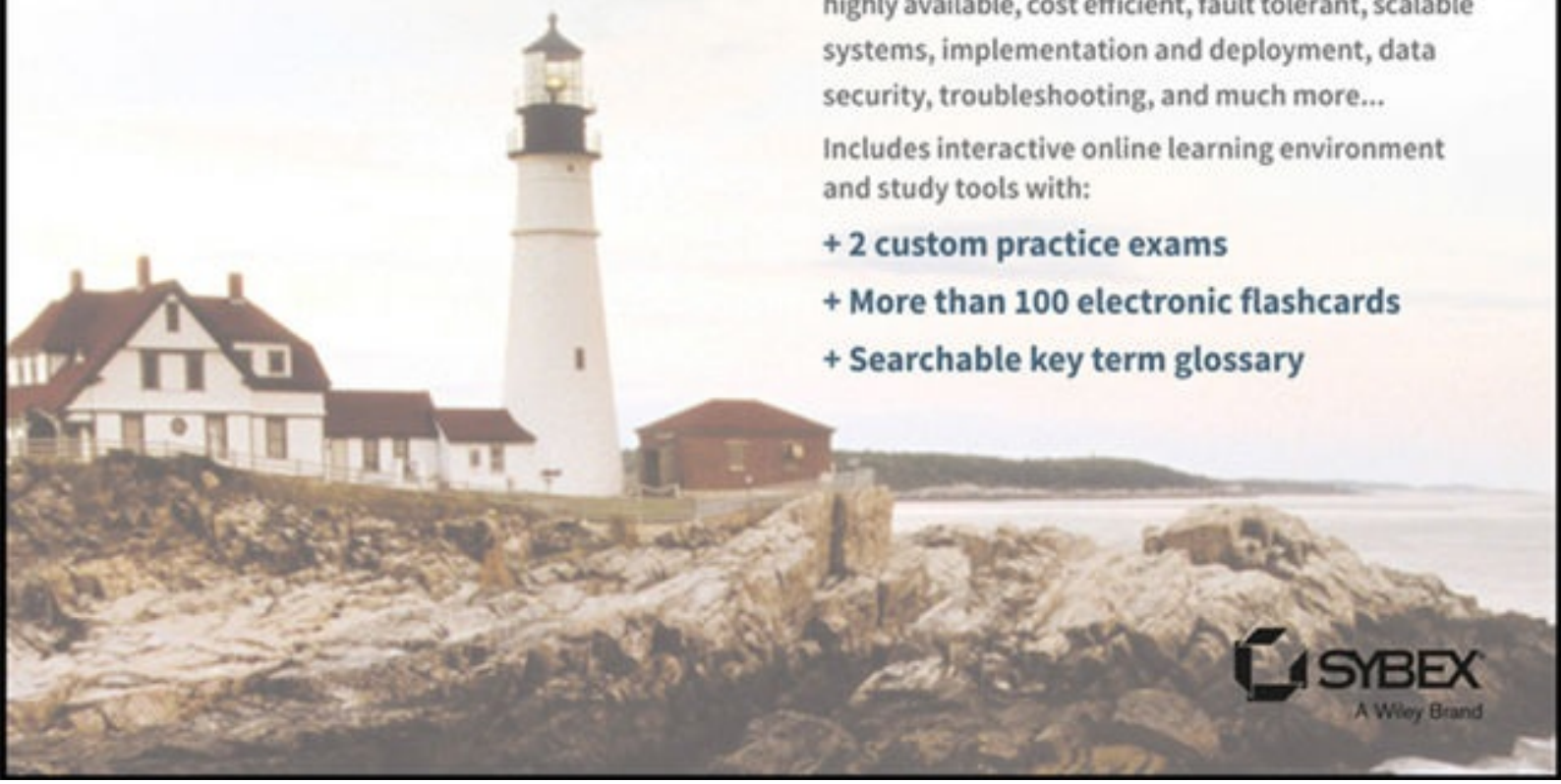
amazon web services

Joe Baron, Hisham Baz, Tim Bixler, Biff Gaut,
Kevin E. Kelly, Sean Senior, John Stamper

AWS Certified Solutions Architect

OFFICIAL STUDY GUIDE

ASSOCIATE EXAM



Covers exam objectives, including designing highly available, cost efficient, fault tolerant, scalable systems, implementation and deployment, data security, troubleshooting, and much more...

Includes interactive online learning environment and study tools with:

+ 2 custom practice exams
+ More than 100 electronic flashcards
+ Searchable key term glossary

SYBEX
A Wiley Brand

network gateways. In addition, organizations can extend their corporate data center networks to AWS by using hardware or software *virtual private network (VPN)* connections or dedicated circuits by using AWS Direct Connect.

AWS Direct Connect

AWS Direct Connect allows organizations to establish a dedicated network connection from their data center to AWS. Using AWS Direct Connect, organizations can establish private connectivity between AWS and their data center, office, or colocation environment, which in many cases can reduce network costs, increase bandwidth throughput, and provide a more consistent network experience than Internet-based VPN connections.

Amazon Route 53

Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. It is designed to give developers and businesses an extremely reliable and cost-effective way to route end users to Internet applications by translating human readable names, such as `www.example.com`, into the numeric IP addresses, such as 192.0.2.1, that computers use to connect to each other. Amazon Route 53 also serves as domain registrar, allowing you to purchase and manage domains directly from AWS.

Storage and Content Delivery

AWS provides a variety of services to meet your storage needs, such as Amazon Simple Storage Service, Amazon CloudFront, and Amazon Elastic Block Store. This section provides an overview of the storage and content delivery services.

Amazon Simple Storage Service (Amazon S3)

Amazon Simple Storage Service (Amazon S3) provides developers and IT teams with highly durable and scalable object storage that handles virtually unlimited amounts of data and large numbers of concurrent users. Organizations can store any number of objects of any type, such as HTML pages, source code files, image files, and encrypted data, and access them using HTTP-based protocols. Amazon S3 provides cost-effective object storage for a wide variety of use cases, including backup and recovery, nearline archive, big data analytics, disaster recovery, cloud applications, and content distribution.

Amazon Glacier

Amazon Glacier is a secure, durable, and extremely low-cost storage service for data archiving and long-term backup. Organizations can reliably store large or small amounts of data for a very low cost per gigabyte per month. To keep costs low for customers, Amazon Glacier is optimized for infrequently accessed data where a retrieval time of several hours is suitable. Amazon S3 integrates closely with Amazon Glacier to allow organizations to choose the right storage tier for their workloads.

Amazon Elastic Block Store (Amazon EBS)

Amazon Elastic Block Store (Amazon EBS) provides persistent block-level storage volumes for use with Amazon EC2 instances. Each Amazon EBS volume is automatically replicated within its Availability Zone to protect organizations from component failure, offering high

availability and durability. By delivering consistent and low-latency performance, Amazon EBS provides the disk storage needed to run a wide variety of workloads.

AWS Storage Gateway

AWS Storage Gateway is a service connecting an on-premises software appliance with cloud-based storage to provide seamless and secure integration between an organization's on-premises IT environment and the AWS storage infrastructure. The service supports industry-standard storage protocols that work with existing applications. It provides low-latency performance by maintaining a cache of frequently accessed data on-premises while securely storing all of your data encrypted in Amazon S3 or Amazon Glacier.

Amazon CloudFront

Amazon CloudFront is a content delivery web service. It integrates with other AWS Cloud services to give developers and businesses an easy way to distribute content to users across the world with low latency, high data transfer speeds, and no minimum usage commitments. Amazon CloudFront can be used to deliver your entire website, including dynamic, static, streaming, and interactive content, using a global network of edge locations. Requests for content are automatically routed to the nearest edge location, so content is delivered with the best possible performance to end users around the globe.

Database Services

AWS provides fully managed relational and NoSQL database services, and in-memory caching as a service and a petabyte-scale data warehouse solution. This section provides an overview of the products that the database services comprise.

Amazon Relational Database Service (Amazon RDS)

Amazon Relational Database Service (Amazon RDS) provides a fully managed relational database with support for many popular open source and commercial database engines. It's a cost-efficient service that allows organizations to launch secure, highly available, fault-tolerant, production-ready databases in minutes. Because Amazon RDS manages time-consuming administration tasks, including backups, software patching, monitoring, scaling, and replication, organizational resources can focus on revenue-generating applications and business instead of mundane operational tasks.

Amazon DynamoDB

Amazon DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit millisecond latency at any scale. It is a fully managed database and supports both document and key/value data models. Its flexible data model and reliable performance make it a great fit for mobile, web, gaming, ad-tech, Internet of Things, and many other applications.

Amazon Redshift

Amazon Redshift is a fast, fully managed, petabyte-scale data warehouse service that makes it simple and cost effective to analyze structured data. Amazon Redshift provides a standard SQL interface that lets organizations use existing business intelligence tools. By leveraging

If you are using Amazon S3 in a GET-intensive mode, such as a static website hosting, for best performance you should consider using an Amazon CloudFront distribution as a caching layer in front of your Amazon S3 bucket.

Chapter 9
Domain Name System (DNS) and Amazon Route 53

THE AWS CERTIFIED SOLUTIONS ARCHITECT EXAM TOPICS COVERED IN THIS CHAPTER MAY INCLUDE, BUT ARE NOT LIMITED TO, THE FOLLOWING:

Domain 1.0: Designing highly available, cost-efficient, fault-tolerant, scalable systems

✓ **1.1 Identify and recognize cloud architecture considerations, such as fundamental components and effective designs.**

Content may include the following:

- How to design cloud services
- Planning and design
- Monitoring and logging
- Familiarity with:
  - Best practices for AWS architecture
  - Developing to client specifications, including pricing/cost (for example, on-demand vs. reserved vs. spot; RTO and RPO DR design)
  - Architectural trade-off decisions (for example, high availability vs. cost, Amazon Relational Database Service [RDS] vs. installing your own database on Amazon Elastic Compute Cloud—EC2)
  - Elasticity and scalability (for example, auto-scaling, SQS, ELB, CloudFront)

Domain 3.0: Data Security

✓ **3.1 Recognize and implement secure procedures for optimum cloud deployment and maintenance.**

✓ **3.2 Recognize critical disaster-recovery techniques and their implementation.**

- Amazon Route 53

Amazon Route 53 Overview

Now that you have a foundational understanding of DNS and the different DNS record types, you can explore Amazon Route 53. *Amazon Route 53* is a highly available and scalable cloud DNS web service that is designed to give developers and businesses an extremely reliable and cost-effective way to route end users to Internet applications.

Amazon Route 53 performs three main functions:

- ***Domain registration***—Amazon Route 53 lets you register domain names, such as `example.com`.
- ***DNS service***—Amazon Route 53 translates friendly domain names like `www.example.com` into IP addresses like `192.0.2.1`. Amazon Route 53 responds to DNS queries using a global network of authoritative DNS servers, which reduces latency. To comply with DNS standards, responses sent over User Datagram Protocol (UDP) are limited to 512 bytes in size. Responses exceeding 512 bytes are truncated, and the resolver must re-issue the request over TCP.
- ***Health checking***—Amazon Route 53 sends automated requests over the Internet to your application to verify that it's reachable, available, and functional.

You can use any combination of these functions. For example, you can use Amazon Route 53 as both your registrar and your DNS service, or you can use Amazon Route 53 as the DNS service for a domain that you registered with another domain registrar.

Domain Registration

If you want to create a website, you first need to register the domain name. If you already registered a domain name with another registrar, you have the option to transfer the domain registration to Amazon Route 53. It isn't required to use Amazon Route 53 as your DNS service or to configure health checking for your resources.

Amazon Route 53 supports domain registration for a wide variety of generic TLDs (for example, `.com` and `.org`) and geographic TLDs (for example, `.be` and `.us`). For a complete list of supported TLDs, refer to the Amazon Route 53 Developer Guide at https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/.

Domain Name System (DNS) Service

As stated previously, Amazon Route 53 is an authoritative DNS service that routes Internet traffic to your website by translating friendly domain names into IP addresses. When someone enters your domain name in a browser or sends you an email, a DNS request is forwarded to the nearest Amazon Route 53 DNS server in a global network of authoritative DNS servers. Amazon Route 53 responds with the IP address that you specified.

If you register a new domain name with Amazon Route 53, Amazon Route 53 will be automatically configured as the DNS service for the domain, and a *hosted zone* will be created for your domain. You add resource record sets to the hosted zone, which define how you want Amazon Route 53 to respond to DNS queries for your domain (for example, with the IP address for a web server, the IP address for the nearest Amazon CloudFront edge location, or

the IP address for an Elastic Load Balancing load balancer).

If you registered your domain with another domain registrar, that registrar is probably providing the DNS service for your domain. You can transfer DNS service to Amazon Route 53, with or without transferring registration for the domain.

If you're using Amazon CloudFront, Amazon Simple Storage Service (Amazon S3), or Elastic Load Balancing, you can configure Amazon Route 53 to route Internet traffic to those resources.

Hosted Zones

A *hosted zone* is a collection of resource record sets hosted by Amazon Route 53. Like a traditional DNS zone file, a hosted zone represents resource record sets that are managed together under a single domain name. Each hosted zone has its own metadata and configuration information.

There are two types of hosted zones: private and public. A *private hosted zone* is a container that holds information about how you want to route traffic for a domain and its subdomains within one or more Amazon Virtual Private Clouds (Amazon VPCs). A *public hosted zone* is a container that holds information about how you want to route traffic on the Internet for a domain (for example, `example.com`) and its subdomains (for example, `apex.example.com` and `acme.example.com`).

The resource record sets contained in a hosted zone must share the same suffix. For example, the `example.com` hosted zone can contain resource record sets for the `www.example.com` and `www.aws.example.com` subdomains, but it cannot contain resource record sets for a `www.example.ca` subdomain.

 You can use Amazon S3 to host your static website at the hosted zone (for example, `domain.com`) and redirect all requests to a subdomain (for example, `www.domain.com`). Then, in Amazon Route 53, you can create an alias resource record that sends requests for the root domain to the Amazon S3 bucket.

 Use an alias record, not a CNAME, for your hosted zone. CNAMEs are not allowed for hosted zones in Amazon Route 53.

 Do not use A records for subdomains (for example, `www.domain.com`), as they refer to hardcoded IP addresses. Instead, use Amazon Route 53 alias records or traditional CNAME records to always point to the right resource, wherever your site is hosted, even when the physical server has changed its IP address.

- The application's failover environment (for example, `fail.domain.com`) has an Amazon Route 53 alias record that points to an Amazon CloudFront distribution of an Amazon S3 bucket hosting a static version of the application.
- The application's subdomain (for example, `www.domain.com`) has an Amazon Route 53 alias record that points to `prod.domain.com` (as primary target) and `fail.domain .com` (as secondary target) using a failover routing policy. This ensures `www.domain.com` routes to the production load balancers if at least one of them is healthy or the "fail whale" if all of them appear to be unhealthy.
- The application's hosted zone (for example, `domain.com`) has an Amazon Route 53 alias record that redirects requests to `www.domain.com` using an Amazon S3 bucket of the same name.
- Application content (both static and dynamic) can be served using Amazon CloudFront. This ensures that the content is delivered to clients from Amazon CloudFront edge locations spread all over the world to provide minimal latency. Serving dynamic content from a Content Delivery Network (CDN), where it is cached for short periods of time (that is, several seconds), takes the load off of the application and further improves its latency and responsiveness.
- The application is deployed in multiple AWS regions, protecting it from a regional outage.

Storage and Content Delivery

This section covers two additional storage and content delivery services that are important for a Solutions Architect to understand: Amazon CloudFront and AWS Storage Gateway.

Amazon CloudFront

Amazon CloudFront is a global Content Delivery Network (CDN) service. It integrates with other AWS products to give developers and businesses an easy way to distribute content to end users with low latency, high data transfer speeds, and no minimum usage commitments.

Overview

A *Content Delivery Network (CDN)* is a globally distributed network of caching servers that speed up the downloading of web pages and other content. CDNs use Domain Name System (DNS) *geo-location* to determine the geographic location of each request for a web page or other content, then they serve that content from edge caching servers closest to that location instead of the original web server. A CDN allows you to increase the scalability of a website or mobile application easily in response to peak traffic spikes. In most cases, using a CDN is completely transparent—end users simply experience better website performance, while the load on your original website is reduced.

Amazon CloudFront is AWS CDN. It can be used to deliver your web content using Amazon's global network of *edge locations*. When a user requests content that you're serving with Amazon CloudFront, the user is routed to the edge location that provides the lowest latency (time delay), so content is delivered with the best possible performance. If the content is already in the edge location with the lowest latency, Amazon CloudFront delivers it immediately. If the content is not currently in that edge location, Amazon CloudFront retrieves it from the *origin server*, such as an Amazon Simple Storage Service (Amazon S3) bucket or a web server, which stores the original, definitive versions of your files.

Amazon CloudFront is optimized to work with other AWS cloud services as the origin server, including Amazon S3 buckets, Amazon S3 static websites, Amazon Elastic Compute Cloud (Amazon EC2), and Elastic Load Balancing. Amazon CloudFront also works seamlessly with any non-AWS origin server, such as an existing on-premises web server. Amazon CloudFront also integrates with Amazon Route 53.

Amazon CloudFront supports all content that can be served over HTTP or HTTPS. This includes any popular static files that are a part of your web application, such as HTML files, images, JavaScript, and CSS files, and also audio, video, media files, or software downloads. Amazon CloudFront also supports serving dynamic web pages, so it can actually be used to deliver your entire website. Finally, Amazon CloudFront supports media *streaming*, using both HTTP and RTMP.

Amazon CloudFront Basics

There are three core concepts that you need to understand in order to start using CloudFront: distributions, origins, and cache control. With these concepts, you can easily use CloudFront to speed up delivery of static content from your websites.

Distributions To use Amazon CloudFront, you start by creating a *distribution*, which is identified by a DNS domain name such as d111111abcdef8.cloudfront.net. To serve files from Amazon CloudFront, you simply use the distribution domain name in place of your website's domain name; the rest of the file paths stay unchanged. You can use the Amazon CloudFront distribution domain name as-is, or you can create a user-friendly DNS name in your own domain by creating a CNAME record in Amazon Route 53 or another DNS service. The CNAME is automatically redirected to your Amazon CloudFront distribution domain name.

Origins When you create a distribution, you must specify the DNS domain name of the *origin*—the Amazon S3 bucket or HTTP server—from which you want Amazon CloudFront to get the definitive version of your objects (web files). For example:

- **Amazon S3 bucket:** `myawsbucket.s3.amazonaws.com`
- **Amazon EC2 instance:** `ec2-203-0-113-25.compute-1.amazonaws.com`
- **Elastic Load Balancing load balancer:** `my-load-balancer-1234567890.us-west-2.elb.amazonaws.com`
- **Website URL:** `mywebserver.mycompanydomain.com`

Cache Control Once requested and served from an edge location, objects stay in the cache until they expire or are evicted to make room for more frequently requested content. By default, objects expire from the cache after 24 hours. Once an object expires, the next request results in Amazon CloudFront forwarding the request to the origin to verify that the object is unchanged or to fetch a new version if it has changed.

Optionally, you can control how long objects stay in an Amazon CloudFront cache before expiring. To do this, you can choose to use Cache-Control headers set by your origin server or you can set the minimum, maximum, and default *Time to Live (TTL)* for objects in your Amazon CloudFront distribution.

You can also remove copies of an object from all Amazon CloudFront edge locations at any time by calling the *invalidation* Application Program Interface (API). This feature removes the object from every Amazon CloudFront edge location regardless of the expiration period you set for that object on your origin server. The invalidation feature is designed to be used in unexpected circumstances, such as to correct an error or to make an unanticipated update to a website, not as part of your everyday workflow.

Instead of invalidating objects manually or programmatically, it is a best practice to use a version identifier as part of the object (file) path name. For example:

- **Old file:** `assets/v1/css/narrow.css`
- **New file:** `assets/v2/css/narrow.css`

When using versioning, users always see the latest content through Amazon CloudFront when you update your site without using invalidation. Old versions will expire from the cache automatically.

Amazon CloudFront Advanced Features

CloudFront can do much more than simply serve static web files. To start using CloudFront's advanced features, you will need to understand how to use cache behaviors, and how to

restrict access to sensitive content.

Dynamic Content, Multiple Origins, and Cache Behaviors Serving static assets, such as described previously, is a common way to use a CDN. An Amazon CloudFront distribution, however, can easily be set up to serve dynamic content in addition to static content and to use more than one origin server. You control which requests are served by which origin and how requests are cached using a feature called *cache behaviors*.

A cache behavior lets you configure a variety of Amazon CloudFront functionalities for a given URL path pattern for files on your website. For example see Figure 11.1. One cache behavior applies to all PHP files in a web server (dynamic content), using the path pattern `*.php`, while another behavior applies to all JPEG images in another origin server (static content), using the path pattern `*.jpg`.

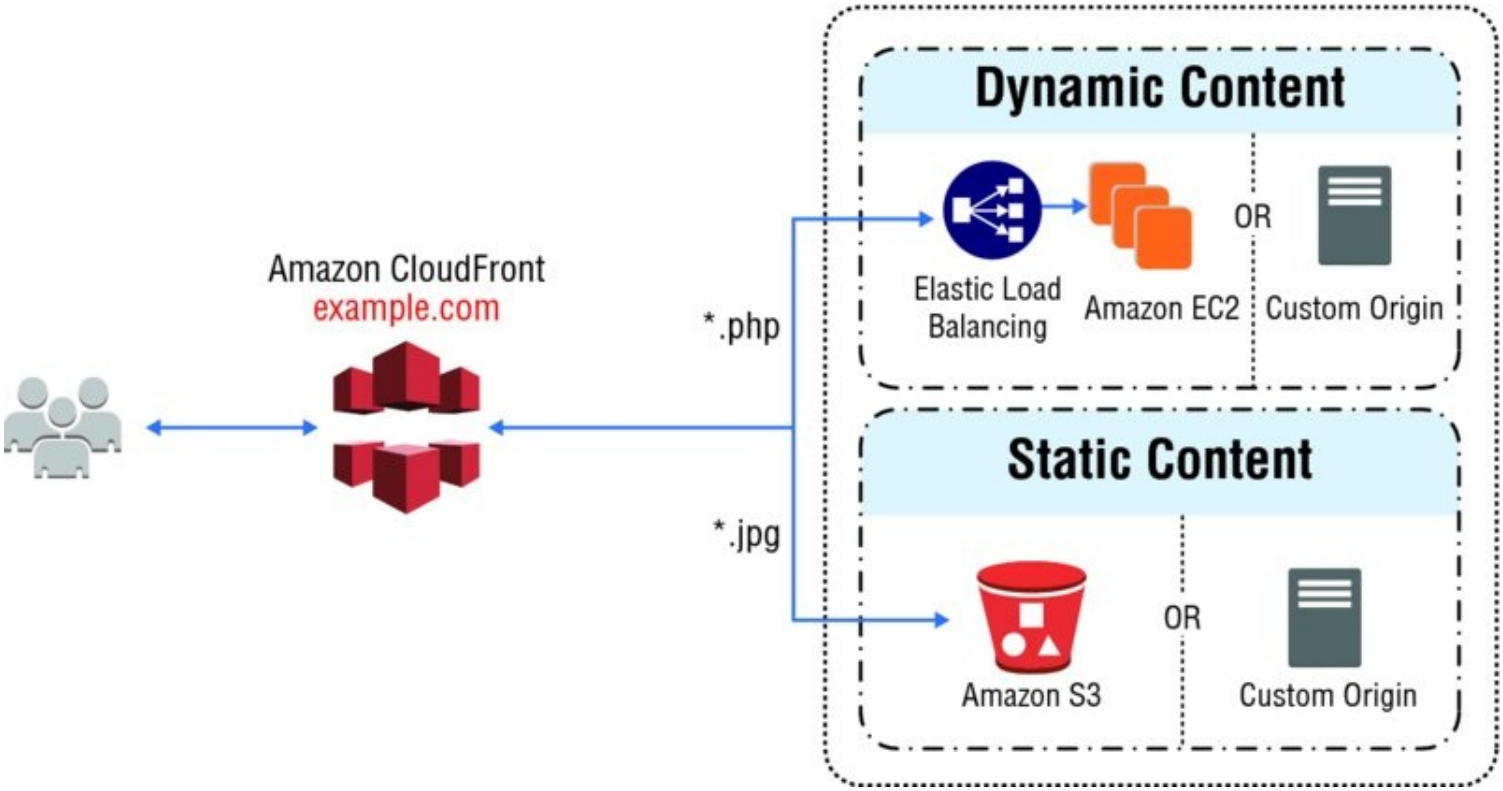


FIGURE 11.1 Delivering static and dynamic content

The functionality you can configure for each cache behavior includes the following:

- The path pattern
- Which origin to forward your requests to
- Whether to forward query strings to your origin
- Whether accessing the specified files requires signed URLs
- Whether to require HTTPS access
- The amount of time that those files stay in the Amazon CloudFront cache (regardless of the value of any Cache-Control headers that your origin adds to the files)

Cache behaviors are applied in order; if a request does not match the first path pattern, it drops down to the next path pattern. Normally the last path pattern specified is `*` to match all files.

Whole Website Using cache behaviors and multiple origins, you can easily use Amazon CloudFront to serve your whole website and to support different behaviors for different client devices.

Private Content In many cases, you may want to restrict access to content in Amazon CloudFront to only selected requestors, such as paid subscribers or to applications or users in your company network. Amazon CloudFront provides several mechanisms to allow you to serve private content. These include:

Signed URLs Use URLs that are valid only between certain times and optionally from certain IP addresses.

Signed Cookies Require authentication via public and private key pairs.

Origin Access Identities (OAI) Restrict access to an Amazon S3 bucket only to a special Amazon CloudFront user associated with your distribution. This is the easiest way to ensure that content in a bucket is only accessed by Amazon CloudFront.

Use Cases

There are several use cases where Amazon CloudFront is an excellent choice, including, but not limited to:

Serving the Static Assets of Popular Websites Static assets such as images, CSS, and JavaScript traditionally make up the bulk of requests to typical websites. Using Amazon CloudFront will speed up the user experience and reduce load on the website itself.

Serving a Whole Website or Web Application Amazon CloudFront can serve a whole website containing both dynamic and static content by using multiple origins, cache behaviors, and short TTLs for dynamic content.

Serving Content to Users Who Are Widely Distributed Geographically Amazon CloudFront will improve site performance, especially for distant users, and reduce the load on your origin server.

Distributing Software or Other Large Files Amazon CloudFront will help speed up the download of these files to end users.

Serving Streaming Media Amazon CloudFront helps serve streaming media, such as audio and video.

There are also use cases where CloudFront is not appropriate, including:

All or Most Requests Come From a Single Location If all or most of your requests come from a single geographic location, such as a large corporate campus, you will not take advantage of multiple edge locations.

All or Most Requests Come Through a Corporate VPN Similarly, if your users connect via a corporate Virtual Private Network (VPN), even if they are distributed, user requests appear to CloudFront to originate from one or a few locations. These use cases will generally not see benefit from using Amazon CloudFront.

AWS Storage Gateway

AWS Storage Gateway is a service connecting an on-premises software appliance with cloud-

Exam Essentials

Know the basic use cases for amazon CloudFront. Know when to use Amazon CloudFront (for popular static and dynamic content with geographically distributed users) and when not to (all users at a single location or connecting through a corporate VPN).

Know how amazon CloudFront works. Amazon CloudFront optimizes downloads by using geolocation to identify the geographical location of users, then serving and caching content at the edge location closest to each user to maximize performance.

Know how to create an amazon CloudFront distribution and what types of origins are supported. To create a distribution, you specify an origin and the type of distribution, and Amazon CloudFront creates a new domain name for the distribution. Origins supported include Amazon S3 buckets or static Amazon S3 websites and HTTP servers located in Amazon EC2 or in your own data center.

Know how to use amazon CloudFront for dynamic content and multiple origins. Understand how to specify multiple origins for different types of content and how to use cache behaviors and path strings to control what content is served by which origin.

Know what mechanisms are available to serve private content through amazon CloudFront. Amazon CloudFront can serve private content using Amazon S3 Origin Access Identifiers, signed URLs, and signed cookies.

Know the three configurations of AWS storage gateway and their use cases. Gateway-Cached volumes expand your on-premises storage into Amazon S3 and cache frequently used files locally. Gateway-Stored values keep all your data available locally at all times and also replicate it asynchronously to Amazon S3. Gateway-VTL enables you to keep your current backup tape software and processes while eliminating physical tapes by storing your data in the cloud.

Understand the value of AWS Directory Service. AWS Directory Service is designed to reduce identity management tasks, thereby allowing you to focus more of your time and resources on your business.

Know the AWS Directory Service Directory types. AWS Directory Service offers three directory types:

- AWS Directory Service for Microsoft Active Directory (Enterprise Edition), also referred to as Microsoft AD
- Simple AD
- AD Connector

Know when you should use AWS Directory Service for Microsoft Active Directory. You should use Microsoft Active Directory if you have more than 5,000 users or need a trust relationship set up between an AWS hosted directory and your on-premises directories.

Understand key management. Key management is the management of cryptographic keys within a cryptosystem. This includes dealing with the generation, exchange, storage, use,