Unix Privileged Account Discovery Report

Prepared for Acme, Inc.

Scan Date: 07/27/2020 20:19:14. Scan completed in 0 minutes.

IP Addresses Scanned: 10.10.202.145

Weaknesses Found: 2368

Unix and its open source cousin, Linux, are high value targets for cybercriminals. If you don't include Unix and Linux in your PAM strategy you're leaving open some of the most vulnerable holes in your attack surface.

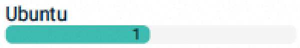
All Unix and Linux systems suffer from the same weak point — the root account. Root account access provides the highest and broadest level of control (god-like privileges), so protecting root accounts must be a priority.

In a Unix/Linux environment, there are many super users. A user must have super-user privileges for many day-to-day operations. A super user (such as root) has access to the operating system in an unrestricted form, and therefore has access to all commands, files, directories and resources.

The typical approach to Unix/Linux privileged security increases risk and incurs administrative time and cost.

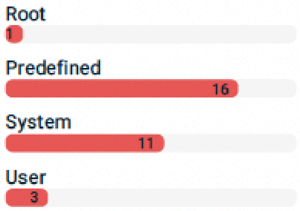
- Every Unix/Linux server has a local database of users and groups. If you have a single host with one user, there's no problem with privilege management. But, when you have multiple servers, a single user becomes split across them and it's difficult to maintain visibility and control across different home directories, user ID's, and passwords.
- Using shared accounts to simplify privilege management is a violation of least privilege principles and makes it difficult to know who is using the system and what they are doing.
- Sudo helps to enforce least privilege by temporarily elevating user accounts to have root privileges, but it introduces risk because it's controlled by local files. Auditors don't like distributed Sudo configuration files because they utilize "static trust." These files are stored in a way that local administrators could easily make modifications.
- Additionally, to properly use Sudo you need highly skilled and highly paid system administrators to spend a great deal of time building sudoers files. Then you have to distribute the files across your organization. If you don't maintain and update Sudo, you may miss security vulnerabilities.

Proactive and ongoing management is essential to ensuring consistent security policies and protecting Unix/Linux privileged accounts and passwords. You can use the results of this report to gain visibility of all Unix/Linux accounts and identify areas of highest risk. The associated file shows additional detail for specific accounts that require immediate attention.



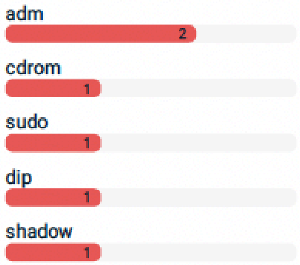
Operating System Breakdown

Many organizations have diverse Unix/Linux systems and platforms in use, increasing the challenge of managing Unix accounts centrally. Older operating systems may present an attack vector for malware if updates are no longer issued. Additional detail on distribution version and Linux kernel version are available in the CSV file associated with the report.



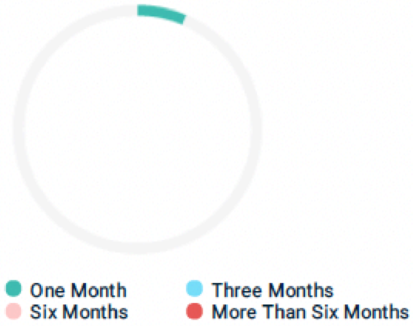
Unix Account Types

A high number of Unix user accounts is a security concern. There's nothing preventing a super user with broad privileges from intentionally or accidentally deleting a system file. It can be difficult to confirm users are who they appear to be without multi-factor authentication. Without built-in accountability, there's no way to tell which person may be responsible for damage. You can see additional detail on Unix/Linux users (user name, home directory, shell, comments) in the associated CSV file.



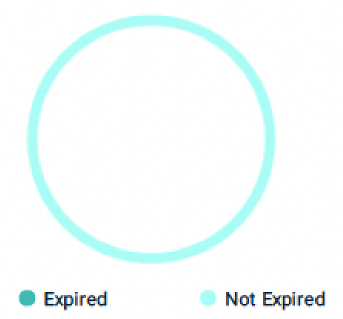
Top Unix Groups

These Groups have the largest number of members. Groups have the possibility to give users elevated privileges (sudoer) that would allow them to perform high-level administrative operations. If there are a large number of Sudo Group members, this could signify misconfiguration or backdoor local accounts.



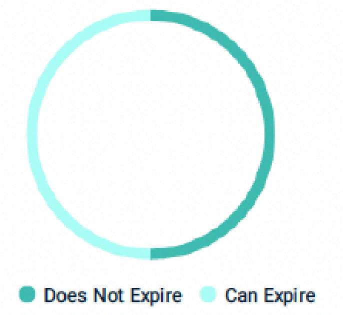
Active Accounts

This chart shows when accounts have last been logged into. If accounts aren't being used regularly, they should most likely be disabled.



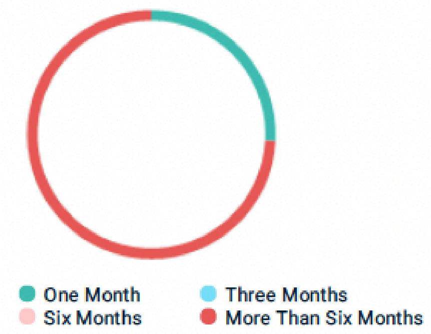
✓ Password Expiration Health

This chart includes the number of Unix local accounts that haven't had their password changed and are now expired. Expired passwords are an attack vector that can be leveraged by both internal and external attackers. If accounts are no longer needed, they should be disabled.



✓ Unix Accounts That Never Expire

Without automation that forces expiration, privileged users are never prompted to change their passwords. Frequent changes are necessary to prevent password theft and abuse of privileged accounts.



! Password Age By Month

Unix local accounts with passwords that change infrequently are a security risk, because former employees or attackers may use old passwords to gain access. Passwords should be changed on a regular basis to prevent abuse.

Unix Computer Scan Summary

Unix Computers Scanned	1
Unix Accounts Found	31

Put this report to use

After reviewing the results of the Discovery Tool, it's easy to see that comprehensive Privileged Access Management (PAM) that includes Unix/Linux accounts would reduce your organization's risk and save management time.

A multi-layered approach to PAM will help you protect Unix/Linux privileged accounts and credentials. Take these steps to simplify management, enforce consistent security policies, and reduce your privileged account risk.

Step 1

To manage Unix security in a sustainable way, you need to get user accounts under control and assigned to a single account. An Active Directory bridge extends Group Policy to non-Windows platforms so you can perform account maintenance and password updates for all systems through a single tool. With an identity bridge you have consistent data across all of your systems. Each user truly has one username, one ID, one password, one home directory, etc.

Step 2

Once this type of identity unification is in place, a central PAM solution lets you authenticate users and assign privileges with ease. PAM solutions enable ongoing discovery of Unix/Linux super user privileges to increase accountability and enforce consistent management. Immutable audit trails and enhanced controls such as session monitoring and recording provide oversight and simplify reporting and compliance.

Step 3

After a user is authenticated and logged in to your IT resources, Unix privilege management controls what actions they can take. Granular control of root credentials limits privileges while meeting compliance regulations and policies for Unix super user Privilege Management (SUPM).

You can reduce the risk of privileged account abuse or accidental error by restricting commands based on defined policies and limited super user permissions. SSH command control allows administrators to log in as root, but only gives them access to a set of predefined commands.

Learn more about how Delinea protects Unix privileged accounts

Delinea empowers thousands of organizations around the globe, from small businesses to the Fortune 500, to manage privileged access.

Delinea Identity Bridge

Identity Bridge utilizes your existing directory service to manage identities across the enterprise regardless of platform and operating system. It connects Unix/Linux root access to a user account so all activity can be logged. As a result, you minimize privilege account sprawl, harden your attack surface, and ensure compliance through detailed reporting on the use of privileged Unix/Linux accounts. Users have only one username and password to remember and IT administrators no longer have to manage Unix/Linux servers separately from other servers and workstations.

Secret Server

Secret Server is the only fully featured PAM solution available both on premise and in the cloud. Delinea's award-winning software gives security and IT ops teams the agility to secure and manage all types of privileges, protecting administrator, service, application, and root accounts from cyber attack. You can set up a secure vault, discover privileges, manage secrets, delegate access, and control sessions – all from a central hub.

Secret Server's AD integration enables users to sign in with their normal domain account to gain access to privileged accounts, such as their domain administrator credential. By assigning access in Secret Server based on security groups, you won't have to manually grant permissions every time a new admin needs access or users change roles.

Try Secret Server and AD integration for yourself with a free, 30-day trial