

Automating system administration tasks – Part2

 michaelfirsov.wordpress.com/automating-system-administration-tasks-part2

September 15, 2017

Part1

II Audit

The next step in my daily monitoring is the reading of the auditing reports – here are the operations that I'd like to be aware of should they ever happen in my network:

I Operations with user/computer accounts:

- 1) Password resets
- 2) User created
- 3) User deleted
- 4) User logon has been denied

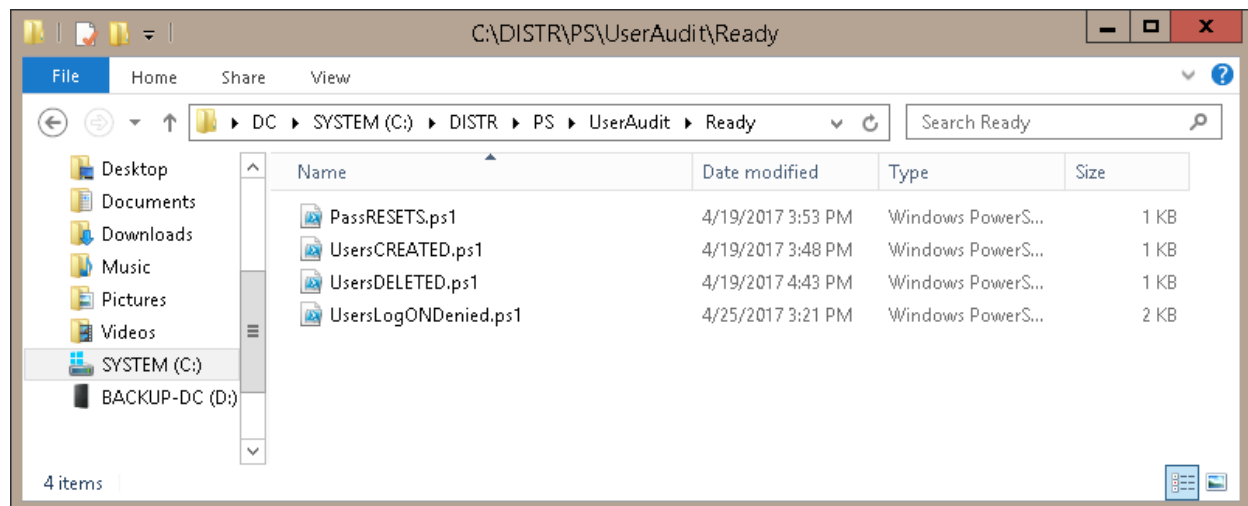
The scripts (please rename the downloaded .docx files to .ps1 files before use):

PassRESETS.ps1 – PassRESETS

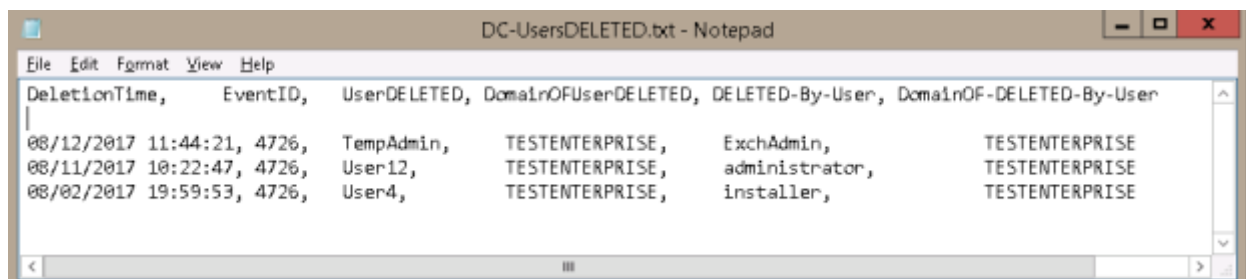
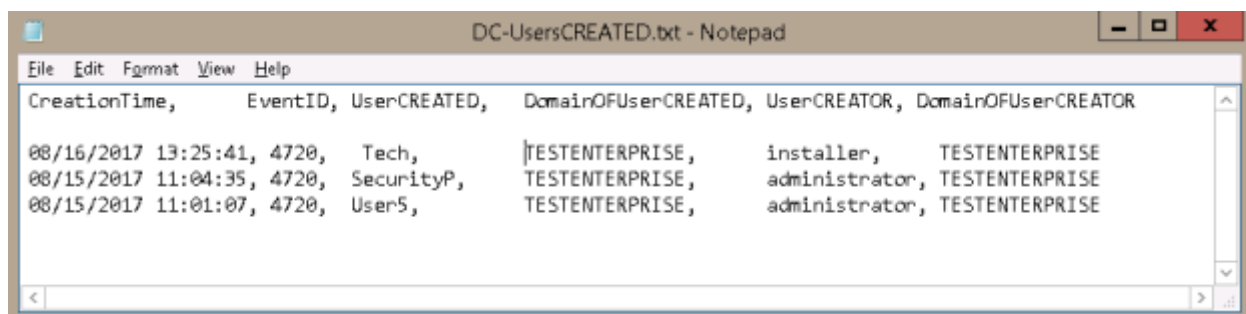
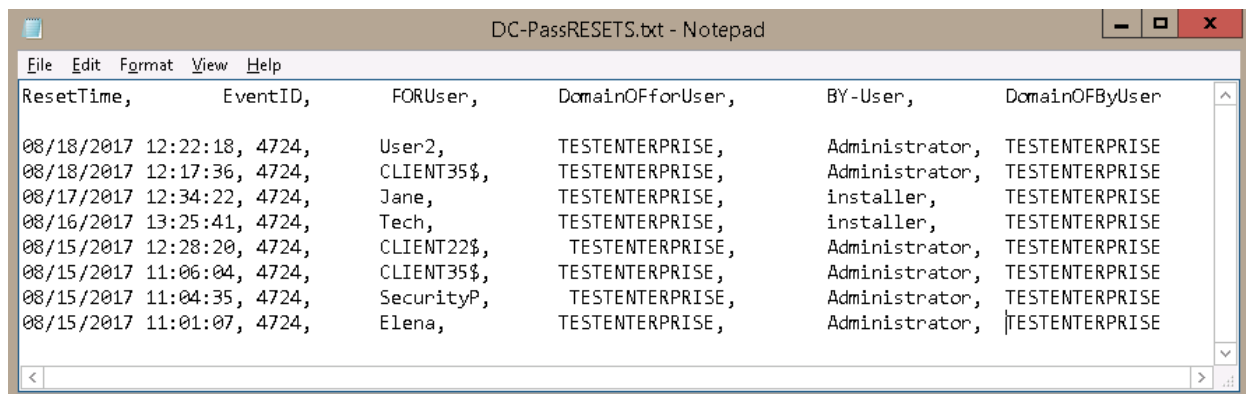
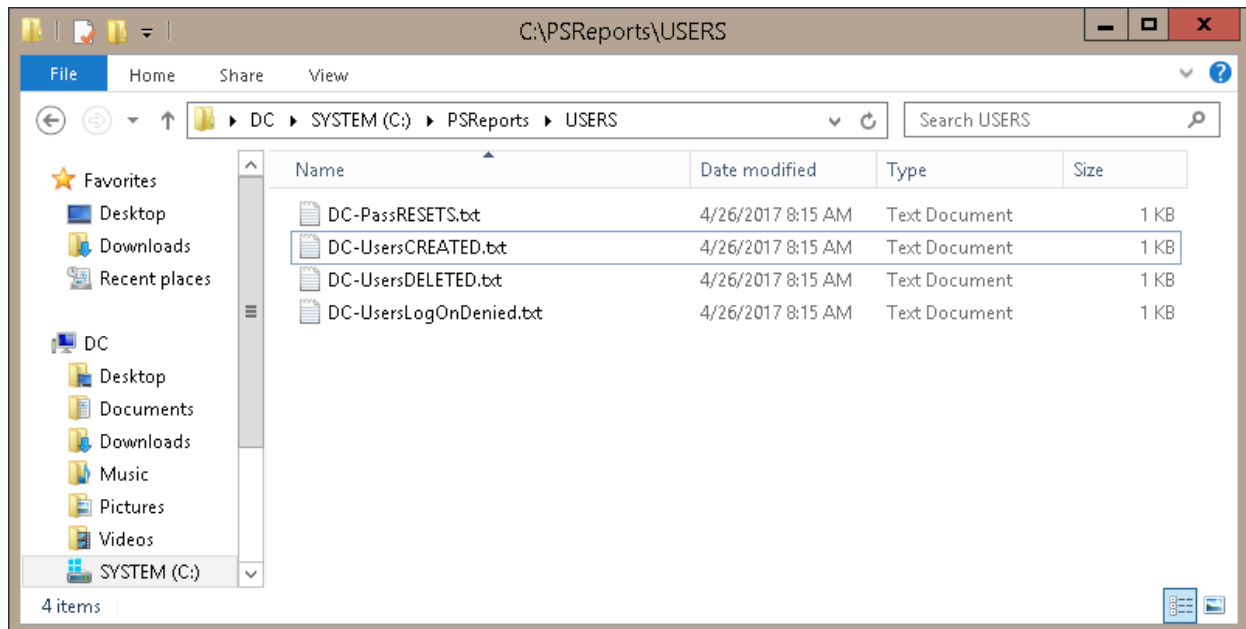
UsersCREATED.ps1 – UsersCREATED

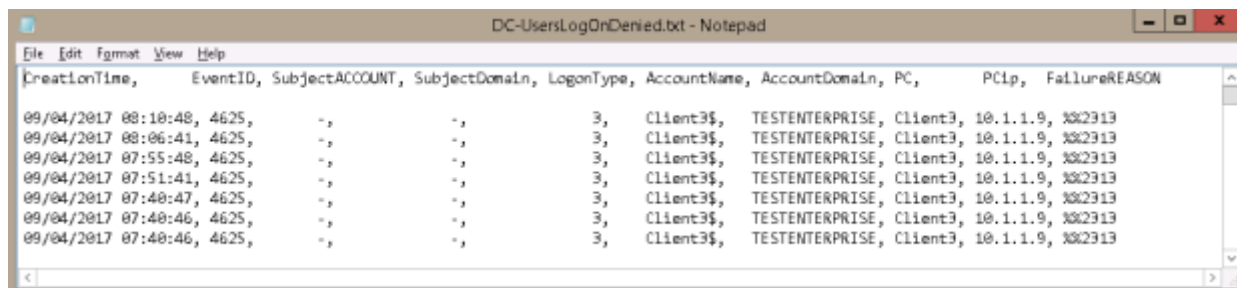
UsersDELETED.ps1 – UsersDELETED

UsersLogONDenied.ps1 – UsersLogONDenied



The example reports:



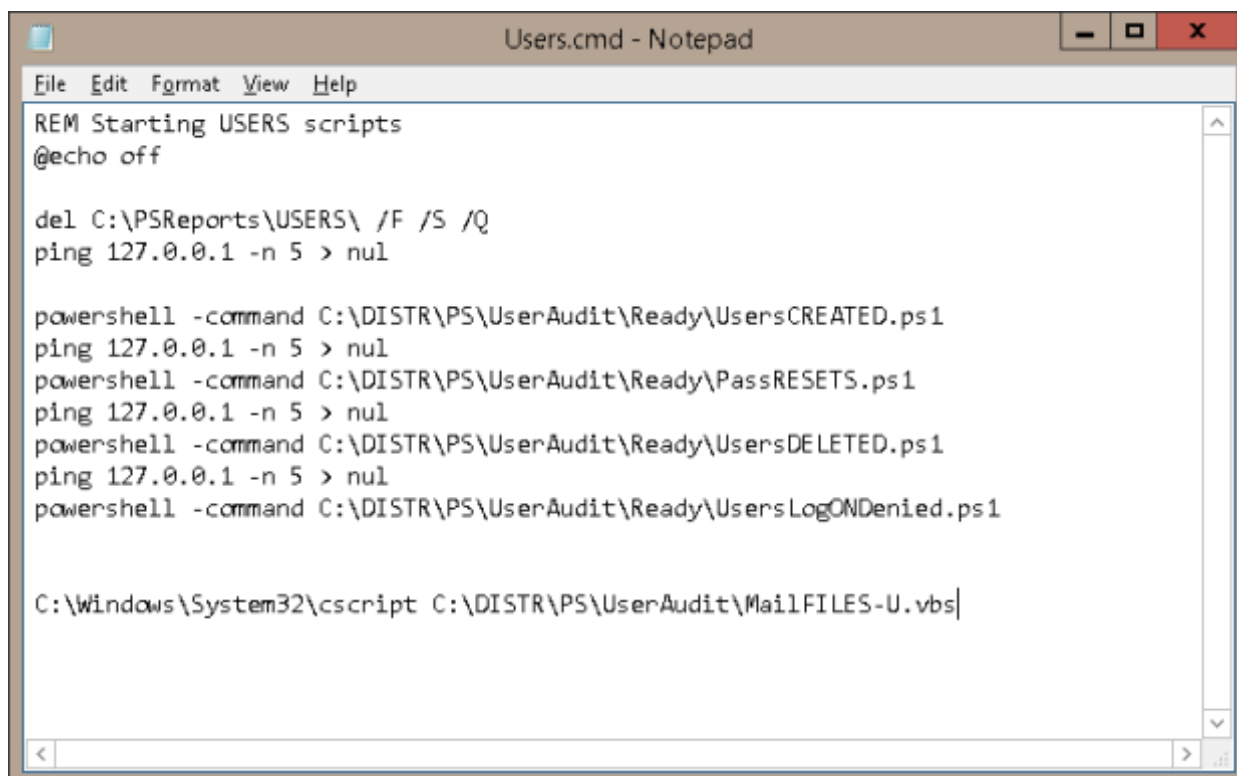


CreationTime,	EventID,	SubjectACCOUNT,	SubjectDomain,	LogonType,	AccountName,	AccountDomain,	PC,	PCIp,	FailureREASON
09/04/2017 08:10:48,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 08:06:41,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 07:55:48,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 07:51:41,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 07:48:47,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 07:48:46,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313
09/04/2017 07:48:46,	4625,	-,	-,	3,	Client3\$,	TESTENTERPRISE,	Client3,	10.1.1.9,	%2313

FailureREASON codes (as found on Internet):

- %%**2305** The specified user account has expired. (532)
- %%**2309** The specified account's password has expired. (535)
- %%**2310** Account currently disabled. (531)
- %%**2311** Account logon time restriction violation. (530)
- %%**2312** User not allowed to logon at this computer. (533)
- %%**2313** Unknown user name or bad password. (529)

All these *.ps1* scripts are run by the single *.cmd* script **Users.cmd – Users** :



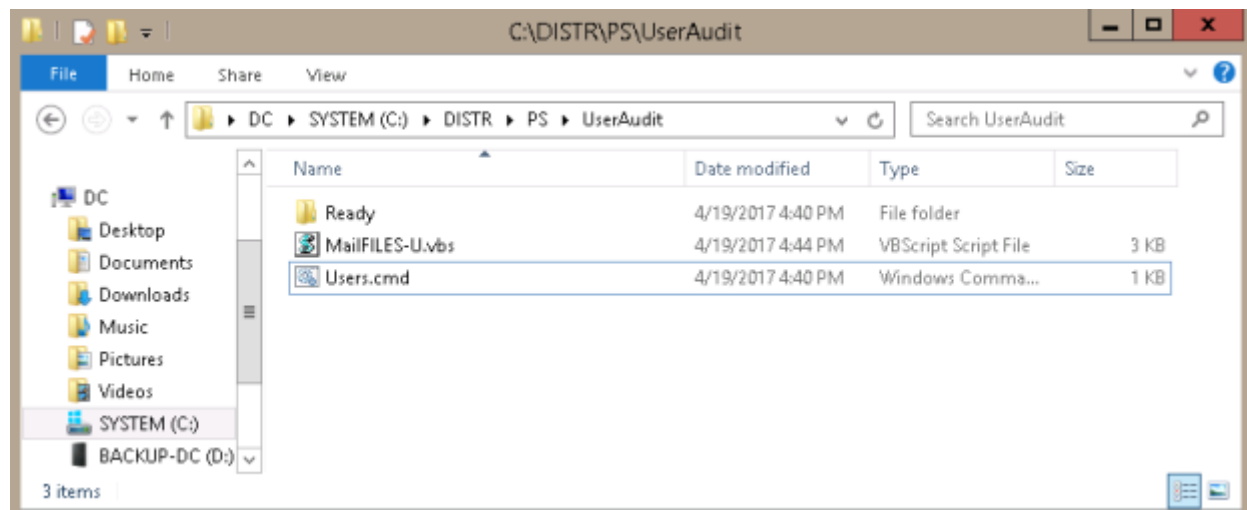
```

REM Starting USERS scripts
@echo off

del C:\PSReports\USERS\ /F /S /Q
ping 127.0.0.1 -n 5 > nul

powershell -command C:\DISTR\PS\UserAudit\Ready\UsersCREATED.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\UserAudit\Ready\PassRESETS.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\UserAudit\Ready\UsersDELETED.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\UserAudit\Ready\UsersLogONDenied.ps1

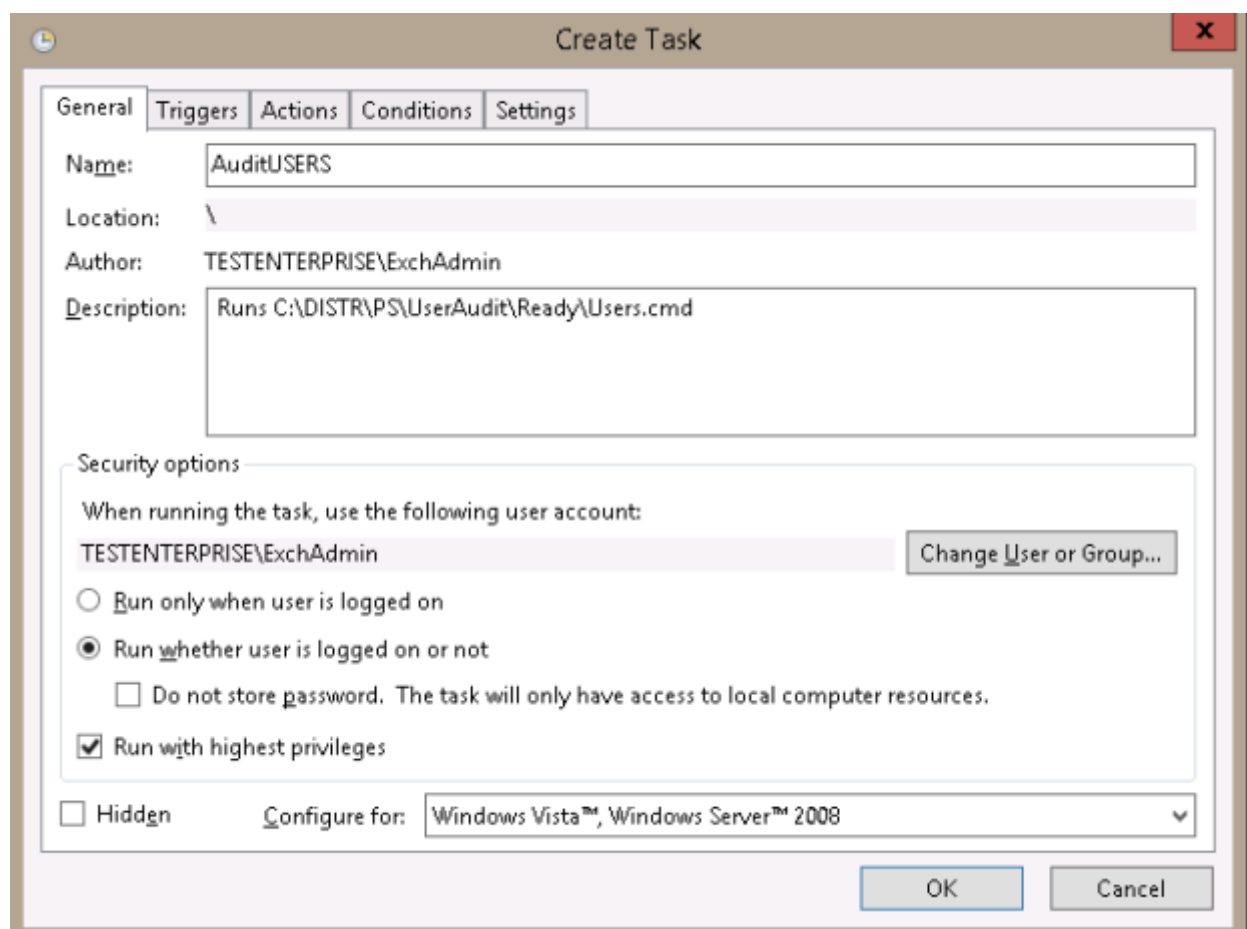
C:\Windows\System32\cscript C:\DISTR\PS\UserAudit\MailFILES-U.vbs
  
```



I'm using `ping 127.0.0.1 - n 5 > nul` here for pausing the scripts for ~5 seconds to allow the preceding command to complete.

MailFILES-U.vbs scripts just sends the resulting *.txt* files to the specified e-mail address – MailFILES-U .

As I want to get the users reports daily I've created the corresponding scheduled task – **AuditUSERS**:



Create Task
X

General
Triggers
Actions
Conditions
Settings

When you create a task, you can specify the conditions that will trigger the task.

Trigger	Details	Status
Daily	At 8:15 AM every day	Enabled

New...
Edit...
Delete

OK

Cancel

Create Task
X

General
Triggers
Actions
Conditions
Settings

When you create a task, you must specify the action that will occur when your task starts.

Action	Details
Start a program	C:\DISTR\PS\UserAudit\Users.cmd

<

|||

>

▲

▼

New...
Edit...
Delete

OK

Cancel

Create Task

General Triggers Actions Conditions Settings

Specify the conditions that, along with the trigger, determine whether the task should run. The task will not run if any condition specified here is not true.

Idle

☐ Start the task only if the computer is idle for: 10 minutes

Wait for idle for: 1 hour

☒ Stop if the computer ceases to be idle

☐ Restart if the idle state resumes

Power

☐ Start the task only if the computer is on AC power

☐ Stop if the computer switches to battery power

☒ Wake the computer to run this task

Network

☐ Start only if the following network connection is available:

Any connection

OK Cancel

Create Task

General Triggers Actions Conditions Settings

Specify additional settings that affect the behavior of the task.

☒ Allow task to be run on demand

☒ Run task as soon as possible after a scheduled start is missed

☒ If the task fails, restart every: 5 minute

Attempt to restart up to: 2 times

☒ Stop the task if it runs longer than: 1 hour

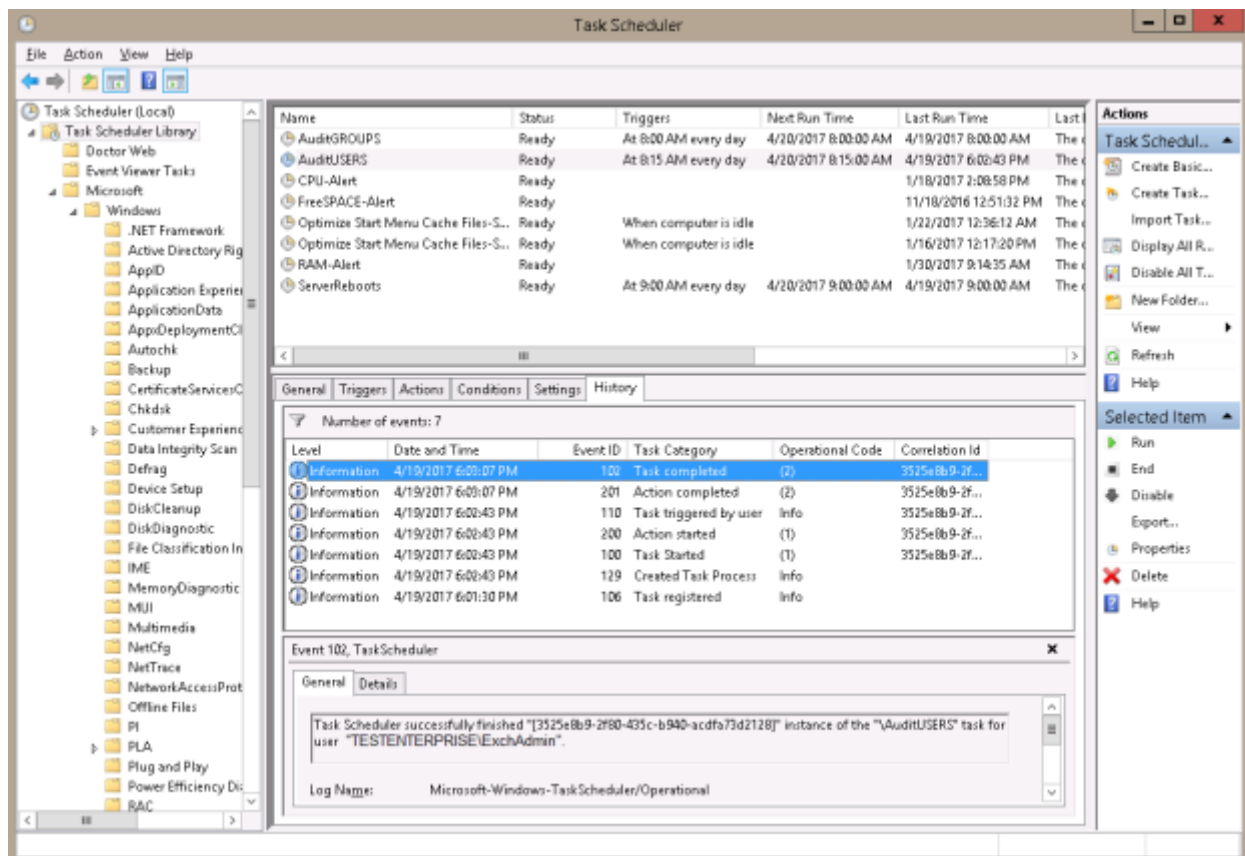
☒ If the running task does not end when requested, force it to stop

☐ If the task is not scheduled to run again, delete it after: 30 days

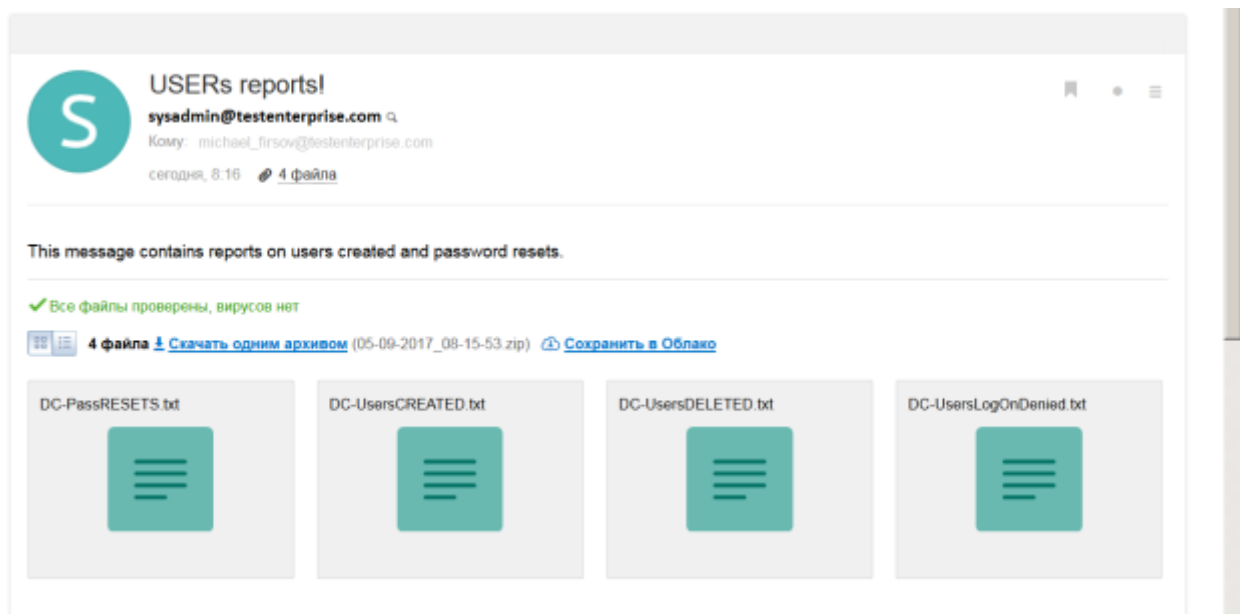
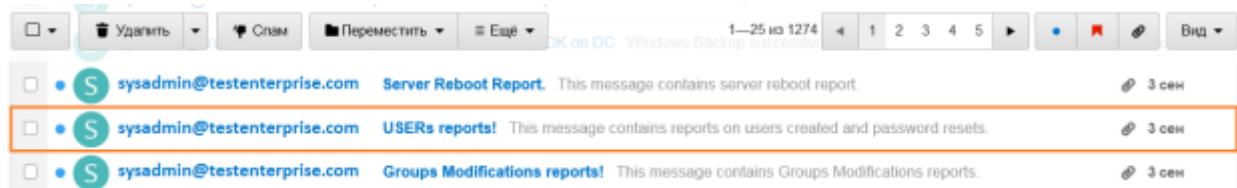
If the task is already running, then the following rule applies:

Run a new instance in parallel

OK Cancel



The e-mail message with the users reports:



II Group modifications operations:

- 1) User/Computer sccount is added to a global group
- 2) User/Computer sccount is added to a local group
- 3) User/Computer sccount is added to a universal group

- 4) User/Computer sccount is deleted from a global group
- 5) User/Computer sccount is deleted from a local group
- 6) User/Computer sccount is deleted from a universal group

The scripts:

AddedToGlobalgroup.ps1 – AddedToGLOBALgroup

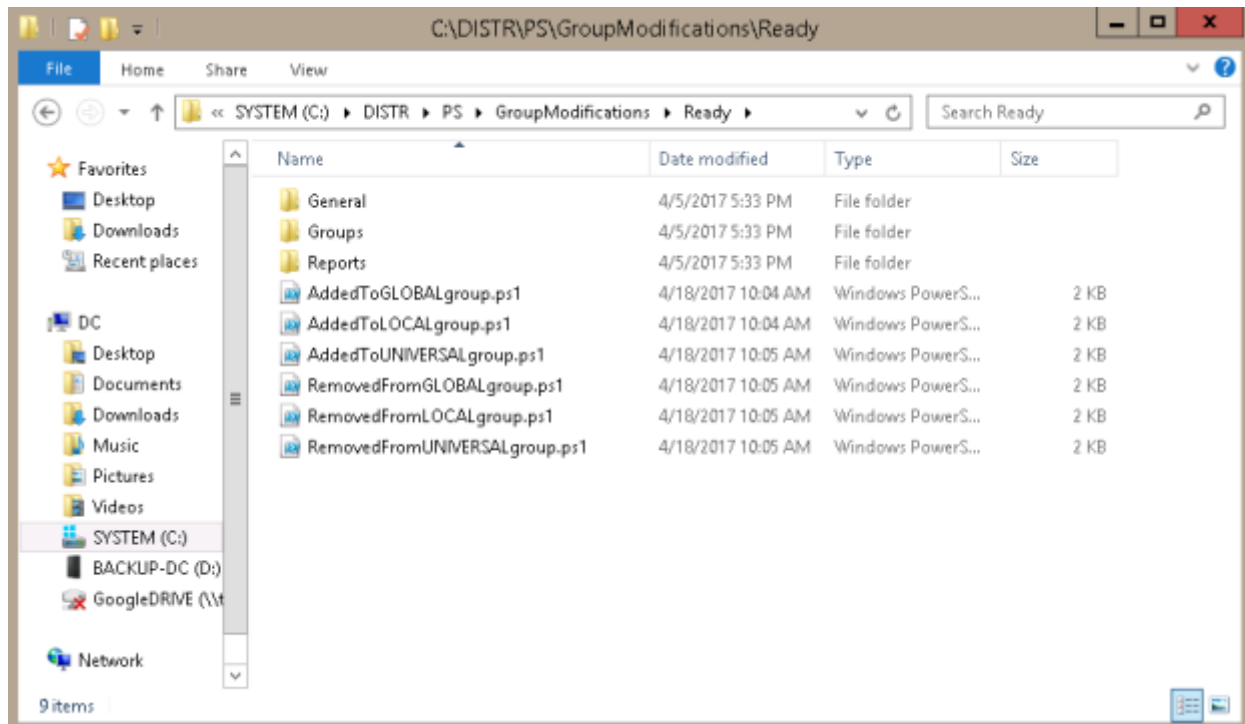
AddedToLOCALgroup.ps1 – AddedToLOCALgroup

AddedToUNIVERSALgroup.ps1 – AddedToUNIVERSALgroup

RemovedFromGLOBALgroup.ps1 – RemovedFromGLOBALgroup

RemovedFromLOCALgroup.ps1 – RemovedFromLOCALgroup

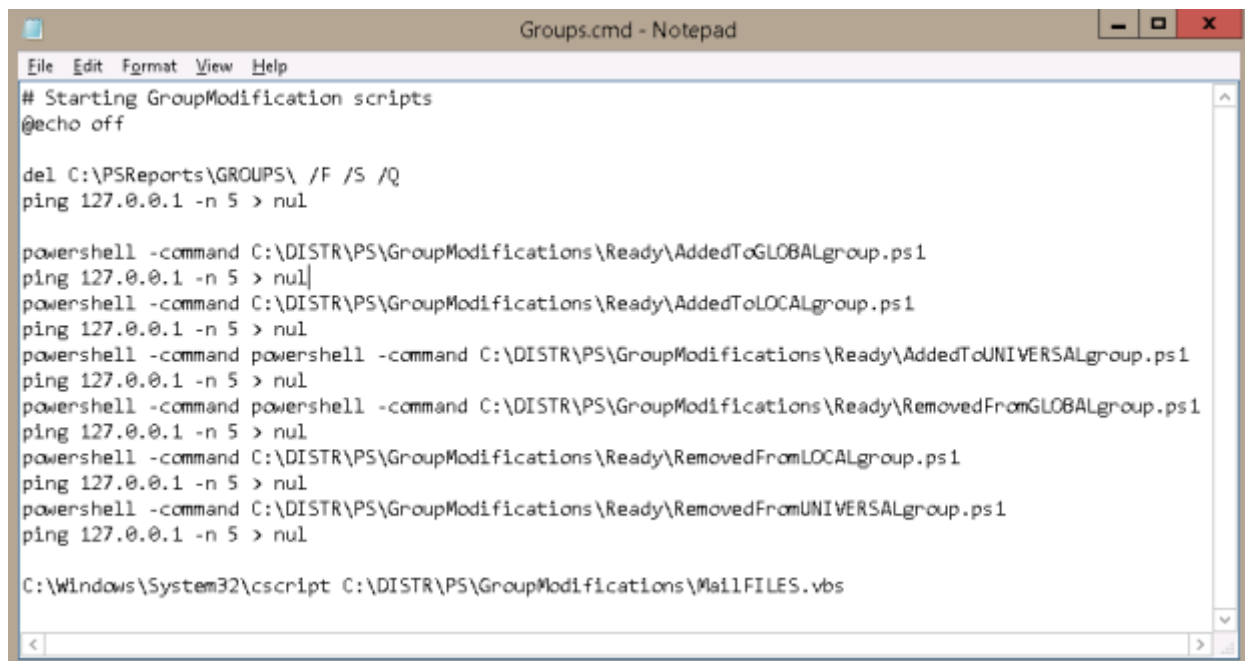
RemovedFromUNIVERSALgroup.ps1 – RemovedFromUNIVERSALgroup



Advertisements

Report this adPrivacy

All these 6 scripts are run by the **Groups.cmd** script: **Groups**

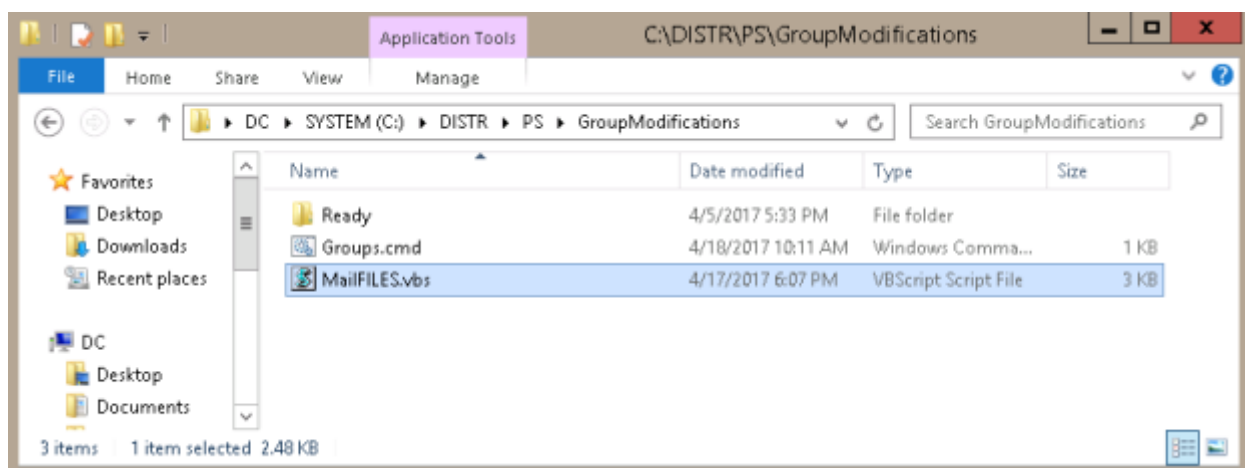


```
File Edit Format View Help
# Starting GroupModification scripts
@echo off

del C:\PSReports\GROUPS\ /F /S /Q
ping 127.0.0.1 -n 5 > nul

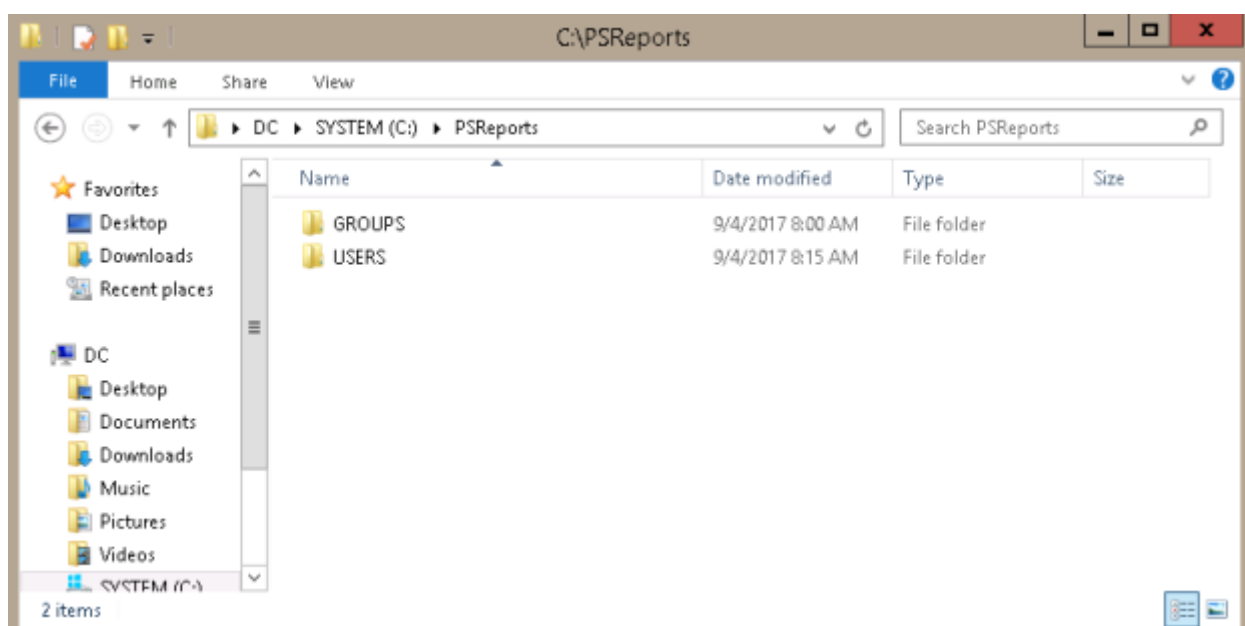
powershell -command C:\DISTR\PS\GroupModifications\Ready\AddedToGLOBALgroup.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\GroupModifications\Ready\AddedToLOCALgroup.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command powershell -command C:\DISTR\PS\GroupModifications\Ready\AddedToUNIVERSALgroup.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command powershell -command C:\DISTR\PS\GroupModifications\Ready\RemovedFromGLOBALgroup.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\GroupModifications\Ready\RemovedFromLOCALgroup.ps1
ping 127.0.0.1 -n 5 > nul
powershell -command C:\DISTR\PS\GroupModifications\Ready\RemovedFromUNIVERSALgroup.ps1
ping 127.0.0.1 -n 5 > nul

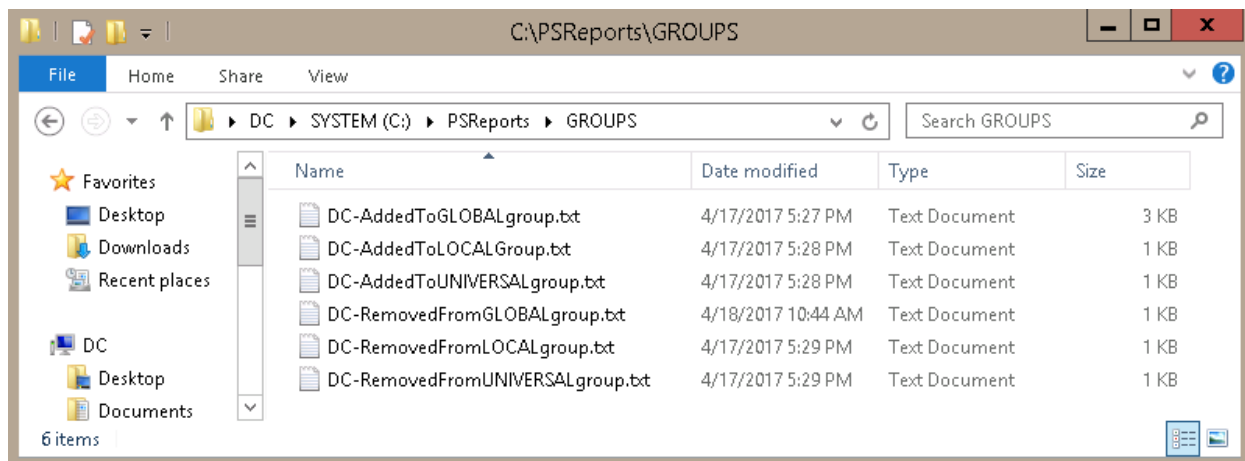
C:\Windows\System32\cscript C:\DISTR\PS\GroupModifications\MailFILES.vbs
```



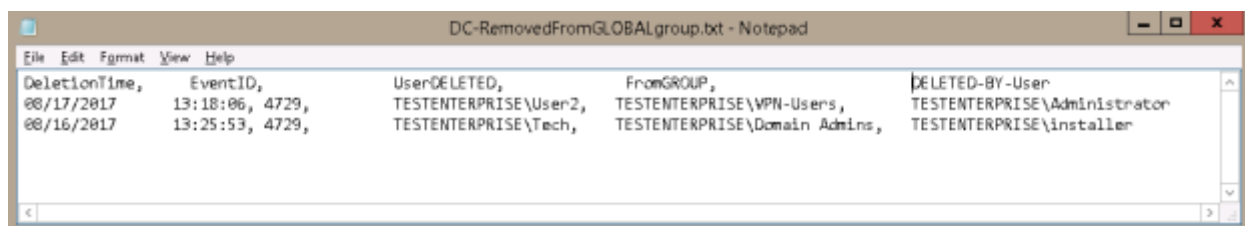
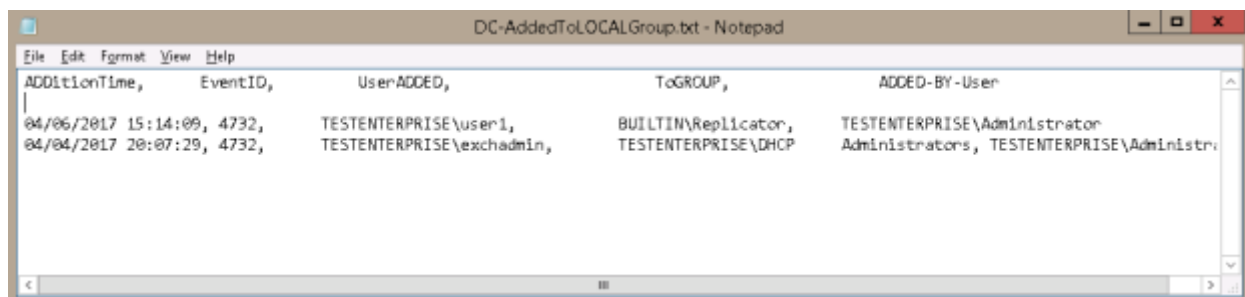
After running all *.ps1* scripts the **Groups.cmd** script calls for **MailFILES.vbs** script – MailFILES – which sends the all produced reports to my e-mail address.

Reports:

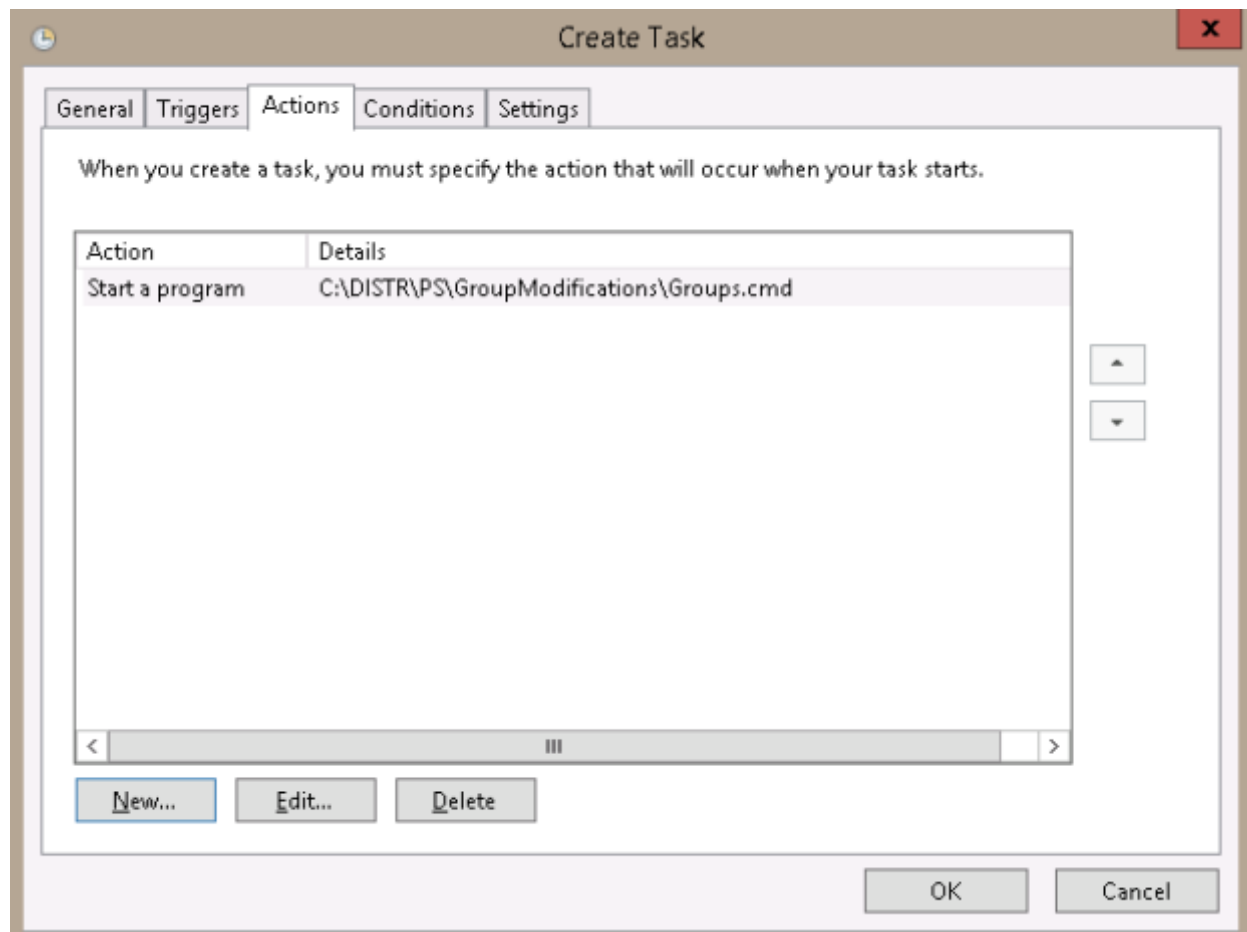




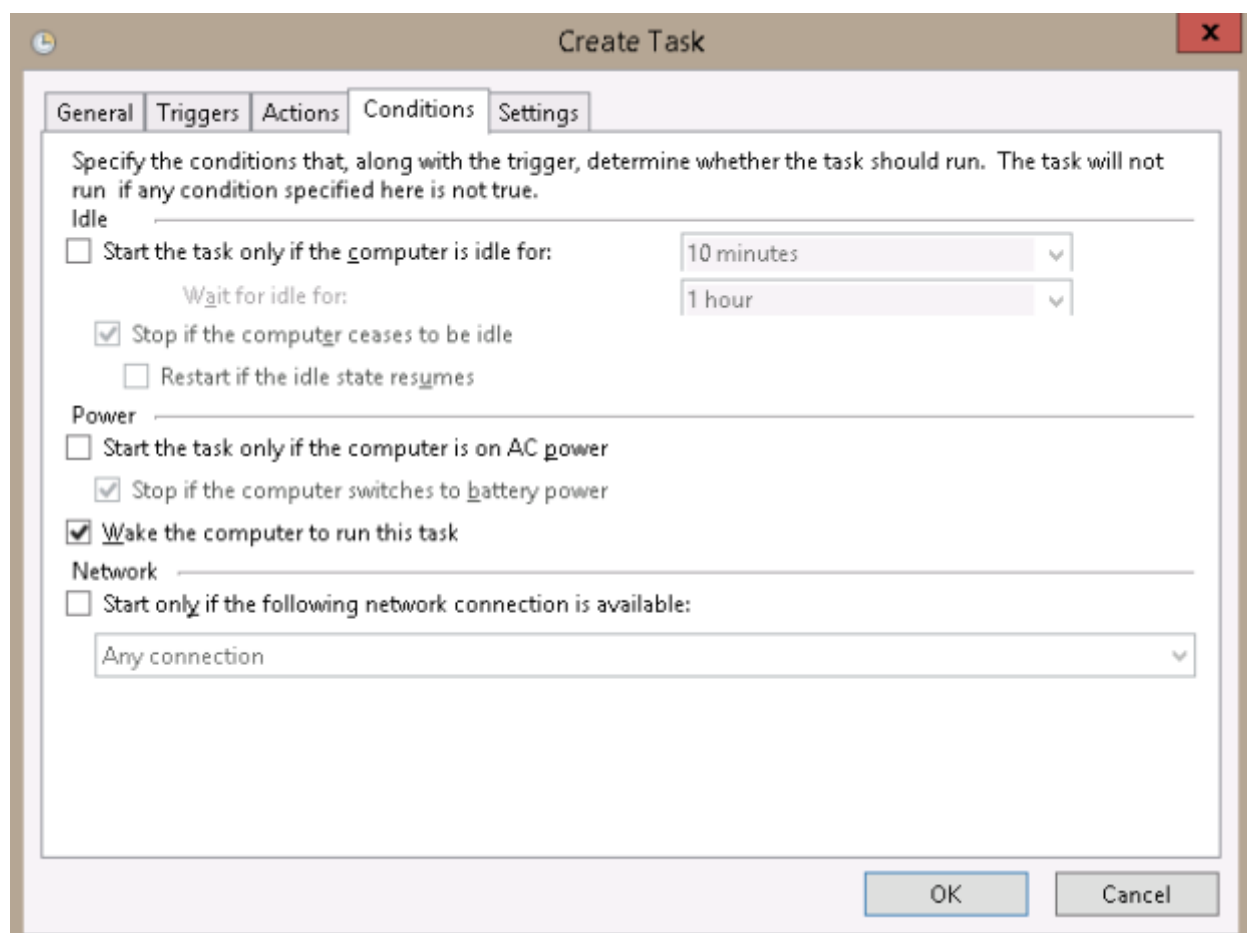
Advertisements
 Report this adPrivacy
 The example reports:

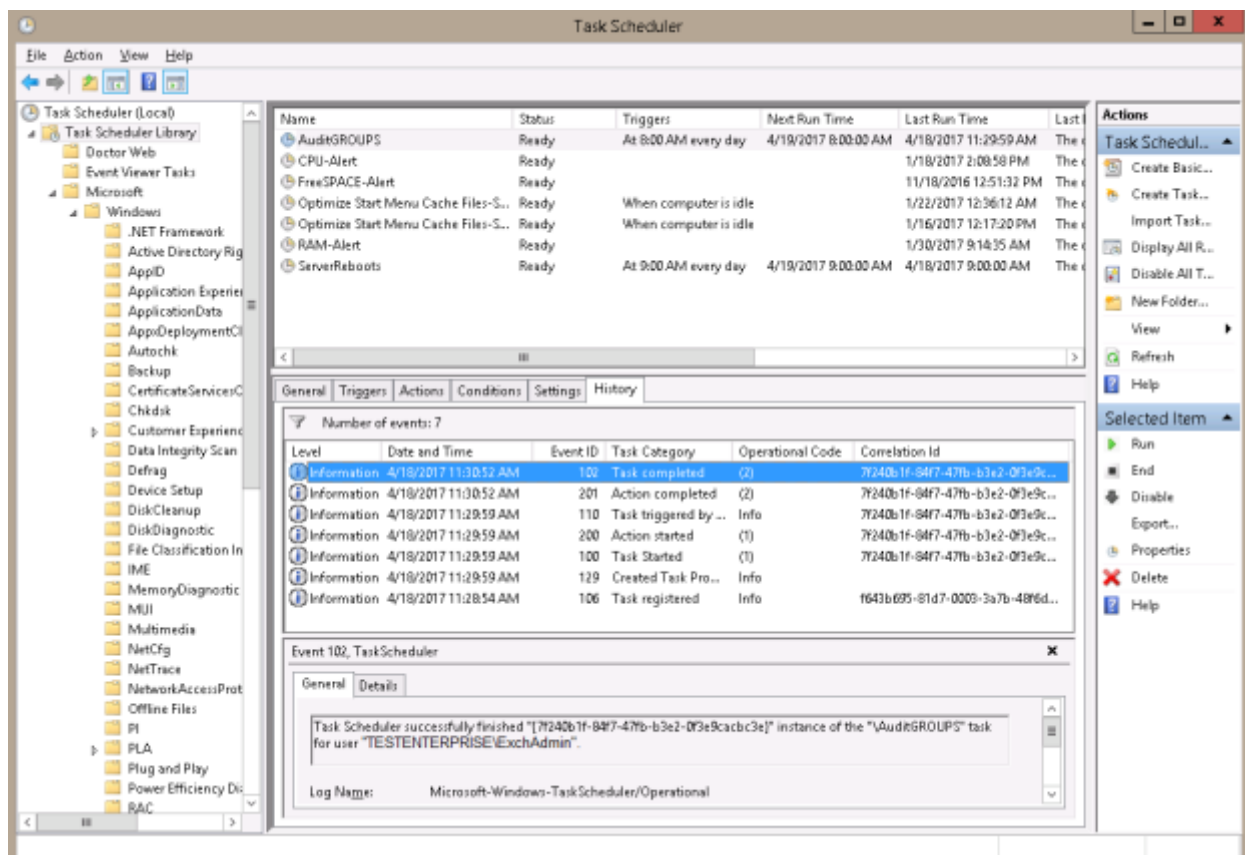
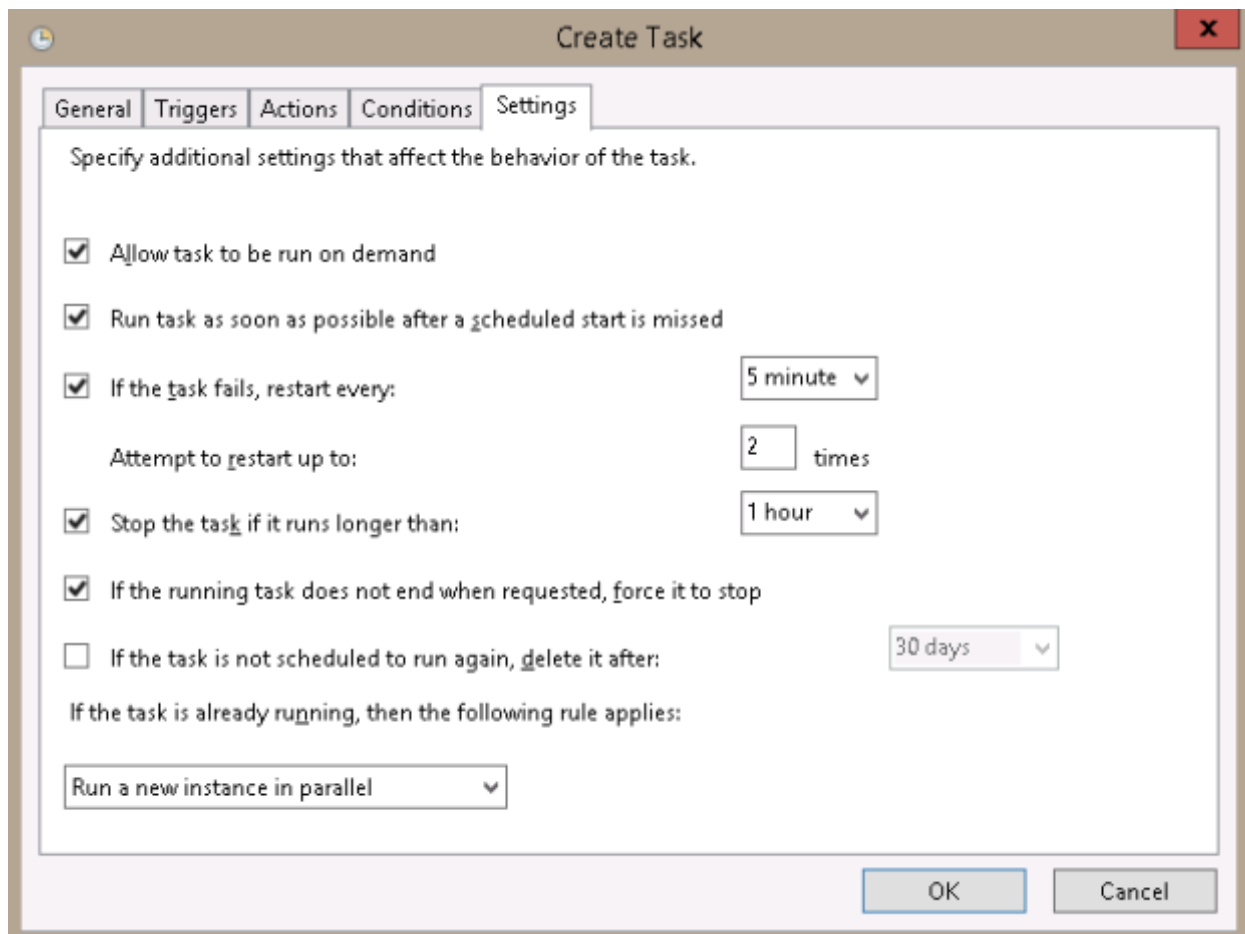


The task **AuditGROUPS**:

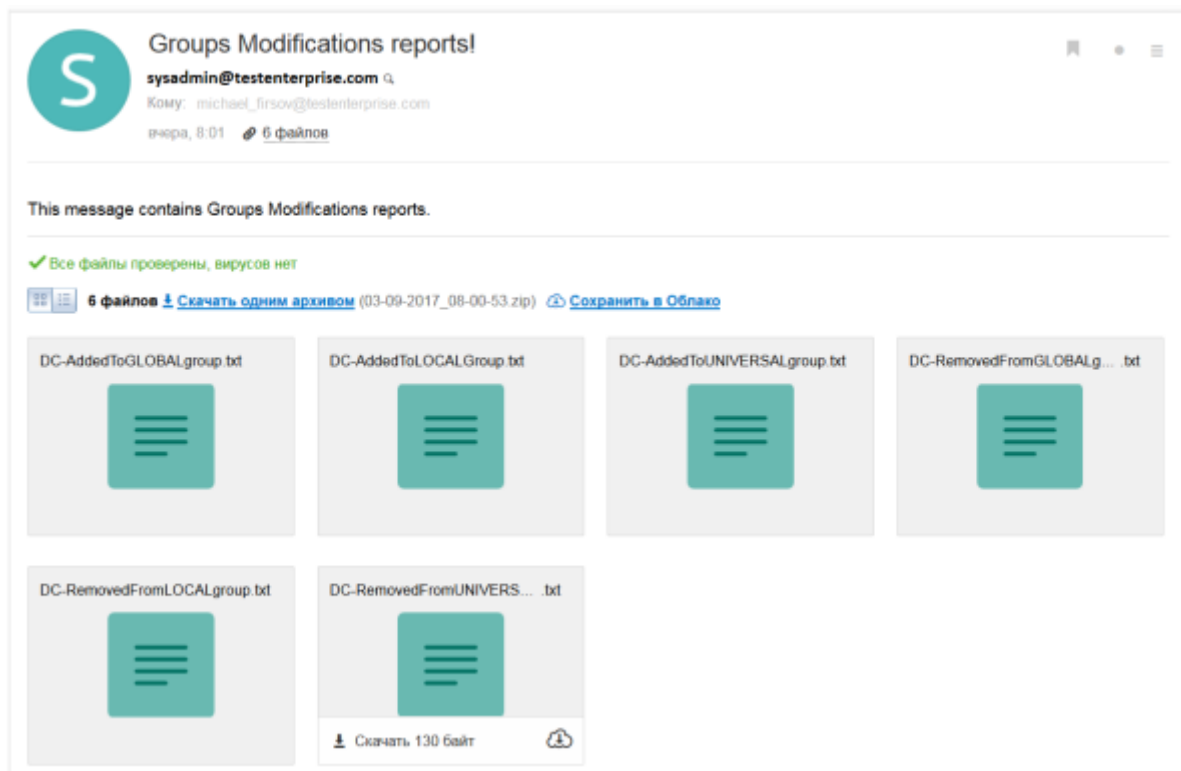
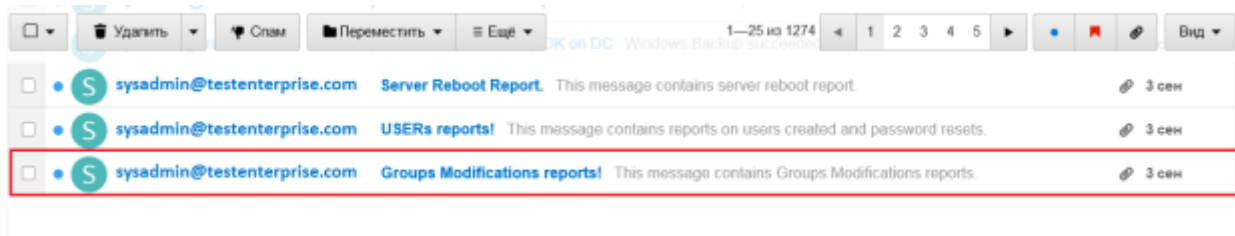


Advertisements
Report this adPrivacy





The resulting email message:



Part3