Zero Trust: From Airports to Identity-Aware Proxies



About Me

- Developer Advocate at Pomerium
- From Montreal, Quebec, Canada
- I’m @nickytonline everywhere
- All the places you can find are on nickyt.online
- Not a big fan of spiders



What We’ll Cover

What We’ll Cover

- My Background
- Why Zero Trust Matters
- The Problem with Perimeter-Based Security
- The VPN Crisis: 2.8 Million Reasons to Worry
- How Airport Security Helps Us Understand Zero Trust
- The "Catch Me If You Can" Vulnerability
- Zero Trust: Airport Security for Your Network

What We’ll Cover

- Core Zero Trust Components
  - Identity-Aware Proxy
  - Context-Defined Access
  - Policy Engine
  - Microsegmentation
- Demo
- Benefits for Legacy Systems
- Implementation Path
- Zero Trust 🤝 Kubernetes
- Key Takeaways

My Background

My Background

I’ve been an application developer most of my career and now I’m diving into the world of infrastructure and security architecture. Lots of new things for me, but hopefully I bring a fresh perspective to the topic.



Why Zero Trust Matters

Why Zero Trust Matters

- Changing landscape:
  - Microservices and cloud architecture expose more access points
  - APIs accessible from anywhere increase attack surface
  - Traditional perimeter security doesn't fit the modern deployment model

Why Zero Trust Matters

- Zero Trust simplifies workflows:
  - Security decisions move outside application code
  - Consistent access patterns across all services
  - No custom security code to maintain in your application logic

Why Zero Trust Matters to Developers

- Security as shared responsibility
  - Security moves earlier in development process ("shifts left")
  - Developers implement auth patterns, not just security teams
  - DevSecOps requiring security integrated into development

The Problem with Perimeter-Based Security

The Problem with Perimeter-Based Security

Imagine showing your ID once at airport security, then being allowed to board any plane, enter any cockpit, or access the control tower...



The Problem with Perimeter-Based Security

- The Reality: Perimeter-based security operates exactly this way - check once, trust forever



The Problem with Perimeter-Based Security

- Remote work, personal devices, and cloud apps expose the limitations of perimeter-based VPNs



The Problem with Perimeter-Based Security

- Perimeter-based security assumed threats were outside your network
- Today's reality: threats can already be inside



The Problem with Perimeter-Based Security

Multi-factor authentication (MFA) strengthens the perimeter, but once authenticated, users still get broad network access.

- VPN appliances themselves remain vulnerable targets
- MFA doesn't prevent lateral movement once inside

The VPN Crisis: 2.8 Million Reasons to Worry

- Alarming Statistics: 2.8 million unique IP addresses are actively attacking VPN infrastructure worldwide
- The Fatal Flaw: Your company's entire security posture collapses if someone compromises your VPN
- The False Promise: VPNs create a secure tunnel but trust everything inside it

The VPN Crisis: 2.8 Million Reasons to Worry

- The New Reality: Remote work, cloud applications, and bring your own devices (BYOD) have rendered the perimeter obsolete

Sources:

- ”2.8 Million Reasons Why You Can't Trust Your VPN” - The New Stack (March 2025)
- "Massive brute force attack uses 2.8 million IPs to target VPN devices" - BleepingComputer (February 2025)

How Airport Security Helps Us Understand Zero Trust

How Airport Security Helps Us Understand Zero Trust

Most of you have probably flown at least once. The airport is a common experience with multiple security layers:

- ID check at entrance → Initial authentication
- Security screening → Device posture verification
- Boarding pass verification → Resource-specific authorization

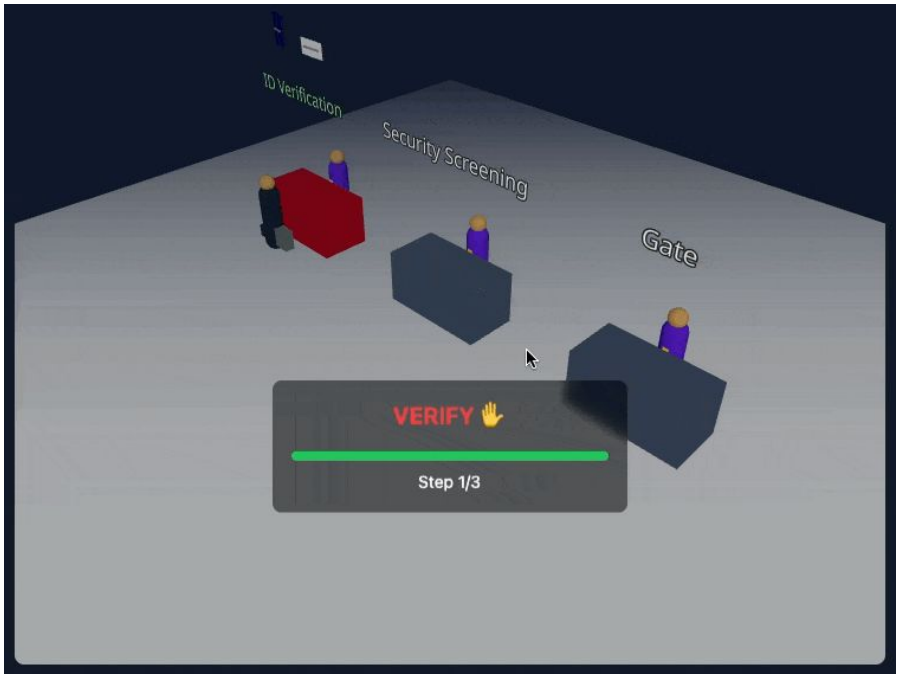


How Airport Security Helps Us Understand Zero Trust

- Gate agent checks → Just-in-time access validation
- Boarding the plane → Accessing the application
- Cockpit access → Privileged functionality requiring additional verification

How Airport Security Helps Us Understand Zero Trust

Key Insight: Multiple verifications based on identity, destination, and context at each security boundary



The "Catch Me If You Can" Vulnerability

The "Catch Me If You Can" Vulnerability

Frank Abagnale Jr. exploited trust-based systems by impersonating pilots and other professionals.



The "Catch Me If You Can" Vulnerability

- His Technique: Obtained uniform and fake credentials → gained access to planes and cockpits
- The Security Failure: Once identity was assumed, no further verification occurred

The "Catch Me If You Can" Vulnerability

- Why It Worked:
  - Systems relied on perimeter security (the uniform) rather than continuous verification
  - No contextual verification (wasn't on flight schedule, no verification of experience)
  - No separation between basic access (boarding plane) and privileged access (cockpit)
  - No continuous monitoring of suspicious behavior patterns

All I saw was a uniform and said to myself, if I had that uniform on, and I walked in a bank, it would be a lot easier to cash a check than me walking in as just some young kid. So the whole thing was to get a uniform and do that."

– *Frank Abagnale*

Source: Frank Abagnale | Catch Me If You Can | Talks at Google

Zero Trust: Airport Security for Your Network

Zero Trust: Airport Security for Your Network

Core Principle: Never trust, always verify. The Airport Model Applied:

- Public internet = airport entrance (untrusted zone)
- Identity-Aware Proxy = security checkpoint
- Device posture = luggage screening
- User roles = different types of boarding passes

Zero Trust: Airport Security for Your Network

Core Principle: Never trust, always verify. The Airport Model Applied:

- Applications = individual planes
- Admin/privileged access = cockpit access (requires additional verification)
- Continuous verification = multiple checkpoints throughout journey
- Behavior monitoring = security personnel watching for suspicious activity

Core Zero Trust Components

Identity-Aware Proxy

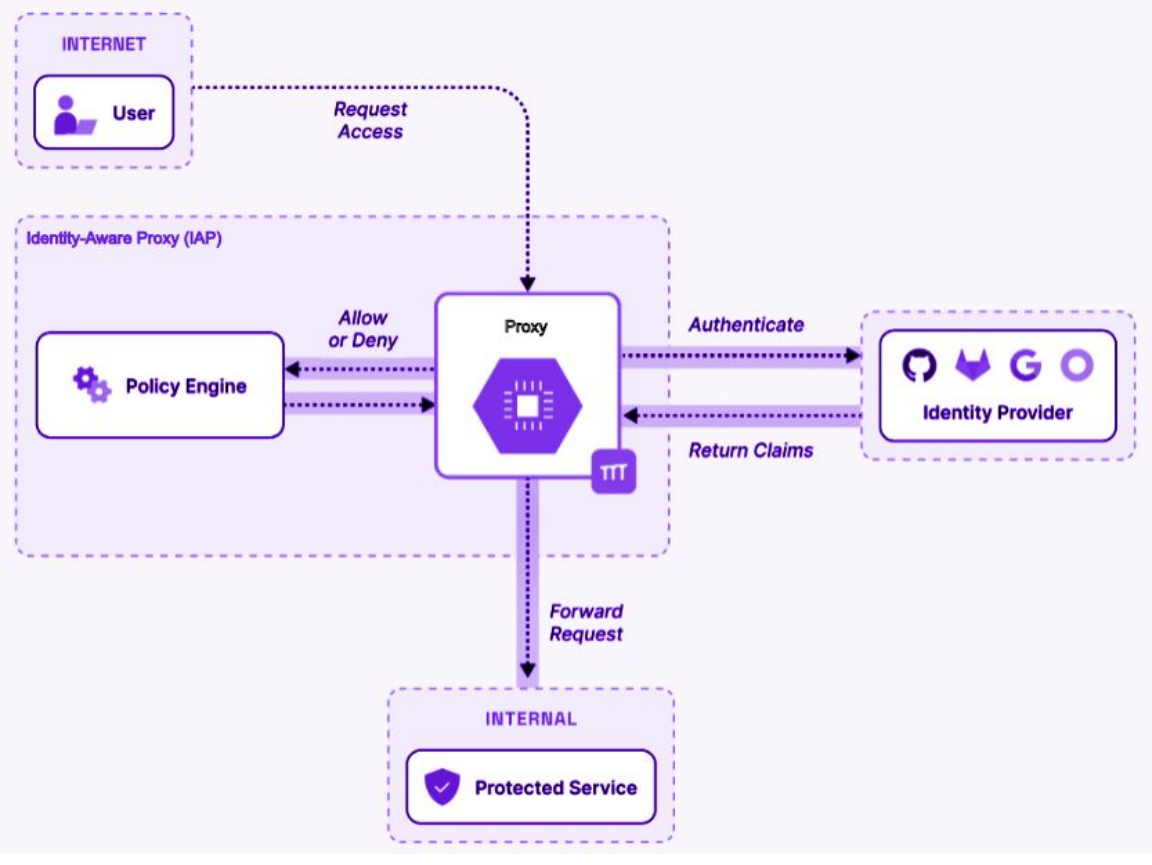
What Is an Identity-Aware Proxy?

- A security service that sits between users and applications at your network's edge
- Acts as a central access control point for all applications
- Operates at layer 7 in the network stack

What Is an Identity-Aware Proxy?

- Authenticates users and verifies their identity before granting access
- Makes access decisions based on context, not network location
- Enforces consistent security policies across all protected resources

What Is an Identity-Aware Proxy?



Context-Defined Access

Context-Defined Access

Zero trust is not just about your identity. It’s about context.

Context Factor	Airport/Airplane Example	Network Example
Who: Identity/Role	Pilot ID vs passenger boarding pass	Admin user vs regular user
What: Resource	Which plane/cockpit	Which application/admin functions
Where: Location	Approved gate/terminal	Corporate office vs coffee shop

Context-Defined Access

Zero trust is not just about your identity. It’s about context.

Context Factor	Airport/Airplane Example	Network Example
Which: Device	Authorized equipment/keys	Managed vs personal device
When: Time	Scheduled flight time	Business hours vs off-hours
Why: Behavior	Normal vs suspicious movement patterns	Typical vs unusual access patterns

Context-Defined Access

Suspicious Context: "Senior engineer accessing production database at 3AM from an unrecognized device in a foreign country"

Airport Equivalent: "Someone with a pilot's ID trying to enter a cockpit during off-hours at a gate where their plane isn't scheduled"



Policy Engine

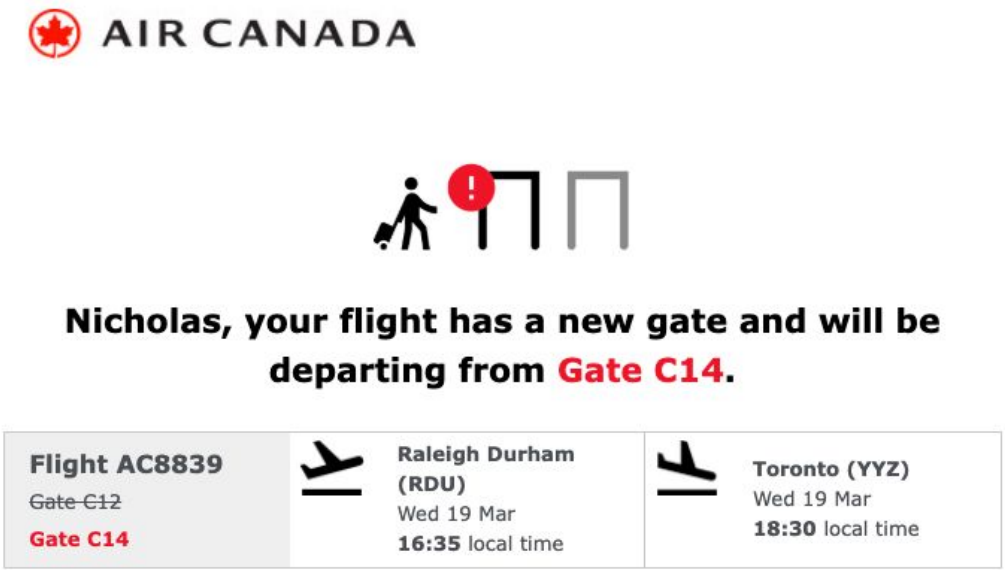
Policy Engine

A policy engine is context-defined rules for access. Let’s look at this in the context of an airport:

- No liquids over 100ml → Clear, specific restrictions
- ID must match boarding pass → Identity verification rules
- Flight crew must have valid credentials → Role-based access rules
- Random additional screening → Risk-based verification

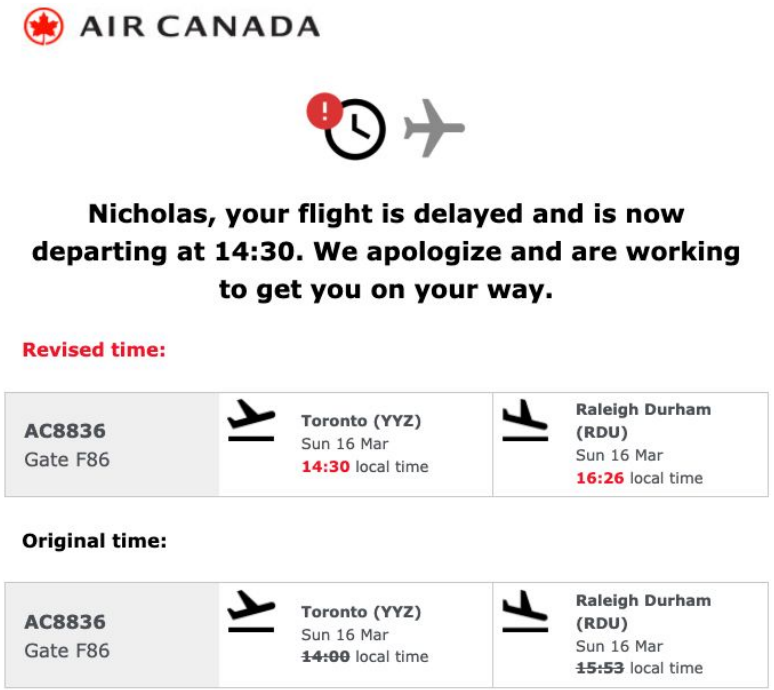
Policy Engine

When things change, your policy needs to be dynamic. In our airport scenario, gates can change. Your boarding pass is no longer valid for the original gate that was assigned to you.



Policy Engine

Maybe your flight's delayed. Your boarding pass is only valid at the new boarding time.



Policy Engine

The list goes on:

- weather emergency
- flight is cancelled
- crew change where new flight crew credentials are authorized and old ones revoked

Policy Engine

Zero Trust Equivalent:

- Resource Relocation: Access policies automatically update when services move
- Maintenance Window: Access permissions adjust during scheduled maintenance times
- Security Incident: Elevated verification required during suspicious activity
- Team Changes: Access revoked immediately when roles change

Microsegmentation

Microsegmentation

Think of microsegmentation as security zones for digital assets. In our airport scenario:

- Public concourse: Anyone with an ID can enter
- Boarding areas: Only passengers with valid boarding passes
- Individual planes: Only passengers for that specific flight
- Cockpit: Only authorized flight crew
- Customs areas: Additional verification for international travel

Microsegmentation

In Zero Trust:

- Creates security boundaries around individual resources, not just network perimeters
- Prevents lateral movement if one application is compromised
- Enables least privilege access at a granular level

Microsegmentation

In Zero Trust:

- Requires continuous verification when moving between segments
- Isolates critical systems with additional security controls
- Makes attackers' jobs exponentially harder by containing their reach

Demo Time

Benefits for Legacy Applications

Benefits for Legacy Applications

You can immediately increase your security posture by putting an Identity-Aware Proxy in front of legacy apps.

Before Zero Trust:

- old application with basic authentication
- No MFA capability
- No device posture checking
- Limited audit logging
- Vulnerable to lateral movement

Benefits for Legacy Applications

After Zero Trust Implementation:

- Modern authentication with MFA
- Device security verification
- Granular access policies
- Microsegmentation preventing lateral movement
- Consistent security model across all applications
- All without changing the application code

Implementation Path

Implementation Path

Assessment Phase (1-2 months)

- Identify critical applications (map your airport terminal)
- Document current access patterns (understand passenger flow)
- Inventory security controls (audit existing checkpoints)

Implementation Path

Pilot Implementation (2-3 months)

- Start with a single application (secure one gate first)
- Implement basic identity verification (ID checks)
- Test with limited user group (small passenger sample)

Implementation Path

Expansion Phase (3-6 months)

- Add context-based policies (more sophisticated screening)
- Incorporate device posture checks (baggage inspection)
- Extend to more applications (expand to additional terminals)

Implementation Path

Maturity Phase (Ongoing)

- Consistent model across all environments (standardized security across all terminals)
- Automated policy management (centralized security operations)
- Continuous verification at all access points (seamless multi-layered security)

Anyone Heading to KubeCon EU?

Zero Trust 🤝 Kubernetes

Zero Trust 🤝 Kubernetes

Zero Trust is a perfect fit for Kubernetes environments. It enhances Kubernetes by addressing:

- API control plane access security
- Pod-to-pod communication (east-west traffic)
- Consistent identity across cluster boundaries
- Authentication without network-level trust

Key Takeaways

Key Takeaways

- Zero Trust Is Intuitive: We now understand it from airport security
- Context, Not Location: Network location is no longer a proxy for trust
- Check Every Access, Every Time: Continuous verification is the new standard
- Incremental Adoption: Start small, expand gradually
- Security Evolution: Improve your posture by adding Zero Trust even with existing perimeter-based infrastructure
- Future-Ready Security: Prepare for distributed workforce and applications

Resources

Resources

- Zero trust
- 2.8 Million Reasons Why You Can’t Trust Your VPN
- Massive brute force attack uses 2.8 million IPs to target VPN devices

Slide Deck

Thank You!

Stay in touch!

- @nickytonline everywhere
- nickyt.online