## Identity vs. Account

In our experience organizations typically fail to draw a distinction between users of accounts and the accounts themselves. This can often lead to disastrous consequences such as making false assumptions that an account can only be used by certain people thereby forgoing necessary controls or falsely accusing an employee of wrongdoing if their account is used in a breach. In order for a PAS program to be effective this concept must be clearly understood by all stakeholders so that necessary steps are taken in the implementation phase to apply appropriate controls. In short, the differences are as follows:

* **An Identity is a flesh and blood human being.** Employees are identities. Contractors are identities. Malicious attackers are identities. Some of these identities are trusted and others are untrusted.
* **An Account is a virtual entity or tool with pre-defined privileges that can be leveraged by an identity to complete a task.** An account could be used by any number of identities and simply learning that a particular account was used to perform a task does not necessarily give us any information on what identity was using the account at that time.

Based on these definitions it is clear that organizations must find a way to ensure that only *trusted* or *authorized* identities have access to our most sensitive accounts. Simply naming an account ‘Identity1’ does not mean that Identity1 is the only person that can use it (This topic will be expanded on in the [Role Based vs. Individual Accounts](#_Role_Based_vs.) section).

One of the strongest ways to actually ensure that only specific identities have access to privileged accounts is by using two-factor (2FA) authentication. By forcing users to authenticate with something they have in addition to something they know impersonation or theft by untrusted identities is much more difficult. Enterprise wide implementation of 2FA however is often impractical or extremely challenging. Not all platforms support 2FA, many solutions require the management and compatibility of agents, and there is always the issue of shared accounts such as service IDs or built-in administrators.

To deal with these challenges, a much more scalable option that organizations are embracing is leveraging CyberArk as the gateway to endpoints. If we restrict one’s ability to access an endpoint directly by taking away access to the target credentials, users will have no choice but to authenticate via CyberArk (where 2FA should be applied) and then check out the session to the target account. Effectively this creates a ‘2FA for all’ paradigm that is universally supported and requires virtually no modification to the endpoint system itself.

## Role Based vs. Individual Accounts

Individual accounts, commonly referred to as ‘personal privileged accounts’ or ‘named administrator accounts’, became widely adopted in corporate IT largely in response to regulatory compliance dictating that actions must be traceable to an individual and that the use of shared accounts should be discouraged. By assigning a person’s name to an account, organizations felt that they could know with absolute certainty who was behind the actions taken with that particular account.

In the wake of many high profile breaches in the early to mid 2010s this notion of being able to directly attribute actions to a human identity simply because of the account name has proven to be obsolete and unfounded. To understand why, we must first understand the difference between an Identity and an Account as explained earlier in the [Identity vs. Account](#_Key_Concept:_Identity) section.

With that understanding in hand it is important to now point out that there are a vast array of tools and techniques that attackers can use to compromise an account ranging from social engineering techniques like phishing to more technical vectors like keyloggers or RAM scrapers. Given the relative ease with which attackers are seen stealing credentials it is virtually irrelevant what an account is named for purposes of understanding which identity is actually using that account.

Considering the general irrelevance of the account name itself, many organizations are now looking to leverage shared role based accounts as they provide many efficiencies over individual accounts when used with CyberArk. Role based accounts are generic IDs that have been provisioned with a set of privileges necessary for a given role in the organization (e.g. Exchange Administrators, SQL Server Administrators, Linux Operators, etc.) that can be used by multiple identities. In the past and without CyberArk these types of accounts would have been discouraged because it was impossible to tell who was using a given account. As discussed previously however, personal privileged accounts also do not solve this problem. By placing these types of role based accounts in a 2FA-enabled CyberArk Vault however, we now know exactly which *identity* used a given role based account, when they used it, and even what they did with it. By solving the identity problem with role based accounts we also gain the following benefits:

* A potential decrease in the total # of privileged accounts in an organization
* A greatly simplified identity management process
  + New hires, terminations, and transfers no longer require creating or tracking down/removing multitudes of individual accounts, but rather simply adding or removing access to the AD groups that enabe access to CyberArk is sufficient.
* A reduction in the scope of influence that any given account has resulting in reduced risk and the establishment of [Credential Boundaries](#_Credential_Boundaries)
* Simplified management in CyberArk

Role based accounts are not perfect however and there will be tradeoffs which is why this is a key concept that should be discussed early on in PAS program building so that the organization has a clear policy on which path will be adopted. To help in this decision we’ve listed some high level pros/cons for each side:

|  |  |
| --- | --- |
| Role Based Accounts | Individual Accounts |
| **+** Decrease privileged account footprint  **+** Simplified identity management  **+** Reduced scope of influence of accounts  **+** Easier to manage in CyberArk  **-** Changes end user workflow  **-** CyberArk, not the endpoint logs, is the system of record  **-** Upfront work to change existing access model  **-** Need to plan for concurrent use | **+** No changes to existing user workflows  **+** Can still see names in endpoint logs  **+** No concerns around concurrent use  **-** Extra overhead to manage new/old accounts  **-** May not be possible to implement in CyberArk with large user populations without performance issues  **-** Simply seeing names in endpoint logs doesn’t guarantee anything  **-** Harder to manage in CyberArk (1 safe per user) |

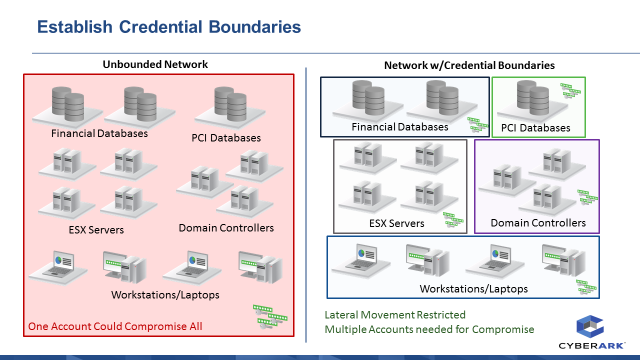
**We strongly recommend role based accounts**, regardless of the tradeoffs. Upon closer examination of the cons for role based accounts the majority of the issues can be categorized as resistance to change and the pros of individual accounts are basically keeping the status quo. We believe the that added efficiencies of role based accounts and potential security gains outweigh any cultural concerns and user behavior can be modified over time. Just because something is the “way its always been done” doesn’t make it the right way.

## Credential Boundaries

When evaluating the risk of a given privileged account one of the main areas of consideration is the account’s *scope of influence*. Simply put, if a single account can influence or access a large number of machines with elevated rights then that account typically represents a greater risk to the organization than one that is restricted to a single device or a small subset. Accounts with a large scope of influence make for easy lateral movement and basically grant attackers a ‘one-stop-shop’ for being able to perform reconnaissance and exfiltrate data.

To restrict an attacker’s ability to move laterally we suggest organizations consider implementing credential boundaries in addition to simply managing existing accounts. This idea of credential boundaries is not unique to CyberArk and other organizations such as Microsoft[[1]](#footnote-1) propose this same idea. Basically much like a firewall can be configured to segment an organization’s network into multiple areas of sensitivity (e.g. PCI Island, ICS Electronic Security Perimeter, DMZ, etc.) so too should we configure our accounts to isolate more sensitive assets from less sensitive ones.

The following example paints a picture of how this concept can help secure an organization. We often find that organizations utilize overpowered accounts that have access to multiple asset classes and create an unbounded network. In an unbounded network a single credential, or a small handful, could be used to compromise a significant portion of the network via easy lateral movement. In a network with credential boundaries in place lateral movement is restricted because no account has access to assets outside of its intended purpose:



As we can see here the idea is that no single account can be used to access workstations, servers, databases and domain controllers but rather at minimum five separate accounts would be required: one with access to just workstations, one with access to just ESX servers, one with access to financial DBs, one with access to PCI DBs, and one with access to just DCs. From an attacker’s perspective it just became much harder to get to a high value asset like a DC since the initial point of ingress is typically a workstation which now will not contain any traces of an account that can access that DC.

This same concept can be applied to all platforms but is especially true at the OS level (e.g. Windows, Unix, Linux, etc.). The fewer assets a single account can access with privileges the better off an organization will be. Controlled use of local administrative accounts can also be an attractive option since if compromised then by definition that account would only be able to access that one system, not every device in the network.

Without a PAS solution like CyberArk keeping track of dozens of credential boundary accounts could certainly be unwieldly and impractical but with a central repository and the ability to easily search for, find, and use an account your trusted identities can still easily traverse the tiers…it is just the attackers that cannot.

1. Mitigating Pass the Hash Attacks and Other Credential Theft Version 2, Microsoft [↑](#footnote-ref-1)