Privilege Access Analysis Security Tool for Windows & Linux

Wasef Anwar

Abstract— In an organization, governing privilege access to their infrastructure, data, and resources is vital to ensure that authorized users have the appropriate level of access to the resources required to do their functional tasks. For example, domain accounts or Local Admin accounts provide privileged access to network resources.

Windows Active Directory is the primary directory service used across the globe by organizations to manage permissions and access network resources by administrators and is available in all Windows Server operating Systems. Initially created for centralized domain management service only, Windows active directory is a title used for a range of identity services in Windows Operating Systems.

Identity and Privilege Access Management (IAM) is implemented with Windows active directory services to ensure enterprises have defined functional roles that enable network users or administrators to have privileged access. For example, a user might be a customer, an employee, or a contractor who can be an administrator, a supervisor, or an Infrastructure support personnel. The core aspect of implementing IAM resolves around a single digital identity per individual. These identities, once established, need to be maintained, modified, and monitored throughout an identity access lifecycle. The primary goal of enabling identity and access management is to readily provide administrators the necessary capability to change roles, monitor identities, create reports for audit and compliance purposes and enforce necessary policies account the enterprise.

Privileged Access Management (PAM) is the process and technology used to control and monitor privileged entities in the network. These accesses are arranged in a hierarchy by domain accounts and domain groups as per their functional roles. Implementing this enables the organization to effectively manage access for privileged users and ensure seamless onboarding/offboarding. In addition, provisioning through groups allows the segregation of permissions based on the functional requirements of the roles.

Functional roles and their permissions increase over time, making it difficult to map them. Nesting occurs on functional groups, and their privileges on the active directory domain and implementation of least privilege on the roles become tedious. These instances increase the vectors for lateral and horizontal privilege escalations, they also increase the difficulty to find the overlap in permissions and map the Privilege entities to their least privileged permissions. Linux Devices have local admin accounts, which are Superuser(root), system users (process and daemons), and regular users. In addition, ordinary users are provided with sudo permissions to perform their functional activities.

The cloud-based security tool will resolve the challenges in managing privileged accounts by identifying account password compliance and mapping privileged users and groups through a graph. The tool queries Windows Active Directory and Linux devices using scripts to fetch the necessary attributes/parameters from users/groups allowing device/Platform specific script-based deployment using natively supported queries to fetch account attributes. It will then map out the users and the groups using the attributes and establish relationships between them to show defined security statistics in a dashboard and a graph. These statistics in the security tool will allow the organization to view actionable insights to understand compliance and vulnerability of the privileged access in the environment.

Index Terms— Privilege Access Analysis, Directory Services, Windows Active Directory, Active Directory, Privilege account, Identity & Access Management, IAM, Privileged Access Management, Windows Directory, Security tool, Cybersecurity.

IJSER

1 Introduction

PRivilege Accounts exist in every organization. It is estimated that a user can have up to an average of 7 privileged accounts in an organization. Even though security policies are in place to ensure that passwords expire after a set period, many accounts like domain service accounts, application accounts, shared privilege accounts, local privileged service accounts, and default administrator accounts, i.e., root, administrator do not have passwords rotated. As a result, they usually remain unmonitored because password rotation of an account is a manual process. The failure of proper management of these accounts also creates security and compliance concerns.

Privileged Access Management tools are necessary for an organization to manage credentials of all privilege accounts and ensure adequate password compliance status. However, these tools are generally expensive to implement, tedious to operate, and require a long time to mature the organization's security posture. SME/SMB organizations do not have the financial bandwidth to implement expensive PAM security tools.

Enterprise security tools available in the industry have the following limitations:

i. They cater to limited security use cases for the organization.

ii. They are costly to implement, maintain and customize.

iii. There is a lack of open-source standard tools and processes throughout the IAM domain.

Privilege Account management relies more on implementation experience.

iv. Simple reporting for PAM is expensive as it requires manual script customization and deployment over a wide range of Operating systems, devices, appliances & platforms.

v. They are unable to map the privilege accounts across the various platforms available.

Windows Active Directory is implemented across the globe by organizations to manage permissions and access to network resources by Privilege users, hence chosen as the primary scope of the proposed security tool.

Linux Operating systems are utilized across the globe due to their robustness and cost-effectiveness. In addition, as Linux powers 90% of all cloud infrastructure [8], it has been chosen as the secondary scope of the proposed security tool.

When organizations use Windows Active Directory [1] to manage their core infrastructure, privileged accesses are provisioned according to functional roles, using domain accounts and domain groups [9]. As a result, nesting occurs in the privileged access as defined permissions overlap with the increased user base and functional roles. As a result, it becomes challenging to find the overlap in permissions and map the privilege accounts to the least privileged permissions required for the functional tasks.

When organizations use Linux devices, local admin accounts like Superuser(root), system users (process and daemons), and regular users [7] are used to manage the services. Once regular users are granted sudo permissions by being added to the sudo list, they have privileged local permissions without any restrictions, audit capabilities, or controls.

The security solution will communicate with the Windows Active Directory using REST APIs over HTTP (port:80), where native PowerShell commands from the Active Directory Module query the domain controller. These collect the necessary attributes of users and groups to be stored in a database in the security tool. In addition, it will analyze the stored data of the users and the groups, establish relationships between them, and show password compliance and statistics of privileged accounts in the windows directory. It will also display the relationship in a graphical format. The security solution will also connect to Linux using REST APIs over HTTP (port:80), where data of local admin users is gathered using standard commands. The attribute data is collected and stored in a database in the security tool.

There is no existing formal research to baseline the project approach, framework, and implementation of Privilege Access Analysis. Therefore, risk reduction in Identity in Access management is based on implementation experience on the technology implemented in the organization. The literature review provides the critical approach with the predominant technologies used to reduce risk and thereby to help formulate the fundamental concepts around privilege access analysis.

2 LITERATURE REVIEW

This study explores Privilege Access Analysis in the Identity and Access Management (IAM) domain of cybersecurity. Unfortunately, there has been no formal research around Privilege Access Analysis for overall cybersecurity risk reduction and mitigation to baseline the research. However, the research papers below advise on the key parameters to achieve risk reduction based on the technology implemented for privilege access. These assist in formating an approach towards Privilege Access analysis based on the predominantly used technologies in the IAM domain.

Privilege access is the core subject of the project. The privileged identity (Privilege account) should exist within a policy-effective domain(s) to implement adequate access controls. Role-based access control (RBAC) regulates access based on the known identity or attributes, such as groups, security clearances, and roles. Role-based access control is a crucial method to configure access in an organization and is used by most organizations in their Directory Services. Segregation of Privileges ensures flexible, functional implementation in providing access to network resources.[4]

Information visualization research deals with unstructured data. However, this is not the Primary subject for this project. Visualization in this project focuses on representations of structured data queried from the Windows Active Directory and Linux Devices as graphs are the fundamental structural representation of the data, we are using to map the Privilege account information. Information visualization is a core aspect in displaying the data analyzed in the context of privilege account in this project.[10]

Windows Active Directory is a directory service used to store user information about network resources and provide a hierarchal access structure to the network infrastructure and other applications. It is a central collection of users, groups, and computers, enabling single sign-on (SSO) for devices and applications joining the AD domain.

It stores information about network resources such as users, user credentials, groups and makes the information available to users and administrators. In addition, the active directory allows the administrator to manage centralized management with the help of group policy.

When a user authenticates into a Workstation that is a member of a windows domain, the active directory validates the username and password and allows access.

Existing tools provide mechanisms for managing user properties, passwords, and group membership in bulk; however, each tool only solves an individual problem.[2], [3], [9]

Privileged Access Management (PAM) is a crucial area of data security in an organization. Privileged accounts are provided to administrators and other users to access critical data and applications. Evolving practices along with digital transformation have ensured that privileged accounts are widespread. If not managed securely, organizations can be exposed to the risks of unmanaged shared accounts, abandoned accounts, and malicious actors are becoming more adept at misusing the credentials to gain access to vulnerable or critical resources. To reduce risk and ensure stringent GRC obligations are met, a cost-effective PAM solution is essential. [11]

Lateral privilege escalation in Windows Infrastructure occurs when attackers create anomalies at the behavior level of privilege access. In a scenario where a domain admin account can only be used from a specific workstation, when used from another workstation becomes a suspicious activity and may indicate lateral movement, where the detection of lateral movements is possible by effectively monitoring the Windows events. Lateral movements are detected by identifying the use of accounts from or to unusual or non-authorized systems. [6]

Prominent lateral movement techniques are dependent on obtained credentials stolen from the attacked network or individual. For example, the Red October hacker activity saw attackers compile a list of all credentials from any available location on the internal network to navigate the IT environment undetected from 2007 to 2013. If malicious activity raised suspicion on one set of credentials, the attacker switched and moved to another network area. [12]

IJSER

In conclusion, the research gap and unavailability of baseline research for Privilege Access Analysis in Identity and Access Management (IAM) domain in cybersecurity confirms that there is no single method or approach to address a security solution that can analyze and effectively reduce risk in the organization. Hence, we have created a clearly defined problem statement to create and implement the security solution proposed for this project.

3 Problem Statement

As users can have up to 7 privileged accounts in an organization, however even though policies are in place to ensure that passwords expire after a defined period of time, however many accounts like service accounts, application accounts, shared privilege accounts, and default administrator accounts go unmonitored as password change is a manual process and fail compliance. As Windows Active Directory is the primary directory service used across the globe by organizations to manage permissions and access to network resources by administrators, our observation advised that this would be the most optimum directory service to build the monitoring capability.

Organizations use Privilege accounts to provide access to their staff to access infrastructure. Windows Activity Directory services are the most widely implemented directory services used to create roles and permissions for Privilege accounts in an organization, enabling them to segregate permissions and apply Group policies to most of their infrastructure seamlessly.

In large organizations, Privileges are overprovisioned and nested in Active Directory. Security policies are enforced to ensure compliance, but there is no periodic validation to check if they have been successfully executed. Furthermore, with an increasing number of privileged accounts (Interactive and non-interactive), regular password updates are not feasible due to the efforts required to ensure compliance.

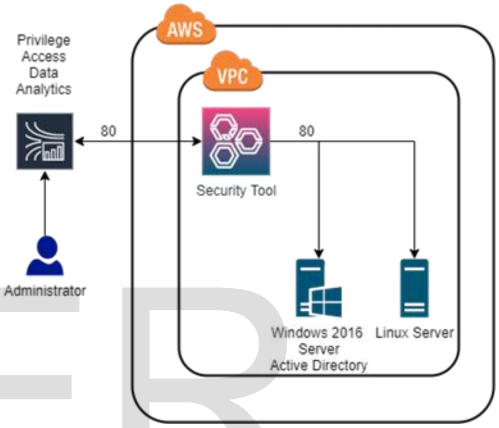

Privilege Access Data Analytics
AWS
VPC
80
80
Security Tool
Administrator
Windows 2016 Server Active Directory
Linux Server

Figure 3.1: Fundamental design of the security tool.

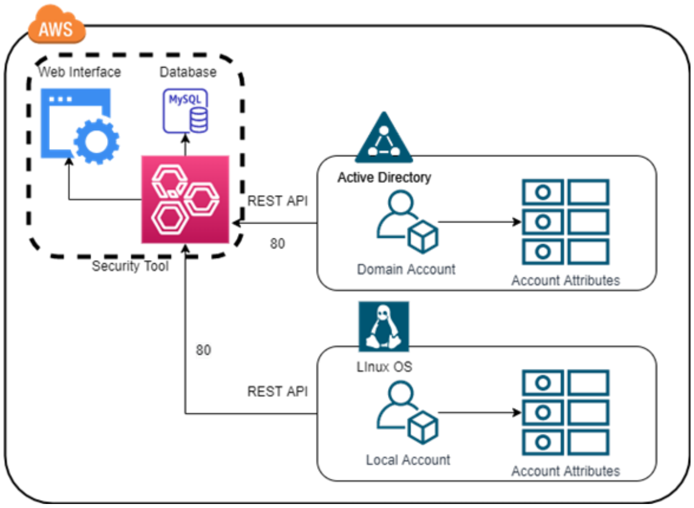
This project proposes to create a cost-effective cloud-based security solution that could analyze and map the Privilege accounts on Windows Active Directory and Linux Devices. It would provide actionable insights to the organization through a dashboard and graph to help in risk reduction. The solution uses REST APIs to collect the Account attributes from the platforms using native commands, i.e., PowerShell in Windows AD and CLI for Linux. It would then store the Account attributes in a database for further analysis. Furthermore, it would map the users and groups and represent them in a graphical format.

As the Identity and Access management domain applies to various technologies, we have formulated the study's objectives to ensure a well-defined approach towards creating the framework for this solution.

4 Objectives of the Study

The project aims to create a security tool to assist the organization by providing insights to reduce risk, which is achieved by conducting Privilege Access analysis and sharing critical information in a concise dashboard.

The below Primary and Secondary objectives are clearly defined to achieve key metrics in Privilege Access Analysis in the security tool.



Functional design of the security tool.

IJSER

Primary Objectives

1. Create a cloud base Privilege Access Analysis Security tool.
2. Map out the Privilege accounts, privilege groups, and their attributes in AD and Linux.
3. Use attributes to analyze accounts and establish relationships in Privilege Access.
4. Show password compliance status and group nesting and establish a vulnerability metric of privilege accounts in a dashboard.

Secondary Objectives

1. Create an API-based scalable approach to store account attributes queried from Active Directory and Linux using APIs, which can be stored in a database.
2. Identify all the users and groups which are nested based on data.
3. Visualize directory Mapping in a graph.
4. The data analyzed will display the below security attributes in a dashboard:
   - i. The password compliance status of all accounts.
   - ii. Sudo users on the servers with their privileges.

The tool utilizes local python scripts to query the Windows Active Directory services for users and groups created in the domain to retrieve all attributes which constitute the Privilege accounts in the environment. The data collected is stored in the security tool server in a MySQL database. The database is analyzed to create the statistics necessary to extrapolate the compliance information.

The compliance statistics of the accounts are displayed through the micro web service or Web micro-framework FLASK, ensuring complex data is available to all the relevant stakeholders to be absorbed, analyzed, and used to improve the organization's security posture. The solution aims to provide the security tool user, analyzed data that is otherwise not easily or readily available or easily understandable. Complex AD queries can fetch the data but organization, and utilization of the data for Privilege access requires a clear understanding of AD.

The design of the security tool enables us to have a dual purpose, to make the data queried from AD available and exportable, along with visualization of the data as per key metrics derived to assist in establishing the PAM security posture of the organization using Windows Active Directory Services.

The solution aims to provide an inexpensive real-time overview of the organization's compliance. These parameters are not available in any previous use cases of expensive security tools available in the market. A new approach to ensure compliance for Privilege Accounts was necessary to create the solution/utility to provide a new perspective to Privileged Access Management tools.

The cloud-based Security tool is created in the Amazon Web Services (AWS) environment with a working AD and Linux infrastructure to replicate real-world scenarios. We have used the infrastructure to set up a windows domain controller, Linux servers, and an Ubuntu server for the security tool that will pull the data from the AD server using API scripts. The Ubuntu server is also a webserver to display the analyzed data to the users.

This solution would be modular and easy to implement in any AD infrastructure and would only need python-generated API queries deployed in the existing domain controllers. In addition, the security tool is created using Web Micro Frame and data displayed in an understandable dashboard.

The critical element of the project is to create a security solution that analyzes the data display of the Privilege Access Analysis metrics to add value to the organization. The Project Methodology further elaborates on the method to achieve the objectives of the security solution.

5 Project Methodology

This security tool creates the mapping utility to query the Windows Activity directory and visually show the Privilege Account statistics. Hence to achieve this objective, we have replicated a real-world infrastructure in the AWS environment. The environment contains a Primary Domain controller, two Linux servers, and the Security tool server.

Active directory services installed in a Windows 2016 server, a Primary Domain Controller in AWS, contains privileged accounts and groups. Python Scripts query the Active directory using PowerShell commands, transfer the user and group attributes using REST API to the security tools server, and store them in a MySQL database. The data, once stored, is analyzed to form PAM parameters displayed in a simple GUI.

This model Privilege account Environment is based on real-world scenarios containing PAM Role-based access control (RBAC) and User-based access control (UBAC) models predominantly used in organizations to manage roles and permissions.

Scripts used in Linux servers query the OS using standard commands to find out the list of local users and users with sudo privileges. These user attributes are analyzed after being stored in the database in the security tool. Once analyzed, the Privilege access parameters are displayed on the dashboard.

The user and group attributes in both Windows Active Directory and Linux devices are analyzed for any correlation to ensure the incoherent data is visualized in a readily available dashboard of the security tool.

The following are considered the most critical aspects of the solution design:

1. Data can be queried from Windows AD and Linux devices using custom python scripts, providing real-time access to the data.
2. The Privilege users and AD Groups attributes can be easily searched from the GUI of the security tool.
3. Critical statistical information providing the Privilege posture of the organization is available in the security tool.

6 Resource requirement Specification

SL No	Resource Type	Function
1	**AWS**	The Working environment is built on AWS to enable easy deployment and accessibility
2	**Windows Server 2016**	This OS is used for the deployment of critical Active Directory services required for the Core infrastructure
3	**Windows Domain Controller**	Windows Domain controller is implemented in the Windows Server to enable Active Directory.
4	**Windows Active Directory**	Active directory is used to create a standard PAM model for Privilege accounts using RBAC/UBAC.
5	**Windows PowerShell**	Active Directory module in PowerShell is used to query AD to gather User and group attributes
6	**Ubuntu 18.04 LTS**	Standard Ubuntu is used as a security tool server to fetch the data, store the data and display the data
7	**FLASK**	Web microservice framework used to build a simple Graphic user interface
8	**MySQL 5**	The database is used to store the gathered account and group attributes.
9	**JavaScript**	Chart.js functions are used to display the data in a statistical formal
10	**Python**	Python scripts have been used to structure the AD queries, retrieve the data using APIs to the utility server.
11	**Ubuntu Devices**	Linux Operating systems servers to replicate PAM infrastructure

Amazon Web Services: This cloud computing service provider was used to build the solution. The core infrastructure created using the Amazon Elastic Compute Cloud service allows deploying virtual servers on the internet with 99% availability and low cost.

Windows Server 2016: The windows server operating systems was selected to ensure reliability in building windows activity directory services. In addition, it allows the most seamless method to build the core infrastructure required to demonstrate the solution.

IJSER

Windows Domain Controller: Windows Domain controller is a built-in Windows Server 2016 server, where it is used to respond to authentication requests within the computer domain. The domain controller is a standalone DC created to replicate the core infrastructure in organizations.

Windows Active Directory: Active directory role enabled in the Windows Service is used to create the network users in the PAM model. The users are based on RBAC/UBAC PAM models and segregated based on groups. Nesting is purposely created to display the capabilities of the security tool.

Windows PowerShell: PowerShell is used as an automation and configuration management framework [13] consisting of a command-line shell and the associated scripting language. When used via administrative rights, PowerShell provides complete access to WMI, enabling us to query the active directory directly and retrieve the necessary data from Windows local and Windows Domain Servers.

Ubuntu: Ubuntu OS is a Linux distribution based on Debian containing free and open-source software.

The easy accessibility of open sources resources required to deploy the solution was the primary reason to select the OS.

FLASK: FLASK is a microframework written in Python for the web, and it does not require any particular tools or libraries; however, as FLASK supports extensions that can add application features, this was chosen to create the GUI.

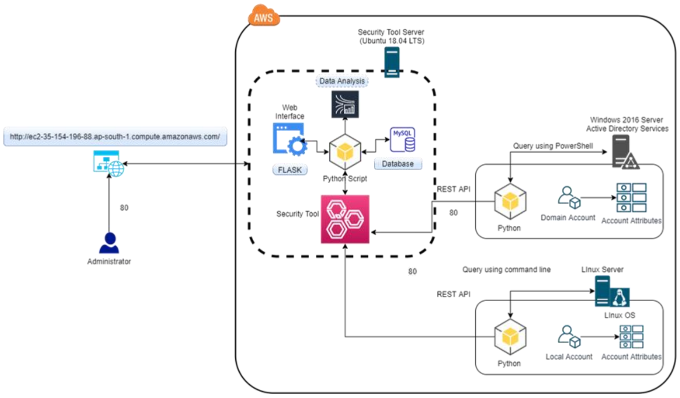
MySQL: MySQL is a free and open-source relational database management system that organizes data into data tables where data types may be related. These relations can further help structure the data.[14]

JavaScript: avaScript is a high-level programming language used in designing the web GUI of this security tool.

Python: It is a high-level Programming language used to create the necessary scripts for the Utility servers.

The above components were carefully chosen after considering the objectives defined for the project to develop the security tool. They are the fundamental blocks for the software design and workflow of the security tool.

7 Software Design



The above diagram represents the low-level design of the Project implementation

Primary functional components:

Core Infrastructure: The core Infrastructure contains the Windows 2016 servers, used as PDC to deploy Windows Active Directory services

CoreDirectory.Internal: Domain was created for the core infrastructure and necessary structure OUs to segregate Access permissions based on Role-based access control and user-based access control.

Based on the best industry practices, users and groups were created as per the diagram below to emulate the Privilege Access Model of an organization.

PAM Database: MySQL database was created to store all queried AD objects and their attributes for further analysis.

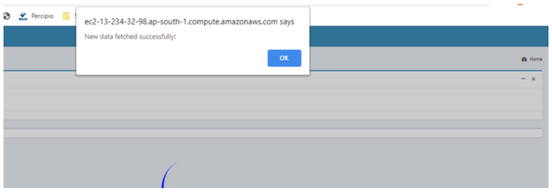
Security tool Server

This server is created using Ubuntu OS to contain all the data, scripts, and the solution's web interface.

Account Compliance and Statistics

The data gathered is then analyzed and display in the GUI dashboard for interpretation.

8 Results



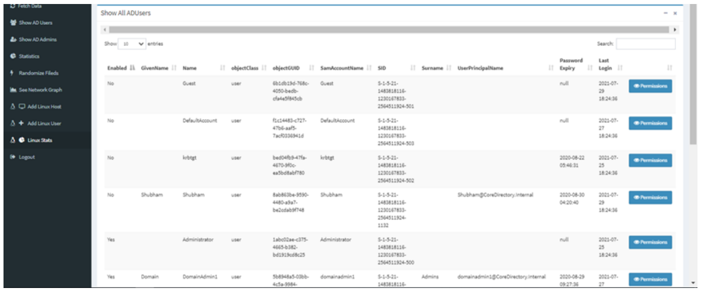
ec2-13-234-32-98.ap-south-1.compute.amazonaws.com says

New data fetched successfully!

OK

The above screenshot displays the Fetch Data capability of the Web Application

IJSER

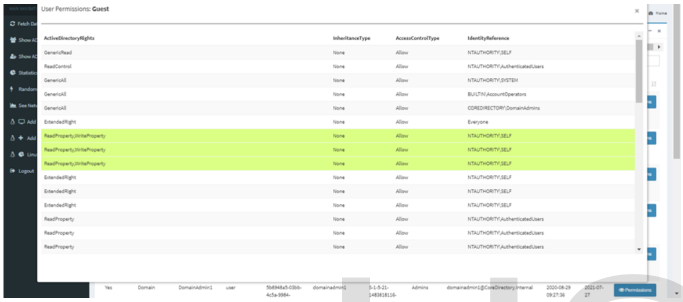


Show All ADUsers

Show 10 entries Search:

Enabled	GivenName	Name	objectClass	objectGUID	SamAccountName	SID	Surname	UserPrincipalName	Password Expiry	Last Login	
No		Guest	user	6b1db19d-768c-4050-bedb-cfa4e5f845cb	Guest	S-1-5-21-1483818116-1230167833-2564511924-501			null	2021-07-29 18:24:36	Permissions
No		DefaultAccount	user	f1c14483-c727-47b6-aaf5-7acf0336941d	DefaultAccount	S-1-5-21-1483818116-1230167833-2564511924-503			null	2021-07-27 18:24:36	Permissions
No		krbtgt	user	bed04fb9-47fa-4670-9f0c-ea5bd8abf780	krbtgt	S-1-5-21-1483818116-1230167833-2564511924-502			2020-08-22 05:46:31	2021-07-25 18:24:36	Permissions
No	Shubham	Shubham	user	8ab863be-9590-4480-a9a7-be2cdab9f748	Shubham	S-1-5-21-1483818116-1230167833-2564511924-1132		Shubham@CoreDirectory.Internal	2020-08-30 04:20:40	2021-07-29 18:24:36	Permissions
Yes		Administrator	user	1abc02ae-c375-4665-b382-bd1919cd8c25	Administrator	S-1-5-21-1483818116-1230167833-2564511924-500			null	2021-07-25 18:24:36	Permissions
Yes	Domain	DomainAdmin1	user	5b8948a5-03bb-4c5a-9984-	domainadmin1	S-1-5-21-1483818116-	Admins	domainadmin1@CoreDirectory.Internal	2020-08-29 09:27:36	2021-07-27	Permissions

The above screenshot displays the Show All AD user Page.



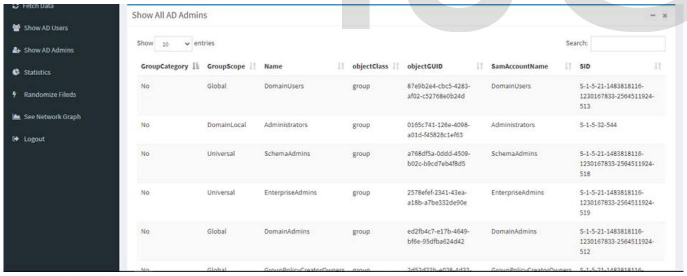
User Permissions: **Guest**

ActiveDirectoryRights	InheritanceType	AccessControlType	IdentityReference
GenericRead	None	Allow	NTAUTHORITY\SELF
ReadControl	None	Allow	NTAUTHORITY\AuthenticatedUsers
GenericAll	None	Allow	NTAUTHORITY\SYSTEM
GenericAll	None	Allow	BUILTIN\AccountOperators
GenericAll	None	Allow	COREDIRECTORY\DomainAdmins
ExtendedRight	None	Allow	Everyone
ReadProperty,WriteProperty	None	Allow	NTAUTHORITY\SELF
ReadProperty,WriteProperty	None	Allow	NTAUTHORITY\SELF
ReadProperty,WriteProperty	None	Allow	NTAUTHORITY\SELF
ExtendedRight	None	Allow	NTAUTHORITY\SELF
ExtendedRight	None	Allow	NTAUTHORITY\SELF
ExtendedRight	None	Allow	NTAUTHORITY\SELF
ReadProperty	None	Allow	NTAUTHORITY\AuthenticatedUsers
ReadProperty	None	Allow	NTAUTHORITY\AuthenticatedUsers
ReadProperty	None	Allow	NTAUTHORITY\AuthenticatedUsers

Yes	Domain	DomainAdmin1	user	5b8948a5-03bb-4c5a-9984-	domainadmin1	S-1-5-21-1483818116-	Admins	domainadmin1@CoreDirectory.Internal	2020-08-29 09:27:36	2021-07-27	Permissions

The above screenshot displays the User attributes and permissions.

IJSER

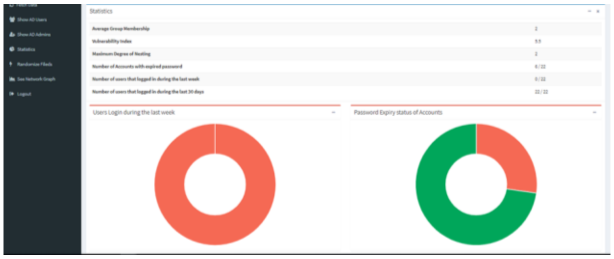


Show All AD Admins

Show 10 entries Search:

GroupCategory	GroupScope	Name	objectClass	objectGUID	SamAccountName	SID
No	Global	DomainUsers	group	87e9b2e4-cbc5-4283-af02-c52768e0b24d	DomainUsers	S-1-5-21-1483818116-1230167833-2564511924-513
No	DomainLocal	Administrators	group	0165c741-126e-4098-a01d-f45828c1ef63	Administrators	S-1-5-32-544
No	Universal	SchemaAdmins	group	a768df5a-0ddd-4509-b02c-b9cd7eb4f8d5	SchemaAdmins	S-1-5-21-1483818116-1230167833-2564511924-518
No	Universal	EnterpriseAdmins	group	2578efef-2341-43ea-a18b-a7be332de90e	EnterpriseAdmins	S-1-5-21-1483818116-1230167833-2564511924-519
No	Global	DomainAdmins	group	ed2fb4c7-e17b-4649-bf6e-95dfba624d42	DomainAdmins	S-1-5-21-1483818116-1230167833-2564511924-512
No	Global	GroupPolicyCreatorOwners	group	2d52d22b-e028-4d33-	GroupPolicyCreatorOwners	S-1-5-21-1483818116-

The above screenshot displays the Show All AD Admins page.



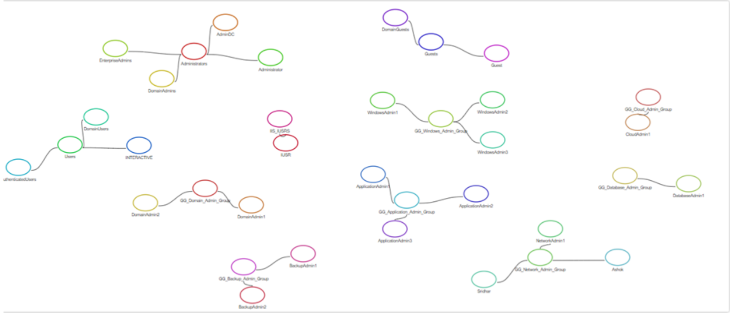
Statistics

Average Group Membership	2
Vulnerability Index	5.5
Maximum Degree of Nesting	2
Number of Accounts with expired password	6 / 22
Number of users that logged in during the last week	0 / 22
Number of users that logged in during the last 30 days	22 / 22

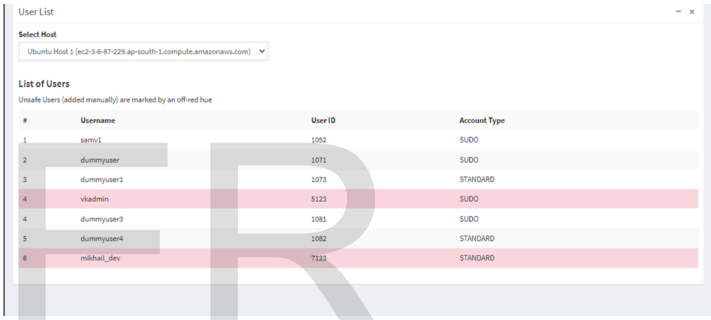
Users Login during the last week

Password Expiry status of Accounts

The above screenshot displays the Statistics dashboard of the security tool.



The above screenshot displays the Windows active groups and users in a graph.



User List

Select Host

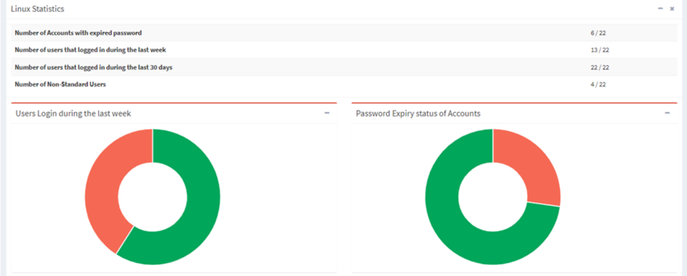
Ubuntu Host 1 (ec2-3-6-87-229.ap-south-1.compute.amazonaws.com)

List of Users

Unsafe Users (added manually) are marked by an off-red hue

#	Username	User ID	Account Type
1	samv1	1052	SUDO
2	dummyuser	1071	SUDO
3	dummyuser1	1073	STANDARD
4	vkadmin	5123	SUDO
4	dummyuser3	1081	SUDO
5	dummyuser4	1082	STANDARD
6	mikhail_dev	7123	STANDARD

The above screenshot displays the standard list of Linux users configured where non-standard users are highlighted in red.



Linux Statistics

Number of Accounts with expired password	6 / 22
Number of users that logged in during the last week	13 / 22
Number of users that logged in during the last 30 days	22 / 22
Number of Non-Standard Users	4 / 22

Users Login during the last week

Password Expiry status of Accounts

The above screenshot displays the Linux device statistics in the environment.

9 Analysis and Result

The problem statement varied as Privilege access compliance is required in each organization that uses a digital resource on a network. Active directory is almost always used to ensure Roles and permissions can be segregated based on the scope of access required.

The solution provides a seamless and accessible interface to extrapolate the data from AD and use it for analysis. Although querying AD and Linux devices is possible manually, using the security tool provides access to data in a single dashboard that can be used to analyze further or design a more efficient PAM model.

By simplifying the task and prioritizing data visualization, data analysis, data gathering, and accessibility to the data, we effectively resolve a challenge faced by administrators of an organization.

The objective is also to discover other data points which can be analyzed and developed better intelligent capabilities for the user.

10 Conclusions

The cloud-based security tool creates a new approach toward privilege access management.

The visualization of data helps to quickly digest the compliance posture of the Privilege accounts in the domain and the Linux devices.

11 Future Work

This solution could be further customized based on the following prospects to add value to the organization.

1. Password rotation and account disabling features built into the security tool.
2. Upon discovering privilege nesting in the Windows active directory after mapping, further intelligence could be added to the security tool to suggest better PAM implementation models to the administrators.
3. Security to the tool is enhanced by server hardening and ensuring all communication from the device to the tool occurs through the encrypted HTTPS protocol.
4. Email notification engine can be implemented to provide notifications to the administrator using SMTP.

References

[1] Microsoft, "Active Directory Domain Services Overview." https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview.

[2] S. Sharad, A. Singh, and A. Rai, "Research Paper on Active Directory," *Int. Res. J. Eng. Technol.*, vol. 3579, no. 4, p. 4, 2008, [Online]. Available: http://www.tech-faq.com/logical-structure-of-an-active-directory.html.

[3] "Active Directory Group Management Best Practices.".

[4] D. Shin and G. J. Ahn, "Role-based privilege and trust management," *Comput. Syst. Sci. Eng.*, vol. 20, no. 6, pp. 401–410, 2005.

[5] M. Poser, "Nesting groups in Active Directory." https://activedirectoryfaq.com/2017/10/nesting-groups-in-active-directory/.

[6] M. Soria-Machado, D. Abolins, C. Boldea, and K. Socha, "CERT-EU Security Whitepaper 17-002 Detecting Lateral Movements in Windows Infrastructure," pp. 0–21, 2017, [Online]. Available: http://technet.microsoft.com/en-us/library/hh994565%28v=ws.10%29.aspx.

[7] T. Carrigan, "Managing local group accounts in Linux," *Redhat*. https://www.redhat.com/sysadmin/local-group-accounts.

[8] D. C. Staff, "Public Cloud Runs on Linux," 2017. https://www.developer.com/news/90-of-the-public-cloud-runs-on-linux/.

[9] Microsoft, "Privileged Accounts and Groups in Active Directory." https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/appendix-b--privileged-accounts-and-groups-in-active-directory.

[10] I. Herman, G. Melançon, and M. S. Marshall, "Graph visualization and navigation in information visualization: a survey," *IEEE Trans. Vis. Comput. Graph.*, vol. 6, no. 1, pp. 24–43, 2000, DOI: 10.1109/2945.841119.

[11] Fisher P, "Understanding Privileged Access Management," 2019, [Online]. Available: https://plus.kuppingercole.com/article/wp80302/understanding-privileged-access-management/.

[12] Raffaella, Sadun, Y. David, and M. Eiran, "CyberArk: Protecting the Keys to the IT Kingdom," 2018, [Online]. Available: https://www.hbs.edu/faculty/Pages/item.aspx?num=53226.

[13] "PowerShell - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Microsoft_PowerShell.

[14] "MySQL - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/My_SQL.

- *Wasef Anwar is currently pursuing master's in science degree program in Cybersecyruty in REVA University, Bengaluru, Karanataka, India, PH-+ 91 9804330523. E-mail: wasef.anwar@gmail.com*

IJSER