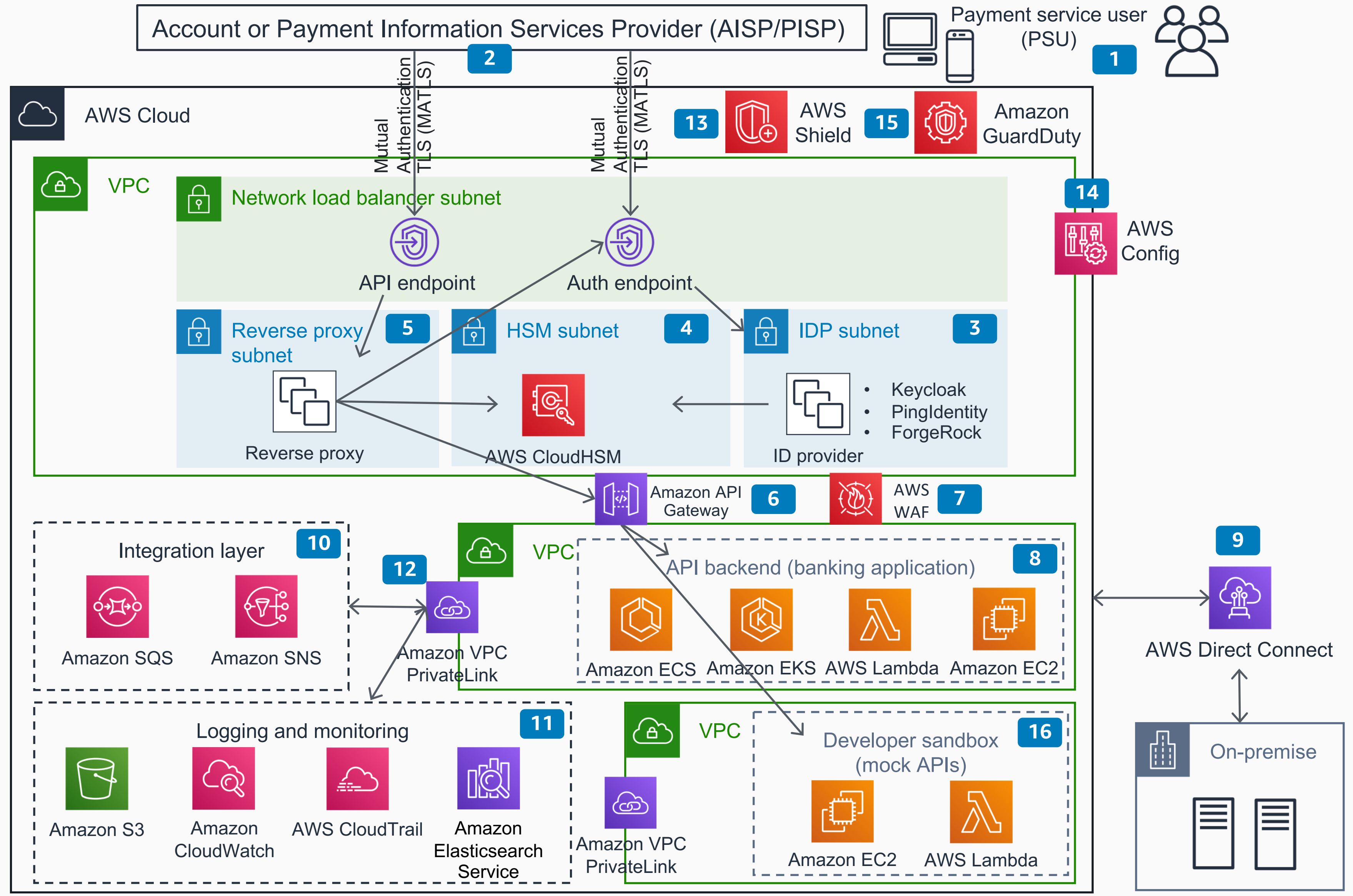
Open Banking on AWS

Implement the right architecture for Open APIs.



1 Payment service user accesses third-party application; can use any service.

2 Third parties—account or payment information services providers (AISP/ PISP)—build applications around payments, money transfer. Aggregating data across banks provides more insights (such as spend analysis, balance across banks).

3 Third-party application obtains an access token from the account servicing payment service provider (ASPSP) to service user requests. ASPSP validates the certificate of AISP/PISP using mutual TLS authentication and provides an access token (from identity providers such as ForgeRock, PingIdentity, and Keycloak).

4 AWS CloudHSM offloads SSL certificates for both API and Auth endpoints.

5 A reverse proxy (such as Nginx) is used to meet mutual TLS requirement of the Open Banking Standard.

6 Amazon API Gateway handles the complete API management of the banking APIs.

7 AWS WAF integrates with API Gateway to protect against common web exploits.

8 Banking logic is implemented using AWS Lambda, containers, or by running Amazon EC2 instances.

9 Banking logic accesses a bank's data center using AWS Direct Connect.

10 Amazon SQS and Amazon SNS provide integration and notification capabilities between different services.

11 Service logs are collected in Amazon S3 and analyzed and monitored using Amazon Elasticsearch.

12 AWS PrivateLink securely connects a VPC to supported AWS services.

13 AWS Shield protects against DDoS attacks.

14 AWS Config provides continuous compliance.

15 Amazon GuardDuty continuously monitors for malicious activity and unauthorized behavior; protecting AWS accounts and workloads.

16 Third parties use a separate developer sandbox to build their applications.

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Reference Architecture