opentext™

WHITE PAPER

PAM Buyer's Guide

A practical guide to assessing and selecting a privileged access management solution

Contents

Privileged access management continues to expand its role as a security gateway for systems, data, applications, and cloud services that run the business.

The question is no longer whether privileged access should exist, but how to make it controlled, just-in-time and temporary, visible, auditable, and safe.

This guide helps security, IT, audit, and executive teams assess their current privileged-access posture and select a solution on the strength of business value rather than a feature checklist. It is organized around a simple progression: understand the exposure, measure where you stand, then close the gaps.

At the center of the guide is a self-assessment. Thirteen questions, scored on a five-point scale, produce a total that places your organization in one of four maturity bands and points directly to your weakest controls. Scoring the assessment with input from IT, security, and audit gives those teams a common view of exposure that can serve as a baseline for setting priorities, justifying funding, and communicating remaining risk.

The sections that follow expand on the capabilities behind the privileged access security score: account discovery, eliminating standing privilege, request and approval workflows, session brokering and monitoring, command-level least privilege, and audit evidence. The guide closes with a four-dimensional PAM evaluation model (risk reduction, operational fit, audit value, and cost), along with warning signs to test for, proof-of-concept guidance, and a phased rollout plan.

The standard to hold any solution to: can it give the right people the right privileged access, only when needed, with full visibility and proof?

Evolving privileged access to the next level means governing it as a lifecycle—not a vault—and putting the right access in the right hands, only when it is needed.

Source: NIST SP 800-53 Rev. 5, AC-2 (Account Management — the full account lifecycle) and AC-6 (Least Privilege). csrc.nist.gov

Raising your privileged access game

Privileged access management (PAM) began as a narrow tool for storing administrator passwords. It has become something far more consequential: a control point for the systems, data, applications, and cloud services that run the business.

Every enterprise depends on privileged access. Administrators need it to maintain servers, database teams to support critical applications, DevOps teams to troubleshoot production, and security teams to investigate incidents. Service accounts, scripts, and automation platforms need it to perform machine-to-machine work. The question is no longer whether privileged access should exist—it must. The question is how to make it controlled, temporary, visible, and auditable, all of which are needed to drive down risk.

A modern PAM program answers five questions: Who has privileged access? Why do they need it? When should they be allowed to use it? What can they do during the session? And can you prove what happened afterward? This guide is designed to help security, IT, audit, and executive teams evaluate PAM through a business-value lens, so rather than simply comparing features, you can put together an approach to build an environment that reduces risk, fits real operating workflows, supports compliance, and grows with the organization.

Privileged access security complexities

Vaulting privileged credentials remains foundational to securing systems, but organizations should view privileged access as a broader lifecycle rather than as a password-storage problem. Perhaps the most time-consuming and complex task is inventorying human, shared, service, application, and automation accounts across on-premises, cloud, SaaS, and legacy environments. Before you can insert a security gateway to your systems, you need to be able to map each account to an owner, business purpose, privilege level, authentication method, and record of actual use, so that unmanaged and excessive access can be identified.

NIST SP 800-53 establishes least privilege (AC-6) and account management (AC-2) as foundational access-control requirements—privileged accounts are where those controls matter most.

Source: NIST SP 800-53 Rev. 5, Access Control family: AC-6 (Least Privilege) and AC-2 (Account Management). csrc.nist.gov

Here are the categories of system exposures that need to be managed against:

- **Standing privilege:** Users often retain elevated rights long after the business need ends. Every permanently available administrator, root, database, or service-account credential is an attack path that persists whether or not anyone is using it. Standing privilege is the single largest and most avoidable source of privileged-access risk.
- **Credential exposure and shared accounts:** When privileged passwords are copied, pasted, stored in scripts, or shared across a team, accountability disappears and rotation becomes impractical. Activity performed behind a shared account cannot be reliably attributed to a person.
- **Hybrid, distributed infrastructure:** Most organizations will run a mix of on-premises systems, cloud services, SaaS applications, databases, network devices, and legacy infrastructure for years. Privileged access has to be governed consistently across all of it, not through a single cloud-only or on-premises-only control plane.
- **Machine and automation identities:** Service accounts, scripts, DevOps pipelines, and application-to-application connections now request privileged access at machine speed and scale, frequently outnumbering human administrators, and frequently unmanaged.
- **Audit and compliance:** Proving that privileged access is controlled, approved, monitored, and reviewed has become an unwieldy exercise. Where financial systems are in scope, SOX and similar mandates raise the stakes, and manually reconciling evidence from multiple systems is slow and error prone.

To address these vulnerabilities, organizations need to be effective at brokering privileged access through central policy, record what happens during privileged sessions, and produce audit-ready evidence as a byproduct of normal operations rather than a fire drill.

Responding to the latest access trends

Cloud adoption, remote administration, and DevOps automation have permanently changed how privileged work gets done. Administrators now connect to critical systems from anywhere, production issues are triaged through pipelines rather than consoles, and access keys to cloud services have become some of the highest-impact credentials in the enterprise. At the same time, credential theft and ransomware routinely turn a single over-privileged account into a full-scale incident.

Ransomware and credential theft turn one over-privileged account into an enterprise-wide event. Reducing standing privilege is the most direct way to shrink that blast radius.

Source: NIST SP 800-207 (Zero Trust Architecture): per-session, least-privilege access limits lateral movement and reduces the blast radius of a credential compromise. csrc.nist.gov

These forces have pushed privileged access to the center of an identity-based zero trust strategy. Verifying identity, reducing standing privilege, enforcing policy at the point of access, and monitoring activity are no longer aspirational, rather, they are the baseline expectation for protecting critical systems.

Measuring your privileged access capabilities

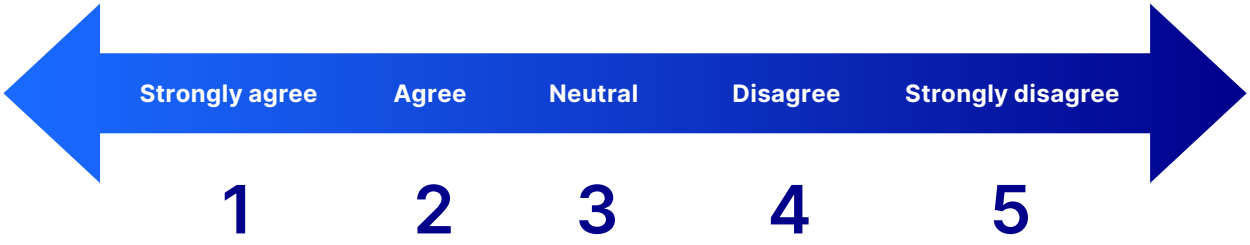
As you evaluate the maturity of your privileged access program, start by assessing the risk your privileged accounts pose to the organization. Your security team likely has artifacts that describe your most sensitive systems, the accounts that can reach them, and the potential impact of misuse. The clearer your understanding of where privileged access exists and what it can do, the more accurate your ratings to the statements on the next page will be.

Accurate privileged-access risk assessment depends on knowing where every privileged account exists—human and machine.

Source: NIST SP 800-53 Rev. 5, AC-2 (Account Management); NISTIR 7966 (Security of Interactive and Automated Access Management Using SSH), which calls for inventory and discovery of privileged access across human and machine identities. csrc.nist.gov

Assessing your environment

Rate your organization on each statement using the scale below, then add your ratings for a total. Lower scores indicate stronger, more mature privileged-access controls.



Statement	Score
We know where privileged accounts exist across our environment, including Windows, Linux, UNIX, databases, network devices, and cloud services; and we onboard them into managed control.	
Administrators can access critical systems without being handed root, administrator, database, or service-account passwords.	
Privileged access is granted just-in-time and time-bound for the approved task, rather than as standing administrative rights.	
We can easily define and enforce request and approval workflows for privileged access, involving the right system and business owners.	
Privileged credentials are centrally vaulted, rotated on policy, and checked out only when policy allows.	
We enforce least privilege at the command or application level, not only at the session level.	
Privileged sessions are monitored and recorded, and we can replay them when an investigation or review requires it.	
We can view live privileged sessions and disconnect or block suspicious activity while it is happening.	
Emergency (“break-glass”) access is controlled, logged, and reviewed after the fact.	
Privileged access is governed through our enterprise identity, roles, and MFA rather than as a separate, disconnected silo.	
We can produce audit-ready evidence—requests, approvals, checkouts, and session activity—without manually piecing together logs.	
Our privileged-access controls span our hybrid reality of on-premises, cloud, SaaS, and legacy systems.	
Our privileged-access records and session logs are tightly secured and monitored to prevent modification or deletion.	
Total	

Assessing the results

Now that you've measured your privileged access capabilities, review the bands below to see where your organization fits. These generalized descriptions don't account for your industry or regulatory exposure, and they aren't a direct indicator of whether you'll be subject to a finding.

Under 26

Your organization likely has a mature, policy-driven PAM program in place. Consider a review of your tooling and processes to ensure your implementation stays future-proof as cloud and automation expand.

27–38

You've put key privileged-access controls in place, but there is still work to do to reduce standing privilege and operational friction, and to lower your risk of privileged misuse.

39–51

You may have several point solutions in place, but demonstrating control and giving administrators fast, reliable access is difficult when those solutions aren't integrated.

52 and above

Your organization is likely exposed to credential theft, insider misuse, and audit findings, with a high potential for privileged-access gaps. Prioritizing a privileged access management program should be a near-term objective.

Investing the time and effort to obtain input from IT, security, and audit teams that have concrete knowledge of each metric provides a consistent scoring process and gives security, IT, audit, and leadership teams a common view of the organization's current exposure. Repeating the assessment over time also provides evidence that the PAM program is improving, or where weaknesses still need to be addressed. A documented baseline makes it easier to set priorities, justify funding, measure progress, and communicate remaining risk to decision makers.

Before you can hand the right privileged access to the right person, you have to see each privileged account—and know what it can do.

Source: NIST SP 800-53 Rev. 5, AC-2 (Account Management) and AC-6(7) (Review of User Privileges — validate the ongoing need for privileges). csrc.nist.gov

Addressing your privileged access gaps

In addition to your overall score, focus on the statements where you rated lowest. The sections that follow explain those capabilities in more depth and describe what a strong implementation looks like. It is quite likely that as you go through the process of assessing solutions for your environment, that based on your business priorities you will be weighing some capability categories more than others.

Discovering and controlling privileged accounts

You cannot control what you cannot see. The first step in reducing privileged risk is knowing where privileged accounts exist; and that inventory is almost always larger and messier than expected. A strong discovery capability surfaces unmanaged accounts, stale and disabled accounts, expired accounts, and the service accounts that quietly accumulate broad rights. Just as important is a repeatable onboarding process, so the program can scale past its first wave of critical systems rather than stalling after the easy wins.

In evaluation, look for discovery that spans your operating systems and platforms, distinguishes account types, and feeds a repeatable onboarding process that brings accounts under managed control.

Eliminating standing privilege

Standing privilege is convenient and dangerous in equal measure. The goal of a modern program is to make elevated access temporary, approved, and scoped to the task so a user holds administrative rights only for the moment they are doing administrative work. Effective just-in-time access combines request and approval workflows, time-bound grants, policy enforcement, controlled emergency access, and revocation, ideally without ever exposing a credential to the user.

When you evaluate just-in-time capabilities, confirm that access can be time-bound and automatically revoked, that approval and emergency paths exist, and that the model extends to cloud and ephemeral scenarios and not just traditional servers.

A well-built workflow presents the approver with exactly what they need to make a fast, defensible decision—at the moment they need to make it.

Source: NIST SP 800-53 Rev. 5, AC-2 (Account Management): requests require approval by defined roles, and privileged access receives additional scrutiny by the system or business owner. csrc.nist.gov

Workflows that work

You know you have a well-implemented privileged access workflow when it is easy for administrators to request and gain the access they need, while system owners retain control. The main vulnerability is the quiet proliferation of standing grants, which is access extended beyond its intended level and never reclaimed.

Build privileged-access workflows by first mapping how administrators request, receive, use, and relinquish access for each major resource type. Define the information every request must include, such as the target system, requested role, business reason, duration, ticket or change reference, and level of urgency. Assign approval responsibility to people who understand both the requester's need and the risk of the resource, and establish escalation and emergency paths for situations in which normal approvers are unavailable. Pilot the workflow with representative administrators and system owners, then adjust it based on completion time, denial reasons, exceptions, and attempts to bypass the process. Key areas to focus on:

- **Encompassing coverage:** The more complete your program, the less often users route around it. Every high-risk system should be reachable through governed access; when people are forced outside an automated request path, it is more work for them and less safe for you.
- **Simplicity for the requester:** The tool needs to be easy to find and usable when and where administrators work, typically any time, on any device, including browser-based access. Resource lists should be granular enough that requests are limited to only what is required.
- **Effective approver:** The critical element is presenting the approver with enough context, such as identity, role, business need, and risk, to make a fast decision and to guard against rubber-stamping. The approver should be able to answer two questions: Does the requester merit this access and what is the total risk of granting it?

Following this guidance makes governed access easier to use than informal alternatives, which increases administrator adoption and reduces policy workarounds. Complete coverage and granular resource selection reduce the likelihood that users will receive broader access than a task requires. Context-rich approvals improve decision quality while helping legitimate requests move quickly enough to support operational needs. The resulting request and approval history also creates a defensible record showing why access was granted, who authorized it, and how long it remained available.

Brokering and monitoring privileged sessions

A PAM platform should not only decide whether access is allowed, it should govern what happens after the session begins. Session brokering lets administrators connect to systems without ever handling the underlying credential while monitoring and recording create a defensible record of activity. Evaluate coverage across SSH, RDP, databases, applications, and cloud services, and look for session recording, keystroke or command visibility where supported, live viewing, replay, and the ability to disconnect a session on policy or on demand.

So they will hold up as evidence when an investigation or audit is required, recordings and logs need to be tamper-resistant, searchable, and retained in line with defined policies.

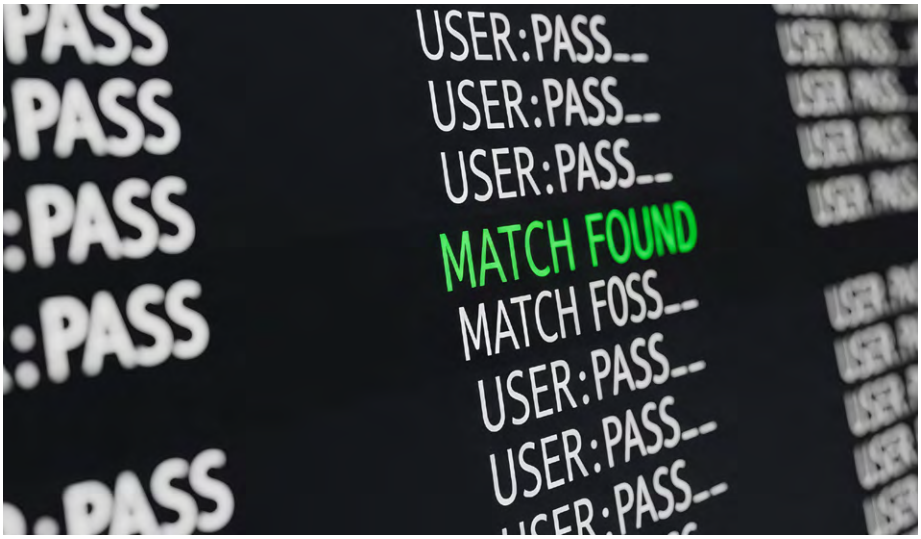
Command control and least privilege

Session-level access can still be too broad. The most tuned environments enforce least privilege at the command or application level: an administrator may be permitted to restart a service but blocked from deleting configuration files or changing security settings. This is where broad, convenient access gives way to precise, defensible control.

Least privilege at the command level is the difference between trusting a session and authorizing an action.

Source: NIST SP 800-53 Rev. 5, AC-6(3)/(9)/(10) (access to privileged commands, logging and restriction of privileged functions); NISTIR 7966 (command restrictions on authorized keys). csrc.nist.gov

In evaluation, test whether the platform can allow, deny, or flag specific commands or actions under policy, and whether those controls are practical to define and maintain at scale, so you can move from “trusted for the whole session” to “authorized for this action.”



Compliance and attestation

Audit readiness should be designed into the privileged-access program rather than assembled shortly before an assessment. Begin by mapping applicable regulatory, contractual, and internal-control requirements to specific PAM evidence, such as requests, approvals, credential events, session activity, policy changes, and emergency access. Assign owners for producing, reviewing, protecting, and retaining each type of evidence, and confirm that timestamps, user identities, resource names, and approval details are understandable to someone outside the operating team. Schedule recurring self-assessments and access reviews so gaps can be corrected before they become formal audit findings. The following objectives typically drive privileged-access attestation:

- **Self-audit** within a team or across the organization to measure the effectiveness of current controls.
- **Internal attestation** as part of risk management, to assure a partner or customer that privileged access meets a defined standard.
- **External audits** by regulators, such as PCI, HIPAA, SOX, GDPR, and similar, where a finding can carry a fine, which is why external audits tend to drive the internal ones.

This approach makes compliance evidence a by-product of normal access operations rather than a separate manual project. It reduces the time that security, IT, and audit teams spend gathering and reconciling information from multiple systems. Regular attestation also surfaces stale access, missing approvals, incomplete monitoring, and policy exceptions before those issues create greater exposure. Reliable evidence strengthens confidence among auditors, customers, partners, and executives that privileged access is being controlled as intended.

Evaluating solutions

Start the evaluation by translating your highest-risk privileged-access scenarios into testable requirements rather than beginning with a generic feature checklist. Build a weighted scorecard that reflects the importance of each area. Score candidates against four dimensions:

- **Risk reduction:** Does it meaningfully reduce attack paths, such as cutting standing privilege, preventing password exposure, enforcing least privilege, and letting you monitor and interrupt risky activity?
- **Operational fit:** Can administrators actually use it? Does it support your SSH, RDP, database, and cloud workflows; browser-based and agentless options where needed; existing directories; and emergency access without forcing every team into one rigid path?
- **Audit value:** Can it produce evidence auditors and risk teams can use, such as clear reports, searchable sessions, and a defensible record of who approved what and why?
- **Cost and complexity:** Look beyond license cost to deployment time, infrastructure, agent maintenance, professional services, integration effort, and the number of modules required to reach the outcome you want.

The strongest privileged-access programs aren't a standalone vault—they're part of a broader identity security strategy.

Source: NIST SP 800-207 (Zero Trust Architecture): identity is the primary control plane; access decisions are identity-centric rather than perimeter- or point-tool-based. csrc.nist.gov

Evaluating solutions in this way reduces the risk of selecting a platform that performs well in a scripted demonstration but fails in day-to-day operations. A balanced scorecard makes trade-offs visible and prevents a single attractive feature or low initial price from dominating the decision. Cross-functional participation improves adoption because the needs of administrators, control owners, auditors, and budget holders are considered before purchase. Testing real scenarios also produces a more credible estimate of implementation effort, operational impact, risk reduction, and total cost of ownership.

Integrating privileged access with identity

Privileged access is strongest when it is connected to identity rather than run as a separate silo. Evaluate whether a platform uses your existing directories, groups, roles, and authentication sources; whether it supports MFA and risk-based, step-up authentication at the moment access is requested; and whether it fits your broader identity governance and lifecycle processes. A tool that duplicates identity administration adds cost and drift; one that participates in your identity program strengthens it.

Building your privileged-access model

As with any organization-wide initiative, a durable privileged-access model depends on commitment from the top and a phased rollout that shows value quickly. Next, establish the operating model before expanding PAM across the organization. Create a phased roadmap based on system criticality, account exposure, regulatory scope, and the feasibility of bringing each environment under control. Review progress at the end of each phase and use the lessons learned to improve policies, workflows, integrations, support processes, and rollout plans for the next group of systems. Rather than attempting every use case at once, start where the risk is highest and expand:

- **Establish visibility and control** over your highest-risk accounts: domain administrators, root, database and cloud administrators, and shared local admin accounts. Discover, onboard, and vault first.
- **Introduce session monitoring** for the systems where misuse would do the most damage: production servers, sensitive databases, security infrastructure, and regulated systems.
- **Reduce standing privilege** by moving users to request-based, time-bound access with defined approval and emergency-access processes.
- **Extend controls** to cloud services, applications, and service accounts, prioritizing accounts with broad permissions or weak ownership.
- **Optimize reporting and governance**, aligning evidence to your control objectives and audit cycles, and establishing regular privileged-access reviews.

A phased model produces measurable security improvements earlier than an attempt to transform every privileged-access process at once. Starting with the most consequential accounts reduces exposure where a compromise has the greatest operational or financial impact. Defined ownership and repeatable rollout practices also prevent the program from stalling after the initial deployment. Over time, the organization gains a consistent control framework that can expand across technologies while continuing to support administration, audit, and business operations.

High-value use cases to prioritize

Before you prioritize, gather the information that shows where privileged risk concentrates. Rank those systems by data sensitivity and business criticality, compare granted privileges against actual usage to expose unneeded standing rights and unmanaged credentials, and confirm your findings with the system owners, IT, DevOps, security, and audit teams who know the exceptions an inventory misses.

With that picture in hand, prioritize the use cases that deliver the clearest risk reduction:

- Secure administrator access to critical Windows servers: broker RDP, enforce policy, and record sessions instead of sharing administrator passwords.
- Secure elevated SSH access to Linux and UNIX: provide controlled, audited access in place of shared root credentials.
- Reduce standing access with just-in-time privilege: grant elevated rights only for an approved task and duration.
- Protect privileged access to cloud services: control high-impact keys and access patterns with temporary, policy-driven access.
- Improve audit evidence for privileged activity: centralize requests, approvals, sessions, and reviews.
- Support emergency access without losing control: fast access with request history, monitoring, and post-event review.

As you weigh the results, you should be able to show concrete risk reduction. These calculated reductions can help build support from both auditors and the leadership who are funding a PAM upgrade.

Automating requests, approvals, and reviews

A capable platform automates the request-to-review lifecycle. Requesters work through a familiar, self-service experience; approvers are presented with the business context and risk they need to decide quickly; and requests escalate to another level when policy requires.

Beyond automation, look for the ability to see who accessed which resources and when, respond when a risk threshold is exceeded, and reliably tie every session to an identity. Access granted outside the standard model, such as one-offs and exceptions, should be easy to surface to owners and security teams and bring back under governance.

Break-glass and continuous review

Even with access onboarded and workflows automated, you still need to react to out-of-bounds situations and emergencies that require immediate access, and activity that falls outside accepted criteria. Controlled break-glass access provides speed without sacrificing accountability: emergency requests are still logged, monitored, and reviewed after the event.

Continuous review complements it, periodically re-examining privileged grants to surface access that is unused, stale, or excessive—reducing both risk and account-based licensing cost. Manual review at this scale is impractical and invites reviewer fatigue; automating it is what lets you maintain your security edge.

Warning signs in a PAM evaluation

Assess warning signs through structured testing rather than relying on vendor assurances. Add each concern below to the evaluation scorecard and require the provider to demonstrate how the solution addresses it in the deployment model you intend to use. Document any prerequisite products, agents, services, custom integrations, manual processes, or additional licenses needed to achieve the stated outcome. Validate important claims through a proof of concept, architecture review, customer references, support documentation, and a multi-year total-cost analysis.

Be cautious when a solution:

- Leans heavily on password vaulting but offers weak session control.
- Requires broad, permanent access for convenience.
- Forces every workflow through a single access method.
- Demands heavy agent deployment with no agentless alternatives.
- Gives limited visibility into what users did after access was granted.
- Makes audit evidence difficult to retrieve.
- Requires multiple add-on products to deliver core PAM outcomes.
- Doesn’t integrate well with your existing identity and MFA investments.
- Creates excessive friction for administrators.
- Cannot support hybrid infrastructure.

Applying these checks helps prevent hidden dependencies and operational limitations from emerging after the purchase has been made. It also protects the organization from underestimating implementation effort, infrastructure needs, administrative overhead, and the cost of required add-ons. Testing usability and integration reduces the likelihood that administrators will resist the solution or bypass its controls. The right platform should improve control, accountability, and auditability without making critical IT work unnecessarily difficult.

Resources

OpenText NetIQ Privileged Access Manager stands out with its holistic approach to privileged access management (PAM), integrating robust security features, advanced automation, and a user-friendly design into a scalable platform.

Find out how OpenText™ NetIQ™ Privileged Access Manager can meet your PAM needs ›

Running a meaningful proof of concept

A proof of concept should test your highest-value scenarios in your real environment, not a generic demo. A strong PoC includes a Windows access scenario, a Linux or UNIX SSH scenario, a credential checkout or password rotation, a just-in-time request and approval, a session recording and replay, a live-monitor or disconnect, an audit report review, and an integration with your directory or MFA.

Score it with the teams who will live with it: security on risk reduction, IT operations on usability and reliability, IAM on integration and policy fit, audit on evidence quality, administrators on workflow impact, and finance on cost and operational overhead. Require evidence, not claims—if a capability matters, test it.

Making your decision

The best PAM solution is not necessarily the one with the longest feature list or the most impressive demonstration. It is the one your organization can deploy, operate, and expand successfully, while meeting your security targets.

Can this platform give the right people the right privileged access, only when needed, with full visibility and proof? That is the standard a modern PAM solution should meet.

Source: NIST SP 800-53 Rev. 5, AC-6 (Least Privilege) and AC-2 (Account Management) for “right access, only when needed”; AC-6(9)/AU-2 (logging of privileged functions) for “visibility and proof.” csrc.nist.gov

It’s hoped that this buyer’s guide has provided you with an informative approach with a set of metrics and KPIs that helps organize your selection process. Following the guide should help you select and implement a solution that delivers strong control without unnecessary friction, produces defensible evidence, and gives your teams a realistic path from initial deployment to broader privileged-access maturity.

opentext™