WHITE PAPER • MAY 2017

Hybrid Privileged Access Management for Hybrid Architectures: A Strategic Approach for Cloud Transformation Risk

Table of Contents

Executive Summary

Challenge

Understanding, managing and containing risk has become a critical factor for many organizations as they plot their hybrid architecture strategy. Access by an expanding array of privileged identities looms large as a risk concern once organizations look beyond tactically using cloud services for cost and agility efficiencies. Existing approaches developed for static infrastructure can address initial risk concerns, but fall short in providing consistent policy enforcement and continuous visibility for dynamic, distributed infrastructure.

Opportunity

Multiple elements factor into how effectively an enterprise can embrace automation and advance the maturity of their transformation. However, security tools are central to enabling a structured and measured approach to managing critical access risks at each stage of the maturity model journey. With the right privileged access platform and set of tools, enterprises can progressively automate and scale access management to align risk mitigation with business needs.

Benefits

By acting as a both a bridge between environments and a dynamic enforcement point for granular authorization policies for all privileged actions, privileged access management designed for hybrid cloud architectures and their specific access designs can enable flexibility, limit vendor lock–in and support proactive detection for attacks exploiting privileged credentials. The path toward aligning risk management with cloud transformation depends on governing processes that can rely on access to all sensitive operations.

SECTION 1

New Architectures, New Risk Management Demands: Consistency, Visibility and Automation for Hybrid Architectures

Many of the current approaches to managing privileged access for hybrid architectures and cloud–based services follow a pattern—they extend their existing password vaulting capabilities to new categories of credentials, whether the keys to the Amazon Web Services (AWS) admin console, SSH keys for developers deploying applications on public cloud services or automated provisioning scripts for AWS EC2 instances that incorporate credentials. Certainly, these approaches allow enterprises to leverage their existing investments in privileged access control to address baseline security and compliance for new use cases.

However, the challenge is that the impact of hybrid architectures and cloud–based services is more profound. Instead, these use cases are the precursors to a transformation in how IT is delivered and functions. Of course, hybrid cloud architectures entail bridging infrastructure that you don't own but manage, and infrastructure that your enterprise both owns and manages. In common with many compliance requirements, the most immediate risk management requirements are ensuring consistency in access and authorization policies for privileged identities across environments, and consolidated visibility into what's happening in all the environments that you do manage.

Who, or what, is accessing privileged credentials to perform an operation is a starting point to address this requirement, but not a complete solution, as adoption maturity generates more complexity and relies on greater degrees of automation. And, with many enterprises concerned about facilitating multicloud deployments and avoiding long–term vendor lock–in, they require a model that is extensible across environments and effective at detecting threats to privileged access security that can be compromised because of lack of oversight or malware extracting credentials from provisioning scripts.

There's another dimension of sustainable risk mitigation for transformation: privileged access security can't be resolved in isolation.

Fit for purpose—not retrofitted.

Generally speaking, the business impetus for adopting cloud architectures is to drive agility and accelerate delivery of applications and services—in addition to reaping the benefits of economies of scale and operational cost savings. The technology implications of these business needs are a growing diversity of privileged identities that have now come into scope (including developers, APIs, code in the form of containers, as well as Internet of Things gateways), as well as software delivery processes (broadly referred to as DevOps) which rely on automation to deliver their full benefits.

For effective privilege access risk management, therefore, the question becomes how authorization for cloud resource access and actions is integrated into automated workflows and processes. Applying well–understood principles of least privilege access and role-based access serve as the foundation for governance, but only if the policy enforcement itself is easily integrated and instantiated in dynamic and distributed infrastructure.

Integrating with cloud services discovery is a useful step to determine which accounts and credentials must be brought under management. However, the fundamental challenge is not simply coverage of new sets of credentials, but how privileged access can be automated in order to:

- Tie actions using privileged credentials back to a specific identity.
- Evaluate specific, discrete actions against role-based access policies in real time.
- Consolidate all activity across multicloud environments for monitoring, reporting and proactive detection via analytics.

If the only enforcement point is restricting access to credentials, rather than at the cloud endpoint, risk management extends only to the equivalent of the front door, not to the actions and changes privileged identities are taking even when the infrastructure is dynamic and the cloud endpoints themselves are ephemeral. The outcome of focusing only on the front door and not on actions at the cloud endpoint (such as AWS EC2 instances) could ironically and counter–productively create a class of over–privileged users with minimal oversight. Retrofitting an approach designed to manage shared accounts for physical servers can address baseline requirements but cannot serve as the foundation for enabling cloud transformation maturity models.

SECTION 2

The Challenge of Hybrid Cloud Privilege User Management: Moving Beyond Password Management

In the identity and access management realm, it's a truism that when you provide a user with access, you've created a risk. In the world of hybrid architectures and cloud–based services, when an administrator or a privileged entity is granted the ability to provision a resource, the risk is even more acute. The compromise of a single, highly privileged hypervisor or cloud service admin console account can potentially impact the entire cloud data center—not just individual server instances.

In contrast to physical datacenters where it's a manual process to provision a resource with plenty of checkpoints, administrators in the cloud world have far more scope to provision, run and then deprovision resources and services. For example, Amazon Web Services points out in its documentation: unless specifically constrained and managed by security groups access polices, security credentials "grant you unlimited use of your AWS resources." Likewise, users granted credentials to the AWS Management Console can access your AWS resources "to the extent that you grant them permission."

However, the burden is on the customer to ensure that the policy logic for those access authorizations is appropriate in terms of least privileged access principles, and consistent with internal requirements such as regulatory compliance or separation of duties. Integrating discovery of existing privileged accounts and credentials, such as SSH keys widely used by application developers and services with automated enrollment, can help reduce the risk posed by an account that sits outside of governance and monitoring frameworks—or that privileged credentials are exposed.

Access control is more than authentication.

Enrolling cloud and application privileged credentials into a password vault does provide a straightforward way to manage who gets access to a shared password, and by extension to a privileged account. The more pressing issue, however, is how the privileged credentials can be used, what level of resource access they authorize and how activity at the cloud endpoint can be controlled and governed against policies, especially as container adoption reduces the average life span of a cloud instance down to days rather than weeks. According to research as of April 2017, "At companies that adopt Docker, containers have an average lifespan of 2.5 days, while across all companies, traditional and cloud–based VMs have an average lifespan of 23 days."[1]

Applying role–based access policies to limit the class of credentials that any one privileged identity can access is a reasonable starting point, but leaves plenty of residual risk if a range of questions are not systematically addressed, including:

- What are the specific, discrete operations that the user or service can perform once they have access to the credentials?
- What roles are appropriate for which privileged credentials, and the user with access to the credentials?
- What is the source of authoritative role–based access policies?

- Are those policies updated and reconciled across environments, especially multicloud deployments?
- When do platform operators get privileged access? What level of authorization is appropriate for them?
- How is developer access managed? How can existing development processes that rely on SSH keys for authorization be integrated into privileged access management governance?

Organizations increasingly face a challenge in ensuring that policies are consistent across all elements of their hybrid architectures—on–premises, virtualized and multiple cloud services environments—and that they're not creating islands of policy logic. Many enterprises already face the phenomenon of privilege creep where administrators will loosen controls or define roles broadly for legitimate operational reasons. The concern here is not only that weak access policies will escalate risk but that the practice makes it more difficult to generate meaningful and granular intelligence through security awareness programs and analytics for proactive detection.

SECTION 3

A Modern Approach to Privileged Management: It's Not What You Know, It's What You Can Do

It's important to note that enforcing authentication to privileged credentials is a necessary building block for hybrid architecture risk mitigation. Requiring authentication before privileged credentials are released— whether administrators accessing the AWS Admin Console, developers leveraging SSH keys to processes in CI/CD environments or APIs accessing credentials to run scripts—is important to establish who the entity is that's gaining access for compliance mandates as well as logging and auditing and reporting. Validating the authentication attempt against an authoritative store is the first step in determining whether the entity is authorized to access the credentials. But authentication doesn't answer a set of questions that are central to both risk containment and effectively integrating access security with the cloud transformation journey

To effectively progress from baseline authentication to authorization that is aligned with cloud environments and enables higher levels of automation in tandem with consistent enforcement of segregation of duties policies, several more elements need to be in place.

In terms of architecture, privileged access management enforcement should be easily instantiated, operate in dynamic environments and be designed to apply polices for a services–based environment that communicate via APIs. In turn, those policies should be contingent on an authoritative source that defines access based on role and least privileged access principles that can be easily abstracted from the native platform tools.

With that architecture in place, enforcement should be able to operate through automated discovery and translate policies to the specific environment and basic operational elements, such as AWS identity management rules and EC2 instances. Maintaining an authoritative data source and then bridging policies to the specific environment limits the potential for privilege creep and reins in the creation of islands of policy logic in multicloud deployments. This approach also facilitates consolidated visibility into activity by a specific identity (whether human or service) and enables federation across environments.

The path toward aligning risk management with cloud transformation depends on governing the actual task being performed, whether through the hypervisor or invoked through APIs, that in turn rely on access to privileged credentials—not just who has access to privileged access credentials.

Rethink Privileged Access Management for Hybrid Architectures

- Create an ability to function within dynamic and distributed environments, with no dependencies and ensure resiliency—run as an AMI or virtual image and be able to scale effectively.
- Leverage APIs to automatically discover virtualized and cloud resources and then provision (or deprovision) appropriate credential and access management policies.
- Leverage APIs as both an enforcement point for granular authorization policies by intercepting API calls and a means of integration into target environments.
- Incorporate privileged identity lifecycle management, policy testing and analytics–driven evaluation for least privilege access and separation of duties.
- Support proactive detection through integrated logging, recording and monitoring, even when dealing with high velocity and ephemeral infrastructure.
- Enable SSO and federation based on corporate credentials to centralize governance and ensure a consolidated, single source of logs and event data for all privileged identities

SECTION 4

Governance for AWS Privileged Identities: Authorization and Policy Enforcement Automation Vs. Access Control

In contrast to approaches that tend to confuse or conflate security for hybrid architectures with securing admin access to a cloud service console using methods designed for static infrastructure, CA Technologies' approach enables customers to implement a more effective and sustainable approach to governing and monitoring privileged access to services and resources. Rather than simply provide a centralized point of access control for AWS, CA enables enterprises to enforce granular, audited authorization policies for all interactions, actions and tasks involving privileged credentials in dynamic, distributed environments.

CA Privileged Access Manager (CA PAM) can provide a critical security and governance bridge to hybrid cloud architectures, especially for Amazon Web Services and virtualized data centers. In addition, the architecture of the CA product allows for greater automation and flexibility within cloud services environments, allowing enterprises to instantiate and scale proxies across cloud data centers.

Ensuring that only authorized privileged identities have access to the AWS Admin console, tasks, resources and APIs, and that all access and activity can be tracked back to a specific corporate–issued identity—whether an administrator, a platform operator, a developer or a service—is critical starting requirement. However, to effectively deliver privileged access governance at scale and facilitate automation for AWS operations, CA PAM builds on the capability to define access and authorization policies centrally and apply them locally. The local (or context–sensitive) application of access and authorization policies has two major dimensions.

Policy automation and propagation.

CA PAM operates seamlessly in a dynamic and ephemeral environment with no dependencies, allowing for the outcome of discovery processes of resources like EC2 instances and provisioning events to be automatically tied to policies. Rather than relying on a two–step process where accounts are discovered and then enrolled, CA PAM facilitates a continuous authorization policy enforcement model.

Policies, in turn, are contingent on an authoritative source, with cloud IAM subsystems relying on the centralized polices. Not only does this approach ensure consistency and limit privilege creep but it supports policy extensibility as processes run across hybrid environments, and reduces the potential for vendor lock–in. Even for enterprises that view AWS as a strategic execution environment, concerns about vendor lock–in are front and center.

Granular, real–time enforcement.

In addition to managing who gets access to the front door, CA PAM provides a deeper layer of authorization enforcement, extending to cloud endpoints (in this case, EC2 instances) and the ability to constrain actions in real time by intercepting and evaluating calls against policies, whether invoked via the cloud admin console or AWS management APIs. Running as a distributed and highly available resource within AWS clouds, the AWS API Proxy acts a dynamically generated broker for the session, governing all interaction with the target resource to ensure consistent control and visibility.

Tackle today’s needs and prepare for tomorrow’s risks.

Because of the power of AWS Admin Console access and the dynamic nature of the infrastructure and cloud endpoints, real–time evaluation of authorization decisions based on role and least privilege access becomes a critical component of risk management and aligning operational maturity. Also, since enterprises are both wary of vendor lock–in, and require some level of abstraction from platform–specific rules and groups, there’s the need for both reuse of centralized policy logic and the ability to automatically integrate with the target environments’ structure.

CA PAM goes one step further than any comparable technologies by providing a proxy–based architecture and the ability to constrain, down to the command level, what actions an administrator, developer or operator or privileged application can take based on predefined, audited policies by evaluating AWS API calls and responses. The CA AWS API proxy extends these controls to operations and tasks that are programmatically invoked.

CA’s approach has an additional advantage for data–driven approaches to threat detection: logs for activity across distributed infrastructure are centrally collected, and the system generates high fidelity.

SECTION 5

Conclusion

The more strategic adoption of cloud–based services and growing focus on hybrid architecture maturity does not simply reiterate the existing challenge of privileged access management in a new set of environments. Rather, the impetus toward transformation and automation means that enterprises must rethink how privileged identities and credentials are managed and secured—not only to ensure that they are not exploited for attacks but to integrate risk management into new processes and dynamic infrastructure.

In these new environments, where APIs are both the means of integration and defining boundaries, privileged access management must now be able to function dynamically while still enforcing a global set of access and authorization polices. However, the core requirements remain of ensuring role–based access to sensitive resources, enforcing granular authorization policies at the process later and maintaining visibility for compliance and proactive detection of compromises or breaches.

For more information, please visit **ca.com/pam**

Connect with CA Technologies

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments.
Learn more at **ca.com**.

[1] Datadog, "8 Surprising Facts About Real Docker Adoption," April 2017.