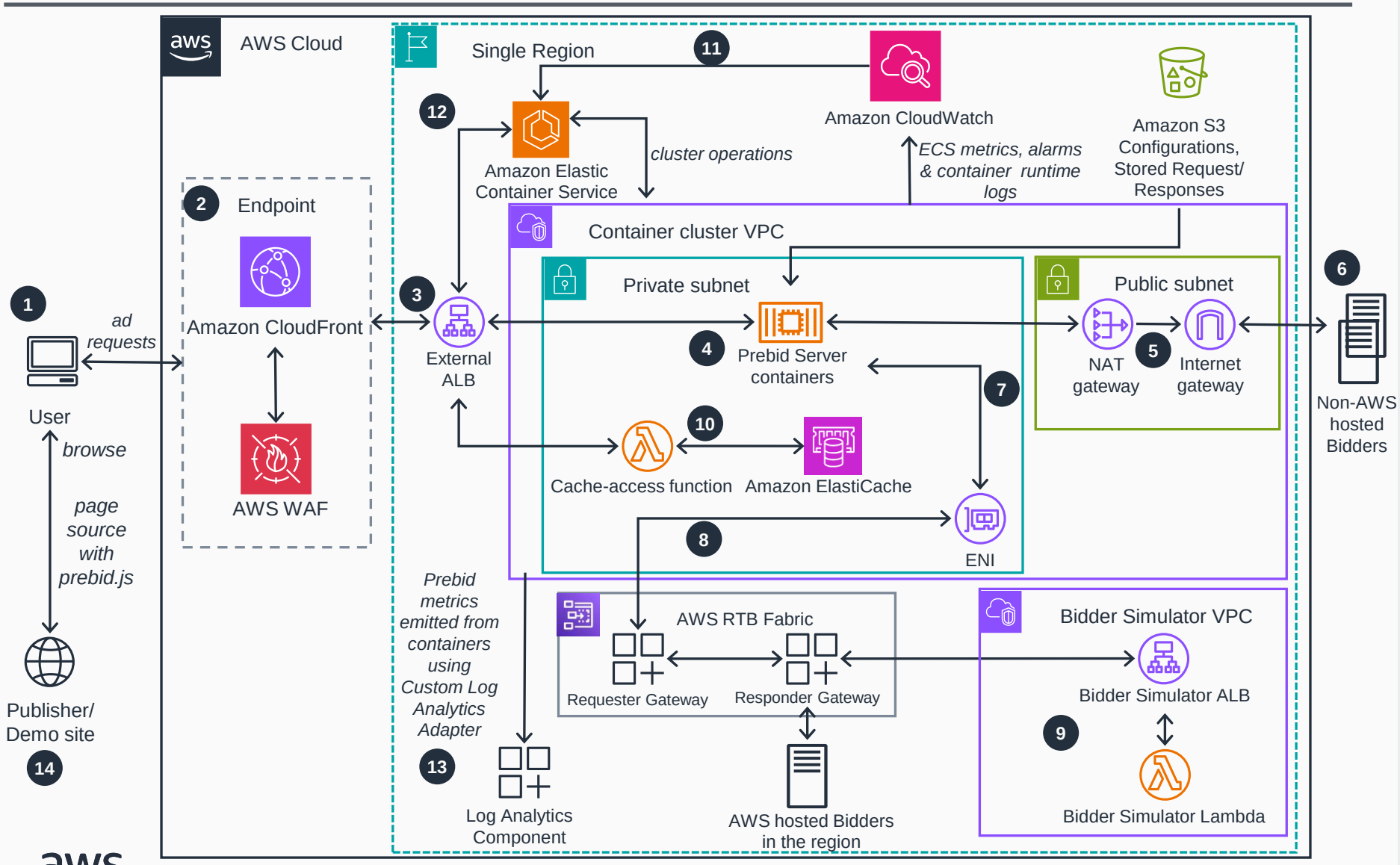
Guidance for Deploying a Prebid Server on AWS

Prebid server deployment

This architecture diagram illustrates how to effectively support Prebid server deployment on AWS. It shows the key components and their interactions, providing an overview of the architecture's structure and functionality.



1. A user browses to a page on a website that hosts ads. The publisher site returns the page source to the browser with resources, and one or more script modules (also called wrappers) that enable real-time bid requests and responses for ads of given dimensions, types, topics, and other criteria.

2. The ad requests are received from the browser at the **Amazon CloudFront** endpoint integrated with **AWS Web Application Firewall (AWS WAF)** for entry into the solution. This step helps validate legitimate traffic from malicious requests, such as penetration or denial-of-service attempts. Traffic can be received here as HTTP or HTTPS.

3. The request is forwarded to **Application Load Balancer (ALB)**. ALB determines which container running Prebid Server in the cluster is at a utilization level that can accept more requests. ALB has a network interface on the public internet and one in each private subnet where containers are hosted within **Amazon Virtual Private Cloud (Amazon VPC)**.

4. The request arrives at an **Amazon Elastic Container Service (Amazon ECS)** container, is parsed and validated, and requests to different bidding services are sent concurrently to the internet through the default internet gateway.

5. The **NAT gateway and Internet gateway** allow containers to initiate outbound requests to the internet and receive responses. These resources are primarily used for Prebid Server containers to request and gather bids for ad auctions.

6. Bidders receive one or more bid requests over the internet from a Prebid Server container. Bidders respond with zero or more bids for the various requests. The response, including the body of the winning creative(s), is sent back to the browser.

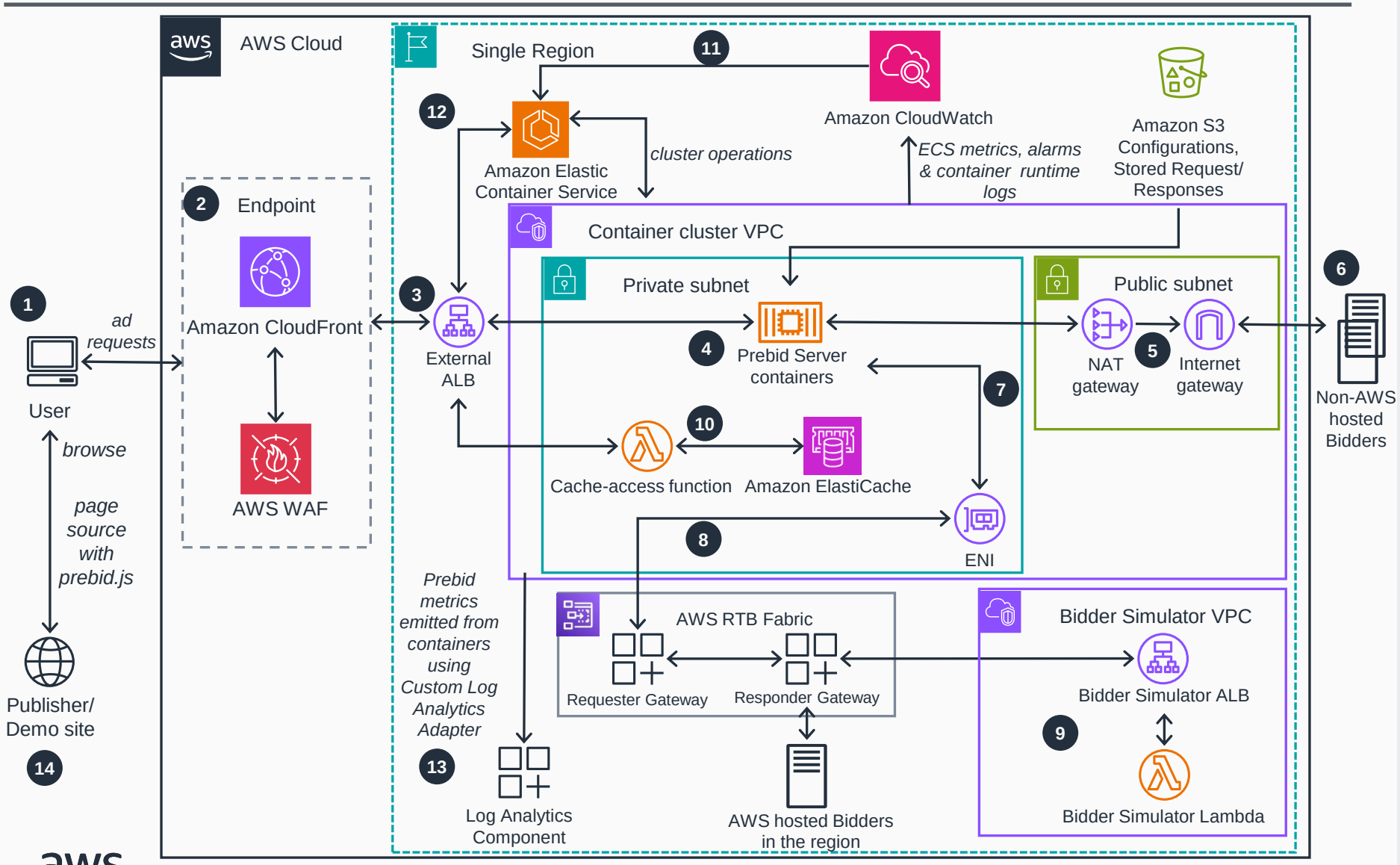
Reviewed for technical accuracy March 19, 2026

AWS Reference Architecture

Guidance for Deploying a Prebid Server on AWS

Prebid server deployment

This architecture diagram illustrates how to effectively support Prebid server deployment on AWS. It shows the key components and their interactions, providing an overview of the architecture's structure and functionality.



7 The optional custom Prebid adapter code sends bid request to a simulated bidder through **AWS RTB Fabric** Service. The adapter sends the requests through the AWS RTB Fabric configured **Elastic Network Interface (ENI)**

8 The bid request passes through the AWS RTB Fabric **Requester Gateway** resource to the **Responder Gateway** of the bidder simulator application

9 The **AWS RTB Fabric** Responder Gateway routes this request to the Application Load balancer of the Bidder simulator application and to the lambda function that processes the request and sends back bid response.

10 The Prebid server completes the auction, stores the bid response information along with Ad creative details in an **Amazon ElastiCache Serverless** cluster. The read and write of cache items from the Prebid server and Prebid.js code through the **AWS CloudFront** and external load balancer and allows storage and retrieval of ad creative content including VAST tags

11 During normal operation, **Amazon CloudWatch** metrics are collected from various resources involved in handling requests and responses through the solution. As the load changes throughout the cluster, CloudWatch alarms are used to determine when to scale-out or scale-in the container cluster.

12 An **Amazon Elastic Container Service(ECS)** service definition (Prebid ECS service) is responsible for tracking the health of the cluster, performing scale-out and scale-in operations, and managing the collection of containers available for ALB. The Prebid ECS service uses **AWS Fargate** instances.

13 The Prebid server auction metrics are collected though an optional Custom Prebid adapter for log analytics component. Detailed architecture of log analytics is explained in the next page.

14 The optional demo website runs Prebid.js and allows publishers to quickly do an end-to-end test of the Prebid server deployment with banner, instream and outstream video ads.

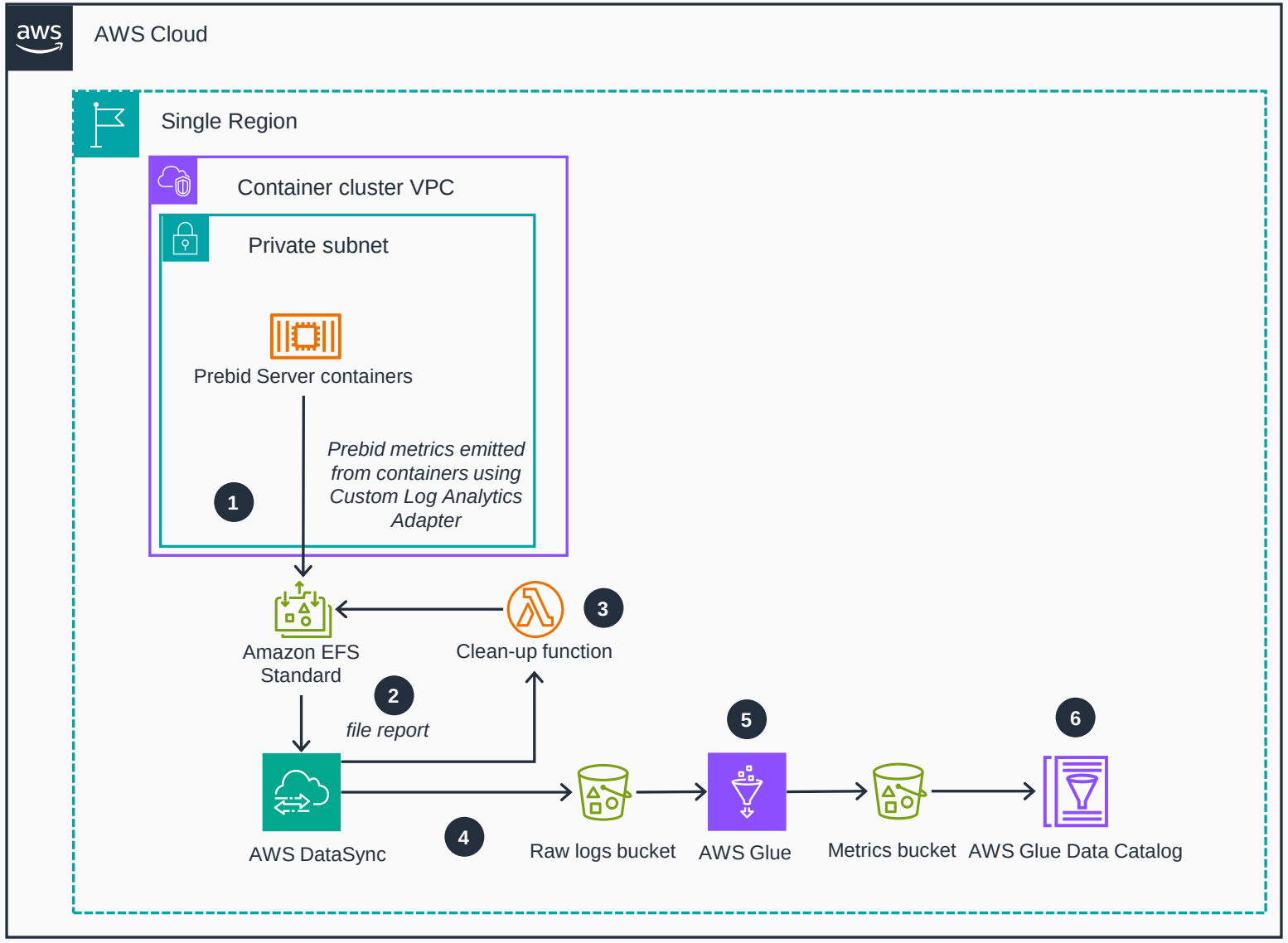
Reviewed for technical accuracy March 19, 2026

AWS Reference Architecture

Guidance for Deploying a Prebid Server on AWS

Prebid server container auction metrics

This architecture diagram illustrates how the Prebid server container auction metrics are collected and organized to allow publishers to have access to deep insights about the auctions and full control of their yield optimization



1. Metrics log files for each container are stored to a shared **Amazon Elastic File System (Amazon EFS)** using NFS protocol. This file system is mounted to each Prebid Server container during start-up. A single metrics log file is written for a limited time and then closed and rotated, so that it can be included in the next stage of processing. EFS is used as temporary location

2. **AWS DataSync** replicates rotated log files from EFS to **Amazon Simple Storage Service (Amazon S3)** on a recurring schedule. DataSync verifies each transferred file and provides a report of the completed work to an **AWS Lambda** function.

3. The DataSync Logs S3 bucket receives the replicated log files from EFS using the same folder structure. Log files arrive in this bucket because of the DataSync process. An **AWS Lambda** function runs after the DataSync process completes in step 12 and removes transferred and verified log file data from EFS.

4. An **AWS Glue** job performs an ETL operation on the metrics data in the DataSync Logs S3 bucket.

5. The ETL operation structures the metric data into a single database with several tables, partitions the physical data, and writes it to an S3 bucket. The Metrics S3 bucket contains the metric log data transformed and partitioned through ETL.

6. The data in this bucket is made available to AWS Glue clients for queries. Many different types of clients use **AWS Glue Data Catalog** to access the Prebid Server metric data.

Reviewed for technical accuracy March 19, 2026

AWS Reference Architecture