PAM
Best Practices Guide

Contents

Introduction

This guide is designed to show system administrators how to install, initialize and run Privileged Access Management (PAM) on a Unix computer.

Technical Support

If at any time you encounter an issue, have questions or need guidance, please contact us using the information provided in our documentation site.

If questions remain or issues arise while using PAM, please contact the Support team using this link: https://support.imprivata.com/communitylogin.

Privileged Access Management

Privileged Access Management (PAM) is an agentless solution that provides a secured database to manage privileged accounts and secrets, establishes secure sessions for users through a standard web browser and automates the execution of jobs or tasks without disclosing or sharing access, passwords or secrets.

PAM contains the following core components:

Privileged Account Management

A secure AES 256-bit encrypted database that contains records which can be stored, shared and used without disclosing the actual privileged account or its secrets (passwords, certificates or keys).

Privileged Session Management

The ability to establish a privileged session to an underlying system (Windows, Unix, Linux, Mac, or Network Device) through a standard web browser or native clients (such as PuTTY, Secure CRT, mstcs, and many others) while providing the means to monitor, join, record or terminate this session.

Privileged Job Management

Schedule, automate or execute on demand jobs to privileged systems without embedding the secrets in scripts or sharing them with untrusted users.

Planning your Build Out

The key to a successful deployment and ultimately user adoption is proper planning.

Before you begin your build out process, please consider the following questions and scenarios.

- What are you trying to accomplish with PAM? Do you plan on using your records for secure vaulting and sharing, session management, task automation or all the above?
- Which (and how many) assets, accounts and secrets do you plan on securing within PAM?
- Will PAM be used by a select group of power users like your IT Department or will it be rolled out across your entire organization?
- How do you plan on categorizing your records in PAM so they can be easily found and managed? Organized by department, relationship, or geographies?
- Are approval workflows (Dual Control, Four-eyes) required on any of your records?
- Should users' login with PAM local accounts or reuse their AD or LDAP accounts? Do you want to implement another layer of security by integrating with multi-factor or two-factor authentication?
- Do you understand your "break glass" scenario?

Answering these questions and understanding your true objective prior to deploying PAM is crucial to starting out on the right foot.

This may require spending time interviewing your various stakeholders and colleagues, gathering requirements and of course collaborating with your team during this process.

While it is possible to change course after the product has been rolled out into production use, it is easier on everyone to start from a solid foundation and build upon it.

Let's get started on building that solid foundation.

Using Folders for Organization and Inheritance

When most computer users think of electronic organization they tend to think about a Windows file system.

While you certainly could keep all your documents in the root of your **My Documents** folder or your *Desktop*, that makes it quite cumbersome and difficult to find, use and share documents when needed.

Instead, users quite rightly create folder hierarchies to organize these files into some logical structure.

Much like these modern file systems, PAM operates with the same underlying structure of folder organization.

PAM folders contain records or folders and provide the *following benefits*:

- Can be used to easily categorize records based on similarities like department, asset, geographies, office locations and the like.
- Can be used to simplify sharing by establishing a permission inheritance model on a parent folder.
- Can be used to simplify workflow bindings by establishing a workflow inheritance model on a parent folder.
- Each folder can be thought of as individual vault with its own permission model.

When planning your folder hierarchy think about these benefits and how they may be applied to your business need. The more you can take advantage of all forms of logical groupings and inheritance, the easier it will be to manage, maintain and understand PAM.

Example of a common IT scenario

You are managing several IT assets in PAM, a domain controller, a development web server and a production web server.

Your IT Manager will need access to all three, your Web Developer will only ever need access to your web server and your AD Admin will only ever need access to your domain controller.

How would you best create a folder hierarchy that would support this scenario (and be extensible to support future growth) while keeping the earlier benefits in mind?

A recommended approach would be to start with a parent folder like IT Infrastructure and then create sub-folders beneath it to organize assets by usage.

For example, a folder for Web Server assets and another for Active Directory assets.

When looking at this hierarchy it makes use of PAM folder benefits by:

- Grouping assets by logical similarities so users can easily find what they need.
- Makes use of permission inheritance by allowing IT Manager(s) access to all assets, Web Developer(s) access to only the web servers and AD Admin(s) access to only AD controllers.
- In a comparable manner to permissions, approval workflows can be applied (as needed) to these same folders so extra safeguards are placed on the child records.
- Allows for future growth with logical extensibility. As you bring your other IT assets into PAM like your PBX servers, Azure, and Amazon Webservices accounts, API keys and more you simply create a new folder under IT Infrastructure and begin to apply the same methodology.

In summary, think about not only how records should be stored in folders, but also how they will be shared (or not shared) with others, if additional safeguards like approval workflows will be used and finally how this hierarchical structure can be expanded for future asset growth.

Once you have a handle on your folders, it's time to begin thinking about your records. But before you jump into creating records, we need to think about a record's foundation which are record types.

Understanding Record Types

Record Types are the foundation of Records simply because when creating a new record, you need to first select which record type to use.

When considering how to structure your Record Types in PAM, keep these concepts in mind:

- PAM comes with many **out of the box** record types built in. We recommend that you do not delete or modify these types.
    - Instead of modifying these types, consider creating a new custom record type using these default types as a Parent to extend their need.
    - Instead of deleting these types, consider using the Hide option to remove them from the list of record types that users can select. This keeps the PAM user interface clean and organized without invoking such a permanent action like Delete.
- If creating new record types, keep the names short and easily recognizable.
- Like folders, record types use inheritance for Formulas, Tasks and Command Control. Consider how these types will be used for records and if any unique or custom tasks can be applied for use with inheritance. Although an AD record and a Web Server record may both use a Windows Host type, the task(s) associated with each may be different so two record types can help in this situation.
- Custom fields can be added to record types to capture additional parameters to records.

How to effectively utilize record types in your PAM deployments is something that needs to be decided early on. Creating too many record types will lead to user confusion and management difficulties, while too few can lead to misuse by users and setup a scenario where they are not flexible enough to meet your future growth or demands.

Managing Assets with PAM Records

You have your folder hierarchy, designed and built your record type hierarchy and now it is time to create records to manage your assets.

When creating records, keep these concepts in mind:

- Can the use of a reference account be used to minimize configuration and task execution?
- Can I make use of the Import function to quickly populate records from CSV files or other connection management and PAM products?
- Which folders should they reside in? Be conscious of inherited folder permissions or work-flows when choosing this location.
- Which record types should be used? Remember record types use inheritance for tasks, for-mulas and command controls policies, so these objects will automatically be applied and could also be executed automatically.
- Should all users be able to create new records? Only users with Owner permission to a folder are permitted to create new records within it.

It is important to remember the principles of inheritance on both folders and types when creating new records. If forgotten or misunderstood, you could unintentionally share an asset with a user.

Sharing and Permissions

Sharing folders, records and access to PAM configuration are very important decisions that need to be made initially and need to be continually updated as necessary.

When planning your PAM security model, please keep these concepts in your design:

- Assign permissions to groups, rather than individual users, whenever possible. Managing group membership is far easier than managing hundreds of individual users.
- The System Administrator role should only be given to a very select group of users. The System Administrator has access to the entire PAM system including all records and folders, regardless of inherited permissions.
- When deciding which permission to grant, start at the lowest (Viewer) and then ask yourself, does this user or group need more access and if so, why? Starting with least privileged is best.
- Most users will not need more than **Viewer** permissions to any record or folder. This grants them the ability to see these objects, but not edit, unlock or compromise them.
- Users with **Owner** permissions to a record or folder have full control of that object (and inherited objects) including the ability to edit, share with others and delete. Limit the number of users who have this level of permissions to any objects.
- Permissions assigned to the PAM **Root Folder** are applied to the All Records view. That means if a user is granted Viewer permissions to Root Folder, they will have View permissions to every object (folder and record) in PAM that has the default inheritance configured.
- Make use of the PAM User and Inventory reports to periodically monitor your security model and ultimately make changes when and where needed.

Tasks, Policies, Execution and Automation

PAM Tasks are objects that contain a script that is executed against the record's host or object and a policy that dictates when or how it is executed.

When considering your PAM deployment, you must also decide if you are going to take advantage of the PAM Task Engine, and if so, on what records.

Because tasks can also be deployed via inheritance, consider the following when building out your plan:

- Most default PAM record types come pre-built with tasks, which means the use of these record types will apply the tasks via inheritance (for example, Windows Host includes the Windows Password Reset task already assigned). Be conscious of this concept because some tasks can be set to automatically execute.

- Use inheritance whenever possible. Rather than organizing tasks on each individual record, consider applying them to record types for ease of use and management.

- Formulas (password complexity) can also be applied via inheritance which provides the ability to decide which types have more or less complex formulas.

Ensure you understand which Tasks will be applied to each record type when inheritance is being used.

A task may be applied without you realizing that could then run a script automatically (or scheduled) against your host record.

Workflows

PAM Workflows provide an extra level of security to both records and actions that can be assigned to users, groups, folders or records.

If you are planning on including approval workflows (dual control, four-eyes) to your PAM deployment, please use the following as guidelines

- Whenever possible, keep the approval as simple as possible. If you require too complex of an approval process, the likelihood of it never being completed increases.
- Think about the who, what, when and where of workflow objects. Putting a workflow request in front of users who are trying to simply complete their work, could lead to frustration if it causes unnecessary slowdowns and layers upon layers of approval.
- Test your workflows (templates, bindings and notifications) with a small group of users before rolling it out to everyone. This will help find any misconfigurations or hiccups in the system before it is put into production use.

Approval workflows are a powerful, and often required, object in many systems throughout an organization, but it is important that they be efficiently built and deployed only where necessary. If you make them too burdensome and overbearing for users, then they will actively look for alternate methods to work around the system.

Alerts and Notifications

The System can be configured to send email notifications and in-app alerts to users who have subscribed to certain events.

If yourself or users are going to configure notifications, understand that:

- Alerts and Notifications are user profile specific. The alerts that you subscribe to are only for your account.
- PAM can be as noisy as you want it to be. When you first start out, think about what events you need to be alerted to. Over time, you can adjust the level of notifications, but like all notifications, the more we receive the less we tend to pay attention. Create alerts for notable events and use PAM reports for review.
- If you already have a Syslog or SIEM product capturing security alerts, consider outputting PAM events to your Syslog for consolidation, reporting and additional alerting.
- If you are a System Administrator, subscribe to system **Error** alerts so that you receive notifications in the event of system issues. Also subscribing to system **Information** or **Warning** events can quickly fill up your inbox or alert listings which can make PAM overly chatty.
- Email notifications (if configured) are sent to the email address associated to the AD account or the Local User account. Be sure these addresses are correct for all your users.
- Email notification templates can be customized. If you would like to change the wording, add or remove placeholders, feel free to do so. Use the test email template to try your changes before applying them to the production templates.

Take advantage of alerts and notifications, over alert yourself in the beginning stages and then gradually scale back on the events that are less important.

Finding that happy medium between alerts and noise is key to effectively managing PAM.

Administrative Responsibilities

Like all solutions that are brought under the IT or Security umbrella, a certain amount of maintenance and monitoring by the System Administrators is recommended.

The following list provides a look into what these duties may entail:

- Do not lose or misplace the master password that is generated during installation. The system is encrypted and it cannot be decrypted without this password.
- Ensure you have an adequate export plan in place (either manual or periodic). Without exports, there is no way to recover from disasters or data loss.
- Monitor system performance on a regular basis, both on and off-peak times. Ensure session connectivity is not lagging, tasks are executed in an expectable amount of time and the user interface is responsive.
- PAM follows an agile release process with weekly updates. When an update becomes available, read the release notes and if you choose, deploy the update. Keep in mind, that this update should be done during an off-peak time and you should check for any active sessions or scheduled tasks before you begin, as services will go offline during the update process.
- Test any changes that may affect other users before making them. If you have a UAT environment, test system changes, including record types, workflows, tasks and updates, there first and then make them available in production.

Conclusion

The purpose of this guide was to introduce certain concepts related to Privileged Access Management solutions, specifically Privileged Access Management and to alert you to specific considerations that should be kept in mind when building out your PAM deployment.

PAM makes broad use of inheritance across many objects so that configuration and management of the system can be simplified, but to achieve this goal you need to plan for it.

It's important to get started in the right direction to minimize the changes that may need to be done later, but it is also important to understand that changes can be made later.

Plan, test and deploy, let users login and start using PAM, then reflect on their behavior and requirements and adopt changes were needed.

For additional guidance, consider reviewing common standards and regulations mandated in your geography or industry. Regulations like NIST, GDPR, ISO and HITRUST can also be helpful when implementing specific policies in PAM.

In the end, a PAM solution deployed in any configuration (even our default settings) is much better than a PAM solution not deployed.