feature

Finding the Dirt in the Windows Control Environment

Vincent Kha, CISM, GPEN, MCTS, OWSP
Is an IT security analyst for a Dutch retail bank where he is responsible for vulnerability management and security event monitoring.

For many information systems auditors, reviewing domain accounts in an Active Directory (AD) environment is sufficient for testing controls around user authentication. However, there is much more to be reviewed in order to gain comfort over the security of business and IT applications that rely on shared Windows and AD infrastructure for authentication services. The good news is that it is very easy to extract data using simple built-in remote administration tools in the Windows world. There is no dark magic in this space.

This article illustrates some further considerations when auditing access control in Windows and AD environments.

The More Data, the Better

Can stakeholders be certain that the data they are given to review form a true representation of the environment? Have stakeholders simply asked for a list of AD accounts and hoped that the administrator is truthfully generating this list? If a request is made for user accounts, the administrator may take that request in the literal sense and include only those accounts in user organisational units (OU). This could potentially omit service or nonuser-based accounts, computer accounts or user accounts that have been placed in incorrect OUs. Stakeholders must be specific in asking for what they want or must supervise the generation of the data. Microsoft's Comma Separated Value Data Exchange (CSVDE) tool[1] is an example of a built-in tool that can be used to extract the entire AD database.

Scrutinise the Existence of Service Accounts

Service accounts (also referred to as generic or system accounts) are plentiful in any Windows environment. The auditor should question how these passwords are managed. Adequate management, at minimum, includes secure storage (e.g., in a password vault), complexity of password composition, as well as password reset frequency (pwLastSet). When was the last time anyone logged into these service accounts (lastLogonTimestamp)? Does the account belong to infrastructure or applications that have long been removed from the environment?

Consideration should be given to the purpose of the account and whether it needs interactive logon rights to servers. If an account needs to exist solely for the purpose of running a scheduled job, it may require only the 'Logon as a batch job'[2] permission, as opposed to full administrator rights on the host. Reducing permissions on service accounts mitigates the risk of inappropriate use of such accounts.

Rather than demanding that these accounts be removed immediately (and causing frustration along the way), a phased approach may be considered. For example, disabling accounts temporarily and monitoring for after-effects or reviewing local and AD audit logs for the existence of account logon events are good measures to assist in slowly phasing out service accounts.

"Disabling accounts temporarily and monitoring for after-effects or reviewing local and AD audit logs for the existence of account logon events are good measures to assist in slowly phasing out service accounts."

Evaluate the Effectiveness of Account Separation Controls

Many organisations implement a separation of accounts for IT administrators by creating a privileged account (such as a domain administrator), and a nonprivileged account, the intention being to limit accidental changes to the environment and mitigate against malware infections. Per Microsoft's best practices on securing AD groups and accounts: '...create two accounts: one regular user account to be used for normal tasks and data administration, and one service administrative account to be used only for performing service administration tasks'.[3]

However, it is useful to consider whether the 'unprivileged' account is truly unprivileged. What AD groups are they members of and what access does that encompass? It might mean that stakeholders need to review the configuration of the source systems that rely on AD for authentication to identify what permission certain groups have. Just because a group is named in such a way that implies the access granted is not sensitive or risky, that does not necessarily make it the case (**figure 1**). It is worth making the effort to identify the true purpose of groups.

Ideally, the default password policy should not be the same for both unprivileged and privileged collection of accounts. Otherwise, what is to stop the administrator from using the same password for both accounts? Doing so defeats the purpose of the separation if the reason for doing so was to mitigate against privilege escalation risk (e.g., if one has the unprivileged account password, one becomes privileged as the passwords to both accounts are the same).

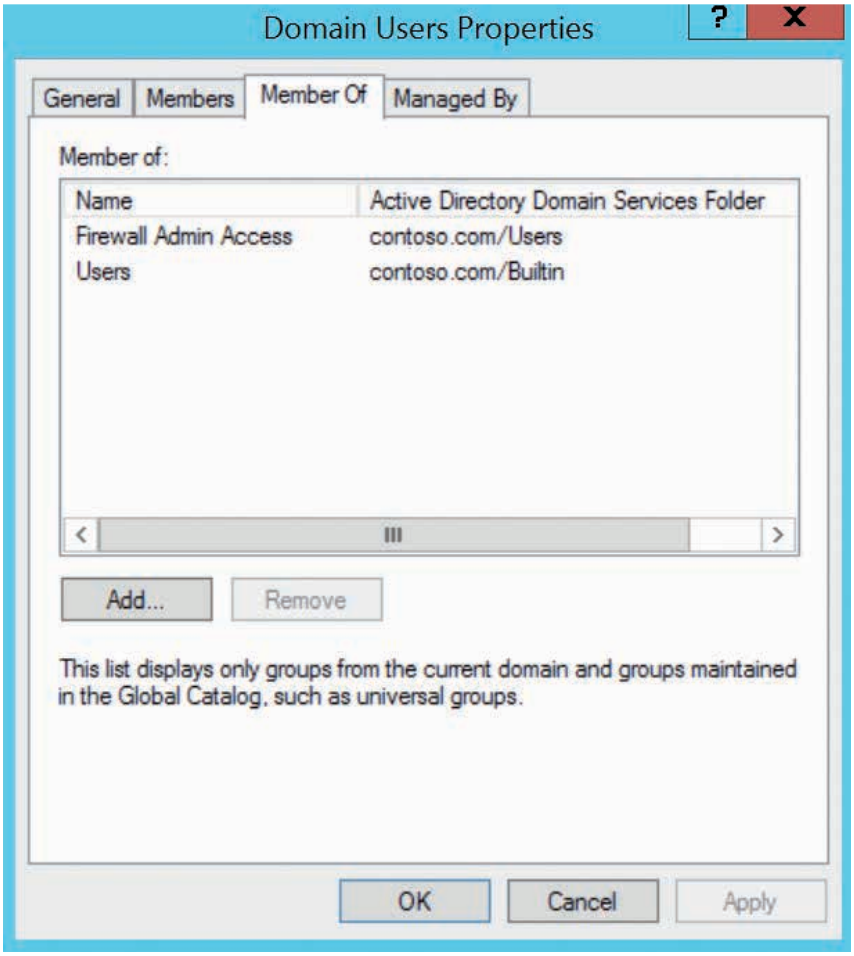
Remember That Access Can Be Provisioned in Multiple Ways

Access is not only granted via AD groups (only in a perfect world, unfortunately) despite what is said. Access can also be granted through other means, including:

- Built-in group memberships locally on a server
- Group policy objects (GPO) via delegation
- Inheritance from nested AD groups

It may be helpful to include a review of at least built-in administrator, power users and remote desktop user membership on all Windows servers to cover the risk of inappropriate remote access to Windows servers. AD groups should be scrutinised to identify where AD groups are used for multiple purposes (including uses for which they were never designed). For a more in-depth look, a review is advised on what AD permissions have been granted to individual user accounts, OUs or AD groups that would not appear if the review focused purely on AD group membership. PowerShell and pivot tables in Excel are excellent tools in these cases. An excellent

Figure 1—An Extreme Example of a Seemingly Harmless Group With Firewall Admin Access



Source: Vincent Kha. Reprinted with permission.

PowerShell script from Microsoft[4] can be used to extract all permissions granted to OUs (**figure 2**).

Ensure the Integrity of Trust Domains or Security Zones

This sounds more complicated than it really is, quite frankly. Some environments are designed with separate domains that have varying trust relationships with one another. For instance, there may be a domain for out-of-band management access to the core infrastructure and a regular domain for the entire company in which to sit.

Despite the best intentions, an environment may contain accounts in a less trusted domain being granted permissions in a more trusted domain. Fortunately, this one is easy to fix by simply looking for any accounts in one domain that do not belong. Again, PowerShell and CSVDE are great tools in this case.

Workgroup Servers Are in Scope as Well

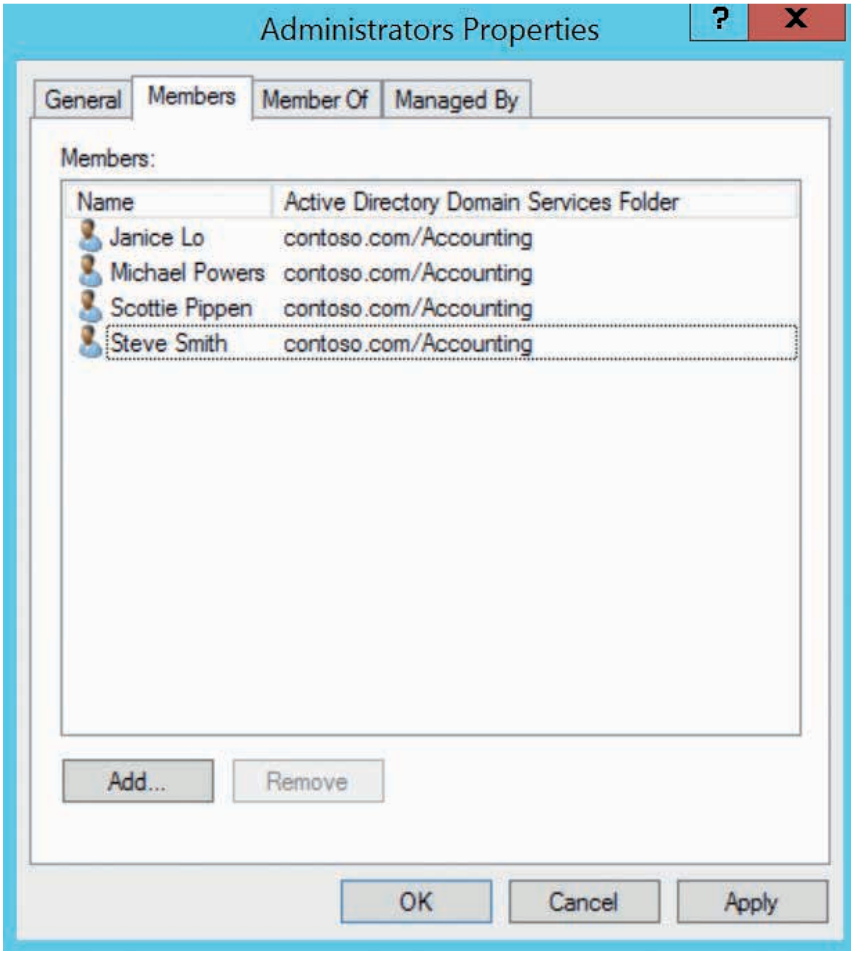
Most of the time, every computer in the environment is a domain member. However, one may come across workgroup computers from time to time. Workgroup computers are often left in isolation and may, therefore, be excluded from general security practices. The important thing to note about workgroup computers is that '...each computer has a set of user accounts. To use any computer in the workgroup, you must have an account on that computer'.[5] This means a regular review over domain accounts will not cover accounts that reside on a workgroup computer.

If the environment contains any workgroup servers, it is important to review the local accounts and local group membership on these servers. Depending on the number of servers and their configuration, it may be easier to simply walk through each server using the Microsoft Management Console (MMC) with the local users snap-in and review the existence of accounts manually.

Conclusion

Gaining comfort over the security of business and IT applications that rely on shared Windows and AD infrastructure for authentication services requires more than reviewing domain accounts in an AD environment. Performing effective reviews begins with collecting the right data in the right amount and scrutinising for the existence, purpose and assigned access for accounts.

Figure 2—Example of Non-IT Departmental Groups With Administrative Level Access



Source: Vincent Kha. Reprinted with permission.

Trust domains or security zones are generally set up with differing levels of permission. It is important to ensure that accounts are assigned to the correct domains to avoid providing accounts more extensive access than is appropriate or secure. Workgroup computers cannot be eliminated from regular review just because they tend to be isolated to specific projects.

These activities are not magic. They do, however, require consistent, focused effort—effort that is highly worthwhile in securing enterprise systems.

Endnotes

1 Microsoft, Csvde, TechNet Library, 17 April 2012, *https://technet.microsoft.com/en-au/library/cc732101.aspx*

2 Microsoft, 'Log on as a BatchJob', TechNet Library, 8 May 2013, *https://technet.microsoft.com/en-au/library/dn221944.aspx*

3 Microsoft, 'Securing Active Directory Administrative Groups and Accounts', TechNet Library, *https://technet.microsoft.com/en-us/library/cc700835.aspx*

4 McGlone, A.; 'Active Directory OU Permissions Report: Free PowerShell Script Download', TechNet.com blog, 25 March, 2013, *http://blogs.technet.com/b/ashleymcglone/archive/2013/03/25/active-directory-ou-permissions-report-free-powershell-script-download.aspx*

5 Microsoft, 'What Is the Difference Between a Domain and a Workgroup', TechNet Library, *http://windows.microsoft.com/en-au/windows-vista/what-is-the-difference-between-a-domain-and-a-workgroup*