Privileged Access Management Policy Template

|  |  |  |
| --- | --- | --- |
| Policy # ############ | Effective DateDD/MM/YYYY | Emailpolicycontact@company.com |
| Version1.0 | ContactPolicy Contact | Phone888.641.5000 |

**ABOUT THE POLICY TEMPLATE**

Privileged accounts present a much greater risk than typical user accounts and thus require a higher level of security controls. To lower risk of privileged account misuse, security policies define how credentials (passwords, keys, and secrets) connected to these accounts should be managed.

You can use this sample policy as a starting point to build a privileged access management (PAM) policy for your organization. The template is divided into several sections according to common governance areas regarding privileged accounts. It contains over 40 pre-written information security policy statements you can select from to match your organization’s governance requirements.

###

[Customizing the Template 3](#_Toc19532534)

[1.0 Purpose 4](#_Toc19532535)

[2.0 Scope 4](#_Toc19532536)

[3.0 Policy 4](#_Toc19532537)

[3.1 System Approval and Authorization 4](#_Toc19532538)

[3.2 Password Categorization 5](#_Toc19532539)

[3.3 Password Composition 5](#_Toc19532540)

[3.4 Password History and Change Interval 7](#_Toc19532541)

[3.5 Account Lockout and Compromised Passwords 7](#_Toc19532542)

[3.6 Acceptable Use of Privileged Accounts 8](#_Toc19532543)

[3.7 Privileged Account Approval 9](#_Toc19532544)

[3.8 Privileged Account Construction 9](#_Toc19532545)

[3.9 Privileged Access Management 10](#_Toc19532546)

[3.10 Third-Party Privileged Accounts 11](#_Toc19532547)

[3.11 Local Administrative Privileges 11](#_Toc19532548)

[3.12 Application Development 12](#_Toc19532549)

[3.13 Privileged Account Logging 13](#_Toc19532550)

[3.14 Application Control 13](#_Toc19532551)

[4.0 Violations 14](#_Toc19532552)

[5.0 Definitions 14](#_Toc19532553)

[6.0 References 15](#_Toc19532554)

[7.0 Approval and Ownership 18](#_Toc19532555)

[8.0 Revision History 18](#_Toc19532556)

[9.0 Appendix – Policy Enforcement 19](#_Toc19532557)

[Secret Server 19](#_Toc19532558)

[Privilege Manager 19](#_Toc19532559)

[Privileged Behavior Analytics 19](#_Toc19532560)

[Account Lifecycle Manager 19](#_Toc19532561)

[DevOps Secrets Vault 19](#_Toc19532562)

[About Delinea 27](#_Toc19532563)

### Customizing the Template

To customize this template, perform the following steps:

1. Remove the “About the Policy Template” and “Customizing the Template” instructions, and “Appendix” and other author comments
2. Replace the term “Company X” with the name of your organization
3. Add your company logo in the upper left corner
4. Update all of the company-specific contact information (highlighted in yellow)
5. Update the effective date
6. Revise policy guidelines to meet your organization’s policies
7. Revise the Violations section to meet your organization’s policies
8. Save your changes
9. Obtain your management and auditors’ approval of the completed policy
10. Distribute the policy according to your management guidance

This document is a template only and should be revised to meet the information security guidelines of your organization. Organizations should not adopt any security policy without proper review and approval by senior management, information security, and legal.

1.0 Purpose

This policy defines the requirements for establishing and maintaining account settings for all privileged accounts on any Company X computer and communications system.

2.0 Scope

This policy applies to all staff and contractors responsible for setting up and maintaining privileged accounts related to Company X electronic information resources. Resources include user workstations as well as servers, databases, applications, and systems managed both on-premise and in the cloud.

3.0 Policy

3.1 System Approval and Authorization

(Default accounts and passwords present one of the greatest risks to IT systems. This section defines specific controls for controlling and managing privileged accounts when systems are placed into production. These controls may also be part of a policy regarding System Configuration Management.)

3.1.1 Default Password Changes

All vendor-supplied default passwords must be changed before any computer or communications system is used for Company X business.

3.1.2 Privileged User ID Review

Before any production multi-user computer operating system is installed at Company X, all privileged user IDs that are not assigned to a specific employee or job role must have their passwords changed to large random values. These should be recorded in the privileged access management system with appropriate permissions for the administrators responsible for managing these accounts.

3.1.3 Unnecessary Software

Software features that could be used to compromise security, and that are clearly unnecessary in the Company X computing environment, must be disabled at the time when software is installed on multi-user systems.

3.2 Password Categorization

(This section defines specific terminology for understanding different categories of passwords, which is helpful for prescribing controls on those passwords. Treating all passwords the same is not effective.)

Passwords fall into two categories:

3.2.1 User Account Passwords

A password is a “secret” that allows the use of an account. A user account is typically tied to a unique individual, for example, an Active Directory user account. Therefore, that password determines a human identity and the password is the secret known by the human that connects that human to the system. A goal is to strive for as few user account passwords per human user as possible; ideally, a single user account password should be maintained per human user.

3.2.2 Privileged Account Passwords

Privileged account passwords provide administrative or specialized levels of access to enterprise systems and sensitive data, based on higher levels of permissions. A privileged account can be associated with a human being or non-human IT system, such as:

* **Service accounts**, which run application services such as Windows Services, scheduled tasks, batch jobs, and Application Pools within IIS
* A**pplication accounts**, which include database logins, certificates for software signing, embedded build script passwords, configuration files, and application services used during software development
* **System administrator accounts used to** manage databases
* **Domain administrator accounts, used to** manage servers and control Active Directory users, as well as local domain accounts at the workstation level
* **Root accounts used** to manage Unix/Linux platforms

3.3 Password Composition

(This section defines specific controls for creating secure passwords for privileged accounts. Passwords for privileged accounts must generally be more secure than for regular user accounts.)

3.3.1 Role-Based Password Length

The minimum length for fixed passwords, or passwords created by users, must be set to six for handheld computers, eight for all network-connected computers, and ten for administrator and other privileged user IDs.

3.3.2 User Account Password Complexity

All user-chosen passwords for user accounts must meet the following complexity requirements:

* Must contain at least one alphabetic, one numeric and one symbol character.
* Must be at least 8 characters in length.
* Ideally passphrases should be used to increase length. Increased length provides more security than complexity and is easier for a human to memorize.

For example:
1) lf@j7asFd! versus 2) Blue5Chandelier2@
The seven extra characters in (2) make it 64 trillion times stronger than (1).

3.3.3 Privileged Account Password Complexity

These passwords should be optimized for the maximum lengths of the platform. Random passwords should be generated between 80 and 127 characters in length to provide the maximum security.

The following requirements should be followed for privileged account passwords:

* Maximize the possible length of password for each platform.
* Passphrases *should not be used* to avoid memorization.
* Should have a complete mix of upper case, lower case, numbers, and symbols.

3.3.4 Seed for Generated Passwords for Privileged Accounts

If system-generated passwords are used, they must be generated using the low order bits of system clock time or some other very frequently changing and unpredictable source.

3.3.5 Null Passwords Always Prohibited

At no time, may any Systems Administrator or Security Administrator enable any user ID that permits password length to be zero (a null or blank password).

3.3.6 Enforce Password Complexity

All passwords must meet the above complexity requirements and this complexity must always be checked automatically at the time that the password is created or changed.

3.4 Password History and Change Interval

(This section defines specific controls for changing passwords for privileged accounts. Passwords for privileged accounts must generally be more secure than passwords for regular user accounts.)

3.4.1 User Account Password Changes

User passwords must be changed at least once every 90 days. Users may be required to change passwords or this may be done automatically.

3.4.2 User Account Maximum Password Changes

Users must not be permitted to change their password within seven days of their previous change. This requirement is only helpful for passwords that users are memorizing (user accounts) and is used to prevent users from changing the password multiple times back to a previously used password (therefore defeating the requirement to change the password).

3.4.3 Privileged Account Password Changes

All privileged accounts must be automatically required to change their passwords at least once every 90 days. This time interval should be set based on an internal risk assessment for any potential disruption to the business. A domain admin account password change would have zero disruption to the business and is very high risk. These accounts should have their passwords changed as often as possible – ideally after every use to reduce exposure to abuse, misuse, or exploits such as Pass-the-Hash attacks.

3.4.4 Password History

On all multi-user Company X computers, system software or security software must be used to maintain an encrypted history of previously chosen fixed passwords. This history must contain at least the previous thirteen passwords for each user ID.

3.5 Account Lockout and Compromised Passwords

(Privileged accounts must be protected against brute-force password guessing techniques just as user accounts are protected. The specific parameters of password attempts and lockout directions should be customized based on the organization’s specific requirements.)

3.5.1 Maximum Login Attempts

All Company X computer systems that employ fixed passwords at log on must be configured to permit only five attempts to enter a correct password, after which the user ID is deactivated.

3.5.2 Lockout Duration

All accounts that have been disabled for incorrect logon attempts must remain inactive for at least 15 minutes.

3.5.3 Lockout Notification

The security team must be notified of all disabling of accounts for incorrect logon attempts so that investigation can occur if necessary and anomalies can be detected.

3.5.4 Password Changes After Privileged User Credential Compromise

If a privileged user credential has been compromised by an intruder or another type of unauthorized user, all passwords on that system and any related systems must be immediately changed.

3.5.5 Fixed Password Change Confirmation

System administrators must be immediately notified when fixed passwords are changed or updated outside of the central privileged access management system.

3.6 Acceptable Use of Privileged Accounts

(Sharing passwords between systems creates great risk. A single compromised system can allow an attacker to quickly move laterally across the network. This section contains controls to limit password sharing across systems.)

3.6.1 User Account Password Sharing

User account passwords must never be shared or revealed to anyone other than the authorized user. If they are shared, then they are no longer considered a user account since the identity of the user is not known.

3.6.2 Privileged Account Password Sharing

Privileged account passwords should not be shared and each privileged account must have a unique password. Passwords for privileged accounts can be shared among administrators only if controls are in place to know which administrator is using the account at any one time. This must include full auditing and non-repudiation mechanisms.

3.6.3 Password Display and Printing

The display and printing of account passwords must be masked, suppressed, or otherwise obscured so that unauthorized parties will not be able to observe or subsequently recover them. Any display of a privileged account password to a user must be audited and the password should be changed after it has been used.

3.7 Privileged Account Approval

(Because of the added risk of compromise, privileged accounts must be strictly controlled. The following sections contain controls for the approval, creation, and maintenance of accounts.)

3.7.1 Privileged Account Requirements

All privileged accounts on Company X systems must employ greater security than non-privileged accounts. This includes longer, more secure passwords and greater audit accountability.

3.7.2 Privileged User Account Approval

The creation or modification of privileged user accounts must be approved by at least two individuals: the system owner and an authorized member of the Information Technology department. System administrators must not be allowed to create other privileged accounts without authorization.

3.7.3 Number of Privileged User IDs

The number of privileged user IDs must be strictly limited to those individuals who absolutely must have such privileges for authorized business purposes.

3.7.4 Role-Based Account Privileges

To facilitate secure management of systems, wherever possible, privileged accounts must be defined based on the specific role of the system administrator.

3.8 Privileged Account Construction

(One way to reduce confusion and gain oversight is to adopt standards for privileged accounts that are created by the organization. For proper separation of duties, administrators must use separate accounts for their day-to-day user activities.)

3.8.1 Privileged User ID Construction

All privileged user IDs on Company X computers and networks must be constructed according to the Company X user ID construction standard, and must conform to one of the following:

* Must clearly indicate the responsible individual’s name
* Must clearly define the account (i.e. purpose of the account, type of account, etc.)
* Must be managed in a system that can clearly associate a single user account to each use of the privileged account in order to document accountability for the use of the privileged ID

3.8.2 Service Account Governance

User IDs for service accounts and other application accounts should also follow the Company X naming convention and requirements outlined in section 3.8.1 above.

3.8.3 Generic User IDs

User IDs must uniquely identify specific individuals. Generic user IDs based on job function, organizational title or role, descriptive of a project, or anonymous must be avoided wherever possible.

3.8.4 Re-Use of User IDs

Each Company X computer and communication system user ID must be unique, connected solely with the user to whom it was assigned, and must not be reassigned after a worker, contractor, or customer terminates their relationship with Company X.

3.8.5 Separate System Administrator User IDs

System administrators managing computer systems with more than one user must have at least two user IDs, one that provides privileged access and is logged, and the other that provides the privileges of a normal user for day-to-day work.

3.9 Privileged Access Management

(The variety of privileged account types across various systems presents unique management challenges. Modern software applications provide ways to automate privileged account discovery, creation, management, and removal. These systems should be used whenever possible to reduce risk and increase visibility.)

3.9.1 Central Automated Management

All privileged accounts on Company X systems must be managed by a central system. This system must provide an audit trail that tracks specific additions, changes, and deletions.

3.9.2 Integration with Native Directories

Any privileged access management system must integrate with native operating system account management systems or directory services, such as Active Directory.

3.9.3 Integration with Strong Authentication Methods

Any privileged access management system must integrate with strong authentication methods, such as multi-factor authentication, to ensure the identity of the user in addition to their directory authentication.

3.9.4 Password Vault

Company X system administrators must have access to a vault system that enables the temporary provisioning of access to privileged accounts and passwords (aka FireID) for emergency maintenance.

3.9.5 Password Vault Encryption

Company X must maintain any credentials stored in a central management system within an encrypted password vault, using strong encryption algorithms that meet compliance and/or regulatory requirements.

3.9.6 Privileged Account Inventory

Company X must maintain an inventory of all accounts with privileged access on production information systems.

3.9.7 Account Inventory Update

The privileged account inventory must be updated at least quarterly to identify new or changed accounts.

3.9.8 Inactive Account Maintenance

All accounts must be created with an expiration date. All inactive accounts over 90 days old must be either removed or disabled.

3.9.9 Disaster Recovery

Any privileged access management system must be configured to utilize robust backup, recovery, and availability methodologies to ensure resiliency and availability of the credentials stored within the system as well as the timely recovery of the system in the event of a system failure.

3.10 Third-Party Privileged Accounts

3.10.1 Third-Party User ID Expiration

Every privileged user ID established for a non-employee or third-party application must have a specified expiration date with a default expiration of 30 days when the actual expiration date is unknown.

3.11 Local Administrative Privileges

(Without a least privilege policy, users typically retain admin access over local endpoints and that makes them an easy target. When a user is logged in with privileged credentials and unwittingly downloads malicious code from an email or website, that malware gains unlimited access to the computer and possibly multiple systems on your network.)

3.11.1 Employee Workstations

Users should not retain local administrator accounts on their individual workstations. In accordance with a least privilege policy, the default local administrator account should be removed prior to providing users with workstations.

3.12 Application Development

(Most applications that require privileged accounts to operate present unique challenges. In many cases these accounts are not disclosed or use default accounts with fixed, hard-coded credentials. Privileged security can be greatly enhanced using secure application design and coding principles.)

3.12.1 Special Application Accounts

All development applications and systems that require privileged access, including DevOps tools, containers, and microservices, must use secure privileged accounts.

3.12.2 Secret IDs or Passwords

Developers must not build or deploy secret user IDs or passwords that have special privileges that are not clearly described in the generally available system documentation.

3.12.3 Hard-Coded Passwords in Software

Passwords must never be hard coded in software developed by or modified by Company X workers or contractors.

3.12.4 Third-Party Repositories

Credentials used in the application development process must never be stored in remote repositories, such as GitHub.

3.12.5 Test Account Removal

Test data and accounts used during development and testing must be removed before a production system becomes active.

3.13 Privileged Account Logging

(To provide audit visibility into privileged accounts, systems must be designed and configured to log events linked to privileged accounts.)

3.13.1 Privileged System Commands Traceability

All privileged commands issued on computer and communication systems must be traceable to specific individuals through the use of comprehensive logs.

3.13.2 Privileged User ID Activity Logging

All user ID creation, deletion, and privilege change activity performed by Systems Administrators and others with privileged user IDs, including third-party vendors, must be securely logged.

3.13.3 Privileged User ID Activity Log Review

All logs recording privileged ID activity must be reviewed at least quarterly via periodic management reports.

3.13.4 Privileged User ID Activity Log Correlation

All logs recording privileged ID activity must be aggregated into a central management tool for privileged accounts or a Security Information and Event Management (SIEM) tool to correlate privileged ID activity to other security events, log entries, and related non-privileged ID activity.

3.12.5 Privileged User ID Session Logging

In addition to event logging, all activity on privileged accounts must be logged via session or keystroke recording.

3.14 Application Control

(Many business applications require administrative access to install or run key processes. Yet, applications may contain malware with the capacity to corrupt data or disrupt key business processes.)

3.14.1 Whitelisting

Only trusted applications should be allowed to be installed or executed automatically. Those not on Company X’s whitelist will be subject to review and approval.

3.14.2 Blacklisting

Specific applications known or suspected to contain malicious code may be added to a blacklist and not allowed to be installed or executed.

4.0 Violations

Any violation of this policy may result in disciplinary action, up to and including termination of employment. Company X reserves the right to notify the appropriate law enforcement authorities of any unlawful activity and to cooperate in any investigation of such activity. Company X does not consider conduct in violation of this policy to be within an employee’s or partner’s course and scope of employment, or in the direct consequence of the discharge of the employee’s or partner’s duties. Accordingly, to the extent permitted by law, Company X reserves the right not to defend or pay any damages awarded against employees or partners that result from violation of this policy.

5.0 Definitions

**Account (User ID or Username)** – A unique string of characters assigned to a user by which a person is identified to a computer system or network. A user commonly must enter both a user ID and a password as an authentication mechanism during the logon process.

**Fixed Password** – A password created by a user for an account or credential.

**Least Privilege** – Least privilege means that for each task or process, the user or system is granted the minimum rights required to perform the task.

**Password** – An arbitrary string of characters that is used to authenticate an account when attempting to log on to prevent unauthorized access to the account.

**Privileged Account** – An account that can either be a user account on any system that has system privileges beyond those of a normal user or an account that does not represent a human use. Privileged accounts are typically not assigned to a user, but can, in some cases, be dedicated user accounts that are given more permissions than a typical user account. Root, local administrator, domain admin, and enable passwords are all examples of privileged accounts that have elevated access beyond that of a normal user.

**System Administrator –** An employee or partner who is responsible for managing a Company X multi-user computing environment. The responsibilities of the system administrator typically include installing and configuring system hardware and software, establishing and managing user accounts, upgrading software, and managing backup and recovery tasks.

**Third Party –** Any non-employee of Company X who is contractually bound to provide some form of service to Company X.

**User –** Any Company X employee or partner who has been authorized to access any Company X electronic information resource.

**User Account** – An account that represents a single human user who is the only person to ever use the account and is their way of authenticating into Company X systems. The password for this account is something they would memorize and would not be shared with any other user.

6.0 References

(This section contains references to the information security laws and governance frameworks applicable to this Privileged Access Management Policy.)

Policy and Regulation Section Mapping (for reference only)

| Section | Policy Description | PCI DSS  | NIST 800-66  | HIPAA  | ISO 17799/27001 | CIS Controls |
| --- | --- | --- | --- | --- | --- | --- |
| 3.3.2 | User Account Password Complexity | 8.2.3 |  | 164.308 | 11.2.2, 11.3.1, 12.1.1 | 5.7 |
| 3.3.3 | Privileged Account Password Complexity | 8.2 |  |  | 11.3.1 |  |
| 3.4.1 | User Account Password Changes | 8.2.4 |  | 164.308 | 11.2.2, 11.3.1, 12.1.1 |  |
| 3.4.3 | Privileged Account Password Changes | 8.2.4 |  | 164.308 | 11.2.2, 11.3.1, 12.1.1 |  |
| 3.4.4 | Password History | 8.2.5 |  | 164.308 | 11.2.2, 11.3.1, 12.1.1 |  |
| 3.5.1 | Maximum Login Attempts | 8.1.6 |  | 164.308 | 11.2.2, 11.3.1, 12.1.1 | 16.7 |
| 3.5.2 | Lockout Duration | 8.1.7 |  | 164.312 | 11.2.2, 11.3.1, 12.1.1 | 16.7 |
| 3.61.1 | User Account Password Sharing | 10.1 |  |  |  | 5.8, 11.6 |
| 3.7.2 | Privileged Account Approval |  | 4.14.6 | 164.312 | 11.2.4, 12.5.1 | 5.2 |
| 3.7.4 | Role-Based Account Privileges | 7.1.2, 7.1.3 |  |  | 11.2.2 |  |
| 3.8 | Privileged Account Construction |  | 4.14.3 | 164.312 | 11.5.2 | 5.4, |
| 3.8.4 | Separate Systems Administrator User IDs |  |  |  |  |  |
| 3.9.1 | Central Automated Management | 8.5 | 4.14.5 | 164.312 | 11.5.3 | 16.14 |
| 3.9.3 | Integration with Strong Authentication Methods | 8.2.2, 8.3 |  |  | 11.5.1 | 5.6, 16.11 |
| 3.9.5 | Password Vault Encryption | 8.2.1 |  |  | 11.5.3 | 16.3, 16.4 |
| 3.9.6 | Privileged Account Inventory | 2.4 | 4.16.1 | 164.312 | 8.3.3, 11.2.4 | 2.1, 2.3, 5.2 |
| 3.9.8 | Inactive Account Maintenance | 8.1.4 |  |  | 8.3.3 | 16.1, 16.2, 16.3, 16.4, 16.6 |
| 3.11 | Local Administrative Privileges | 7.1, 7.2 |  |  |  | 5.1 |
| 3.12 | Application Development | 2.1, 6.3 |  |  | 12.5 | 5.3, 18.9 |
| 3.13 | Privileged Activity Logging | 8.1.2, 10.2, 10.3, 10.5, 10.6, 10.7 | 4.15.1, 4.14.2, 4.14.4 | 164.312 | 11.5.2 | 5.4, 5.5, 14.6, 6.616.10 |
| 3.14 | Application Control | 5.4 |  | 164.308 | 10.4 | 2.2 |

7.0 Approval and Ownership

|  |  |  |  |
| --- | --- | --- | --- |
| Owner | Title | Date | Signature |
| Policy Contact | Title | MM/DD/YYY |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Approved By | Title | Date | Signature |
| Executive Sponsor | Title | MM/DD/YYYY |  |

8.0 Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version | Description | Revision Date | Review Date | Reviewer/ApproverName |
| 1.0 | Initial Version | 10/25/19 |  |  |
|  |  |  |  |  |

9.0 Appendix – Policy Enforcement

Having a written Privileged Access Management policy is a great first step, but now you need to enforce it —making sure privileged accounts are created, vaulted, encrypted, changed, monitored, and governed according to your policy.

Delinea makes enterprise-grade privilege management accessible for everyone by eliminating the need for complex security tools and prioritizing productivity, flexibility, and control. With Delinea you can adopt a multi-layered approach that covers your privilege security from endpoints to the cloud, ensuring protection at every point of the privileged attack surface. Our suite of privileged access management solutions includes:

**Secret Server**

Set strict, granular permissions for each user who has access to systems that store information. Securely create, manage, and store credentials used by users and systems to access and share data.

**Privilege Manager**

Implement and maintain least privilege best practices to segment and protect data. Policy-based application controls allow users to do their jobs without requiring unneeded access or local administrative rights.

**Privileged Behavior Analytics**

Analyze use and distribution of privileged accounts and access across your organization. Detect anomalous behavior to prevent breaches before they cause damage.

**Account Lifecycle Manager**

Manage non-human privileged accounts that access services, applications, data, and network resources. By discovering hidden accounts and understanding their purpose, you can turn them off when they outlive their use without risk of service interruption.

**DevOps Secrets Vault**

Centrally manage passwords used to access applications and services without causing friction in the development process.

**Connection Manager**

Manage and interact with multiple remote sessions for both RDP and SSH in a unified environment.

| Guideline # | Policy Guideline | Delinea Enforcement of this Guideline |
| --- | --- | --- |
| 3.1.1 | Before any computer or communications system is used, all vendor-supplied default passwords must be changed. | Secret Server can be configured through its Discovery capabilities to find new systems on the network and automatically secure their passwords and store them in the vault with access for the appropriate admins. |
| 3.1.2 | Before any production multi-user computer operating system is installed, all privileged user IDs that are not assigned to a specific employee or job role must have their passwords changed to large random values and should be recorded in the privileged access management system with appropriate permissions for the administrators responsible for managing these accounts.  | Secret Server offers Discovery capabilities with configurable rules for how to find systems and accounts, how to securely randomize their passwords, and whom to grant access to the secure credentials in the Secret Server vault. The entire process is customizable and automated. |
| 3.1.2 | Software features that could be used to compromise security and that are clearly unnecessary in the computing environment must be disabled at the time when software is installed on multi-user systems. | Privilege Manager includes Discovery capabilities to identify software application usage with admin rights on servers and workstations. Real-time threat intelligence identifies high-risk and unknown software applications so that they can be reviewed before they are installed and blacklisted or removed if necessary.  |
| 3.2 | Privileged account passwords that must be protected fall into two categories:User Account PasswordsNon-human Account Passwords  | Secret Server discovers, stores and manages passwords and credentials for all types of privileged accounts, including user accounts with elevated or administrative privileges and non-human accounts such as service, application, root, and other system accounts. |
| 3.3.1 | The minimum length for passwords must be set to six for handheld computers, eight for all network-connected computers, and ten for administrator and other privileged user IDs. | Secret Server sets rules for password length and complexity that are enforced automatically.  |
| 3.3.3 | The following requirements should be followed for privileged account passwords:Should maximize the possible length of password for each platform.Should not be memorized.Passphrases should not be used since memorization is not desirable.Should have a complete mix of upper case, lower case, numbers, and symbols | Secret Server has policies, templates, and password requirements to maximize the length and complexity of privileged account passwords. Secret Server can easily generate, enforce, and rotate passwords that are 100 random characters or more for these accounts. There is no need for users to memorize passwords. |
| 3.3.4 | If system-generated passwords are used, they must be generated using the low order bits of system clock time or some other very-frequently-changing and unpredictable source. | Secret Server uses secure random numbers to generate large random passwords. The libraries used are approved by Microsoft to produce cryptographically secure random numbers. |
| 3.3.5 | Null passwords are always prohibited. | Secret Server can find weak passwords using the Discovery capability. These passwords can then be automatically changed by Secret Server to ensure policy requirements are met. |
| 3.3.6 | All passwords must meet the above complexity requirements and this complexity must always be checked automatically at the time that the password is created or changed. | Secret Server enforces requirements on new passwords as they are created and on existing accounts. Customizable alerts and reporting ensure that all passwords meet requirements and can be proven to an auditor. |
| 3.4.3 | All privileged accounts must be automatically required to change their passwords at least once every 90 days.  | Secret Server offers an expiration policy capability to ensures that all privileged account passwords are changed on a configurable schedule. Calendar specifics can also be set – for example: only change this password on Sundays at 2 am once the 90-day threshold has been met. |
| 3.4.4 | On all multi-user computers, system software or security software must be used to maintain an encrypted history of previously chosen fixed passwords. This history must contain at least the previous 13 passwords for each user ID. | Secret Server automatically tracks history of all passwords, keeping a full log of previous passwords. This is helpful when restoring a system from backup and an earlier password is needed. |
| 3.5.4 | If a privileged user ID has been compromised by an intruder or another type of unauthorized user, all passwords on that system and any related systems must be immediately changed. | Secret Server provides mechanisms to easily identify vulnerable passwords and automatically change thousands of passwords in minutes.Privileged Behavior Analytics identifies anomalous privileged account activity, enabling swift incident response, including immediate password changes.  |
| 3.5.5 | System administrators must be immediately notified when fixed passwords are changed or updated outside of the central privileged account management system. | Secret Server provides the Heartbeat capability to constantly test and validate passwords kept in the vault. If an administrator or an intruder changes a password outside of the vault or creates a backdoor account on a system, then alerts can be generated to the administrator team or Security Office. |
| 3.6.2 | Passwords for privileged accounts can be shared among administrators as long as controls are in place to know which administrator is using the account at any one time. This must include full auditing and non-repudiation mechanisms. | Secret Server provides full auditing on all use of passwords from the vault. If a password is given to an administrator, the CheckOut capability can be used to ensure that the password is changed when the admin is finished using it, increasing accountability on the usage of privileged accounts. |
| 3.6.3 | The display and printing of account passwords must be masked, suppressed, or otherwise obscured so that unauthorized parties will not be able to observe or subsequently recover them. Any display of a privileged account password to a user must be audited and the password should be changed after it has been used. | Secret Server provides controls to allow the authorized use of a password without revealing the password itself. This protects the password from disclosure while still auditing its use. If the password has to be provided to the administrator (for example, for legacy systems) then Secret Server provides the CheckOut capability to ensure that the password is changed after it is used. |
| 3.7.2 | The creation or modification of privileged user accounts must be approved by at least two individuals: the system owner and a member of the Information Technology department. System administrators must not be allowed to create other privileged accounts without authorization. | Secret Server provides workflow capabilities with dual approval. Backdoor accounts can also be detected, secured and the Security Office can be notified to ensure that administrators never create unauthorized accounts.Account Lifecycle Manager establishes workflows for tighter control over service account creation.  |
| 3.7.3 | The number of privileged user IDs must be strictly limited to those individuals who absolutely must have such privileges for authorized business purposes. | Granular, policy-based controls within Secret Server and Privilege Manager allow role-based access and other rights to be limited to specific individuals and groups, allowing teams to follow the security best practice of least privilege with ease. |
| 3.7.4 | To facilitate secure management of systems, wherever possible, privileged accounts must be defined based on the specific role of the system administrator. | Secret Server allows specific access to privileged accounts, and access can be easily provisioned to administrators based on their membership in Active Directory groups. This allows privileged access to be granted only for certain functions based on role. |
| 3.8.1 | User IDs must uniquely identify specific individuals. Generic user IDs based on job function, organizational title or role, descriptive of a project, or anonymous must be avoided wherever possible.  | In the cases where generic user IDs can’t be avoided, Secret Server can be used to manage access using these accounts to ensure they are both secure and users remain accountable. |
| 3.8.2 | User IDs for service accounts and other application accounts should also follow the naming convention. | Account Lifecycle Manager delegates ownership of service accounts with role-based permissions to enforce governance and create accountability and ownership.  |
| 3.8.4 | System administrators managing computer systems with more than one user must have at least two user IDs: one that provides privileged access and is logged, and the other that provides the privileges of a normal user for day-to-day work. | Secret Server can be used to effectively manage named privileged accounts (A common example is an Active Directory Domain Admin account. This can be vaulted, ensuring passwords are very strong, changed frequently and Pass-the-Hash attacks are mitigated). While these accounts are assigned to a user, the high levels of access and privilege they are given require that they are governed as privileged accounts for purpose of this policy. |
| 3.9.1 | All privileged accounts must be managed by a central system. This system must provide an audit trial that tracks specific additions, changes, and deletions. | Secret Server tracks all usage of privileged accounts being managed. There is extensive auditing of all changes and usage, which can be configured to provide alerts, daily reports, or ad-hoc reporting for auditors. |
| 3.9.2 | Any privileged access management system must integrate with native operating system account management systems, such as Active Directory. | Secret Server integrates with Active Directory for authentication, granular permissions (role-based access control), and automation for password changing. |
| 3.9.3 | Any privileged access management system must integrate with strong authentication, such as multi-factor authentication, to ensure the identity of the user in addition to their directory authentication. | Various multi-factor technologies can be used with Secret Server, including RSA SecurID, Duo Security, Google Authenticator (TOTP), and any technology that is RADIUS compatible. Secret Server supports push notifications directly to the user’s phone, as well as hardware tokens such as YubiKey. If the user’s app or token isn’t available, they can also receive a phone call or text message for out-of-band authentication. |
| 3.9.4 | Company system administrators must have access to a vault system that enables the generation of temporary privileged accounts and passwords (aka FireID) for emergency maintenance. | Secret Server provides Unlimited Administrator Mode which can be used in emergency situations to gain access to restricted credentials. All actions are notified to appropriate groups and everything is fully audited. This ensures that an organization is prepared for any catastrophe from a password perspective and includes the ability for dual control to activate the firecall mode. |
| 3.9.6 | Company must maintain an inventory of all accounts with privileged access on production information systems.  | Secret Server provides Discovery capabilities to find new accounts on the network and automatically track those accounts against those in the vault. Accounts can also be automatically imported into the vault using configurable discovery rules. This ensures that the vault automates the process of inventory and reconciliation of accounts.Privilege Manager provides granular local group membership management to manage local admin rights on workstations. Account Lifecycle Manager builds a detailed inventory of service accounts, including owner, purpose, and dependencies.  |
| 3.9.7 | The privileged account inventory must be updated at least quarterly to identify new or changed accounts. | Secret Server provides an updated inventory of accounts on an ongoing basis through its automated discovery, which is continually reviewing the environment for changes. There is no need for human intervention or manual work. |
| 3.9.8 | All inactive accounts over 90 days old must be either removed or disabled. | Secret Server provides extensive APIs to allow configuration of rules such as this. However, the risk is much lower with managed privileged accounts since Secret Server is setting strong passwords, controlling access, and rotating passwords on a regular basis. |
| 3.9.9 | Any privileged account management system must be configured to utilize robust backup, recovery and availability methodologies in order to ensure resiliency and availability of the credentials stored within the system as well as the timely recovery of the system in the event of a system failure. | Secret Server supports load-balanced front-end application server configurations and back-end database clustering to ensure the highest levels of resiliency and system uptime. Secret Server also supports all standard backup and recovery methods for database backups and application recovery on standard platforms. |
| 3.10 | Every privileged user ID established for a non-employee or third-party application must have a specified expiration date, with a default expiration of 30 days when the actual expiration date is unknown. | Secret Server provides workflow approval for temporary access. The access is approved by an appropriate manager and the non-employee / third-party application can then use the credentials for the approved period of time, such as 30 days. |
| 3.11.1 | In accordance with a least privilege policy, default local administrator account should be removed prior to providing users with workstations. | Privilege Manager manages and removes local admin rights on domain and non-domain machines. |
| 3.12.1 | All development applications and systems that require privileged access, including DevOps tools, containers and microservices, must use secure privileged accounts. | DevOps Secrets Vault uses secure API calls that developers can use to create credentials and automatically pull them from a central, secure vault. This allows the credentials used in application development to be secured, audited and rotated. |
| 3.12.3 | Passwords must never be hard coded in software developed by or modified by Company workers. | DevOps Secrets Vault eliminates the need to hard-code credentials in software. |
| 3.12.4 | Credentials used in the application development process must never be stored in remote repositories, such as GitHub. | DevOps Secrets Vault eliminates the need to store credentials in remote repositories. |
| 3.13.1 | All privileged commands issued on computer and communication systems must be traceable to specific individuals through the use of comprehensive logs. | Using the SSH proxy/jump host, Secret Server intercepts all commands to the target system and records them in the audit trail for the session for that user. These sessions can then be searched by Secret Server users with the correct role permissions. This ensures that users are accountable for their activity. |
| 3.13.2 | All user ID creation, deletion, and privilege change activity performed by Systems Administrators and others with privileged user IDs, including third parties, must be securely logged. | Secret Server can control access to all privileged accounts and can monitor and log activity.  |
| 3.13.3 | All logs recording privileged ID activity must be reviewed at least quarterly via periodic management reports. | This can be done in Secret Server through ad-hoc or scheduled reports to be sent to and reviewed by the appropriate people. |
| 3.13.4 | All logs recording privileged ID activity must be aggregated into a central log management or Security Information and Event Management (SIEM) tool in order to correlate privileged ID activity to other security events, log entries and related non-privileged ID activity. | Secret Server supports native integration with several SIEM tools and can also output log files in standard CEF or Syslog formats for easy integration into most any log management or SIEM tool. |
| 3.13.5 | In addition to event logging, all activity on privileged accounts must be logged via session and keystroke recording. | Secret Server provides event logging and can also capture session activity through session recording and keystroke logging. |
| 3.14.1 | Only trusted applications should be allowed to be installed or executed automatically. | Privilege Manager’s policy-based controls allow trusted applications to be elevated and allowed to run, while maintaining a least privilege model.  |
| 3.14.2 | Specific applications known or suspected to contain malicious code may be added to a blacklist and not allowed to be installed or executed. | Privilege Manager creates policy-based controls such that applications known or suspected to be malicious can be added to a blacklist or a greylist for investigation before being allowed to run. |

**About Delinea**

Delinea is a leading provider of privileged access management (PAM) solutions for the modern, hybrid enterprise. We make privileged access more accessible by eliminating complexity and defining the boundaries of access to reduce risk, ensure compliance, and simplify security. Delinea empowers over 14,000 customers worldwide, including over half the Fortune 100. Our customers include the world’s largest financial institutions, intelligence agencies, and critical infrastructure companies. [delinea.com](https://www.delinea.com/)