okta

The Eight Principles of Modern Infrastructure Access

Okta Inc.
100 First Street
San Francisco, CA 94105

info@okta.com
1-888-722-7871

Introduction

Your infrastructure resources are some of the most sensitive and valuable assets across your corporate network. Whether in the cloud or on-prem, controlling access to servers and databases is a top priority for IT and Security departments. Traditional methods are laser-focused on "protecting the keys", yet admin credential breaches continue to slam businesses year over year. Something has to change.

This whitepaper examines the core challenges with securing access to infrastructure, and why we need to revisit the approach our industry has taken to date. Building on Forrester's Zero Trust model, we've developed a modern methodology for infrastructure access. This methodology is bound by eight principles that set the foundation for a more secure environment—one that's fit for the modern cloud era.

Traditional Measures Don't Cut It

With static credentials, possession is 100% of the law—there is no direct link to identity

We have a credential problem. Any password or key-based system—which currently represents a majority—presents a serious issue. The core challenge with securing infrastructure lies in the credential mechanism used to log in to servers. Any user with the correct login key or password can access the system, no matter how that credential was acquired. Stolen credentials then become a carte blanche for any attacker. This has brought about a number of products and practices that attempt to address the credential problem. These solutions mostly center on wrapping a management layer around credentials so they can't be lost, stolen, or misused. Despite being a step up from self management, these solutions are still rooted in the concept that the credential holds the key to access the system, not the user.

Problems with traditional solutions:

- Static credentials. Even with a management layer protecting credentials, their inherent properties do not change. Multiple users can hold the same credential, and there's no way to guarantee or track identity.
- Painful to operate. Products in this space are widely recognized as a burden on operations, especially in highly automated, elastic cloud environments. This can hold back automation, which is counterproductive to adopting cloud infrastructure.
- Poor end-user experience. Out-of-band processes to check out credentials are painful for systems administrators, and notoriously slow. Technical users who are blocked from doing their job will find workarounds, rendering the security controls ineffective.

The Problem with Server Credentials

Static in nature: A key, once generated, is effectively immutable. This cryptographic property doesn't necessarily make it more secure. Credentials are often lost, stolen, and misused.

Manual to provision: Removing a key, in the case of an employee leaving the company or changing teams, is a manual process. The administrator needs to know which servers and which keys belong to which user. How can they be sure they've completely revoked someone's access?

No ties to identity: Despite the way we've learned private key infrastructure (PKI) from the Alice and Bob example, a private key is not associated with an identity profile. Possession is 100% of the law, and anyone can pick it up.

Shared across systems: Credential sprawl is no different than password sprawl, but the stakes are higher. When you add a key to a system, you're simply accumulating privileges over time. Do you know how many keys are out there that can access systems?

Throw Out the Keys With Zero Trust

The Zero Trust security model changes how we look at securing the company network. Rather than a binary decision based purely on the network, access represents a contextual decision that factors in dynamic user, device, network, and location information. The framework is simple: verify before trusting, every time.

Google's BeyondCorp initiative, a marquee example of Zero Trust done right, was designed around the following premises:

- Connecting from a particular network must not determine which services you can access.
- Access to services is granted based on what we know about you and your device.
- All access to services must be authenticated, authorized, and encrypted.

BeyondCorp was a transformative effort that impacted Google's entire organization. Not every company is ready for a shift that drastic, which is why Okta recommends taking it one step at a time. Infrastructure access should be the first Zero Trust use case implemented. Because it applies to a subset of technical users who are capable of handling the architectural and procedural changes, our customers have found great success when they narrow their focus to this one initiative.

The Eight Principles of Modern Infrastructure Access

It is time for companies of all sizes to have access to a better identity-led architecture. The Okta Identity Cloud is a foundational platform to support this methodology, with its Advanced Server Access product as the solution.

The key to an overhaul of infrastructure access is to break away from traditional methods and products. At Okta, we've been working with our customers to implement this new methodology across fleets of large-scale infrastructure deployments, with significant and trackable success. Through this real-world experience we've narrowed in on eight principles that together form a cohesive architecture suited to any modern organization. Each principle is outcome oriented, focused on what companies really need and want from their identity and access solutions.

1 Automation over Manual Operations

Configuration management automatically provisions access controls

Traditional access management methods can require a significant amount of heavy lifting. As cloud adoption grows, these methods show their age. Imagine the case of a server administrator leaving the company—it is a considerable pain point to ensure all of their credentials and accounts are disabled. Companies get the most out of cloud infrastructure when they embrace automation, at any level of scale and elasticity. But security products can hold cloud adoption back because they generally don't support automation.

With the rise of DevOps tools and processes, security controls must "shift left" to get in sync with the developer and operations teams who build and deploy "infrastructure as code". When done right, teams can configure the environments once, then let automation take over. Getting configuration right is critical, and best done through test environments. Once that initial work is complete, scaling becomes efficient and effective.

In terms of access controls, the user and group accounts that exist on the machine are automated. Respective access policies are then enforced during the authorization process. Any change in user status, group membership, or policy specifications must be captured in near real-time, so every request is evaluated based on the most up-to-date information. In the case of a server administrator leaving the company, the action from the system of record should trigger a series of workflows that immediately disable any access.

Key Takeaway

Everything about Okta is exposed as an API. Enrollment, provisioning, and configuration can all be fully automated, making it incredibly easy to use.

2 Ephemeral Credentials over Static Keys

Login credentials are limited in both scope and time to only allow for single-use

A key tenet of Zero Trust is the shift of access controls from the network layer to the application layer, where more dynamic context allows for smarter decision-making. This moves away from pure binary access decisions like "in the network/ not in the network". Now imagine all that effort is made to gather context and enforce it in real-time, only to hand the user a shared static credential. That would be a waste. In order to adhere to the access decision that was made, the credential mechanism must match exactly.

Mitigating the risk associated with credentials is less about protecting them through a management plane, and more about limiting their value. A credential that is limited in time and scope carries no inherent value outside of said time and scope.

Modern technologies allow for a more flexible credential mechanism where you can control its scope and time. Leverage a client certificate architecture so login credentials can be minted on-demand to exactly match the access decision. The scope is the surrounding context—a user on a device accessing a server. The certificate is only minted once fully authenticated and authorized, and each has such a short expiration time that it can only be effectively used once.

Key Takeaway

Operating the end-to-end PKI to support client certificate authentication is no trivial task. Okta abstracts the complexities with Advanced Server Access, operating a programmable Certificate Authority under the hood. Servers backed by Okta are configured to trust signed certificates, which are minted to users on demand.

3 User Identities over Shared Accounts

System accounts are directly attributable to a user source via an Identity Provider

With traditional access management products, the system administrator has inherent privileges. Common practice is to leverage separate, shared accounts that are each locked down. The thinking is to follow the principle of least privilege, and limit the activities a user can perform on each system. While the desired outcome of least privilege is in line with the Zero Trust model, the use of separate accounts is counter to the notion of People as the Perimeter.

Contextual access policies are explicit about who should and should not be granted access. To effectively adhere to that policy, access controls need to tie directly to an identity within your system of record, eliminating the use of shared accounts.

This model can only be accomplished with a strong foundational identity layer, reflected on downstream systems via automation. Accounts on the machine link directly to account within the Identity Provider, and any changes are picked up automatically.

Key Takeaway

Okta believes in identity-led access controls, and extends all authentication workflows directly from the user's account. That is then provisioned downstream to your servers. All activity is attributable to the user, making for a clean and consistent audit log.

4 Role-Based Access over Privilege Escalation

System-level permissions are a function of the user's role, and are enforced during initial authorization

Shared accounts delegate privileges for specific activities on the machine, acting as "guard rails" for system paths and commands. Another form of enforcing least privilege, this model of escalation is a very common practice. The problem with this approach is the enforcing policy. Because the management plane is local to the system, it can be extremely difficult to truly know which users should have which rights on any given machine. Managing whitelists and blacklists is painful, and difficult to keep up with at any level of scale.

The more policy and enforcement you can extract from local systems, the better you can adhere to those policies via a central control plane. A shift towards an identity-led access control mechanism means permissions are clearly attributed to the user's role, which is subject to change. For example: A member of the TechOps team is granted 'sudo' privileges on a Linux server, while a member of the DataScience team is only allowed to run read-only SQL queries against a database server.

With this approach, roles are a function of both the user and group membership in the Identity Provider. Systems need to be able to understand the user's role once they are logged in, and grant local permissions accordingly. This is not only a function of automation, but also a function of the local system permission model.

Key Takeaway

Okta simplifies policy adherence by providing a central control plane where group membership is pushed to downstream servers. Command-level whitelisting and blacklisting then becomes a direct function of the user's role, and policy is managed at the access layer.

5 Local Accounts over Directory Interfaces

System accounts are provisioned and deprovisioned directly to the machine

Servers have their own local account and file systems, and it can be challenging to link them to your system of record. A common approach is to run a directory interface on the machine, which then syncs with a backing Identity Provider. On Linux, this can be done with an LDAP PAM module. These interfaces are a headache to build and operate, and break down quickly at scale. It can very quickly turn into a distributed systems challenge, with little guarantee of consistency.

A more effective approach is to directly provision local accounts from the Identity Provider. This method eliminates the need to run a directory interface on the machine, making a more direct link to the role-based access principle, where system permissions are associated with the local account.

Directory interfaces are replaced by a local agent that has control of local accounts, and a direct link back to the system of record. This agent can pick up changes in user status or group membership, and create, update, or delete the local accounts accordingly.

Key Takeaway

Okta manages the local user and group accounts on a machine via a Server Agent, and provides end-to-end automated lifecycle management. If a user is deactivated from Okta, the local user account is instantly disabled, so you don't have to worry which servers that user had access to.

6 Bastions over VPNs

Private systems are protected via a bastion architecture with Layer 7 access controls

Securing your infrastructure environments has traditionally been an exercise in protecting the network. In the modern cloud era, however, we've witnessed the network perimeter break down in favor of the Zero Trust model. As a leading example, Google's BeyondCorp eliminated the use of VPNs for its entire workforce across the globe.

Network segmentation is a highly encouraged defense-in-depth measure, but it should be independent of the access control mechanism. A more effective cloud-native approach to protecting private infrastructure resources is through the use of lightweight bastion hosts. Users authenticate to, and jump through, these hosts to reach the target system. A properly configured bastion architecture eliminates the need for a VPN, extending seamless authentication workflows from any location.

The best approach is to configure private systems at the network layer to access inbound connections via the bastion hosts. Then, deploy bastions to the public Internet, generally as a group of instances for high availability. Once authenticated, there are a number of methods to jump to the target system. Port-forwarding is a common approach, but it remains a security risk because traffic is decrypted. Okta recommends proxying traffic through the bastion, preserving the encrypted channel all the way to the target system.

Key Takeaway

Okta treats bastions as first class citizens, allowing you to configure your target machines with bastion hosts where the authentication and transport happens transparently.

7 Single Sign-On over Checkout Processes

Identity-led login workflows are native to the underlying transport protocol

With a shared account model backed by static credentials, the most common workflow is to authenticate, check out the shared credential for use, and then use it to log in to the system. This out-of-band process can be a painful and slow experience for system administrators, especially during an incident. Because system administrators are highly technical, they will try to circumvent any security controls put in place that get in the way of doing their job.

Single sign-on has become commonplace for accessing business applications, and the same principles and seamless experience are delivered at the infrastructure layer.

With infrastructure resources, this is accomplished by injecting authentication workflows inline to the underlying transport protocol. When a user logs into a Linux box via SSH, it initiates an authentication workflow backed by the Identity Provider. Should a multi-factor authentication policy exist, it is introduced as part of this workflow.

Key Takeaway

Okta built its Advanced Server Access product to interface directly with your local tools, and to work inline with the SSH and RDP protocols for Linux and Windows servers, respectively. All authentication and authorization happens behind the scenes, delivering a more secure method of access control without compromisingthe end user experience.

8 Structured Logs over Session Recordings

Audit events are captured as searchable and alertable structured logs

Forensics analysis is a common line item with any compliance standard, requiring all admin activity on a system to be recorded for playback. The security benefits of having this information readily available are greatly outweighed by the burden of recording, storing, and delivering that data.

From a security outcome perspective, organizations need a clear audit of who has access to what system, from which device, and when—and what they did once logged into that system. In order to build actionable intelligence on that data, it is better captured and delivered as a structure log via a session recording. This allows you to quickly index, search, and alert on this information.

There are two primary ways to achieve this level of audit capabilities: via a gateway that proxies all traffic, or via an agent that runs a capture process. Either method will output structured logs that can be delivered to a logging service or SIEM for further inspection. In both cases, the processing is asynchronous, so as not to interfere with the user session.

Key Takeaway

Okta runs a lightweight Server Agent on each machine, capturing login activity as a log entry that can be further analyzed via the dashboard or third party SIEM service.

Conclusion

As the modern cloud era fundamentally changes the infrastructure landscape, access controls must also change. Follow these eight identity-led principles and your company will be better suited to implement highly secure, automated environments that can scale. Whether as part of a cloud migration or greenfield deployment, getting the architecture right early on saves time, money, and manual headaches in the future.

Assembling an identity and access management system that covers each of these eight principles, across your entire infrastructure fleet, is a big task. Okta Advanced Server Access abstracts those complexities under a single control plane backed by the Okta Identity Cloud.

Learn more about Okta Advanced Server Access here:
https://www.okta.com/products/advanced-server-access/