Listar las cuentas de usuarios inactivos en AD DS

Problemática

La constante es bastante similar a las cuentas de equipos: los administradores están siempre al corriente cuando se trata de crear una cuenta de usuario pero nunca cuando se trata de eliminarla. Esto entraña necesariamente unas desviaciones que pueden terminar siendo costosas. En efecto, el número de objetos en el servicio de directorio Active Directory no cesa de crecer y los recursos permanecen monopolizados para nada (espacio de disco donde se alojan los «home directory», buzones de correo, etc.).

Por otro lado, una mala gestión de las cuentas de usuarios puede causar problemas de seguridad ya que, como sabemos, una contraseña que no cambia nunca puede «petarse» fácilmente...

Una única solución: ¡Hacer limpieza!

¡Hacer limpieza, la intención es loable, pero sin un script difícil lo tendremos! En el estudio del caso anterior, hemos focalizado nuestra atención en las cuentas de equipos. Sabemos que la administración de la cuentas de usuarios se realiza sobre el mismo principio, y que nuestras explicaciones respecto de los atributos «LastLogon» y «LastLogonTimeStamp» serán validas. Sabemos que el atributo «LastLogon» no se replica entre los controladores de dominio y que «LastLogonTimeStamp» se replica pero se actualiza cada catorce días. Para simplificarnos la vida, como en el caso de las cuentas de equipos, nos contentaremos con usar «LastLogonTimeStamp»; podremos tener por tanto, en el peor de los casos, una diferencia máxima de catorce días en relación con el dato real. Aunque, ¿será verdaderamente importante?

«LastLogonTimeStamp» sólo está disponible a partir de un dominio Windows Server 2003. Si su dominio es un dominio Windows 2000 no tendrá otra opción que consultar a cada controlador de dominio y tomar la información mas reciente de «LastLogon». Aunque, con PowerShell, ¡no será una tarea insuperable! Observe que esto es lo que debería hacerse si no desea tener un delta de fecha de catorce días.

El script que vamos desarrollar juntos nos permitirá encontrar las cuentas inactivas desde un cierto número de días. Luego tendrá la libertad de adaptarlo para que se adhiera mejor a sus necesidades. Podrá desactivar las cuentas o, por qué no, eliminarlas (¡pero no sería muy prudente!), o mejor, archivar los datos de los usuarios antes de cualquier otra cosa.

```
# Get-userAccounts.ps1 - v1.0
[datetime]$date = '01/01/1601'

$adsPath = 'LDAP://OU=Usuarios,' + ([ADSI]'').distinguishedName
$objDominio = [ADSI]$adsPath
$objBusqueda = New-Object
                  System.DirectoryServices.DirectorySearcher($objDominio)
$consulta = '(&(objectCategory=person)(objectClass=user))'
$objBusqueda.Filter=$consulta
$cuentas = $objBusqueda.FindAll()

$cuentas |
  select-object @{e={$_.properties.cn};n='Nombre común'},
                @{e={$_.properties.whencreated};n='Fecha de creación'},
                @{e={$_.properties.homedrive};n='HD'},
                @{e={$_.properties.homedirectory};n='HomeDirectory'},
                @{e={$date.AddTicks($($_.properties.lastlogontimestamp))};
                   n='Última conexión'}
```

Resultado:

```
./Get-userAccounts.ps1 | Format-Table

Nombre común   Fecha de creación    HomeDirectory       Última conexión
----------     -----------------    -------------       ---------------
Jose Bar       12/01/2007 10:52:33   F: \\Srv3\JoseB     14/01/2007 14:28:05
Eduardo López  24/05/2007 19:53:35   M: \\Srv1\Ed        01/01/1601 00:00:00
Jose Laredo    30/06/2007 21:53:58   E: \\Srv2\JoseL     01/01/1601 00:00:00
Jaime Lafuente 01/07/2007 11:57:26   E: \\Srv2\Jaime     06/10/2007 15:53:38
Juan Carlos
Durante        05/08/2007 22:54:33   M: \\Srv1\Juan      01/01/1601 00:00:00
Pablo Ponce    30/09/2007 23:11:02   M: \\Srv1\Pablo     12/11/2007 18:05:19
```

Observe que hemos aprovechado para mostrar información adicional que puede ser útil, como la fecha de creación, la ruta del «home directory» así como su letra asociada.

Ahora sólo nos queda definir un parámetro que marcará el número de días a partir del cual un usuario verá su cuenta desactivada. Por ejemplo, si se define este número en noventa, desactivaremos las cuentas de los usuarios que no han iniciado sesión durante este período.

He aquí el script modificado:

```
# Get-userAccounts.ps1 - v1.1
param ($NoConectadoDespuesNDias)

[datetime]$date = '01/01/1601'

$adsPath = 'LDAP://OU=Usuarios,' + ([ADSI]'').distinguishedName
$objDominio = [ADSI]$adsPath
$objBusqueda = New-Object
                    System.DirectoryServices.DirectorySearcher($objDominio)
$consulta = '(&(objectCategory=person)(objectClass=user))'
$objBusqueda.Filter=$consulta
$cuentas = $objBusqueda.FindAll()

if ($NoConectadoDespuesNDias -eq $null) {
  $cuentas |
    select-object @{e={$_.properties.cn};n='Nombre común'},
       @{e={$_.properties.whencreated};n='Fecha de creación'},
       @{e={$_.properties.homedrive};n='HD'},
       @{e={$_.properties.homedirectory};n='Home Directory'},
       @{e={$date.AddTicks($($_.properties.lastlogontimestamp))};
         n='Última conexión'}
}
else {
  $cuentas |
    select-object @{e={$_.properties.cn};n='Nombre común'},
       @{e={$_.properties.whencreated};n='Fecha de creación'},
       @{e={$_.properties.homedrive};n='HD'},
       @{e={$_.properties.homedirectory};n='Home Directory'},
       @{e={$date.AddTicks($($_.properties.lastlogontimestamp))};
         n='Última conexión'} |
    Where-object {
     (new-timespan $_.'Última conexión' $(get-date)).days -ge
       $NoConectadoDespuesNDias
    }
}
```

Resultado:

```
./Get-userAccounts.ps1 90 | Format-Table

Nombre común   Fecha de creación      HomeDirectory     Última conexión
----------     -----------------      -------------     ---------------
Jose Bar       12/01/2007 10:52:33    F: \\Srv3\JoseB   14/01/2007 14:28:05
Eduardo López  24/05/2007 19:53:35    M: \\Srv1\Ed      01/01/1601 00:00:00
Jose Laredo    30/06/2007 21:53:58    E: \\Srv2\JoseL   01/01/1601 00:00:00
Juan Carlos
Durante        05/08/2007 22:54:33    M: \\Srv1\Juan    01/01/1601 00:00:00
```

Una fecha de la última conexión a «01/01/1601» indica que el usuario no ha abierto sesión nunca. Para más información sobre el atributo «LastLogonTimeStamp» consulte la documentación Microsoft Technet disponible en: http://www.microsoft.com/technet/scriptcenter/topics/win2003/lastlogon.mspx y http://technet.microsoft.com/es-es/library/cc772829(WS.10).aspx

Veamos como quedaría el script si utilizamos los nuevos commandlets proporcionados en el módulo Active Directory de Windows Server 2008 R2:

```
# Get-userAccounts.ps1 - v2.1
param ($NoConectadoDespuesNDias)

[datetime]$date = '01/01/1601'

$cuentas = Get-ADUser -Filter * `
    -properties Name,WhenCreated,HomeDrive,HomeDirectory,LastLogonTimestamp

if ($NoConectadoDespuesNDias -eq $null) {
  $cuentas |
    select-object @{e={$_.Name};n='Nombre común'},
       @{e={$_.whencreated};n='Fecha de creación'},
       @{e={$_.homedrive};n='HD'},
       @{e={$_.homedirectory};n='Home Directory'},
       @{e={$date.AddTicks($($_.lastlogontimestamp))};
         n='Última conexión'}
}
else { $cuentas |
    select-object @{e={$_.Name};n='Nombre común'},
       @{e={$_.whencreated};n='Fecha de creación'},
       @{e={$_.homedrive};n='HD'},
       @{e={$_.homedirectory};n='HomeDirectory'},
       @{e={$date.AddTicks($($_.lastlogontimestamp))};n='Última conexión'} |
    Where-object {
     (new-timespan $_.'Última conexión' $(get-date)).days -ge
       $NoConectadoDespuesNDias
    }
}
```