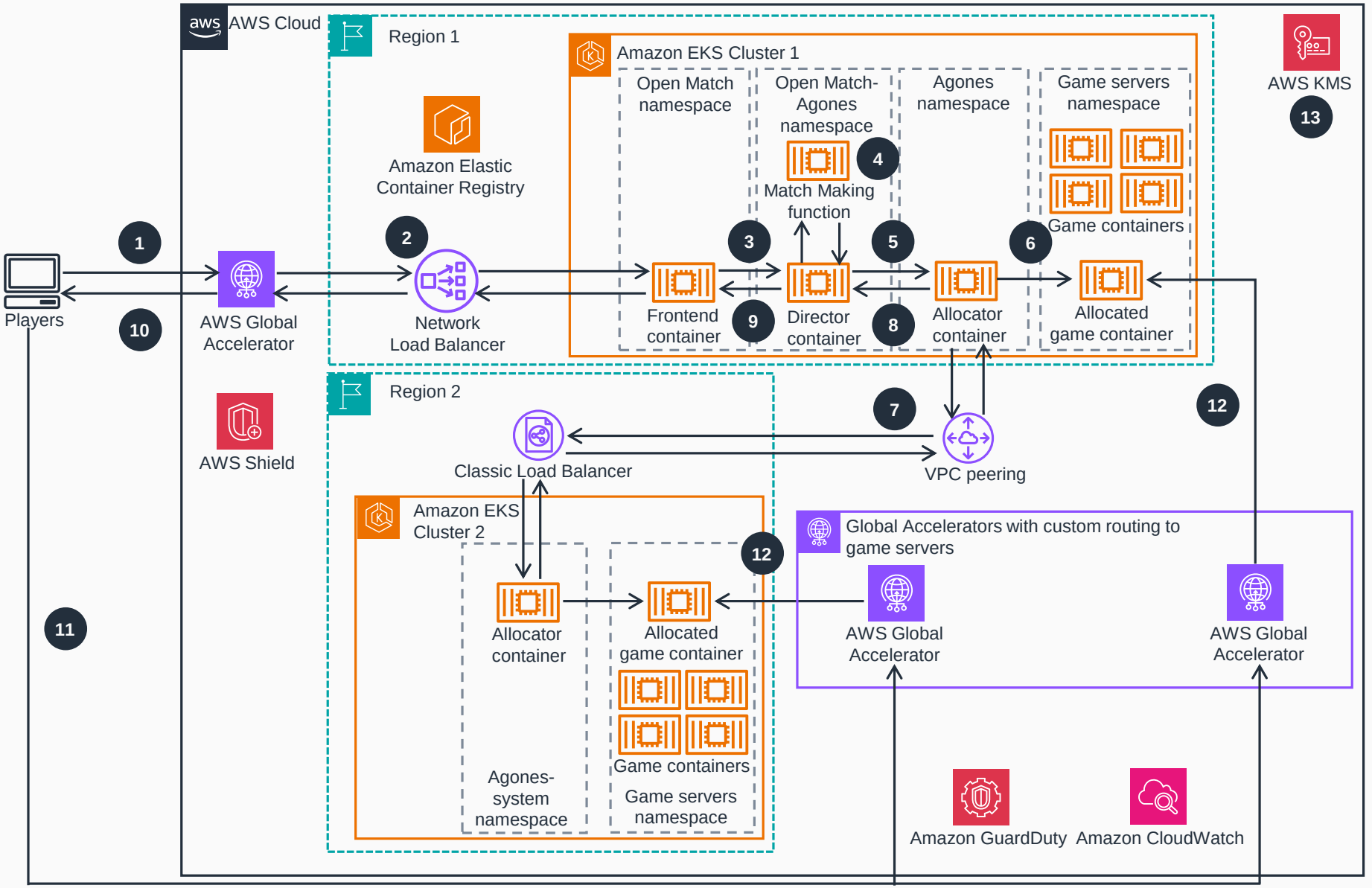
Guidance for Game Server Hosting Using Agones and Open Match on Amazon EKS

This architecture diagram shows how to use Agones and Open Match to build a global matchmaking and game server structure. Steps 1-9 are outlined below. For details on steps 10-13, refer to the next slide.



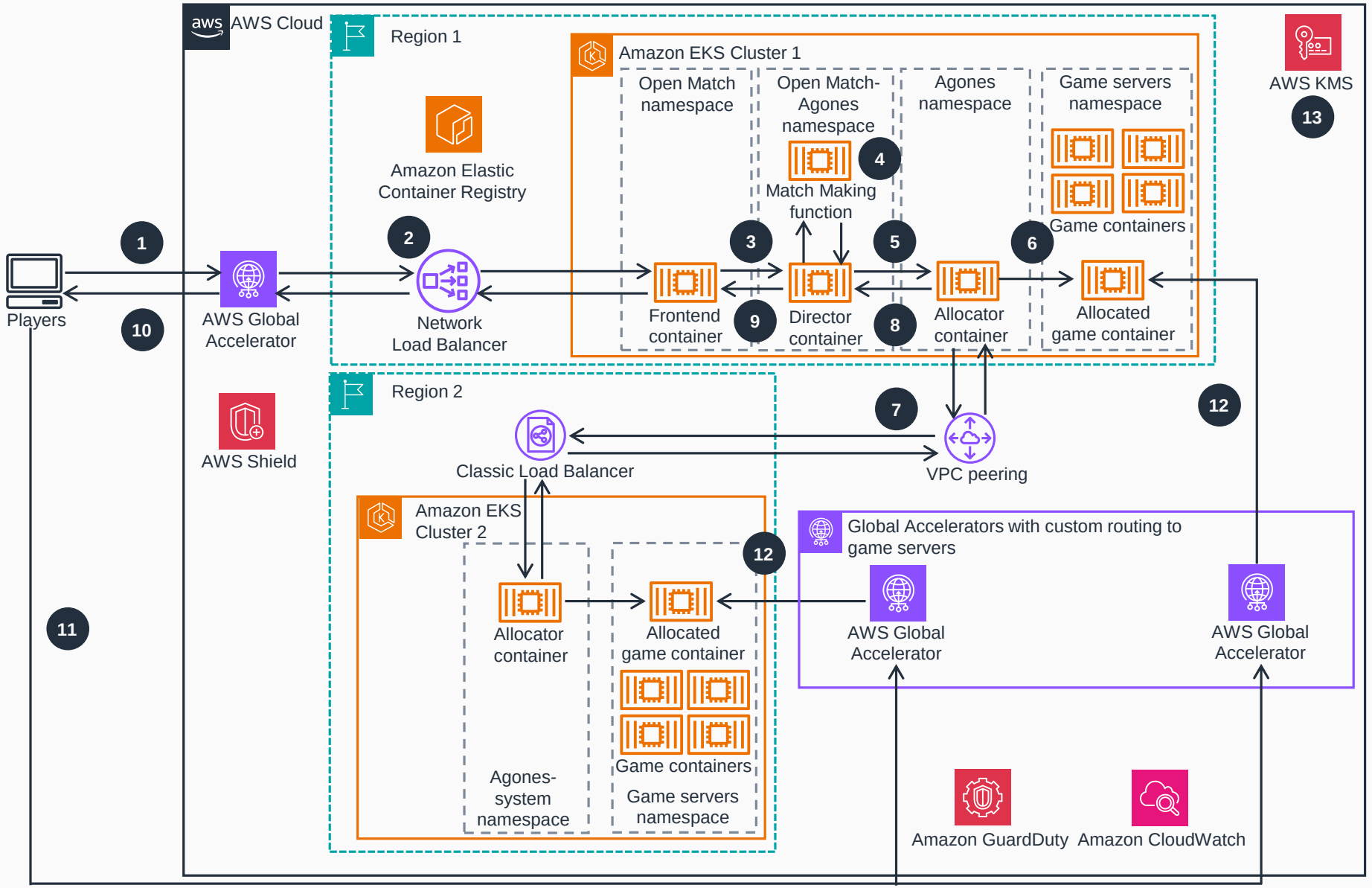
1. Connect to **AWS Global Accelerator** endpoint asking for match allocation.
2. Route the request through a Network Load Balancer to the Open Match Game Frontend container.
3. **Amazon CloudWatch** enables observability as the Director container receives and processes match requests with player data.
4. Group the players' tickets with the Match Making function, according to the function's criteria, and return a match ticket to the Director container.
5. Request a match allocation from the Agones Allocator container. The request can specify a different Region, defined by the Match Making function.
6. If the Region 1 has the lowest latency to the players, allocate a game server on the same Region with the Agones Allocator container.
7. Alternatively, route the request to the Agones Allocator on Region 2, through a virtual private cloud (VPC) peering connection and Classic Load Balancer, and allocate a server in that Region.
8. Return the internal IP address and port of the game server to the Director container.
9. Translate the internal IP address and port to the equivalent **Global Accelerator** address and port for the Director container. Then, send the translated internal IP address to the Frontend container.

Reviewed for technical accuracy February 12, 2024

AWS Reference Architecture

Guidance for Game Server Hosting Using Agones and Open Match on Amazon EKS

Steps 10-13



10 Send the game server connection details, containing a **Global Accelerator** address and custom routing port, back to the players.

11 Connect to the **Global Accelerator** for the designated cluster in a port that gets routed to the allocated game container for the match.

12 Route the connection to the allocated game container.

13 Encrypt Agones and Open Match certificates with [AWS Key Management Service](#) **(AWS KMS)**.

Note:

- Steps 1-10 use gRPC with TLS/SSL.
- Steps 11-12 use the game specific protocol.
- All the containers are pulled from [Amazon Elastic Container Registry](#) **(Amazon ECR)**.

Reviewed for technical accuracy February 12, 2024

AWS Reference Architecture