Living on Borrowed Time

Why Zero Trust & Least Privilege Management Are Critical To Your Company's Future

When it comes to security, there are two core concepts to know: zero trust, and least privilege. Zero trust is the core concept, but branches into subgroups, one of which contains the principle of least privilege.

Den Jones, CISO at Banyan Security

Lumos

Zero Trust and Least Privilege – Breaking down the key concepts

Let’s get a few critical terms straight first:

Privileged Access is special access or abilities above and beyond that of a standard user, such as admin access in AWS or Salesforce.

Least privileged access grants the least amount of privilege necessary, depending on who is requesting access and the context of the request. It’s the difference between having a skeleton key that opens every door, and a key that only opens certain rooms depending on who you are, like a guest staying at a hotel. This means that a key (or access permission) should only be given to users based on their role or responsibilities, for example a software engineer doesn’t need to have the permission to see salaries in the HR system.

Zero Trust enforces a “never trust, always verify” approach to privileged access. It basically follows the principle, “What should I do if I assume that every single person could be a malicious actor?”

More formally, zero trust removes any assumption of implicit trust, regardless of who is asking for access, what network they happen to be on, or what resource is being requested. Since no one is assumed trusted in a zero trust model, access needs to be verified each time a user wants to access a certain system, and in the best systems, re-verified periodically during access.

Here’s the deal, though – logging in isn’t zero trust, nor is passwordless authentication, nor is adding or removing permissions. Zero trust means we don’t inherently trust the user, device, network, or access. Trust is constantly earned, constantly calculated.

The Core Principles of Zero Trust Implementation

First: let's acknowledge the fact that there are multiple interpretations of zero trust. For the purpose of this article, we're zeroing in on zero trust as it pertains to your workforce accessing applications and services from anywhere in the world.

Based on NIST 800-207 (some say, the gold standard), the zero trust model includes these three core principles:

- Continuous verification
- Limiting the 'blast radius'
- Automated context collection and response.

Properly executed, continuous verification ensures that both authentication and authorization are continuously re-visited and re-verified. For example, after gaining access to a sensitive corporate finance server, the user may have been multitasking and inadvertently clicked on a link that deployed malware on their device. As soon as this is detected, their device trust level should drop, and per policy, they should no longer have access to that sensitive resource, even if they were successfully granted access 5 minutes ago.

Risk-based conditional access and scalable and agile policy deployment are crucial: without sacrificing the user experience, verification is consistent, shifting when risk levels do. It's important to note that zero trust doesn't go around compliance requirements. For that reason, policies must take risk into consideration (in addition to compliance).

With zero trust, limiting the 'blast radius' lessens the impact of a breach. The range of paths to access or credentials for a bad actor is limited. This supplies additional time for those tasked with defense (both systems and people) to mitigate the attack and form a response. Identity-based segmentation should be leveraged, as credentials and data can change often when traditional network-based segmentation can fall short.

The least privilege principle takes a front seat here. Whenever credentials are in the mix, they must be given minimal access – just enough to carry out the required task. It’s critical that the scope changes as tasks do; we’ve seen that many attacks happen when accounts are over-privileged and under-monitored.

NIST lists guidance for automating context collection and response. An increased level of data can be a positive, but it must be processed and leveraged in real-time. Some sources that NIST can provide further perspective on are: user credentials, workloads, and endpoints (APIs could include SSO, SIEM).

Best Practices and Obstacles With Zero Trust Adoption

Let's take a real-world example of how not having a solid zero trust infrastructure can backfire. Okta is using a company called Sykes Sitel for outsourcing customer support. Hackers targeted an employee within Sykes Sitel who had privileged access from their role in customer service and dealing with Okta clients and data.

They compromised that account. That account was empowered to reset passwords and reset multi-factor authentication. So, the hacker was able to reset passwords for literally hundreds of companies, which is a problem because they can say "I'm just going to set a new password, and I'm going to remove this multifactor authentication and set my own multi-factor authentication."

How to prevent this?

Do regular access reviews. Privilege creep happens very quickly. Make sure managers and app admins revoke access for their employees once not needed. I think most large enterprises assume they are incapable of doing regular access reviews.Thousands of groups, network based IP tables, local accounts and access....it's too much effort.

To reduce exposure as well as the amount of work to perform reviews, automate the removal of users from groups. A simple start is to write a script that reviews the authentication logs and if a user hasn't logged into an application for 90 days then remove them from the group.

Implement Just-in-Time Access and Dynamic Permission Elevation (time-based access). Basically, every person is a standard user in the system, say AWS. When they need to access privileged roles, they

need to ask for it and indicate for how long they need it. If granted, they would only have the additional access for the indicated time period.
Implement Activity-Based access removal. When a person stops using an app or certain permissions within the app for a defined period of time, make sure to remove the unneeded access.

Companies still struggle to implement zero trust or even get started with it. In our discussions it seems people are overwhelmed with all the information out there (especially conflicting details). They also have concerns around funding and resource constraints. The only way forward is to simplify the problem into bite-sized chunks.

At Banyan, we start by modifying the authentication workflow to include device posture, registration and certificates for authentication. Then we overlay the remote access component (in parallel to any existing VPN).

In an ideal world there's a workflow with your helpdesk that has access requests bundled into “pre-approved” and “requires approval” with an easy catalog (an Amazon-like experience).

Then there would be an automated way to remove people who aren't using the application. It's easy to create a script that looks at the authentication logs and if a user does not login within XX days they are removed from the directory group. This assumes you create application groups. At Banyan, we also assigned group owners and designated if it was pre-approved or required approval.

‘Can’ vs. ‘Should’ Have Access: A Closer Look at JIT

Before taking the plunge into adopting a zero trust model, many companies implement Role-Based Access Control (RBAC). RBAC assigns roles as a group – grouping similar users and granting access.

The challenge here is that if the admin isn’t consistently monitoring, access can go stale very quickly as users change roles but their permissions don’t follow suit.

Where RBAC stipulates whether a user can have access, justified and just-in-time (JIT) access dictates if the user should have access, for what aspect, and within what timeframe. JIT authorizes dynamic querying of access to a certain asset or resource.

I’m a big fan of JIT if done right. But in reality it needs fine grained ownership so that the workflow knows how to approve the access. However, this can be expensive and slow down a large organization with thousands of requests per day. Imagine an IT organization with 500+ admins who at different times need access. If it requires a human to approve, that can slow down the business and increase costs. If it’s automated, then that doesn’t prevent a bad actor during account take-over. So, great in principle, but like most of the IAM workflows, they break down fairly quickly at scale.

The issue with JIT is if the user is approved to have the access but they are compromised it doesn’t exactly make it hard for the bad actor to gain access (as they are impersonating the user). Of course you can add an MFA, but even that doesn’t always slow the bad actor down much. That’s why we opted for the method above, it was a simple step and not too disruptive to the business.

Adopting a zero trust model can pay off in a myriad of ways that impact the entire organization.

Spend Management

Tying zero trust to cost savings = very relevant to today’s economy. Basically if you remove access once a person doesn’t need it, you also save on costs. The thing about zero trust is it depends on the meaning..if it’s network based then you can reduce operational costs.

If it’s identity based you can save costs on subscriptions by removing users; but one could say that’s not exactly zero trust, that’s just good IAM hygiene since you’re removing users who don’t need access. At Banyan we created a script to remove people from groups based on them not logging into the app.

We did calculate cost savings for the workforce not VPN’ing in (productivity gain); but our big one (hard cost) was not changing passwords every 90 days and moving to an indicator of compromise (IOC).

This was a combination of using certificates instead of passwords as well as UEBA and workflows related to authentication events. The same way a bank will notify customers of suspicious activity, we’ve done that for logins with our users.

This reduced service desk tickets related to password changes by around 80%. It also saved our workforce around 15 minutes per person every 90 days. For an organization with 40k users, this saves around 40,000 hours per year.

Time Savings

Done right, a decent zero trust deployment leverages your existing investments and that includes your team. There shouldn’t be a need for exhaustive training and if you partner with the right companies with great products your deployment should take days, not months or years.

There’s huge benefits to employees as they access applications and services in a more seamless way, no passwords, no VPN.

Increased Security Posture

There’s an immense benefit to security as we don’t expose the entire network, just access to the specific applications themselves. This access is managed via your existing directory-based groups which you would already do as part of your application provisioning.

The traditional VPN access method often means that full-time employees are provided full VPN access to your office network. This means that a single compromised endpoint with an established VPN connection would have free reign to launch a broader attack.

This shift to granular application and resource access removes the network level access and as such this attack vector. In addition it removes the need to manage network-level VPN access which significantly reduces operational costs.

Let’s Recognize Attackers Masquerading as Insiders

Let’s face it: we’re living in a perimeter-less security world. Adopting a zero trust model powered by least privilege and identity access management (IAM) is the goal to help secure your company.

Between the expansion of remote work and a snowball effect of SaaS growth, malicious actors have become undaunted and even more daring in their attempts to threaten enterprise assets for their own self interests.

The reality is that most companies maneuver in an atmosphere where employees have more access than they need, which has adverse effects on cost and security. Re-think how your organization handles access and authorization; the future of your company depends on it.

Great Technology Needs Great Control.

You have grown and so have your risks: Software cost explosion. Excessive admin access. A flood of JIRA tickets.

Don't let managing hundreds of apps and permissions slow down your business. Lumos is the first app governance platform that automates access requests, enforces least privilege, speeds up user access reviews, and eliminates extra SaaS app spending.

Gone are the technology silos that left IT, Security, Compliance, and Finance in the dark. With Lumos, you have visibility into app usage, entitlements, and spending - and the power to take action on that data.

The impact? Disappearing IT support costs, Just-in-Time Access (JIT) with no audit spreadsheets and VLookups. All this equals guaranteed software savings.

To learn how Lumos can help your organization, **let's chat.**