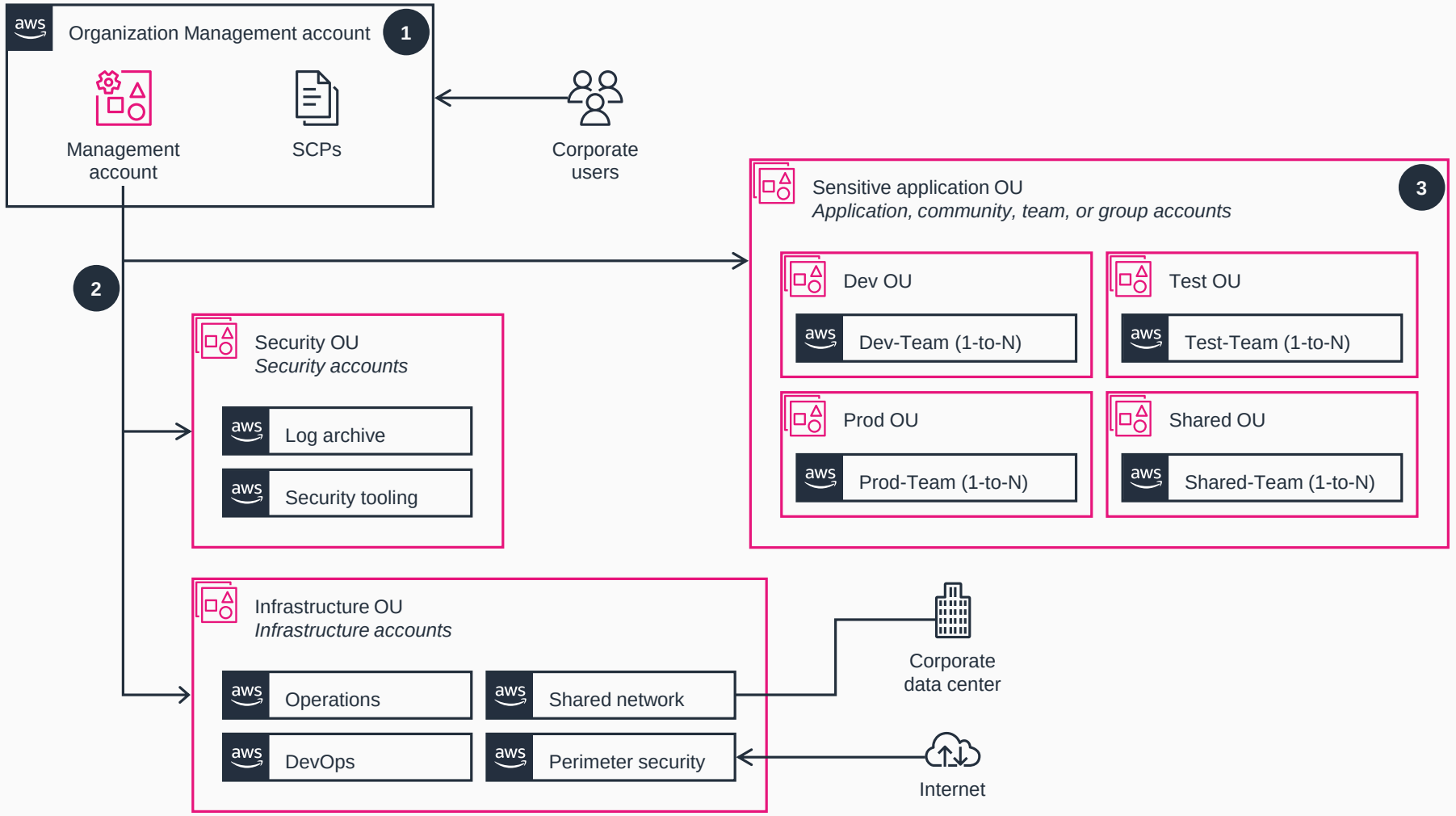
Guidance for Trusted Secure Enclaves on AWS

Overview

This architecture diagram shows how to configure comprehensive, multi-account workloads with unique security and compliance requirements.



1. An organization in **AWS Organizations** with multiple accounts, guided by service control policies (SCPs): The organization groups multiple separate AWS accounts that are controlled by a single customer entity. Separate AWS accounts provide strong control-plane and data-plane isolation between workloads or environments, as if they were owned by different AWS customers.

2. The management account is used to create the organization. From the organization's management account, you can do the following:
   - Create accounts in the organization and manage policies for all organizational units (OUs).
   - Join the following OUs to the organization:
     - Security OU
     - Infrastructure OU
     - Sensitive application OU

   Each OU will have one or more member accounts or nested OU, per design.

3. The application OU will have several nested OUs dedicated to application delivery and lifecycle management and will include the following:
   - Dev OU
   - Test OU
   - Prod OU
   - Shared OU

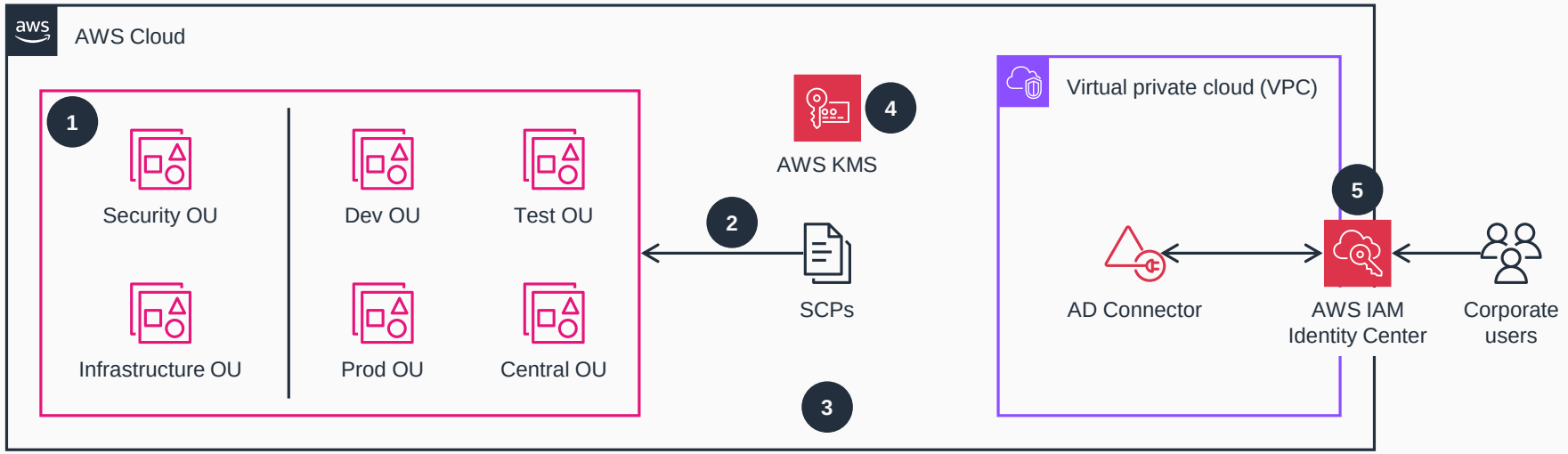
   Additionally, sandbox OUs can also be provisioned as nonsensitive workloads.

Reviewed for technical accuracy October 25, 2023

AWS Reference Architecture

Guidance for Trusted Secure Enclaves on AWS

Organization Management Account



1. **An organization with multiple accounts:** The organization groups multiple separate AWS accounts, which are controlled by a single customer entity. This consolidates billing, groups accounts using OUs, and facilitates the deployment of an organizations preventative controls using SCPs.

2. **Preventative security controls:** These controls, implemented by SCPs, protect the architecture, prevent guardrail disablement, and block undesirable user behavior. SCPs provide a guardrail mechanism principally used to deny specific or entire categories of API operations at an AWS account, OU, or organization level. These can be used to make sure workloads are deployed only in prescribed AWS Regions or deny access to specific AWS services.

3. **Automation:** Automation makes sure that guardrails are consistently applied when the organization adds new AWS accounts as new teams and workloads are brought onboard. It remediates compliance drift and provides guardrails in the root organization account.

4. **Encryption: AWS Key Management Service (AWS KMS)** with customer-managed keys encrypts data stored at rest using FIPS 140-2–validated encryption, whether in **Amazon Simple Storage Service (Amazon S3)** buckets, **Amazon Elastic Block Store (Amazon EBS)** volumes, **Amazon Relational Database Service (Amazon RDS)** databases, or other AWS storage services. It protects data in transit using TLS 1.2 or higher.

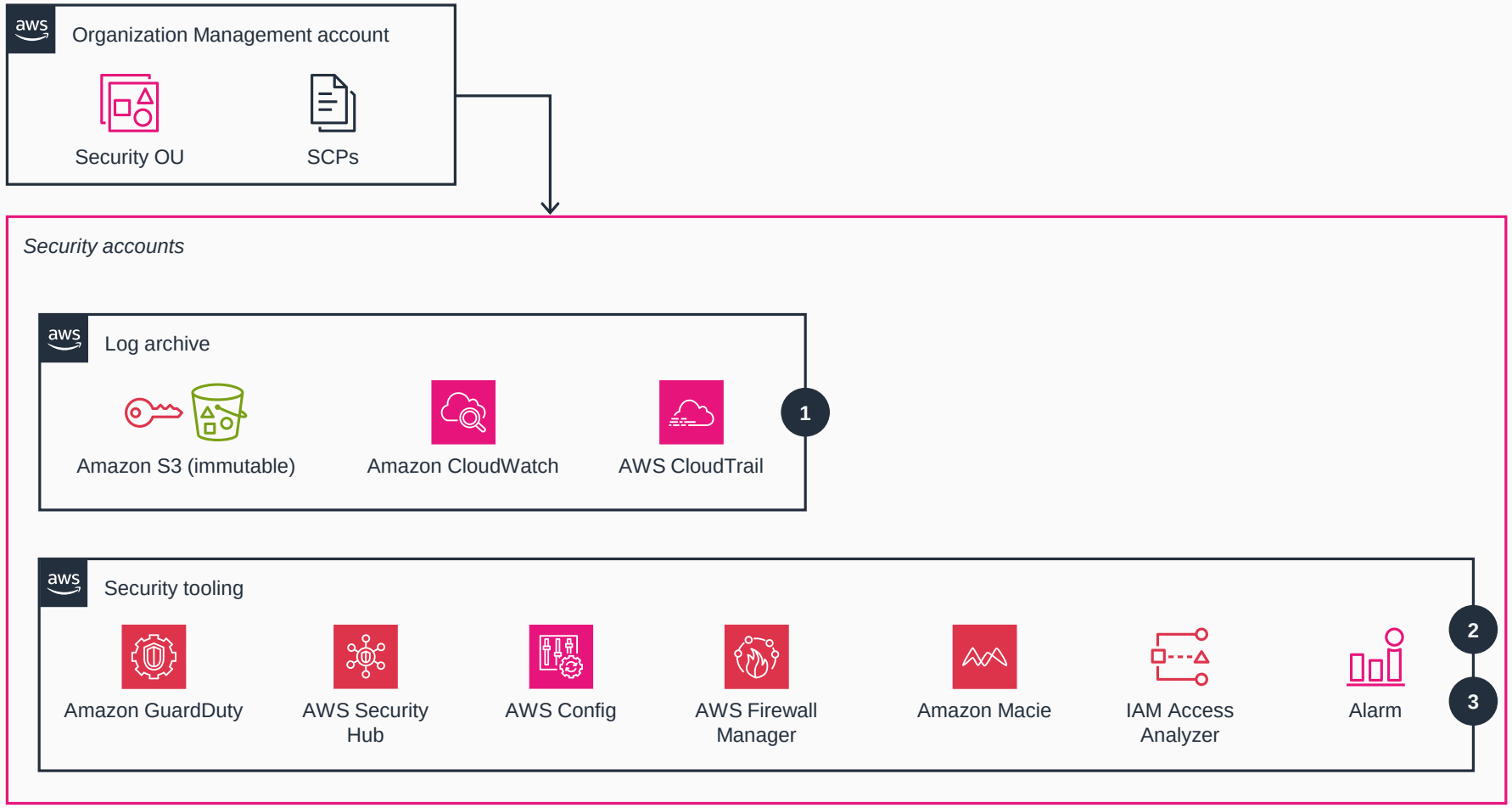
5. **Single sign-on:** A feature of **AWS Identity and Access Management (IAM), IAM Identity Center** is used to provide centralized **IAM** role assumption into AWS accounts across the organization for authorized principals. An organization's existing identities can be sourced from a customer's existing Active Directory (AD) identity store or another third-party identity provider (IdP). AWS facilitates multifactor authentication enforcement using authenticator apps, security keys, and built-in authenticators, supporting WebAuthn, FIDO2, and Universal 2nd Factor (U2F) authentication and devices.

Reviewed for technical accuracy October 25, 2023

AWS Reference Architecture

Guidance for Trusted Secure Enclaves on AWS

Security Accounts



1. **Centralized logging:** This architecture prescribes comprehensive log collection and centralization across AWS services and accounts. **AWS CloudTrail** logs work organization-wide to provide full control-plane auditability across the cloud environment. **Amazon CloudWatch** logs, a cloud-native AWS logging service, is used to capture a wide variety of logs including operating system and application logs, VPC flow logs, and domain name system logs, which are then centralized and only available to defined security personnel.

2. **Centralized security monitoring:** Compliance drift and security threats are surfaced across the customer's AWS organization through the automatic deployment of a multitude of different types of detective security controls. This includes activating the multitude of AWS security services in every account in the organization. These security services include **Amazon GuardDuty**, **AWS Security Hub**, **AWS Config**, **AWS Firewall Manager**, **Amazon Macie**, **IAM Access Analyzer**, and **CloudWatch** alarms. Control and visibility should be delegated across the multi-account environment to a single central security tooling account for easy organization-wide visibility to all security findings and compliance drift.

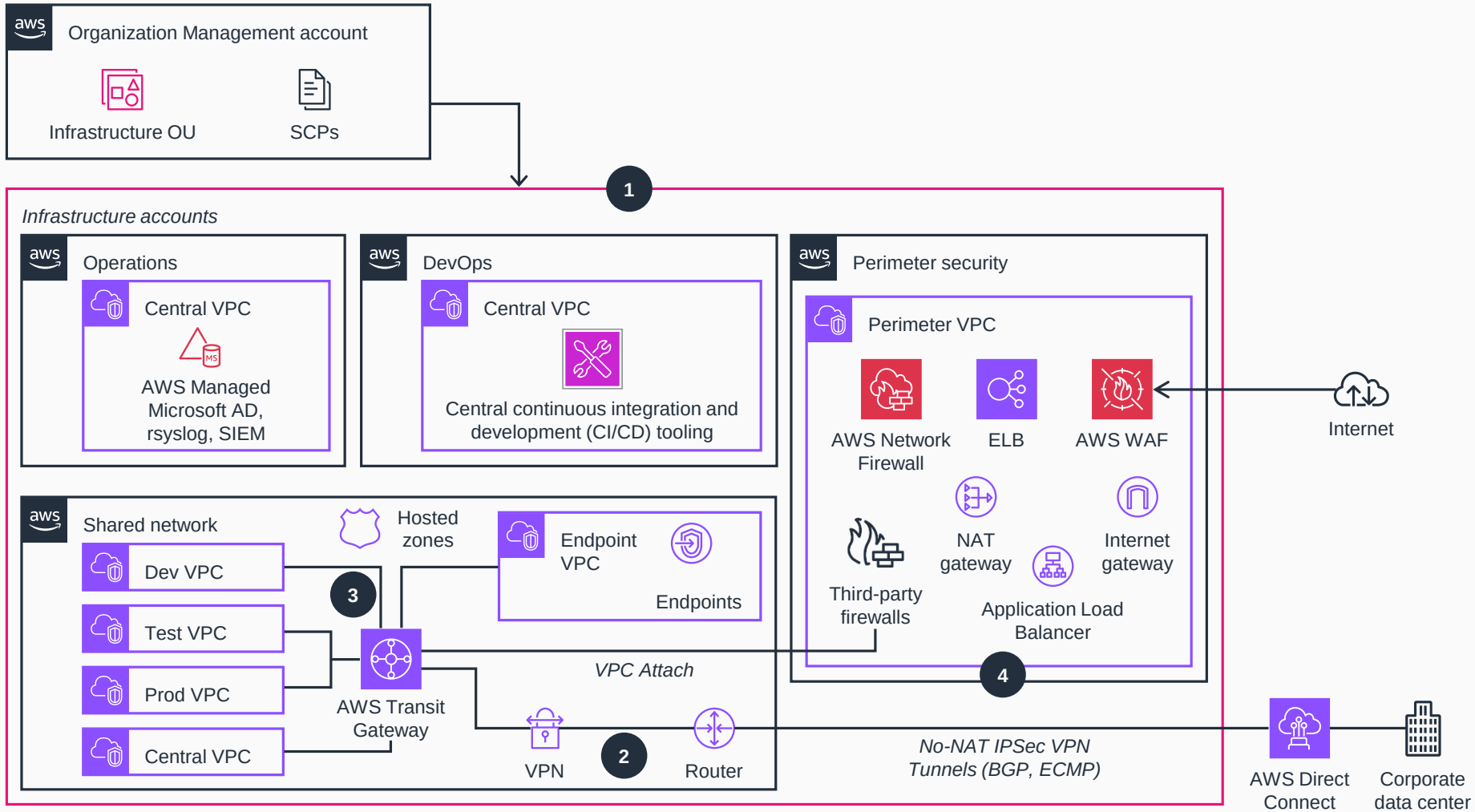
3. **View-only access and searchability:** The security account is provided view-only access across the organization (including access to each account's **CloudWatch** console) to facilitate investigation during an incident. View-only access is different from read-only access in that it does not provide any access to any data. An optional add-on is available to consume the comprehensive set of centralized logs to make them searchable, providing correlation and basic dashboards.

Reviewed for technical accuracy October 25, 2023

AWS Reference Architecture

Guidance for Trusted Secure Enclaves on AWS

Infrastructure Accounts



1. **Centralized, isolated networking:** VPCs built through **Amazon Virtual Private Cloud (Amazon VPC)** are used to create data-plane isolation between workloads, centralized in a shared-network account. Centralization facilitates strong segregation of duties and cost optimization.

2. **Mediated connectivity:** Connectivity to on-premises environments, internet egress, shared resources, and AWS APIs are mediated at a central point of ingress and egress through the use of **AWS Transit Gateway**, **AWS Site-to-Site VPN**, next generation firewalls, and **AWS Direct Connect** (where applicable).

3. **Alternative options:** The centralized VPC architecture is not for all customers. For customers less concerned with cost optimization, an option exists for local account–based VPCs interconnected through **Transit Gateway** in the central shared-network account. Under both options, the architecture prescribes moving AWS public API endpoints into the customer's private VPC address space, using centralized endpoints for cost efficiency.

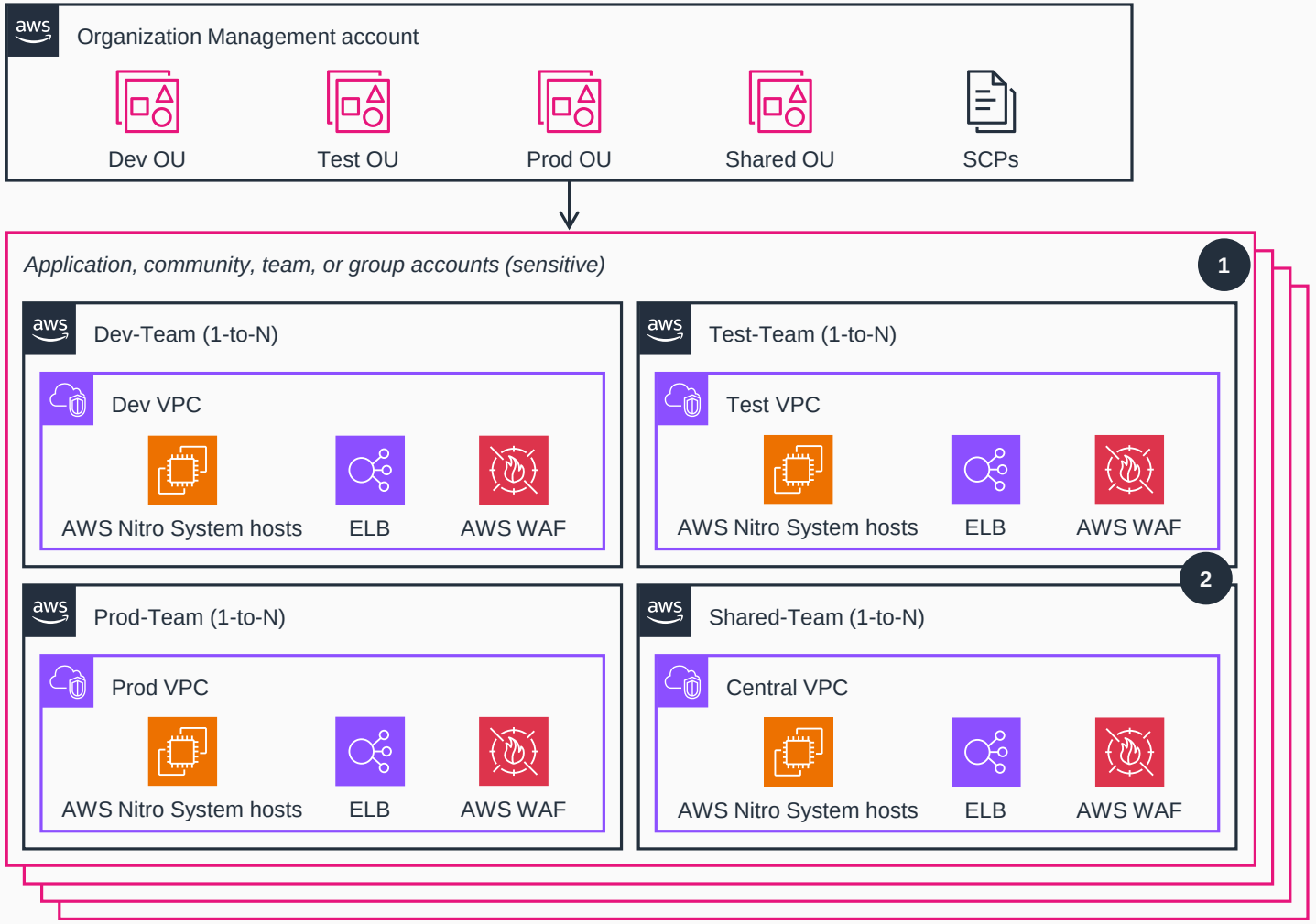
4. **Centralized ingress and egress infrastructure-as-a-service (IaaS) inspection:** It is common to see centralized ingress and egress requirements for IaaS-based workloads. The architecture provides this functionality, so customers can decide if native AWS ingress and egress firewall inspection services—such as **AWS Network Firewall**, **AWS WAF**, or **Application Load Balancer** through **Elastic Load Balancing (ELB)—**meet their requirements. If not, customers can augment those capabilities with third-party firewall appliances. The architecture supports starting with an AWS firewall and switching to a third-party firewall or using a combination of ingress and egress firewall technologies.

Reviewed for technical accuracy October 25, 2023

AWS Reference Architecture

Guidance for Trusted Secure Enclaves on AWS

Application, Community, Team, or Group Accounts (Sensitive)



1. **Segmentation and separation:** The architecture does not merely provide strong segmentation and separation between workloads belonging to different stages of the software development lifecycle or between different IT administrative roles (like between networking, ingress and egress firewalls, and workloads). It also offers a strong network zoning architecture, microsegmenting the environment by wrapping every instance or component in a stateful firewall that is enforced in the hardware of the **AWS Nitro System**, along with services such as **ELB** and **AWS WAF**.

2. All network flows are tightly enforced, with lateral movement prevented between applications, tiers within an application, and nodes in a tier of an application unless explicitly allowed. Further, routing is prevented between Dev, Test, and Prod with recommendations on a CI/CD architecture to enable developer agility and ease code promotion between environments with appropriate approvals.

Reviewed for technical accuracy October 25, 2023

AWS Reference Architecture