BeyondTrust

# The Guide to Just-In-Time Privileged Access Management

What It Is, Why You Need It &
How to Implement It

# TABLE OF CONTENTS

# INTRODUCTION

## Taking "Just Enough" Access To The Next Level

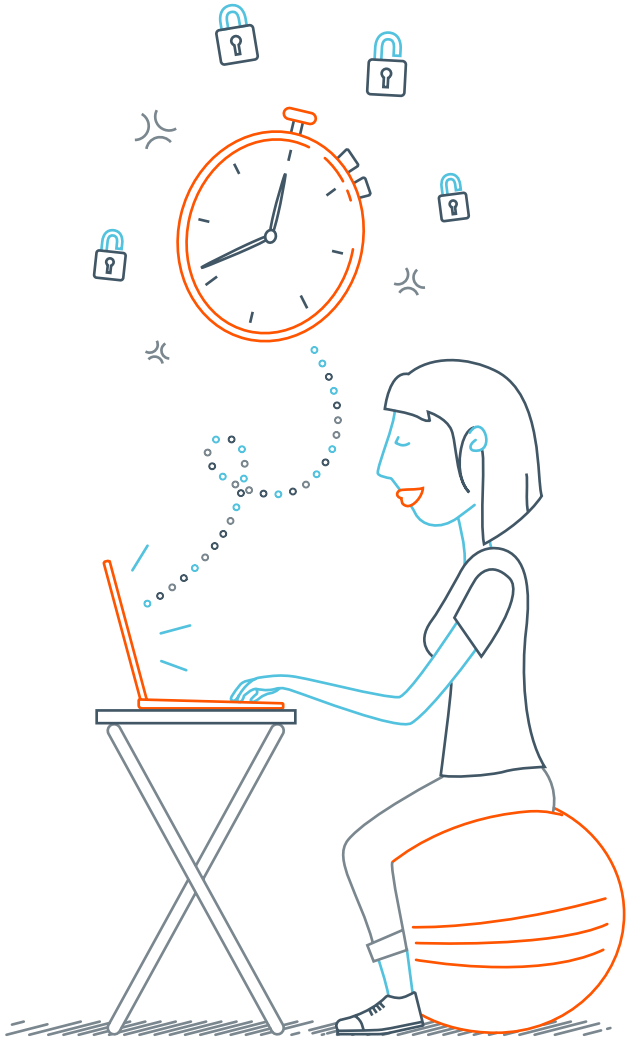
A true least-privilege security model requires users, processes, applications, and systems to have "just enough" rights and access to perform tasks —and for no longer than necessary.

Organizations are increasingly effective at applying the "just enough" access piece using privileged access management (PAM) solutions, but they have largely neglected the time-limited part and persistent risk portion of the equation for privileged user accounts.

For the last 40 years, "always-on" privileged accounts have been the default mode for administrative access and have proliferated across enterprises, presenting a massive risk surface. Privileged access, rights, and permissions that are always in an active mode are ready to be exercised at any time—for legitimate activities as well as for illicit ones. And this risk surface is rapidly expanding with the expansion of virtual, cloud, and DevOps environments, internet of things (IoT) devices, as well as in emerging areas, such as robotic process automation (RPA).

Against this backdrop, it's no surprise that the abuse and/or misuse of privileges play a role in almost every cybersecurity breach incident today. With privileged access in hand, an attacker essentially becomes a malicious insider, an alarming scenario for any IT professional, all the way up through the C-level and the Board.

Privileged accounts are now truly everywhere across your organization, but traditional, perimeter-based security technologies can only protect privileged accounts within their boundaries. Just-in-time (JIT) privileged access management (PAM) can help drastically condense the privileged threat surface and reduce risk across the enterprise. Implementing JIT PAM means that identities only have the appropriate privileges when necessary, and for only the least time necessary. This process can be entirely automated so that it is frictionless and invisible to the end user.



**Organizations are increasingly effective at applying the "just enough" access piece using privileged access management (PAM) solutions, but they have largely neglected the time-limited part and persistent risk portion of the equation for privileged user accounts.**

## What is Just-In-Time Privileged Access Management

Just-in-time (JIT) privileged access management (PAM) is a strategy that aligns real-time requests for usage of privileged accounts directly with entitlements, workflow, and appropriate access policies. Companies use this strategy to secure privileged accounts from the flaws of continuous, always-on access by enforcing time based restrictions based on behavioral and contextual parameters.

A privileged account is defined as an account that is granted privileges and permissions above a standard user, and includes the following:

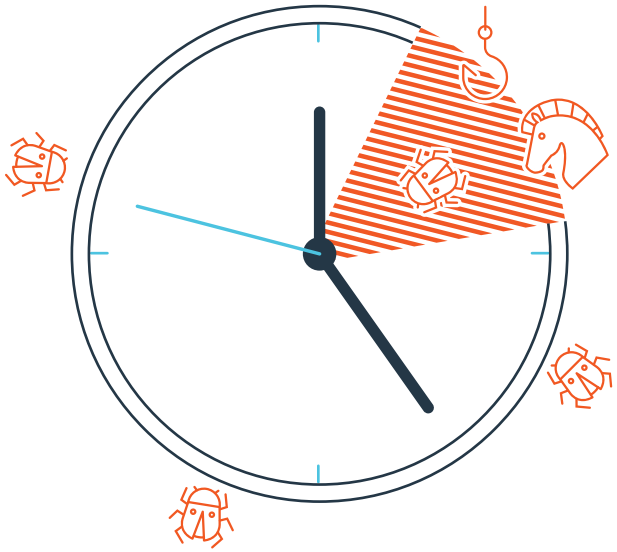
- Superuser account with immense privileges, such as administrator (in Windows environments) or root (in Unix/Linux environments)
- Power users with privileges that fall between a superuser and a standard user account (also called a non-privileged, or least-privilege user account).

JIT PAM sharply limits the duration for which an account possesses elevated privileges and access rights, drastically reducing the window of vulnerability during which time a threat actor can exploit account privileges. JIT helps enforce the principle of least principle to ensure that privileged activities can be performed in alignment with acceptable use policies, while forbidding privileged activities that fall outside of the right context.

When a privilege is requested, it must meet the required contextual parameters before being checked out—the privilege is never owned by the account. This mitigates the risk for their misuse when potentially leveraged outside of a privileged access management deployment. No longer are privileged accounts essentially fully armed and ripe for abuse.

As an example – consider a typical always-on privileged account that may be "privilege-active" 168 hours a week. By shifting to a JIT PAM approach, you could reduce that privilege-active state from 168 hours down to just a couple dozen minutes. Multiplying this effect across all your organization's privileged user accounts will have a truly massive impact on risk-reduction.

Adopting just-in-time as part of your privilege management approach means you can implement a true least-privilege model enterprise-wide. And, the exposure is not just based on time. Attack vectors that utilize techniques like lateral movement are also mitigated since there is no "always-on" privileged account to leverage across resources.



**Companies use this strategy to secure privileged accounts from the flaws of continuous, always-on access by enforcing time based restrictions based on behavioral and contextual parameters.**

# AUTOMATING JIT PAM

## JIT Methods

A JIT approach to administration of privileges requires organizations to establish criteria for just-in-time privileged access and accept that the accounts that fall within this policy are not available outside of potentially break-glass scenarios.

While similar, well-established concepts for JIT exist across other use cases, such as manufacturing, applying the model for a security and operations solution presents some unique technical considerations during an implementation.

The goal of a JIT privileged account is to automatically assign the necessary privileges "on the fly" based on an approved task or mission and subsequently remove them once the task is complete or the window or context for authorized access has expired.

The modeling required to take a standard user account and apply the appropriate privileges can be implemented by using any of the following six JIT Methods:

### 1 JIT PRIVILEGES

The account has individual privileges, permissions, or entitlements added to perform a mission once all criteria are met, but only for a limited duration. These rights need to be revoked once the mission is complete and should include certification that no other privileges were inappropriately altered.

### 2 JIT ACCOUNT CREATION & DELETION

The creation and deletion of an appropriate privileged account to meet mission objectives. The account should have traits to link it back to the requesting identity or service performing the operation for logging and forensics.

### 3 JIT IMPERSONATION

The account is linked to a preexisting administrative account(s) and, when a specific application or task is performed, the function is elevated using those credentials. This is commonly done using automation or scripting with Windows "RunAs" or *Nix SuDo. Typically, the end user is unaware of the impersonation account for this type of operation, and the process may overlap with always-on privileged account delegation.

### 4 JIT TOKENIZATION

The application or resource has its privileged token modified before injection into the operating system kernel. This form of least privilege is commonly used on endpoints to elevate the privileges and priority of an application, without elevating privileges for the end user.

### 5 JIT GROUP MEMBERSHIP

The automatic addition and removal of an account into a privileged administrative group for the duration of the mission. The account should only be added to an elevated group when the appropriate criteria are met. Group membership should be revoked immediately upon completion of the mission.

### 6 JIT-DISABLED ADMINISTRATION ACCOUNTS

Disabled administrator accounts are present in a system with all the permissions, privileges, and entitlements to perform a function. They are enabled to perform a specific mission and then subsequently disabled again once operational criteria have been satisfied. This concept is no different than having always-on administrative accounts, with the exception that native enablement functionality is leveraged to control JIT access.

## JIT Trigggers

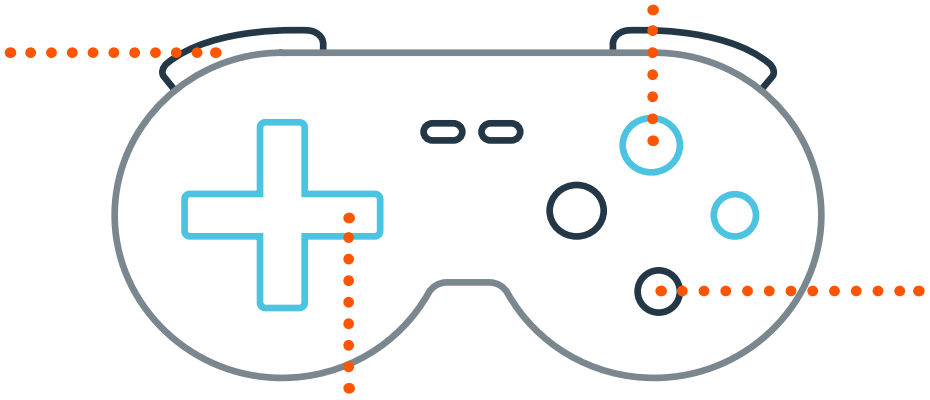
For any of these privileged account elevation methods to work according to the principles of just-in-time privileged access management, the following criteria should be considered as *triggers*. These should also include variables such as time and date for change control windows, as well as suspension or termination criteria if indicators of compromise are detected.

### TWO-FACTOR (2FA) OR MULTI-FACTOR AUTHENTICATION (MFA)

A common method for authorizing privileged access to always-on or JIT privileged accounts is 2FA or MFA. While this does not distinguish between the two access techniques, it does provide additional risk mitigation by validating that the identity has proper access to a privileged account. These authentication methods can, however, be used as a JIT trigger for an account using any of the techniques listed above.

### WORKFLOW

The concept of workflow approval is commonly associated with call centers, help desks, and other information technology service management (ITSM) solutions. A request is made for access and, using a defined workflow of approvers, access is either granted or denied. Once the workflow satisfies an approval, a JIT account can be enabled. This typically corresponds to the user, asset, application, time/date, and associated ticket in a change control or help desk solution. Privileged session monitoring is typically enabled by PAM solutions in this scenario to verify that all corresponding actions were appropriate.



### CONTEXT-AWARE

Context-aware access is based on criteria like source IP address, geolocation, group membership, host operating system, applications installed or operating in memory, documented vulnerabilities, etc. Based on any logical combination of these traits, JIT account access can be granted or revoked in order to satisfy business requirements and mitigate risk.

### ENTITLEMENTS

When privileged access management is integrated with identity access management (IAM) solutions, entitlements between solutions can be synchronized for privileged access. To that end, JIT access can be assigned directly via PAM solutions, or alternatively, programmatically through IAM entitlements. While a typical IAM entitlement workflow is a longer process for synchronization and may not be real time, it does provide a vehicle for account certifications based on privileges. This may be void when linking IAM with PAM solutions to control access.

**JIT automation triggers are conditions for an account to be placed in a state for privileged access.**

## JIT Policies

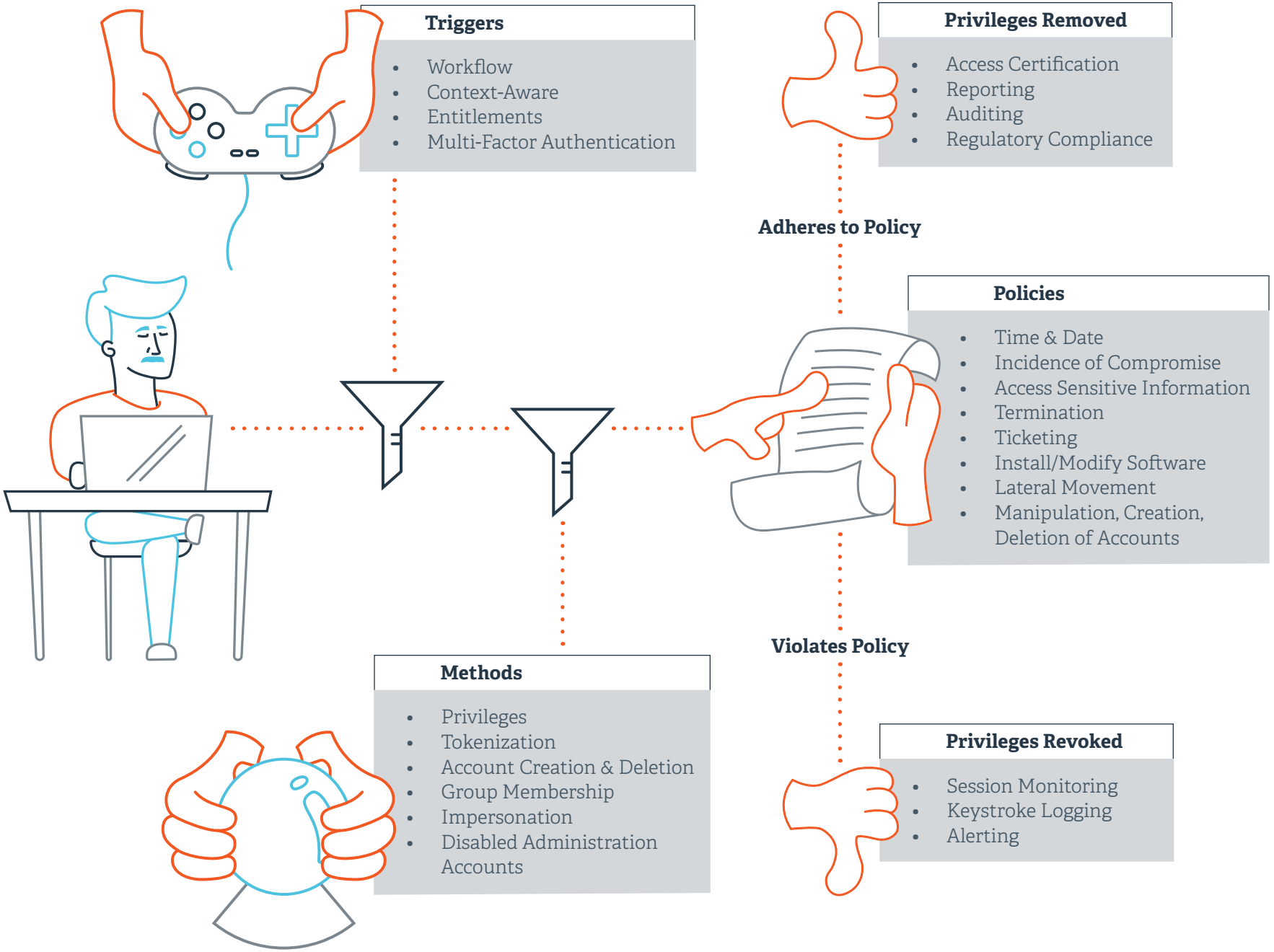
The two key questions for teams to consider are *"What policies govern a JIT account for proper privileged access,"* and *"What conditions should be met for its revocation?"*

These policies could include:

1. **Time and date windows** for access and change control
2. **Commands or applications** that may be an indicator of compromise
3. **Detection of access** to sensitive information
4. **Termination** of the primary session
5. **Existence of corresponding collateral** in a ticketing solution
6. **Inappropriate modification** of resources, including installing software or modifying files
7. **Inappropriate attempts** at lateral movement
8. **Manipulation, creation, or deletion** of user accounts or data sets

While this is by no means an exhaustive list of all policy variables, it can help filter the criteria for a JIT account to be made available or terminated based on corresponding triggers.

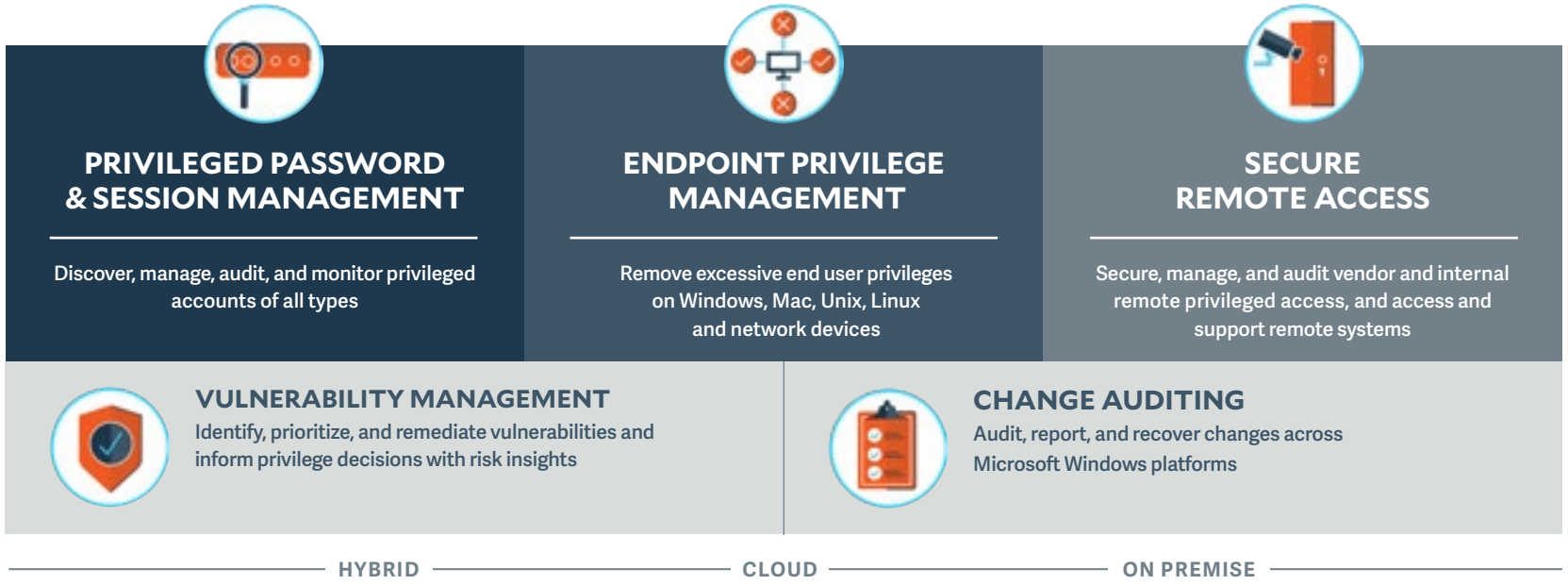
## JIT PAM in Action



## How BeyondTrust Solutions Enable Just-In-Time Privileged Access Management

The BeyondTrust Privileged Access Management Platform can make the JIT PAM security model a reality for your organization, while also helping you enhance security by enforcing other critical aspects of your IAM/PAM strategy, such as privileged account discovery and management, least privilege, credential management, privilege separation and separation of duties, privilege auditing, and privileged threat analytics.

BeyondTrust supports the just-in-time model by enabling the centralized management of privileged credentials and sessions, privilege elevation/delegation across endpoints (desktops, servers, and more), and remote access (employees, vendors, etc.) with a wide variety of triggers. The BeyondTrust PAM Platform includes multiple, integrated solutions.




**BeyondTrust supports the just-in-time model by enabling the centralized management of privileged credentials and sessions, privilege elevation/delegation across endpoints, and remote access with a wide variety of triggers.**

| BeyondTrust Solutions | Features & Capabilities Supporting JIT PAM | | | | | |
|---|---|---|---|---|---|---|
| **Privileged Password & Session Management** | **Continuous Automated Account Discovery & Auto-Onboarding** | **Secure SSH Key Management** | **Application-to-Application Password Management** | **Enhanced Privileged Session Management** | **Adaptive Access Control** | **Advanced Privileged Threat Analytics** |
| Discover, manage, audit, and monitor privileged accounts of all types | Leverage a distributed network discovery engine to scan, identify, and profile all assets. Dynamic categorization allows auto-onboarding into Smart Groups for efficient management and JIT access when new accounts are detected. | Automatically rotate SSH keys according to a defined schedule and enforce granular access control and workflows. Leverage private keys to securely log users onto Unix/Linux systems through the proxy, with no user exposure to the key, and with full privileged session recording. | Eliminate hard-coded or embedded application credentials through an adaptable API interface that includes an unlimited number of password caches for scalability and redundancy. This allows JIT access to the latest passwords for any application. | Live session management enables true dual control, enabling admins to record, lock, and document suspicious behavior—without killing sessions or productivity—based on any JIT activity. | Evaluate just-in-time context and simplify access requests by considering the day, date, time, and location when a user accesses resources to determine their authorization to access those systems. | Measure asset characteristics and user behaviors from one day to the next, assessing the scope and speed of any changes to alert you to suspicious deviations. |
| **Endpoint Privilege Management** | **Least Privilege Enforcement** | **Seamless Application Control** | **Complete Auditing & Reporting** | **Privileged Threat Analytics** | **Security Ecosystem Integrations** | |
| Enforce least privilege and remove excessive end user privileges on Windows, Mac, Unix, Linux and network devices | Elevate privileges to applications for standard users on any operating system through fine-grained, policy-based controls, providing just enough access to complete a task just-in-time. | Deliver trust-based application whitelisting, blacklisting, and greylisting with a flexible policy engine to set broad rules. Choose automatic approval for advanced users – protected by full audit trails – or utilize challenge-response codes for just- in-time application control. | Provide a single, unimpeachable audit trail of all user activity that speeds forensics and simplifies compliance. | Correlate user behavior against asset vulnerability data and security intelligence from best-of-breed security solutions to provide an overall picture of end-user risk. | Built-in connectors to a host of third-party solutions, including help desk applications, vulnerability management scanners, and SIEM tools, ensure that organizations realize a return on their security investments. | |

| BeyondTrust Solutions | | Features & Capabilities Supporting JIT PAM | | | | | |
|---|---|---|---|---|---|---|---|
| **Secure Remote Access** | **Remote Vendor & Third-Party Access** | **Privileged Access Control** | **Monitor Sessions** | **Reduce the Attack Surface** | **Integrate with Password Management** | **Mobile & Web Consoles** | **Audit & Compliance** |
| Secure, manage, and audit vendor and internal remote privileged access, and access to support remote systems | | Enforce least privilege by giving users the right level of access for any remote session. | Control and monitor sessions using standard protocols for RDP, VNC, HTTP/S, and SSH connections. | Reduce the threat surface by consolidating the tracking, approval, and auditing of privileged accounts just in time, in one place, and by creating a single access pathway. | Inject credentials directly into servers and systems with just one click, just in time, so users never need to know or see plain text credentials. | Use mobile apps or web-based consoles anytime, anywhere, and just in time to perform remote access tasks. | Create audit trails, session forensics, and other reporting features by capturing detailed session data in real-time or post-session review and provide attestation reports to prove compliance. |
| | **Remote Support & HelpDesk Personnel** | **Chat Support** | **Broad Platform Support** | **Granular Permissions & Roles** | **Collaboration** | **Session Recording & Audit Trail** | **Support from Chrome, Firefox, IE, & more** |
| | | Enable live support from your website with Click-to-Chat with just-in-time escalation to screen sharing and remote control, without ever losing contact with the end user. | Support and provide support from Windows, Mac, Linux, iOS, and Android devices. Also support legacy devices using RDP, Telnet, SSH, and VNC. | Granularly manage teams, users, roles, and session permission settings to enforce a least-privilege security posture. | Resolve support incidents faster and defining escalation paths to skilled resources, while improving customer satisfaction by including the appropriate team members, just in time. | Track team performance, as well as log session activity, to serve as an audit trail for security, compliance, and training. | Our HTML5 Web Rep Console lets you offer secure remote support just-in-time from any browser – no downloads required – to immediately begin fixing issues. |

## Mapping JIT Methods & Triggers

The following matrix maps JIT PAM Triggers and Methods to each BeyondTrust solution.

| | Privileged Password & Session Management | Endpoint Privileged Management | Secure Remote Access |
|---|---|---|---|
| **Triggers** | | | |
| Entitlements | ● | | ● |
| Workflow | ● | ● | ● |
| Context Aware | ● | ● | ● |
| Multi-Factor | ● | ● | ● |
| **Methods** | | | |
| Account Creation & Deletion | ● | | |
| Group Membership Privilege | ● | | |
| Impersonation | ● | ● | ● |
| Disabled Administration Accounts | | ● | |
| Tokenization | | ● | |

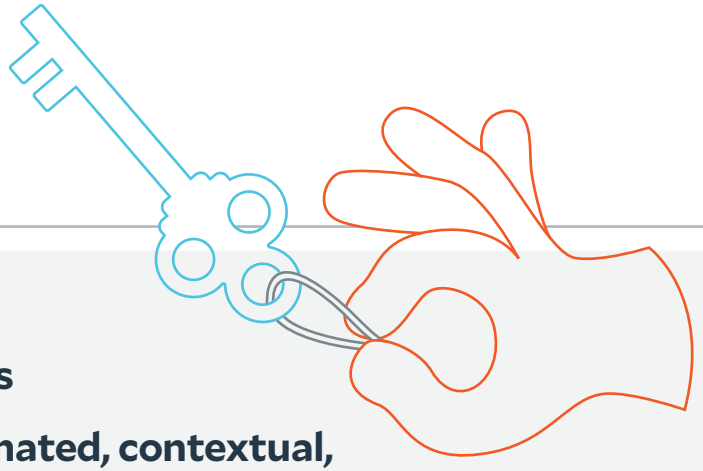
## Moving Your JIT PAM Implementation Forward to Reduce Cyber Risk

For many organizations, implementing a JIT strategy in synchrony with a just enough access model is the next, most impactful step they can take toward protecting their valuable IT estate.

JIT privilege management should be considered an essential component of a true least-privilege strategy. In lieu of enabling accounts all the time once authenticated, exert further control over when and how they can be used by expanding the security model to deny all privileged activity until the appropriate business criteria is satisfied for their usage. This entails not only restricting account access, but the actual privileges, permissions, and entitlements that an account can use in real-time.

By enabling privileged access management just in time using to contextual triggers, and ensuring the user behavior of the privileged account is appropriate based on real-time policies, JIT PAM dynamically addresses the substantial, enterprise-wide risks posed by always-on accounts. This represents not just the natural evolution of privileged access management, but a considerable leap forward in IT risk management.

BeyondTrust Privileged Access Management enables organizations to address the challenges and risks of always-on privileged access sprawled across increasingly complex and heterogeneous IT environments, while keeping your end-users productive and secure.



### JIT PAM Key Benefits

1. **Centralized, automated, contextual, and time-based provisioning/de-provisioning of privileges massively reduces the window of vulnerability during which privileges may potentially be exploited**

2. **Enforcement of true least-privilege equates to fewer privileged users and privileged sessions, which in addition to improving security, simplifies auditing and compliance initiatives**

3. **Invisible and frictionless experience for the end user, enabling productivity without disrupting workflows**

4. **The elimination of always-on privileged accounts and the attack vectors associated with them**

## Related Concepts & Terminology

**Break-glass:** In the context of computing, break-glass refers to checking out a system account password to bypass normal access controls procedures for a critical emergency, generally when other access methods have failed or are inaccessible. Break-glass provides the user immediate, but typically, time-limited access to an account that they may not normally be authorized to access.

**Just-In-Time (JIT) Privileged Access Management:** The goal of JIT privilege management is to assign the necessary privileges "on the fly" based on an approved task or mission, and subsequently remove them once the task is complete or once the window or context for authorized access has expired. JIT privilege management enables organizations to secure privileged accounts from continuous, always-on access by enforcing restrictions based on behavioral and contextual parameters.

**Least Privilege:** Least privilege refers to the concept and practice of restricting access rights for users, accounts, and computing processes to only those resources absolutely required to perform routine, authorized activities. A least privilege security model entails enforcing the minimal level of user rights, or lowest clearance level, that allows the user to perform his/her role. Least privilege also applies to processes, applications, systems, and devices (such as IoT), in that each should have only those permissions required to perform an authorized activity.

**Privilege:** Privilege provides the authorization to override, or bypass, certain security restraints, and may include permissions to perform such actions as shutting down systems, loading device drivers, configuring networks or systems, provisioning and configuring accounts and cloud instances.

**Privileged Access Management (PAM):** Alternatively referred to as privileged account management, privileged identity management (PIM) or simply privilege management, PAM refers to solutions and strategies to manage and secure privileged accounts, and control privilege delegation and escalation activities for users, applications, services, processes, tasks, etc. PAM solutions enable organizations to remove admin rights from users (across both servers and desktops), and instead, elevate privileges for authorized applications or tasks as-needed.

**Privileged Account:** A privileged account is considered to be any account that provides access and privileges beyond those of non-privileged accounts (e.g. standard accounts and guest user accounts). A privileged user is any user currently leveraging privileged access, such as through a privileged account. Because of their elevated capabilities and access, privileged users/privileged accounts pose considerably larger risks than non-privileged accounts/ non-privileged users.

**Privileged Session:** A privileged session is a computing session that involves the execution of activities requiring privileges that are typically beyond those of a standard user. A privileged session could be initiated by a user, system, application, or service.

**Privileged Session Management (PSM):** Privileged session management (PSM) entails the monitoring and management of all sessions for users, systems, applications, and services that involve elevated access and permissions. PSM allows for advanced oversight and control that can be used to better protect the environment against insider threats or potential external attacks, while also maintaining critical forensic information that is increasingly required for regulatory and compliance mandates.

**Standard User Accounts:** Standard user accounts, sometimes called least-privileged user accounts (LUA) or non-privileged accounts, have a limited set of privileges. In a least-privilege environment, these are the type of accounts that most users should be operating in 90 – 100% of the time. A standard user is a non-privileged user in computing environments (Windows, Mac, Linux, Unix, etc.) with basic access rights. This type of account/user, has limited ability to access resources and settings, as opposed to a privileged or superuser account (such as root or admin), which may have vast administrative rights and privileged access.

**Superuser Accounts:** Superuser accounts are highly privileged accounts primarily used for administration by specialized IT employees and provide virtually unrestrained power to execute commands and make system changes. Superuser accounts are typically known as "Root" in Unix/Linux and "Administrator" in Windows systems. Superuser account privileges can provide unrestricted access to files, directories, and resources with full read / write / execute privileges. Superuser accounts can also render systemic changes across a network, such as creating or installing files or software, modifying files and settings, and deleting users and data. Superusers may also provision and de-provision access and permissions for other users.

## About BeyondTrust

BeyondTrust is the worldwide leader in Privileged Access Management, offering the most seamless approach to preventing data breaches related to stolen credentials, misused privileges, and compromised remote access. Our extensible platform empowers organizations to easily scale privilege security as threats evolve across endpoint, server, cloud, DevOps, and network device environments. BeyondTrust gives organizations the visibility and control they need to reduce risk, achieve compliance objectives, and boost operational performance. We are trusted by 20,000 customers, including half of the Fortune 100, and a global partner network.

**beyondtrust.com**

V2019_07_ENG