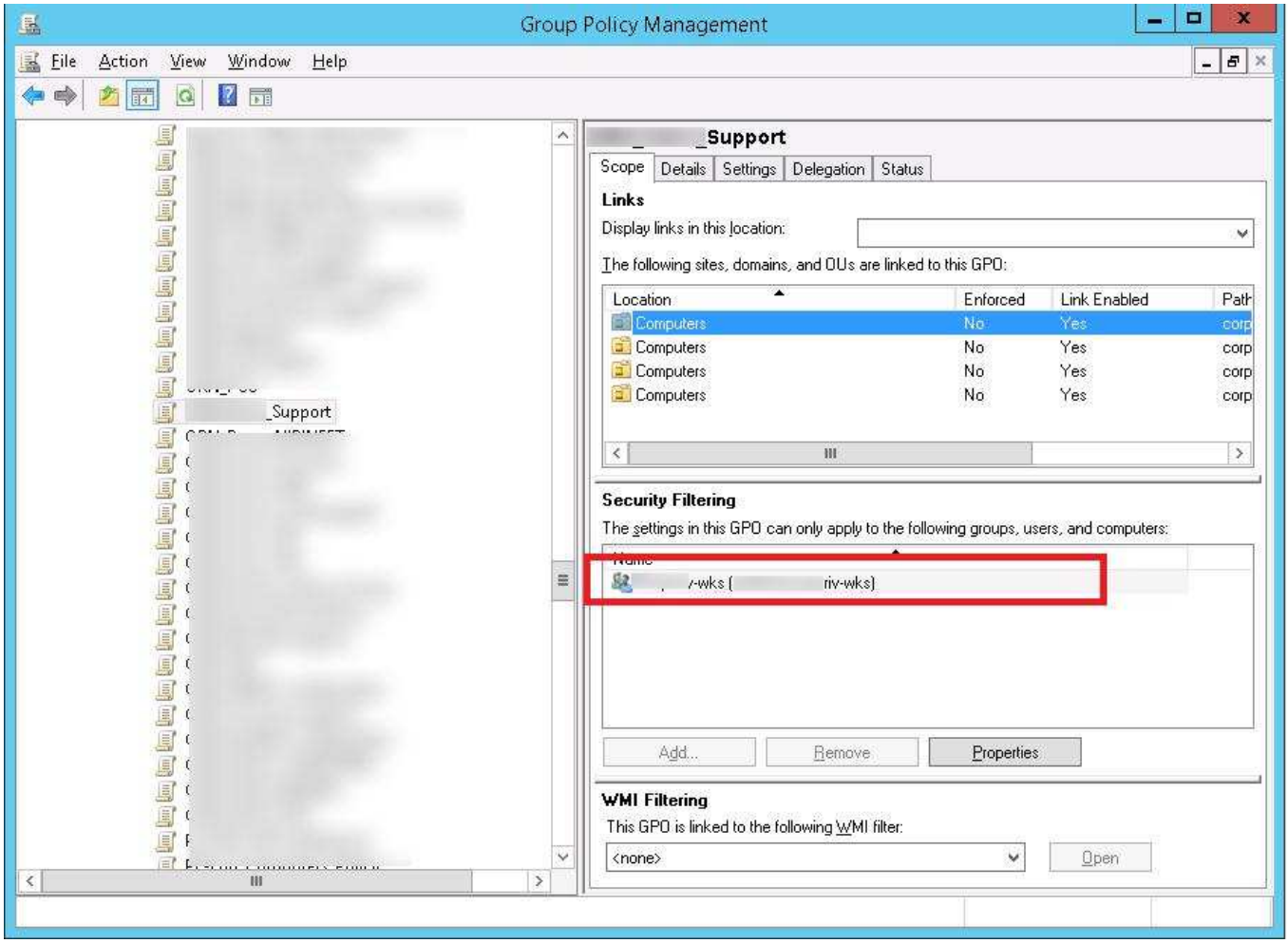
Почему перестали работать некоторые GPO после установки MS16-072

http://winitpro.ru/index.php/2016/06/20/pochemu-perestali-rabotat-nekotorye-gpo-posle-ustanovki-ms16-072/

На прошлой неделе Microsoft выпустила обновления безопасности, меняющих стандартную схему работы механизма применения групповых политик Windows. Речь об обновлениях, выпущенных в рамках бюллетеня MS16-072 от 14 июня 2016 года, который предназначен для устранения уязвимостей в механизме GPO. Разберемся для чего выпущено это обновление и что нужно знать системному администратору об изменениях в применении групповых политик.

Обновления из MS16-072 устраняют уязвимость, позволяющую злоумышленнику реализовать атаку Man in the middle (MiTM), и получить доступ к трафику, передаваемому между компьютером и контроллером домена. Для защиты от уязвимости разработчики MS решили изменить контекст безопасности, в котором получаются политики. Если ранее, пользовательские политики получались в контексте безопасности пользователя, то после установки MS16-072, пользовательские политики получаются в контексте безопасности компьютера.

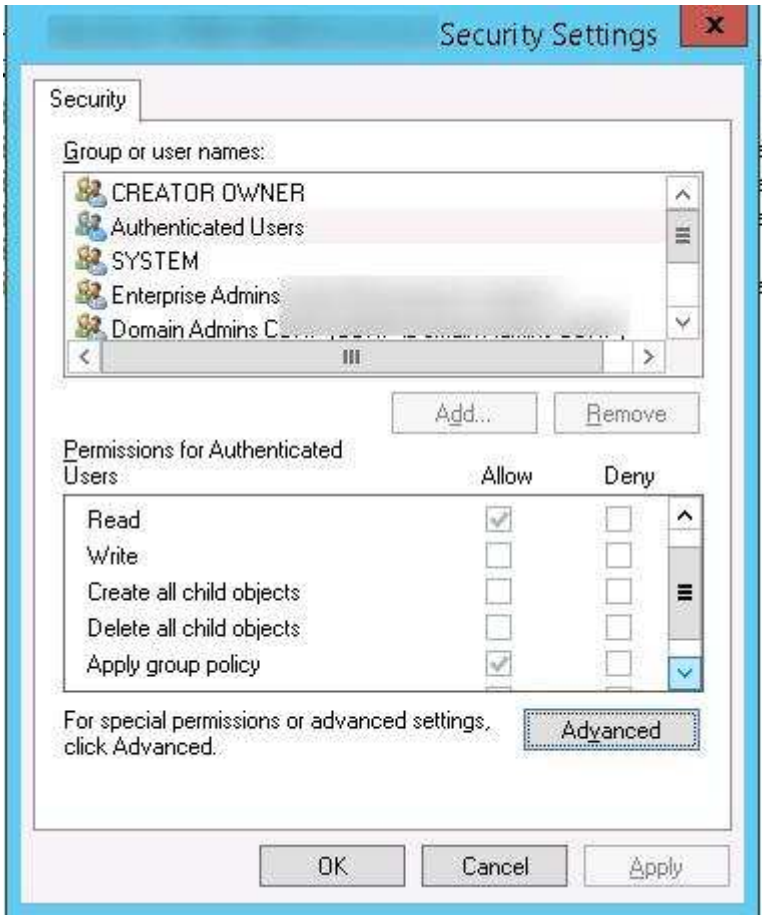
В результате многие пользователи обнаружили, что после установки обновлений из данного бюллетеня, перестали применяться некоторые политики. GPO со стандартными разрешениями, у которых в **Security Filtering** даны права на Read и Apply Group Policy для группы **Authenticated Users**, применяются как обычно. Проблема наблюдается только с политиками, на которых настроена фильтрация безопасности (Security Filtering) и из разрешений которых удалена группа Authenticated Users .



Во всех предыдущих рекомендациях при необходимости использовать Security Filtering MS всегда советовали удалять группу Authenticated Users и добавлять группу безопасности пользователей с правами Read и Apply.

После установки обновления MS16-072 /KB3159398 теперь для успешного применения политики, права Read на доступ к объекту GPO должны быть также и у учетной записи самого компьютера.

А так как под Authenticated Users подразумеваются, как пользовательские, так и компьютерные учетки, то удаляя эту группу, мы тем самым блокируем доступ к GPO.



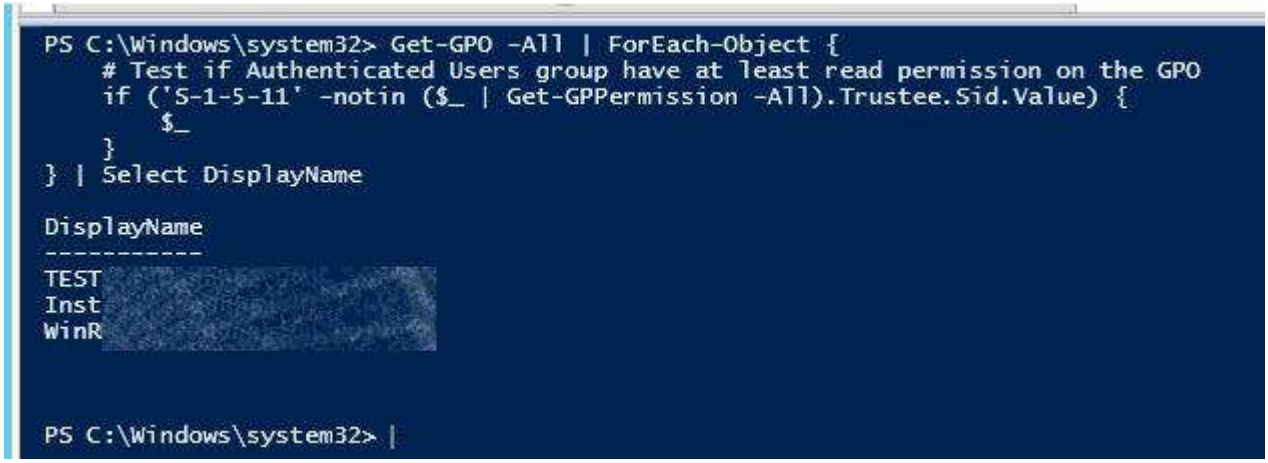
Чтобы решить проблему, нужно удалить обновление (не верный, но действенны способ) или с помощью GPMC.MSC для всех политик, у которых используется фильтрация безопасности по пользовательским группам, добавить в Security Filtering группу **Domain Computers**.

Таким образом, у компьютеров домена появится право на чтение этой политики (нужны права только на Read).

Примечание. Пользовательские группы должны по-прежнему иметь Read и Apply права на политику.

Чтобы найти все объекты GPO в домене, у которых в Security Filtering отсутствует группа Authenticated Users, можно воспользоваться таким скриптом:

```
Get-GPO -All | ForEach-Object {
if ('S-1-5-11' -notin ($_ | Get-GPPermission -All).Trustee.Sid.Value) {
$_
}
} | Select DisplayName
```



Для больших и сложных инфраструктур с запутанной структурой групповых политик для поиска проблемных политик можно воспользоваться более удобным PowerShellскриптом [MS16-072 – Known Issue – Use PowerShell to Check GPOs]

https://blogs.technet.microsoft.com/poshchap/2016/06/16/ms16-072-known-issue-use-powershell-to-check-gpos/

MS16-072 – Known Issue – Use PowerShell to Check GPOs

There is a known issue with the application of particular GPOs once MS16-072 is applied. Click the following link and browse to 'Known Issues' for more information:

MS16-072: Security update for Group Policy: June 14, 2016 (https://support.microsoft.com/en-us/kb/3163622)

In response, I've put together the below PowerShell example to help identify GPOs, from the current domain, that might experience the issue once the update is applied. The output should be the basis for further investigation, i.e. it lists GPOs that may need the 'Authenticated Users' read permission or 'Domain Computers' read permission adding.

In the above image, a red 'WARNING:' message indicates a GPO that may experience the known issue.

There are also three types of 'INFORMATION' message*:

1) yellow – the GPO does not have an 'Authenticated Users' permission, but does contain a 'Domain Computers' permission

2) yellow – the GPO has an 'Authenticated Users' permission that is not 'GpoApply' (Read / Apply) or 'GpoRead' (Read)

3) white – the GPO has the expected 'Authenticated Users' permission.

**NB – all three 'INFORMATION:' messages can be commented out in the script to reduce the output to screen, although the first two may require further investigation*

You should also take a look here:

New Group Policy Patch MS16-072– "Breaks" GP Processing Behavior

---

```
#Load GPO module
Import-Module GroupPolicy

#Get all GPOs in current domain
$GPOs = Get-GPO -All

#Check we have GPOs
if ($GPOs) {
```

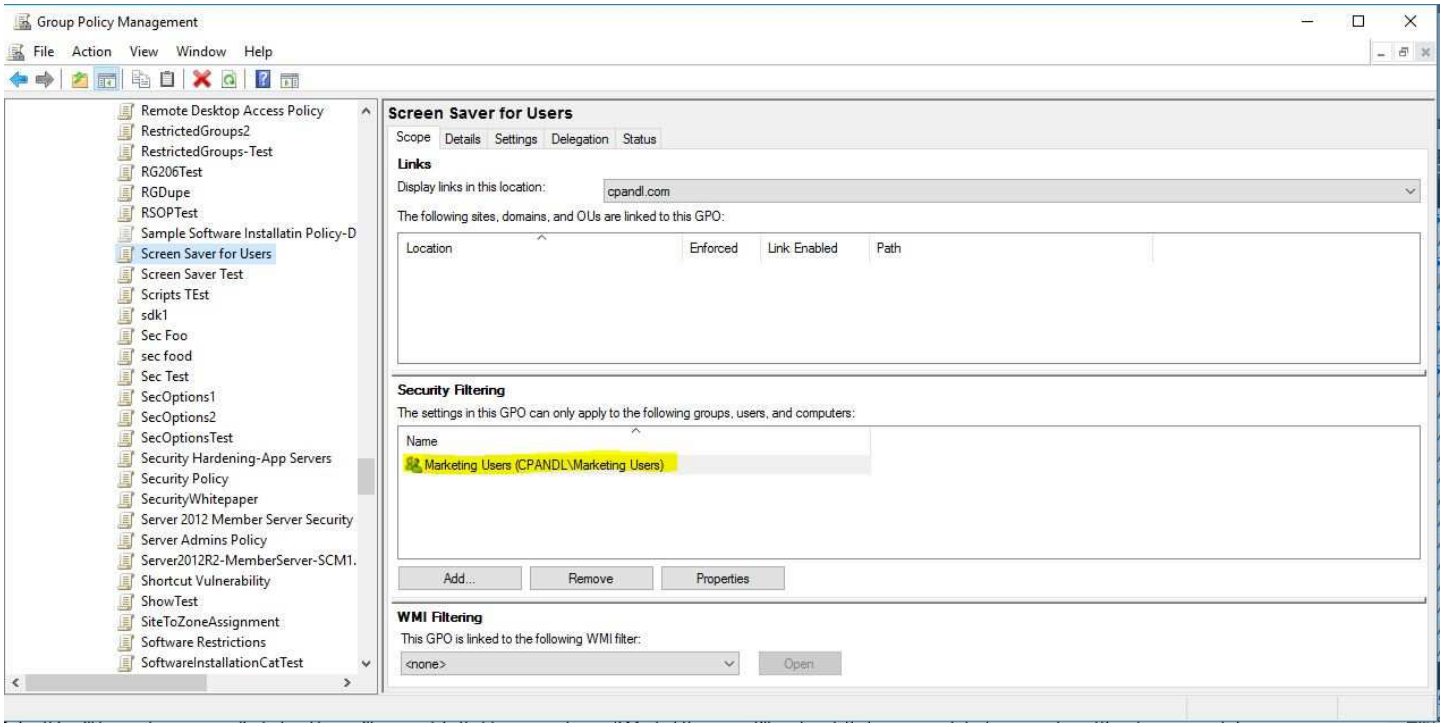
```
#Loop through GPOs
foreach ($GPO in $GPOs) {

#Nullify $AuthUser & $DomComp
$AuthUser = $null
$DomComp = $null

#See if we have an Auth Users perm
$AuthUser = Get-GPPermissions -Guid $GPO.Id -TargetName “Authenticated Users” -TargetType Group -
ErrorAction SilentlyContinue

 #See if we have the ‘Domain Computers perm
$DomComp = Get-GPPermissions -Guid $GPO.Id -TargetName “Domain Computers” -TargetType Group -
ErrorAction SilentlyContinue

 #Alert if we don’t have an ‘Authenticated Users’ permission
if (-not $AuthUser) {

#Now check for ‘Domain Computers’ permission
if (-not $DomComp) {

                Write-Host “WARNING: $($GPO.DisplayName) ($($GPO.Id)) does not have an
‘Authenticated Users’ permission or ‘Domain Computers’ permission – please investigate” -
ForegroundColor Red

}   #end of if (-not $DomComp)
else {

#COMMENT OUT THE BELOW LINE TO REDUCE OUTPUT!

               Write-Host “INFORMATION: $($GPO.DisplayName) ($($GPO.Id)) does not have an
‘Authenticated Users’ permission but does have a ‘Domain Computers’ permission” -ForegroundColor
Yellow

}   #end of else (-not $DomComp)

}   #end of if (-not $AuthUser)
elseif (($AuthUser.Permission -ne “GpoApply”) -and ($AuthUser.Permission -ne “GpoRead”)) {

#COMMENT OUT THE BELOW LINE TO REDUCE OUTPUT!
Write-Host “INFORMATION: $($GPO.DisplayName) ($($GPO.Id)) has an ‘Authenticated Users’ permission
that isn’t ‘GpoApply’ or ‘GpoRead'” -ForegroundColor Yellow

}   #end of elseif (($AuthUser.Permission -ne “GpoApply”) -or ($AuthUser.Permission -ne “GpoRead”))
else {

   #COMMENT OUT THE BELOW LINE TO REDUCE OUTPUT!

            Write-Output “INFORMATION: $($GPO.DisplayName) ($($GPO.Id)) has an ‘Authenticated
Users’ permission”

        }   #end of else (-not $AuthUser)

    }   #end of foreach ($GPO in $GPOs)

}   #end of if ($GPOs)
```

New Group Policy Patch MS16-072– "Breaks" GP Processing Behavior

Written by Darren Mar-Elia Posted on Wednesday, June 15, 2016 15 Comments

https://sdmsoftware.com/group-policy-blog/bugs/new-group-policy-patch-ms16-072-breaks-gp-processing-behavior/

This morning I woke up to an email from a fellow Group Policy MVP–Martin Binder–warning that folks were seeing GP Processing issues after the recent slew of Patch Tuesday updates were applied. Indeed, I had noted late on Tuesday via Twitter that there was a fix to GP in this latest round of patches, that prevents a privilege elevation vulnerability in GP processing. Great! Well, not so great. It turned out that the fix was a bit problematic for folks who had set per-user security group filtering in their GPOs, as shown in the figure below. GPOs set up this way were no longer being processed after the patch was applied to client systems.



A GPO with no Authenticated Users in Security Filtering

Specifically, if you'd set security group filtering for GPOs that contain per-user settings, and you'd **removed** Authenticated Users completely from the GPO's delegation, then GPO processing for per-user settings would fail after applying MS16-072. As the day went on, I mostly ignored this issue, until tonight I read the KB article surrounding this patch in detail. Specifically, there's a section called **Known Issues** where it says the following:

"MS16-072 changes the security context with which user group policies are retrieved. This by-design behavior change protects customers' computers from a security vulnerability. Before MS16-072 is installed, user group policies were retrieved by using the user's security context. After MS16-072 is installed, user group policies are retrieved by using the machines security context"

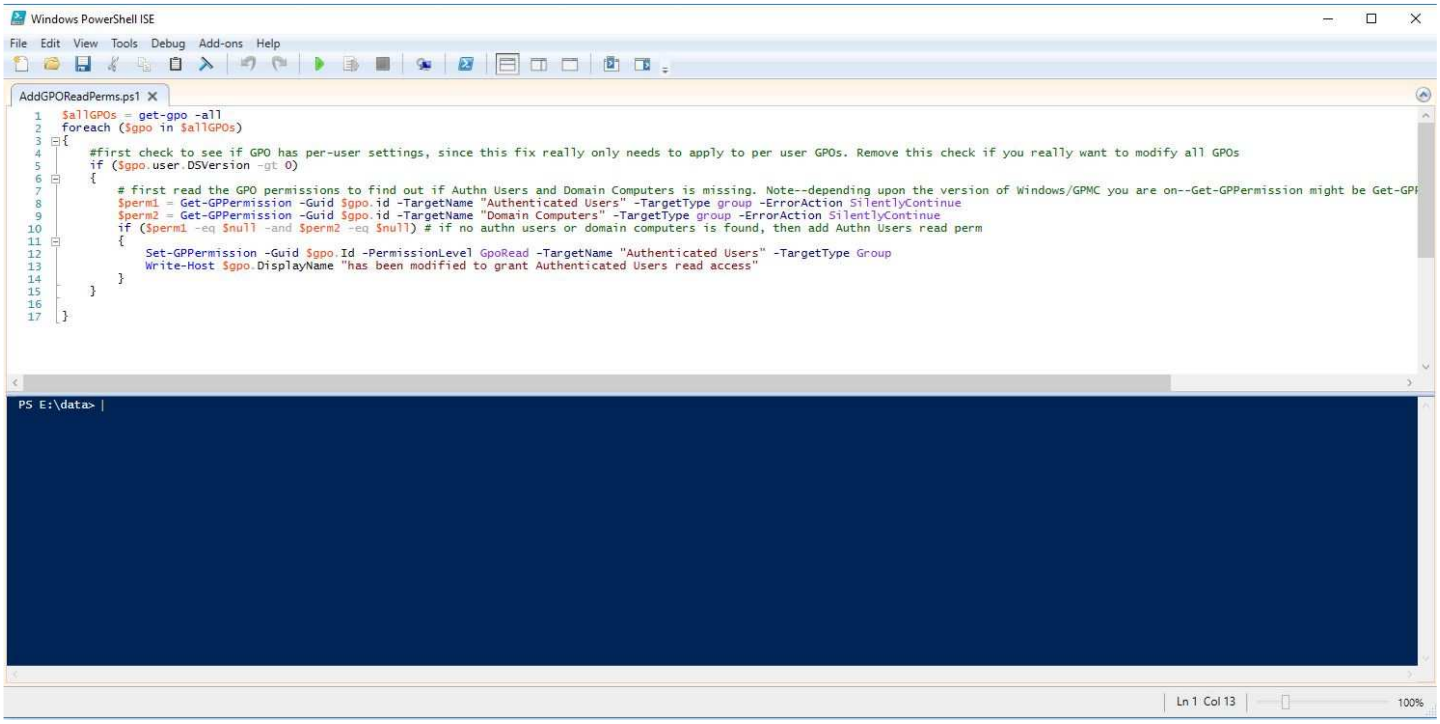
Um….that's big. What it's saying is that per-user GP processing has fundamentally changed. It goes on to further say:

"This issue may occur if the Group Policy Object is missing the Read permissions for the Authenticated Users group or if you are using security filtering and are missing Read permissions for the domain computers group."

Indeed, many people found that by adding back the Authenticated Users Access Control Entry (ACE) to the GPO's delegation with Read access (NOTE: I AM SAYING **READ** ACCESS–THIS IS **DIFFERENT** THAN READ AND "APPLY GROUP POLICY", which will have the affect of nullifying any security group filtering you are using on the GPO) per-user GP processing will go back to working. The above referenced article says that you can add either Authenticated Users or **Domain Computers** with Read access on the GPO to solve this, because the per-user settings are running in the computer's security context, so adding Domain Computers should give the computer the access it needs to continue processing those per-user settings.

Mitigation

OK, again, this is a BIGGGGG change, and I'm sure a lot of folks got broken by this. What I've done is created a quick PowerShell script for those who have a lot of GPOs in your environment and don't want to manually make this change. What the script does is get a list of all of your GPOs in the current domain. It then iterates through them, checks to see if the Authenticated Users or Domain Computers groups are found in the GPO's delegation. If not found, then the script adds the Read (only) permission to the GPO for Authenticated Users. You might decide you'd rather use Domain Computers, because some people have purposefully prevented Authenticated Users from reading their GPOs to prevent unwanted security posture discovery. You can easily modify the script to add Domain Computers instead of Authenticated Users by modifying line 9 of the script. Note that this script needs the Group Policy PowerShell module that is part of GPMC to be installed to function:



GPO Permission script for MS16-072

```
$allGPOs = get-gpo -all
foreach ($gpo in $allGPOs)
{
    #first check to see if GPO has per-user settings, since this fix really only needs to apply to per user GPOs. Remove this check if you really
want to modify all GPOs
    if ($gpo.user.DSVersion -gt 0)
    {
        # first read the GPO permissions to find out if Authn Users and Domain Computers is missing. Note--depending upon the version of Windows/GPMC
you are on--Get-GPPermission might be Get-GPPermissionS
        $perm1 = Get-GPPermission -Guid $gpo.id -TargetName "Authenticated Users" -TargetType group -ErrorAction SilentlyContinue
        $perm2 = Get-GPPermission -Guid $gpo.id -TargetName "Domain Computers" -TargetType group -ErrorAction SilentlyContinue
        if ($perm1 -eq $null -and $perm2 -eq $null) # if no authn users or domain computers is found, then add Authn Users read perm
        {
            Set-GPPermission -Guid $gpo.Id -PermissionLevel GpoRead -TargetName "Authenticated Users" -TargetType Group
            Write-Host $gpo.DisplayName "has been modified to grant Authenticated Users read access"
        }
    }

}
```

PLEASE NOTE: THIS SCRIPT CHANGES PERMISSIONS ON YOUR GPOs. **Test first** in a non-production environment before running it against your live GPOs. It's provided for you as-is, with no warranty!

June 16 Edit: I made a change to the script, to have it check for GPOs that contain user settings, since we're only interested in doing this fix for GPOs with per-user settings. Also note that Microsoft has just released an assessment-only script here.

June 17 Edit: I added a blog post to show how you can modify the default permissions that get stamped on a newly created GPO, to include Domain Computers with Read access

Next Steps

I've been asked if this is a bug that Microsoft will fix. If you read the article I mention above, it sure doesn't seem like they see it as a bug, but rather a change in behavior in the interests of security. I agree that making GP secure is critical to ensuring it can do it's job of, well, securing your Windows systems. I wish they had given a little bit more notice on this so it didn't break people's GP environments, but, hey, at least NOW we know :-).

If you have any feedback or questions on the script, feel free to email us at info@sdmsoftware.com

Darren

> Erik de Vries says:
>
> Changed it a little for to make it work in Server 2008(r2) :

```
$allGPOs = get-gpo –all
foreach ($gpo in $allGPOs)
{
# first read the GPO permissions to find out if Authn Users and Domain Computers is missing
$perm1 = Get-GPPermissions -Guid $gpo.id -TargetName “Authenticated Users” -TargetType group -ErrorAction
SilentlyContinue
$perm2 = Get-GPPermissions -Guid $gpo.id -TargetName “Domain Computers” -TargetType group -ErrorAction
SilentlyContinue
if ($perm1 -eq $null -and $perm2 -eq $null) # if no authn users or domain computers is found, then add Authn
Users read perm
{
Set-GPPermissions -Guid $gpo.Id -PermissionLevel GpoRead -TargetName “Authenticated Users” -TargetType Group
Write-Host $gpo.DisplayName “has been modified to grant Authenticated Users read access”
}

}
```