of your architecture includes experimenting more often, using serverless architectures, and designing systems to be able to go global in minutes.

**Cost optimization**

–

**Cost optimization** is the ability to run systems to deliver business value at the lowest price point.

Cost optimization includes adopting a consumption model, analyzing and attributing expenditure, and using managed services to reduce the cost of ownership.

Quiz links :

<https://youtu.be/IvvD13aNO68>

<https://youtu.be/X5w7OTapVsY>