# Zero Trust Architecture

A Comprehensive Guide to Enterprise Security Implementation

# Core Principles: Never Trust, Always Verify

### Verify Explicitly

Always authenticate and authorize based on all available data points, including user identity, location, device health, service or workload, data classification, and anomalies.

### Least Privilege

Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive polices, and data protection to secure both data and productivity.

### Assume Breach

Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

# Authentication Models & Access Control

## Multi-Factor Authentication (MFA)

MFA requires more than one distinct authentication method for successful access, mitigating compromised credentials.

- **Knowledge:** Something you know (Password, PIN).
- **Possession:** Something you have (Phone, Hardware Token).
- **Inherence:** Something you are (Fingerprint, Face ID).

## Role-Based Access Control (RBAC)

Access decisions are based on the roles that individual users have as part of an organization.

- **Role Assignment:** Users are assigned roles (e.g., Admin, HR, Dev).
- **Role Authorization:** Roles are granted specific permissions.
- **Permission Authorization:** Users acquire permissions via their roles.

# Designing the Zero Trust Framework

01

### Identify

Define the "Protect Surface"—critical data, applications, assets, and services (DAAS).

02

### Map Flows

Understand how traffic moves across the network to identify dependencies.

03

### Architect

Design the Zero Trust network with micro-segmentation and next-gen firewalls.

04

### Policy

Create granular policies using the "Kipling Method" (Who, What, When, Where, Why, How).

# Simulating Enterprise Security

### Containerization

Use Docker to spin up isolated microservices representing different business units (Finance, HR, Engineering) to test segmentation.

### Virtualization

Deploy VMware ESXi to host virtual firewalls (like pfSense) that act as the Policy Enforcement Point (PEP) between zones.

### Network Segmentation

Configure Virtual LANs (VLANs) to strictly isolate traffic. Ensure no default "Any-Any" rules exist between segments.

```
# Docker Compose Example: Isolated Networks services: finance-
db: image: postgres:14 networks: - secure_zone web-app: image:
nginx:alpine networks: - secure_zone - public_zone networks:
secure_zone: internal: true # No internet access public_zone:
driver: bridge
```

# IAM & Access Control Policies

## Identity Policies

Policies focused on verifying **Who** is requesting access.

```
> Require MFA for all users.
> Block 'Root' login from external IPs.
> Rotate API keys every 90 days.
> Enforce strong password complexity.
```

## Context Policies

Policies focused on **Where** and **How** access is requested.

```
> Deny access from High-Risk Geos.
> Device must have OS Patch > v12.4.
> Device requires active Antivirus.
> Session timeout after 15 mins idle.
```

# Testing Effectiveness & Results

| THREAT SCENARIO | SIMULATION METHOD | ZERO TRUST DEFENSE | OUTCOME |
|---|---|---|---|
| **Phished Credentials** | Attacker attempts login with stolen password. | MFA Challenge (Adaptive Policy) | **BLOCKED** |
| **Insider Threat** | Compromised endpoint scans for open DB ports. | Micro-segmentation (Deny All Inbound) | **CONTAINED** |
| **Unmanaged Device** | Personal laptop attempts VPN connection. | Device Posture Check (IAM) | **DENIED** |
| **Data Exfiltration** | Large file upload to unknown public cloud. | DLP (Data Loss Prevention) Rules | **FLAGGED** |

*"By implementing Zero Trust, we moved from a static perimeter to dynamic, identity-based security, reducing the attack surface by 80%."*