Nessus Compliance Generator

Using Nessus to audit Oracle SQL database configurations is fairly straightforward and only requires a few pieces to get moving. The Nessus Compliance Generator (NCG) program has been supplied to assist in building compliance audits to plug in to Nessus. The audit files themselves have a simple xml-like format. Currently supported by our generator tool are the following types of audits:

1. Oracle database compliance: requires a SQL query and an expected output.

2. Windows group membership compliance: requires a group name and expected members.

3. Windows file contents compliance: requires a filename to check on a remote system and a regular expression to validate contents.

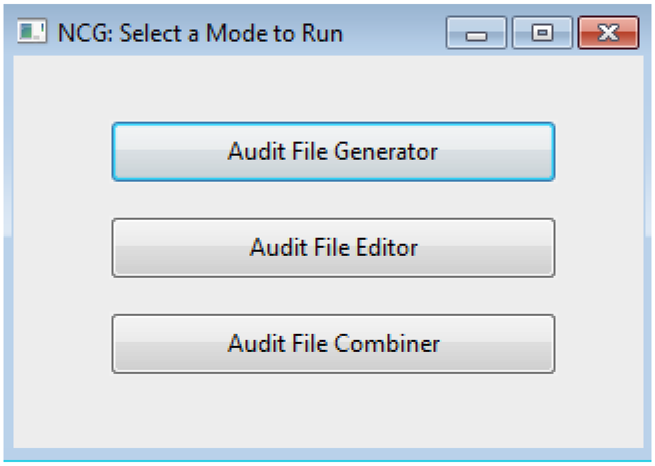
Compliance type 1 above requires database credentials to be entered into Nessus. Types 2 and 3 require that SMB file sharing be enabled and require administrator credentials be entered into Nessus. Note that all of the compliance checks we do are read-only queries and that the credentials need not be entered by the person conducting the audit.

This documentation will walk through (I) installing the NCG program, (II) generating an example audit file of each type, (III) editing an existing audit file, (IV) combining smaller audit files into a single file for import into Nessus, and (V) importing the audit into nessus and running the audit. We will conclude with (VI) an examination of the finished audit and some notes on limitations.

I. Installing NCG

The NCG program can be installed on a Windows computer using a simple installer file. The program can also be run on linux computers (via python) and can be built for Mac OS X. Additionally, programmatic access to all functions is available via the ncg_lib.py file. We will be showing examples from the windows version here, but runtime operation should be essentially the same on all platforms.

Launching the program will give you a “selector window” that allows you to select which mode of the program you'd like to run, seen below.

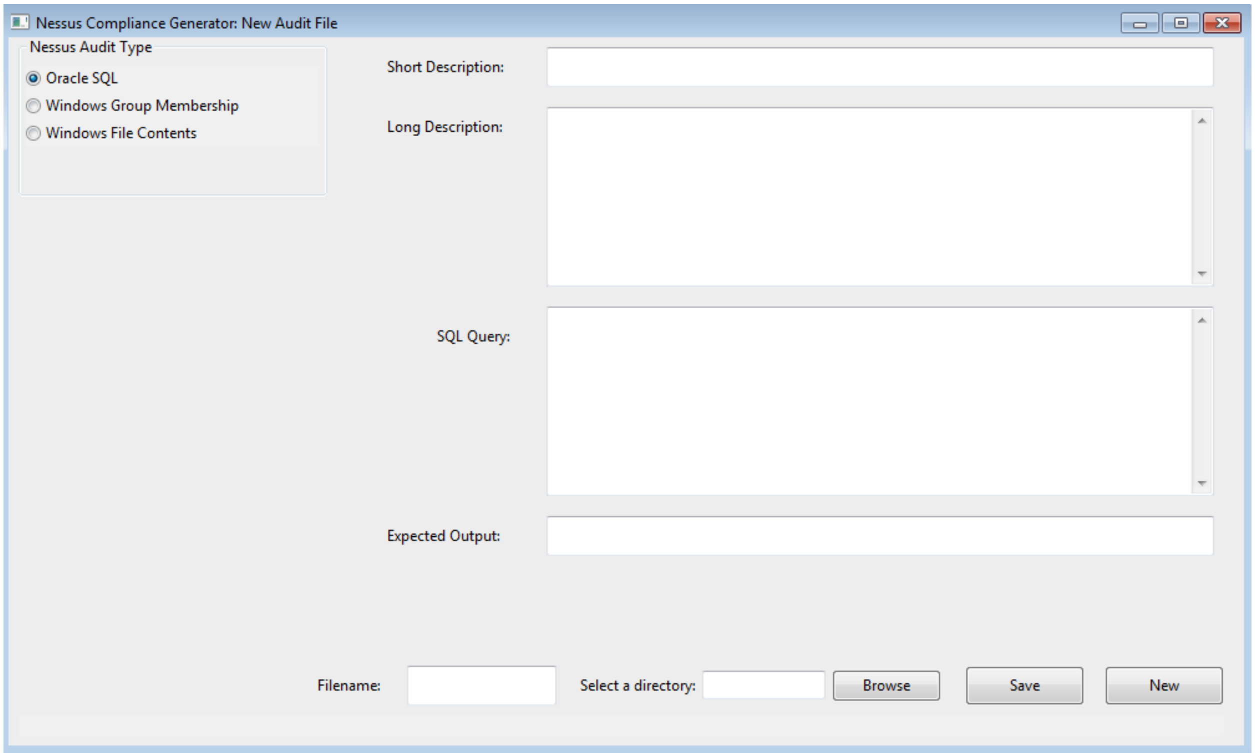


We'll start by choosing “Audit File Generator.”

II. Nessus Audit File Generation

1. Oracle SQL Audits

The audit file generator has 3 different modes as you'll see are selected by a radiobox selection in the top left of the main generator window:



Nessus Audit Type

● Oracle SQL

○ Windows Group Membership

○ Windows File Contents

The fields here are fairly straightforward – the short description should be one line and will show up as a heading in the final audit file. The long description will show up when looking for more information on a passed or failed audit item. SQL query and expected output are exactly what they seem like. We’ll start with a simple example that is considered a “Pass” if the query returns NULL or no rows.

Please note that this helper tool can be used even for queries that are not just Pass/Fail – To do this, enter NULL as your desired output. The Nessus check will always show as “Failed” but will show you the query results allowing an auditer to examine them manually and determine if they correspond with expected output.

Expected output here can be anything that SQL can output, but best practice is to simplify queries as much as possible so that Nessus can more easily match output. For instance, if your current query includes “select name, value ...” and matches for “SYS   TRUE”, a better query for Nessus would be “select value where name = 'SYS' ...” and match only on “TRUE.” Of course, this is not always possible or ideal. We will show in section VI how to evaluate queries that have no expected output or varying expected output. For now, for these types of queries, leave expected output as NULL.

After you choose a filename and a directory to save in, you can click “Save” and you'll see a message saying where the file was successfully saved:

Now you can click “New” to clear all fields and start a new audit file. The selected directory will remain, as will the notice of where the previous file was saved to help you track where you're at as you're creating multiple audits.

Also Notice I've included a filename with a .audit extension. This is not necessary, but might help keep track of these files. To make things simpler for later steps, its advised that you keep audit files in a folder with no other files, and keep them separated by Windows and SQL types. As we'll see later, this will make combining them and installing them into Nessus easier. Lets take a look at what the file this generated looks like:

```
<check_type: "Database" db_type:"Oracle" version:"1">
  <group_policy: "Test">

<custom_item>
  type: SQL_POLICY
  description: “Check (default users only)"
  info: "All DEFAULT accounts that are unused and not required for database operations are LOCKED
and/or EXPIRED.  Passwords are set for each account based on the password verification policy"
  sql_request: "Select username, account_status, profile from dba_users where profile IN
('DEFAULT','MONITORING_PROFILE') order by profile, account_status, username;"
```

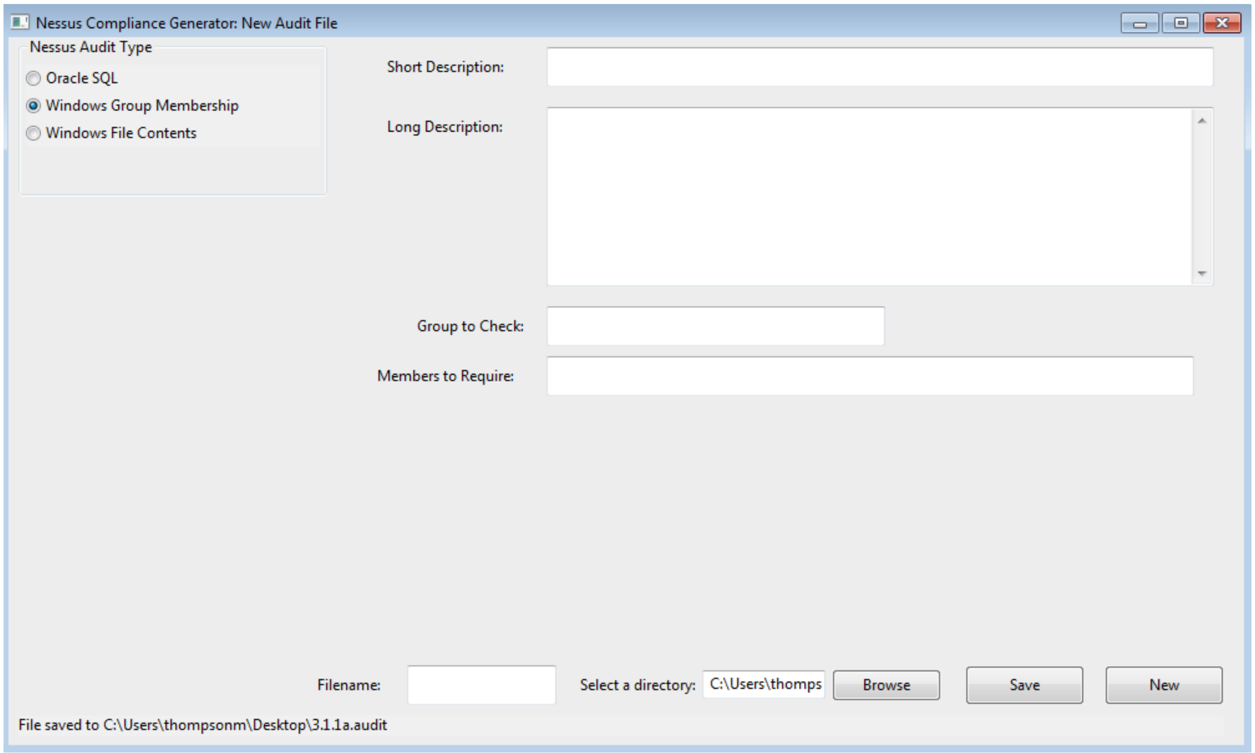
```
  sql_types: POLICY_VARCHAR
  sql_expect: NULL
</custom_item>

  </group_policy>
</check_type>
```

Most of the fields don’t vary much outside of what the python script already queries you for. We could obviously generate these by hand, but since this is mostly a data entry task, the NCG tools are meant to make it easier to do exactly that: make it a data entry task and make it harder to make a mistake. Since the goal of this tool is to avoid having to mess with the xml-like syntax of the audit files, this is the only time we'll look in detail at the generated file.

When we run the GUI in combination mode (which we will in section IV below), it will generate an XML file much like the one above, but with several “custom_item” entries. Nessus can use this to batch several files into one.

2. Windows Group Membership Audits

In the main generator window, if you click the radio box for Windows Group Membership, you'll notice slightly different choices for entering data:



Nessus Compliance Generator: New Audit File

Nessus Audit Type

○ Oracle SQL
● Windows Group Membership
○ Windows File Contents

Short Description:

Long Description:

Group to Check:

Members to Require:

Filename:

Select a directory: C:\Users\thomps

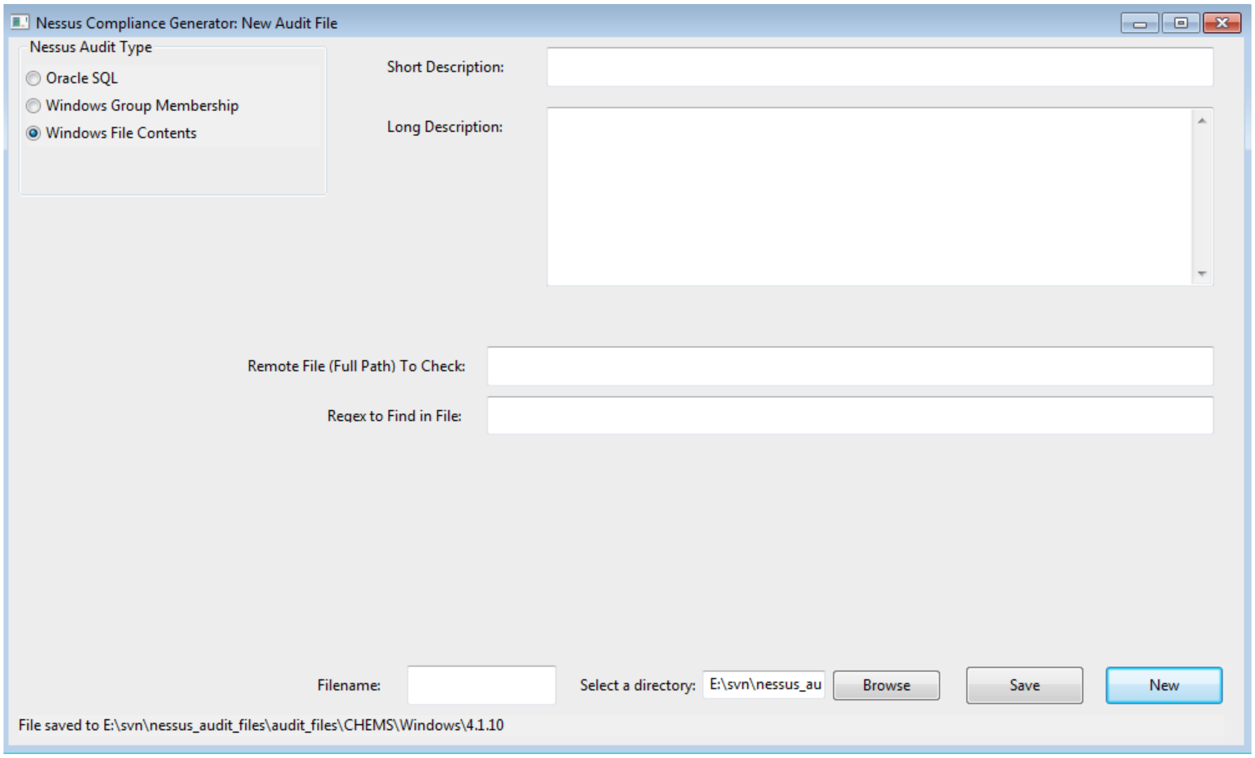
File saved to C:\Users\thompsonm\Desktop\3.1.1a.audit

Options here are again, fairly straightforward. As before, the short description should be one line and will show up as a heading in the final audit file. The long description will show up when looking for more information on a passed or failed audit item. Group to check should be a windows group, and members to require aregroup members that should be in that group for the test to pass. If there are members in the group that are not in this list the test will fail. Also if all the members put here are not in the group, the test will fail.

The script will attempt to sanitize any input, so spaces and tabs are ok. In Group members to require, items should be comma separated (additional spaces around commas are ok but not necessary).

3. Windows File Contents Audits

In the main generator window, if you click the radio box for Windows File Contents, you'll notice slightly different choices for entering data:



○ Oracle SQL
○ Windows Group Membership
● Windows File Contents

Select a directory: E:\svn\nessus_au

File saved to E:\svn\nessus_audit_files\audit_files\CHEMS\Windows\4.1.10

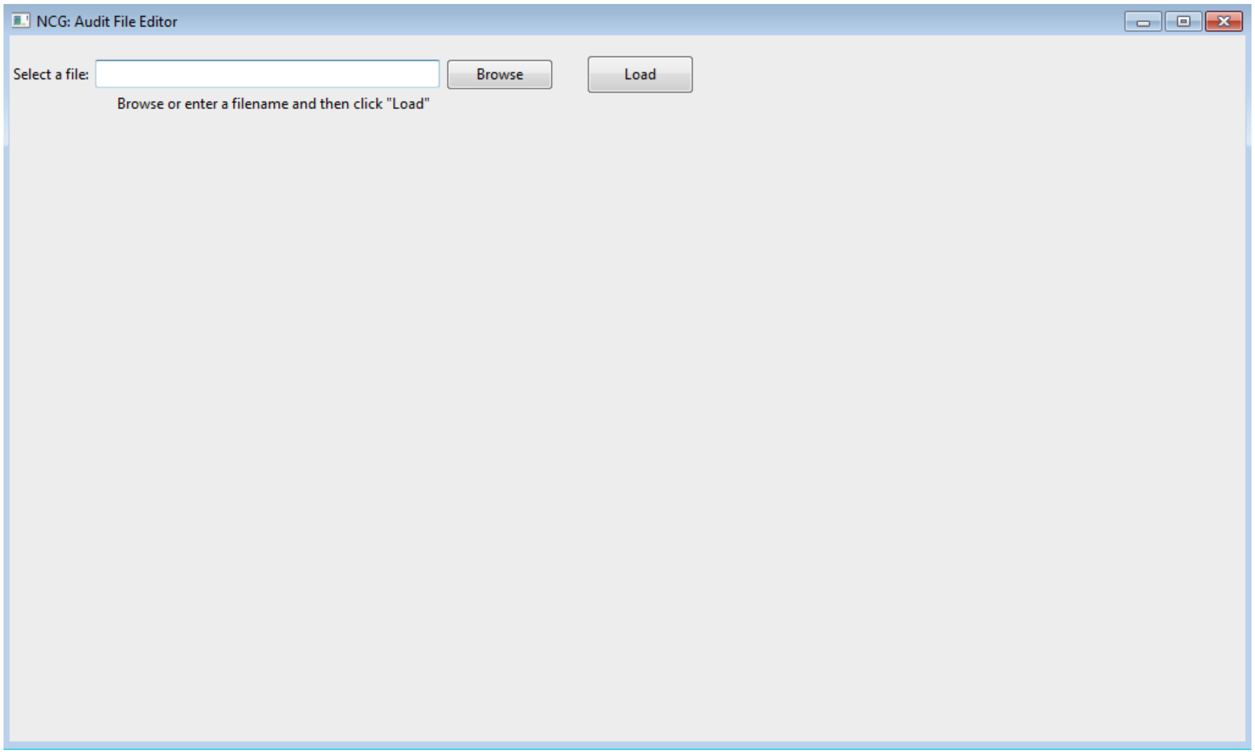
Options here are again, fairly straightforward. As before, the short description should be one line and will show up as a heading in the final audit file. The long description will show up when looking for more information on a passed or failed audit item. Remote File to Check is a filename on the windows system you're auditing that you want to verify some content of. Regex to Find in File will take a regular expression to verify that the file contains. If that regular expression successfully matches some content in the file, the check will show up as passed in Nagios.

Regular expressions follow a fairly standard format.

III. Editing an Existing Audit File

Editing an existing audit file is fairly simple if that audit file was generated using this program. Note that it is best practice to save individually generated audit files and put them in revision control so that editing and regenerating combined audits (as we'll see in section IV of this document) is easy, consistent, and reproducible. This time after launching the main selector window, click the "Audit File Editor" mode.

When you first launch the file editor, you'll see a blank window with a file selection dialog, as below.



Browse or enter a filename and then click "Load"

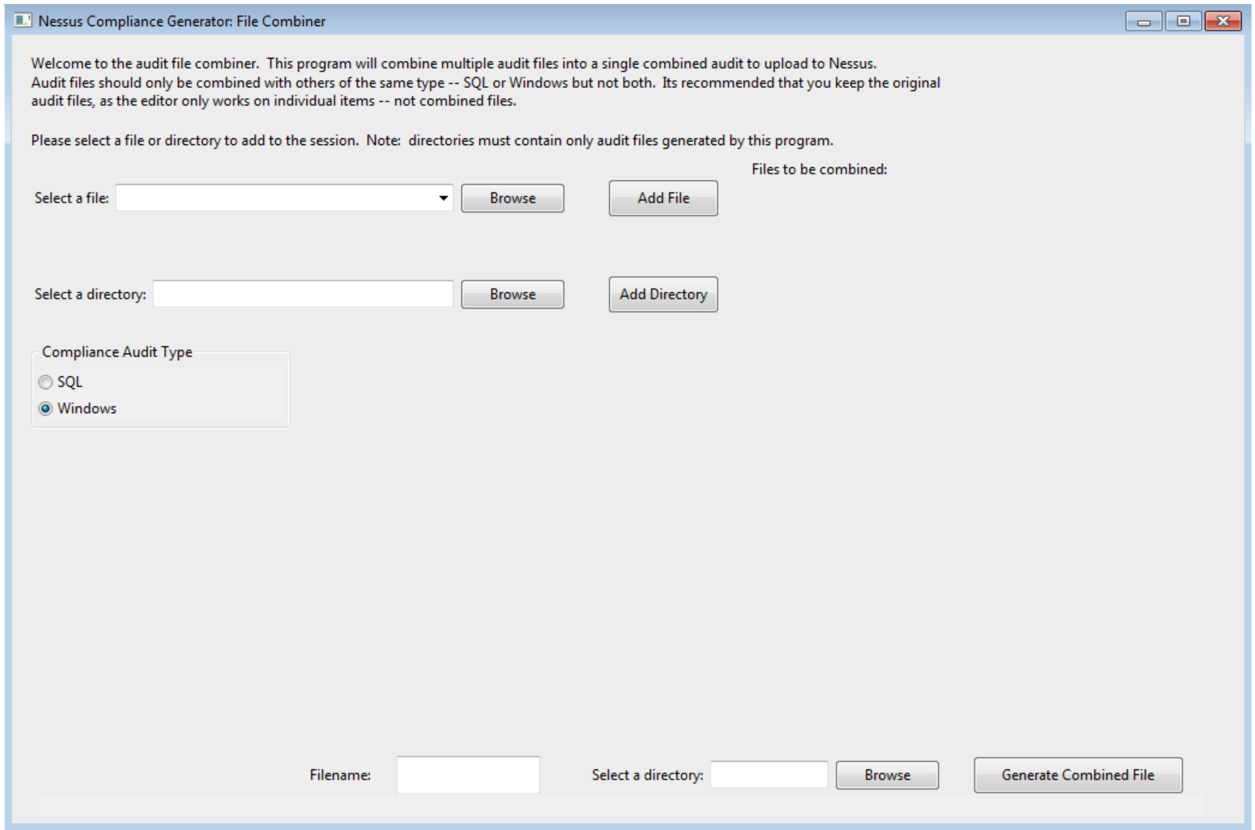
Simply browse for a file, and once its location is entered in the box, click "Load." The editor will do its best to parse the file given and display it to you much the same way as it was displayed in the generator window. This is why its especially important to use the NCG tool for creation and save the originals. If you have an audit file created by hand or by another tool, there are no guarantees that the editor will work properly. Also note, if you have made changes to the file that are not in the scope of what the editor allows you to change, those changes **will not** be reflected back when you click update.

When you click “Update,” the original file will be overwritten with whatever changes you've made.

IV. Combining smaller audits into one large audit file.

NCG also has a “combiner” mode – since Nessus will take a limited number of audit files for a single scan and since adding files to the web interface is cumbersome, we'll combine them into one large file to add to Nessus. As noted several times before, it is best practice to keep the separate audit files for editing and/or recombination and to only use the combined file for uploading to Nessus.

From the NCG mode selector, choose “Audit File Combiner” and you'll see the below window:



Welcome to the audit file combiner. This program will combine multiple audit files into a single combined audit to upload to Nessus. Audit files should only be combined with others of the same type -- SQL or Windows but not both. Its recommended that you keep the original audit files, as the editor only works on individual items -- not combined files.

Please select a file or directory to add to the session. Note: directories must contain only audit files generated by this program.

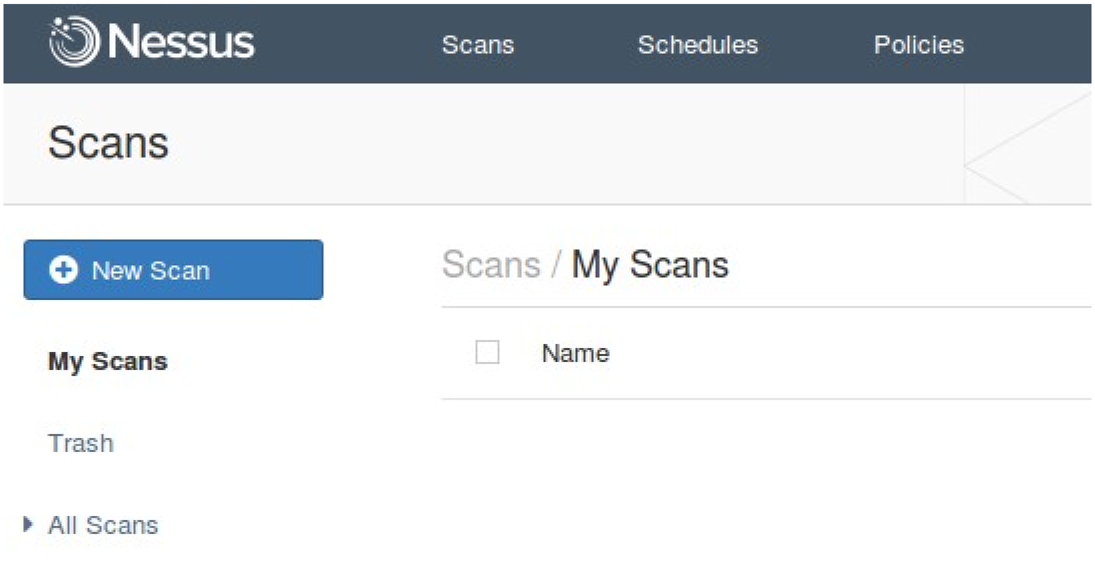
Here you see controls to select and add a file or directory and to choose a compliance audit type. There are a few things to note here. If adding a full directory, make sure that directory **only** contains audit files. When choosing compliance type, “Windows” refers to both group membership audits and file content audits. They can be grouped together, but because of the way we'll upload to Nessus (shown in section V. of this document), SQL and Windows audits cannot be grouped with each other.

As you add files and/or directories, you'll see them show up on the right hand side. When you're finished, you go through the same simple process to enter a filename and choose where to save and click “Generate Combined File”

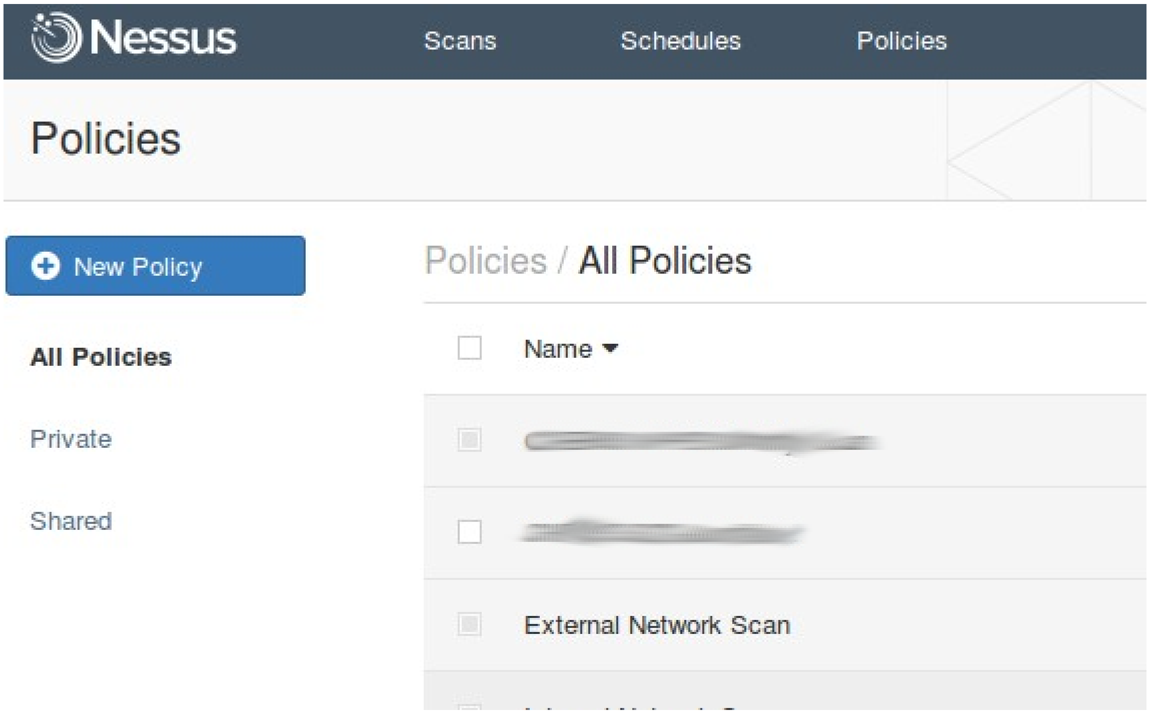
As already stated, we have to choose the compliance type. At this stage, the program doesn't do any checking on the contents of the files, so it depends on you to only add SQL or Windows files to a combined audit (and to not accidentally add any other types of files). Now that we have the combined file, lets see how to make it interact with Nessus.

V. Using Audit Files with Nessus.

Using these audit files with the Nessus software is again, not difficult, once you know where to look for it. When initially logging into Nessus, you’ll see a screen like the one below… You’ll want to click the button on the top that says “Policies”:



When you get to the policies screen, as shown below, you’ll click “+ New Policy”

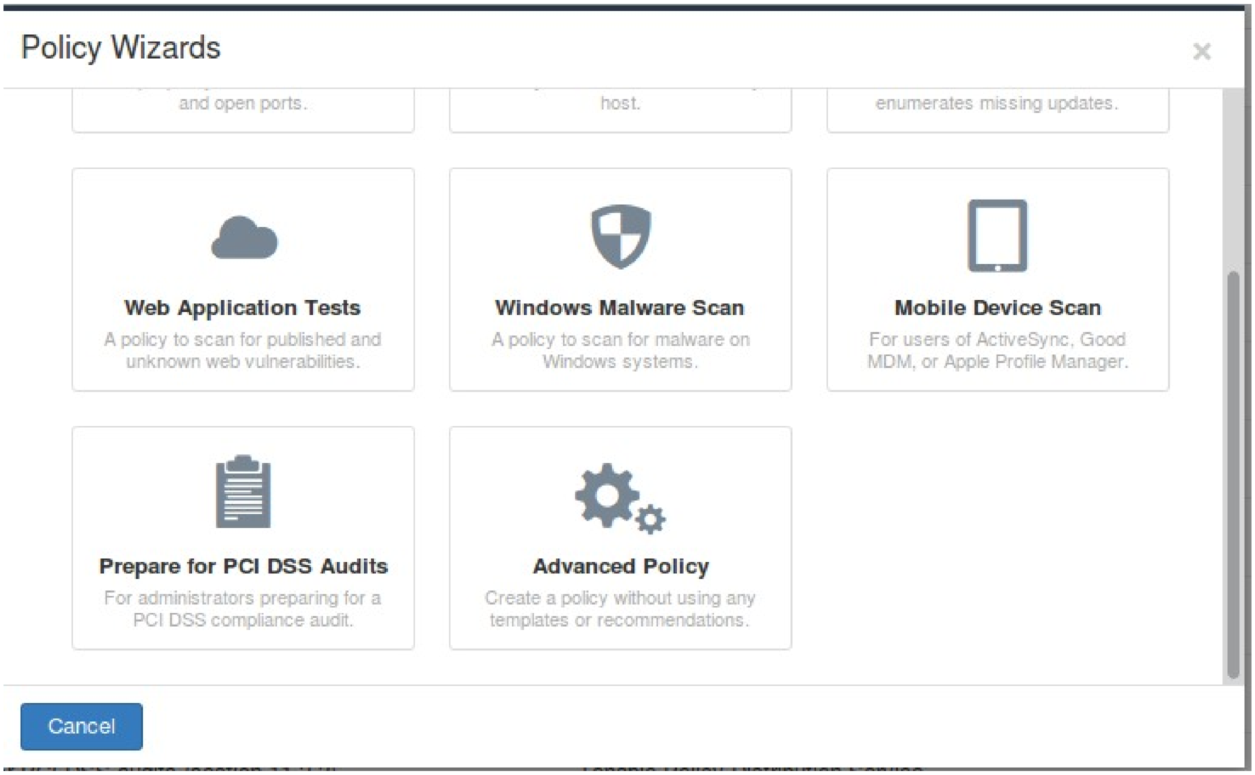


□ [redacted]

□ [redacted]

□ External Network Scan

Now you'll see the Policy wizards popup, as below. Just scroll down and choose "Advanced Policy."



and open ports.

host.

enumerates missing updates.

Web Application Tests
A policy to scan for published and unknown web vulnerabilities.

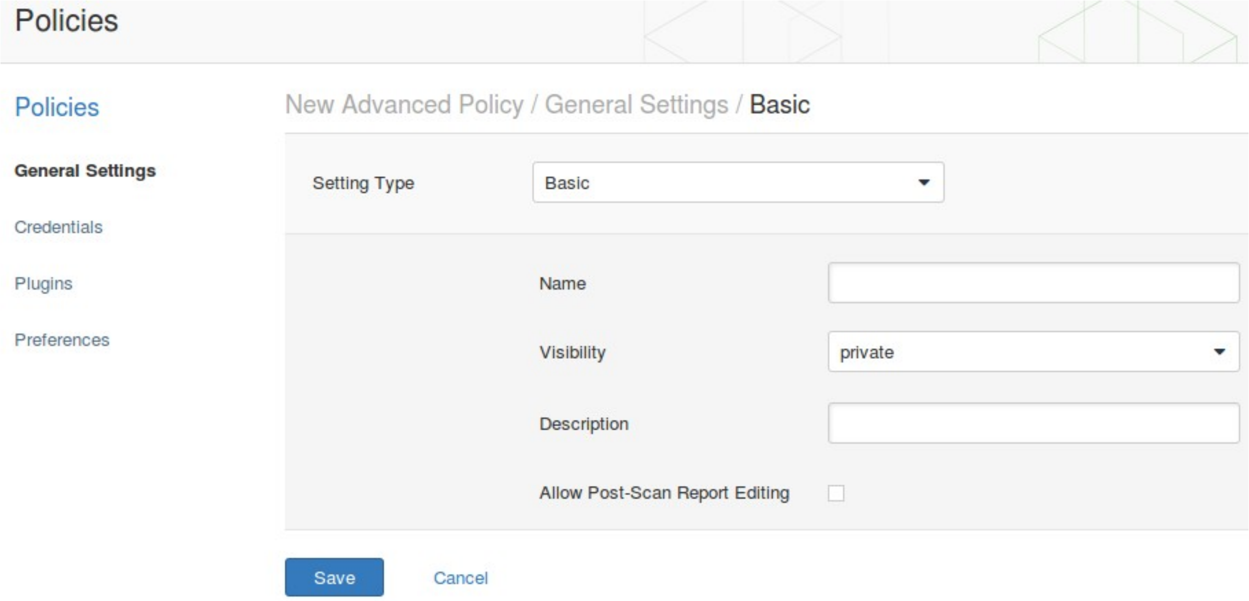
Windows Malware Scan
A policy to scan for malware on Windows systems.

Mobile Device Scan
For users of ActiveSync, Good MDM, or Apple Profile Manager.

Prepare for PCI DSS Audits
For administrators preparing for a PCI DSS compliance audit.

Advanced Policy
Create a policy without using any templates or recommendations.

On the “Policy General Settings” page that you’ll see next (shown below), you’ll want to name the policy, select “Shared” if you want others to be able to run it, and uncheck any tests that you don’t want to run (In the dropdown menu, there’s an option for Port Scanning, unless you have a reason to run any of those scans, you can safely uncheck them).



New Advanced Policy / General Settings / Basic

Setting Type: Basic

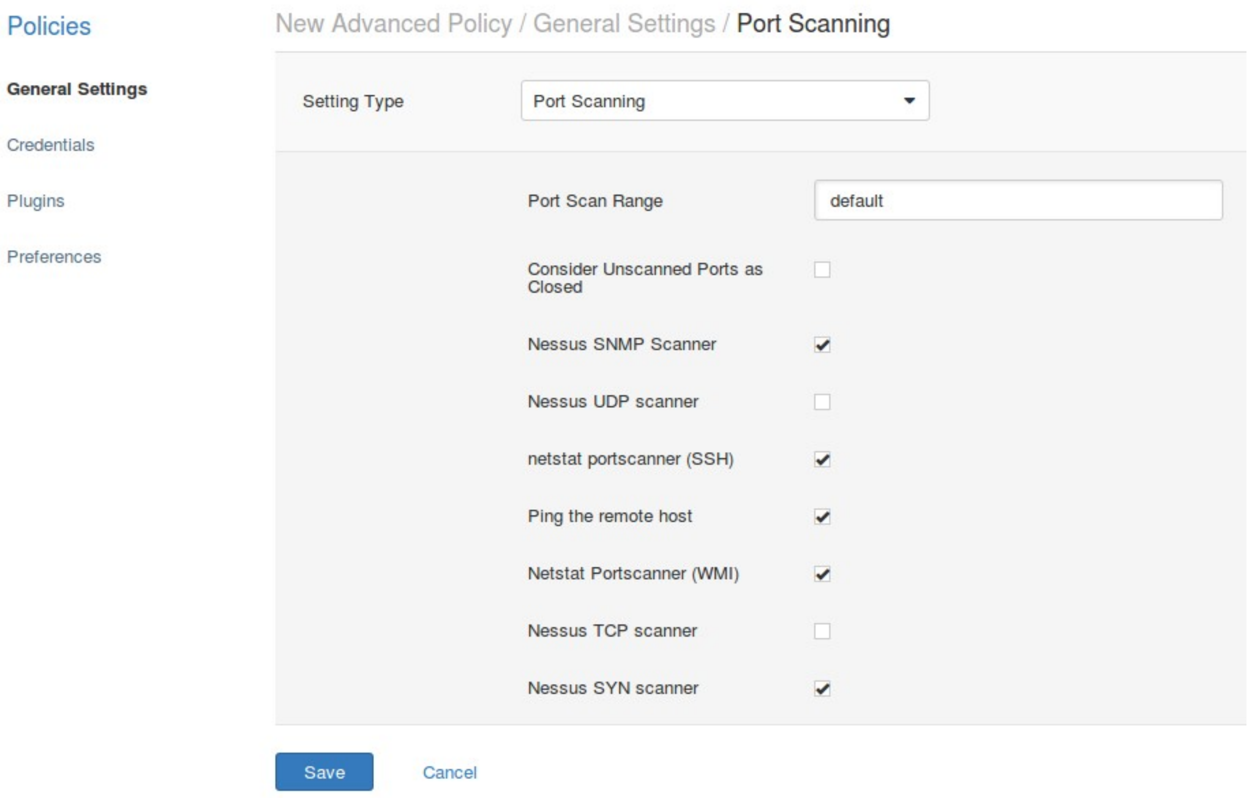
Name:

Visibility: private

Description:

Allow Post-Scan Report Editing □

Port scanning options can be disabled below:



Policies

General Settings

Credentials

Plugins

Preferences

New Advanced Policy / General Settings / Port Scanning

Setting Type: Port Scanning

Port Scan Range: default

Consider Unscanned Ports as Closed □

Nessus SNMP Scanner ☑

Nessus UDP scanner □

netstat portscanner (SSH) ☑

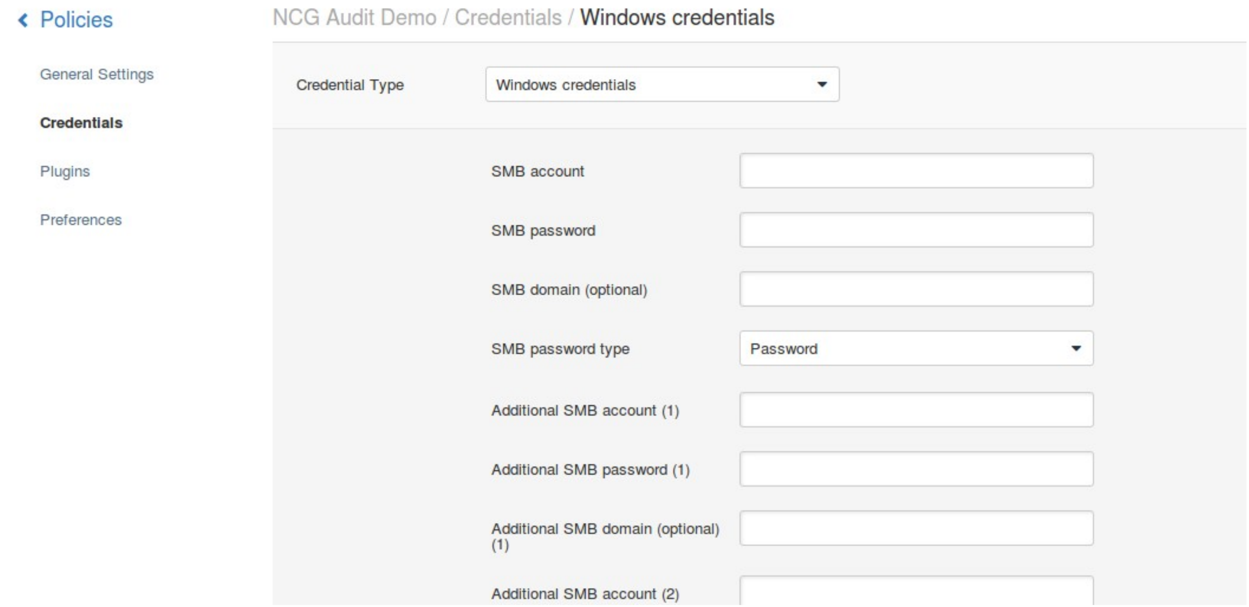
Ping the remote host ☑

Netstat Portscanner (WMI) ☑

Nessus TCP scanner □

Nessus SYN scanner ☑

Once you're finished on the "General Settings" page, click "Credentials" on the left side. This is where we're enter Windows credentials if we're doing windows audits (we'll enter Oracle credentials a bit further down in a different spot in the Nessus interface).



‹ Policies

General Settings

Credentials

Plugins

Preferences

NCG Audit Demo / Credentials / Windows credentials

Credential Type: Windows credentials

SMB account:

SMB password:

SMB domain (optional):

SMB password type: Password

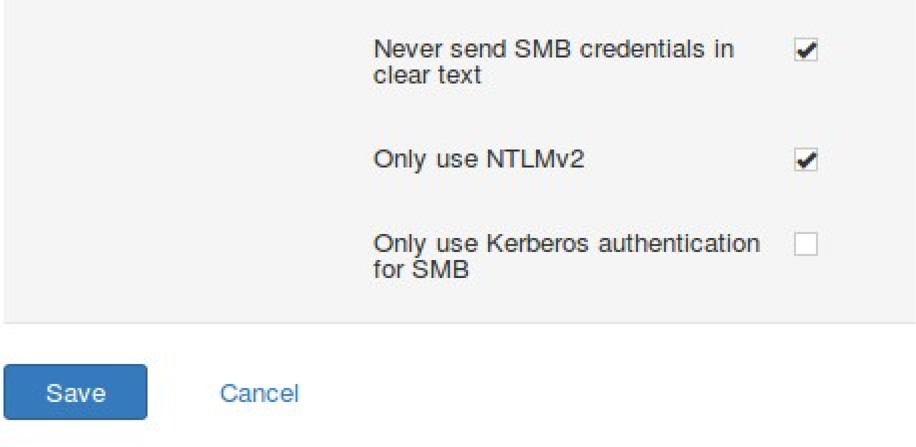
Additional SMB account (1):

Additional SMB password (1):

Additional SMB domain (optional) (1):

Additional SMB account (2):

We tested this using Administrator credentials. It should be investigated if these checks can be run with lower privileges. You should also click to only “Never send SMB credentials in clear text” and “Only use NTLMv2” as shown below:

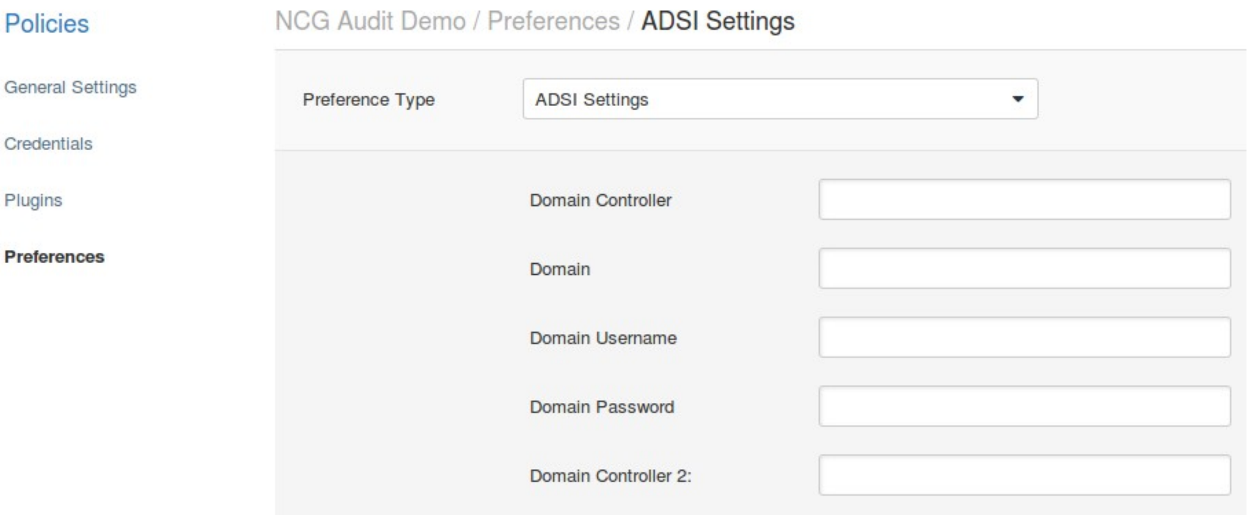


Never send SMB credentials in clear text ☑

Only use NTLMv2 ☑

Only use Kerberos authentication for SMB □

To add Oracle credentials, click the “Preferences” button on the left.   You’ll see a screen like the one below, and you’ll want to click the dropdown menu and select “Database Settings.”



NCG Audit Demo / Preferences / ADSI Settings

Preference Type: ADSI Settings

Domain Controller

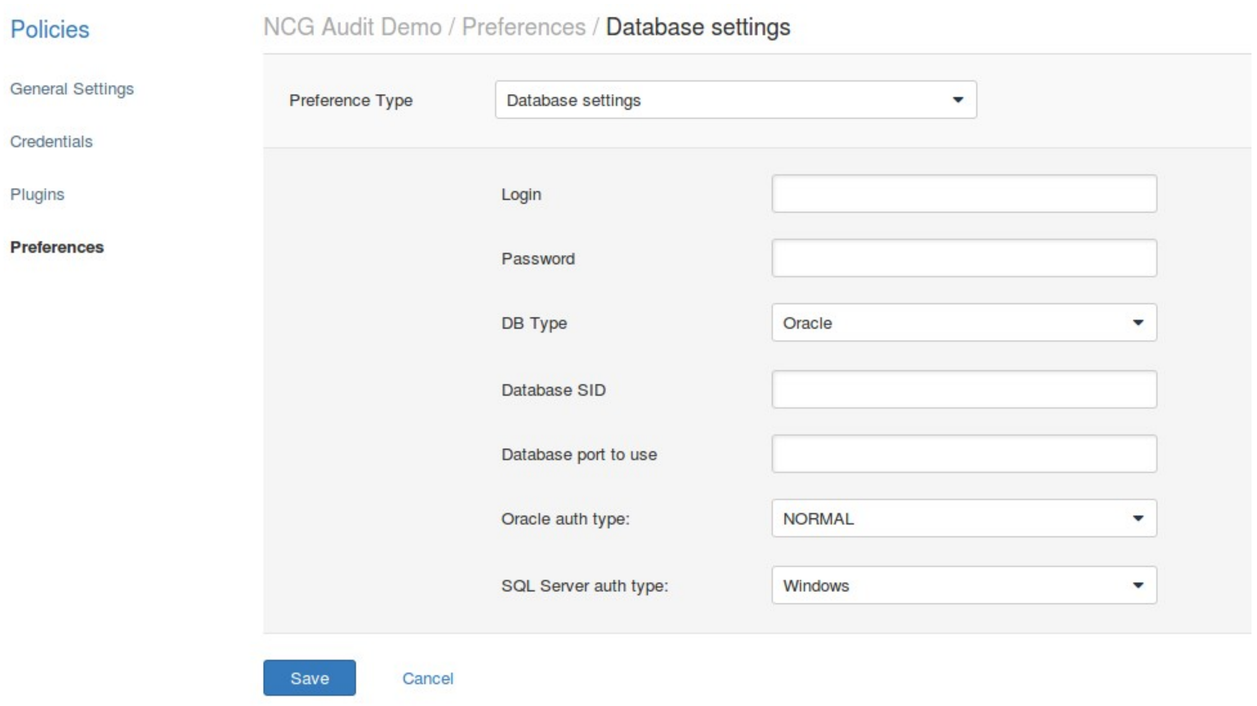
Domain

Domain Username

Domain Password

Domain Controller 2:

On the database settings page shown below, you’ll want to enter the relevant information for the databases that you want to check. Checks can only be batched using Nessus if they have a consistent username and password among them that has privileges to issue the relevant select statements (the same is true of the windows checks).



NCG Audit Demo / Preferences / Database settings

Preference Type: Database settings

Login:

Password:

DB Type: Oracle

Database SID:

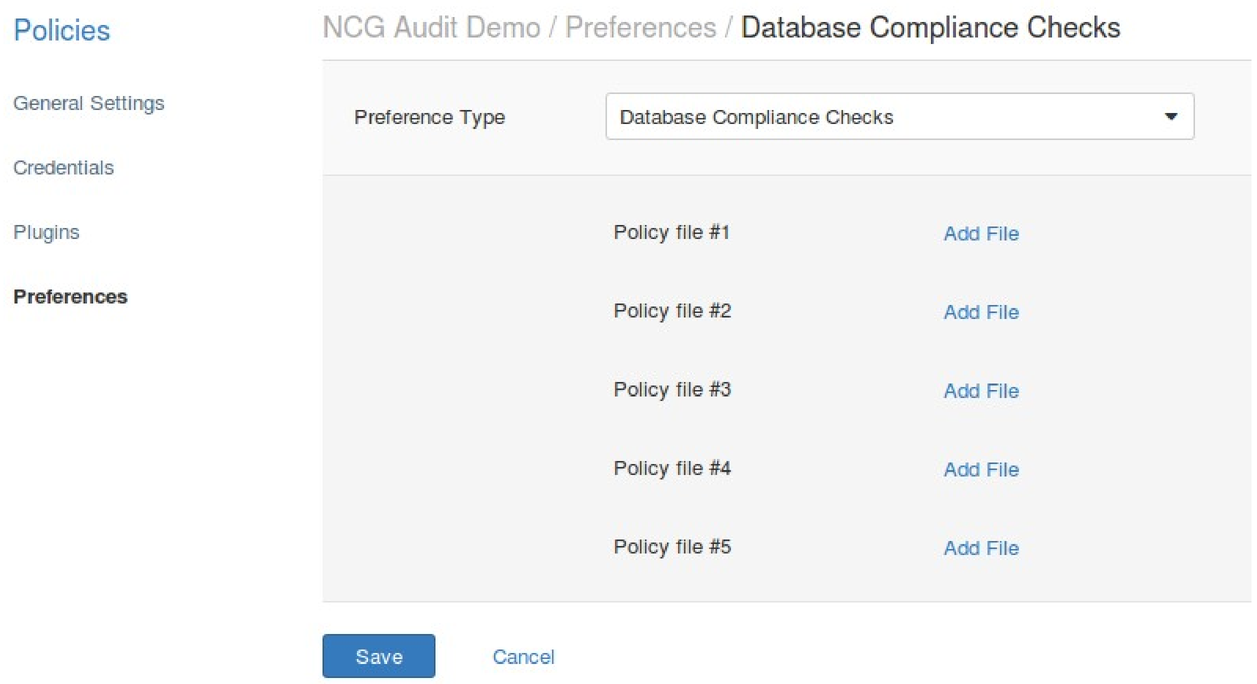
Database port to use:

Oracle auth type: NORMAL

SQL Server auth type: Windows

This is probably a good time to remind the user to be aware of security concerns. Once these credentials are entered, anyone with access to this check on Nessus can effectively run **any** sql query they'd like. So its important to both tightly control Nessus access and make sure that either you use an unprivileged db user to run audits or that the passwords are only entered before the audit is run.

Once you’ve entered the credentials, go back to the “Preference Type” pulldown menu and choose “Database Compliance Checks.” You’ll see the following screen:



Policies

- General Settings
- Credentials
- Plugins
- **Preferences**

NCG Audit Demo / Preferences / Database Compliance Checks

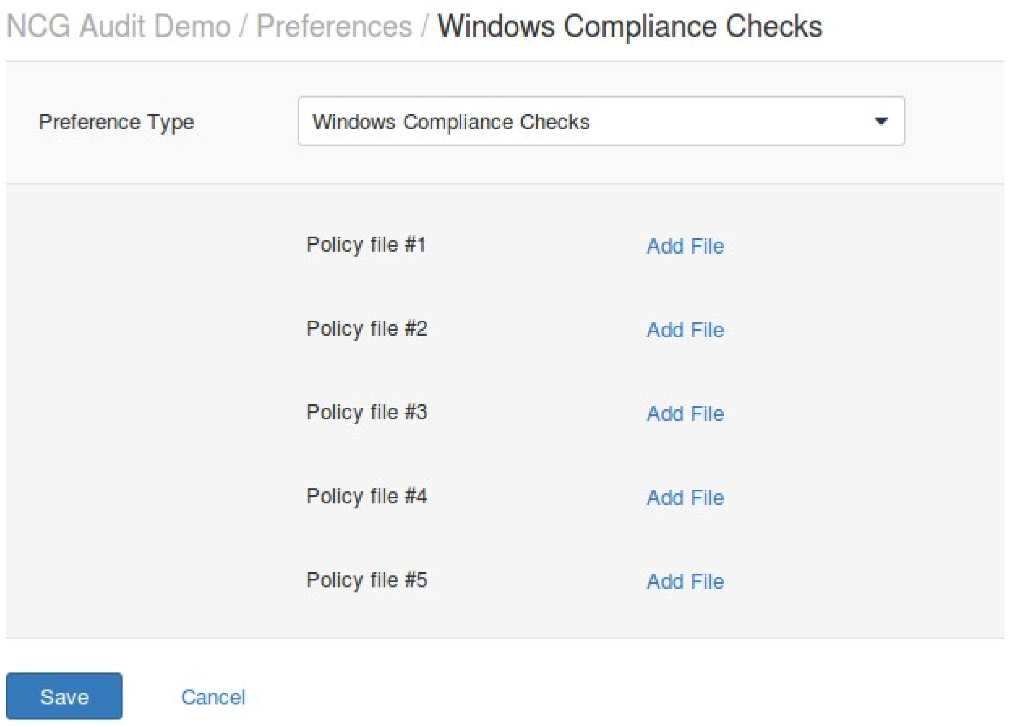
Preference Type: Database Compliance Checks

Policy file #1	Add File
Policy file #2	Add File
Policy file #3	Add File
Policy file #4	Add File
Policy file #5	Add File

Save Cancel

As you can see, Nessus allows you to choose up to five audit/policy files for a single policy. This is why we combined our separate audits into one large file.

We'll need to do the same thing for our windows audits. From the pulldown menu, choose “Windows Compliance Checks” (there is an option for Windows File Compliance Checks, and though we are doing those, Nessus v2 audit files allow us to lump them all together). You'll see a similar screen here.

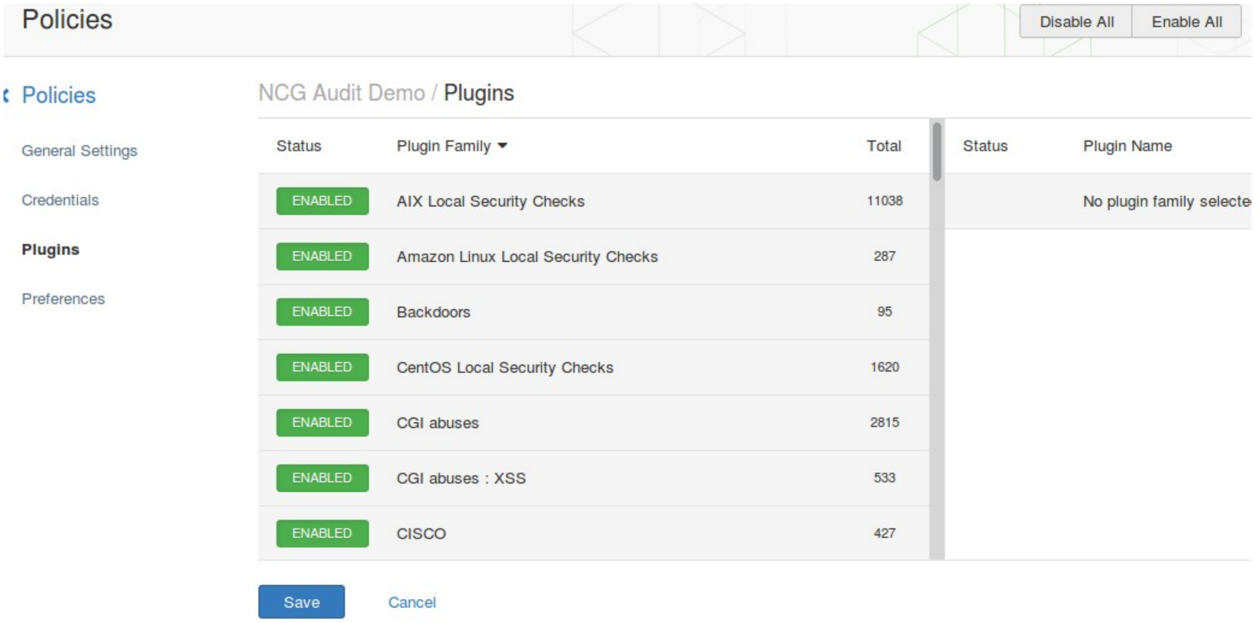


NCG Audit Demo / Preferences / Windows Compliance Checks

Preference Type: Windows Compliance Checks

In both cases, once the files are uploaded, they stay with the audit until you remove or add to them. Remember to delete the old one and upload new ones each time you make a change.

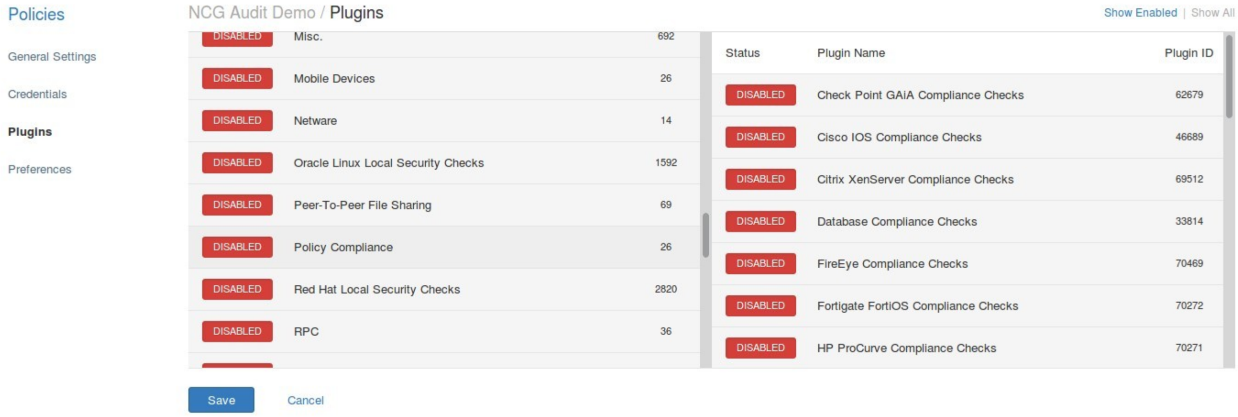
Before we go on to run our scans, we want to optionally disable all other tests, otherwise Nessus will try to run *every* test with our compliance check, which is probably not what we want. To do this, Click “Plugins” on the left hand side. You’ll see the screen below:



Status	Plugin Family	Total
ENABLED	AIX Local Security Checks	11038
ENABLED	Amazon Linux Local Security Checks	287
ENABLED	Backdoors	95
ENABLED	CentOS Local Security Checks	1620
ENABLED	CGI abuses	2815
ENABLED	CGI abuses : XSS	533
ENABLED	CISCO	427

Status	Plugin Name
	No plugin family selecte

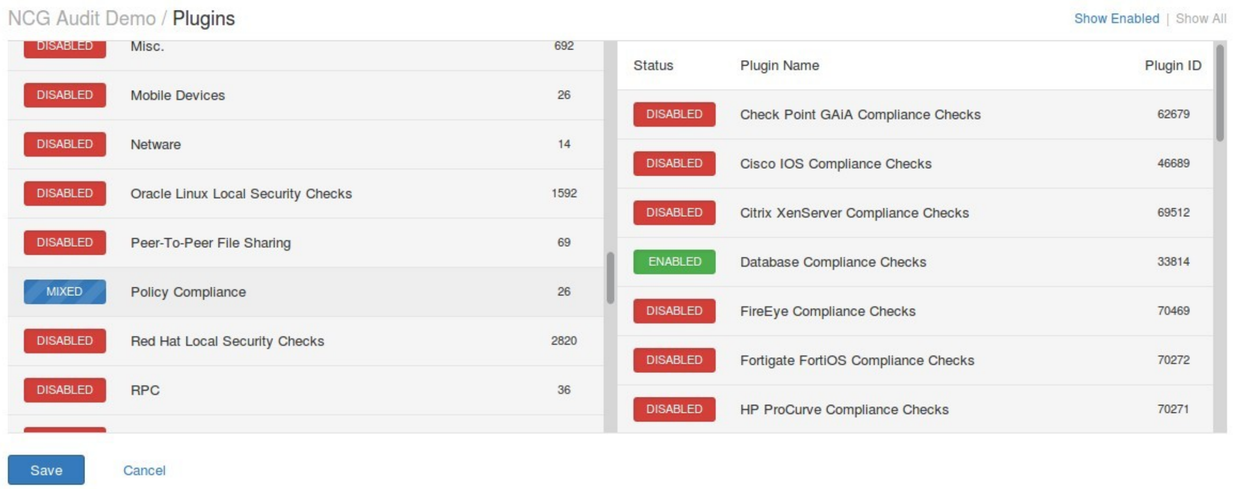
Click the button on the top right that says “Disable All”, then scroll down and click on “Policy Compliance.” You’ll see a screen like below:



DISABLED	Misc.	692
DISABLED	Mobile Devices	26
DISABLED	Netware	14
DISABLED	Oracle Linux Local Security Checks	1592
DISABLED	Peer-To-Peer File Sharing	69
DISABLED	Policy Compliance	26
DISABLED	Red Hat Local Security Checks	2820
DISABLED	RPC	36

Status	Plugin Name	Plugin ID
DISABLED	Check Point GAiA Compliance Checks	62679
DISABLED	Cisco IOS Compliance Checks	46689
DISABLED	Citrix XenServer Compliance Checks	69512
DISABLED	Database Compliance Checks	33814
DISABLED	FireEye Compliance Checks	70469
DISABLED	Fortigate FortiOS Compliance Checks	70272
DISABLED	HP ProCurve Compliance Checks	70271

In the list on the right, you’ll see an item called “Database Compliance Checks” with a button next to it that says “disabled”… Click “disabled” and it should become “enabled”:

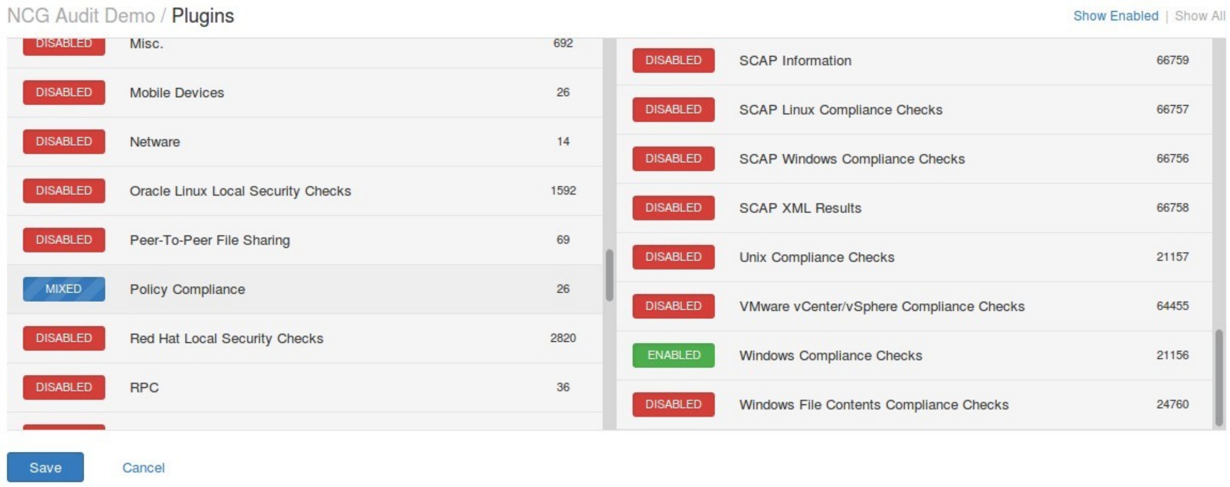


NCG Audit Demo / Plugins

DISABLED	Misc.	692
DISABLED	Mobile Devices	26
DISABLED	Netware	14
DISABLED	Oracle Linux Local Security Checks	1592
DISABLED	Peer-To-Peer File Sharing	69
MIXED	Policy Compliance	26
DISABLED	Red Hat Local Security Checks	2820
DISABLED	RPC	36

Status	Plugin Name	Plugin ID
DISABLED	Check Point GAiA Compliance Checks	62679
DISABLED	Cisco IOS Compliance Checks	46689
DISABLED	Citrix XenServer Compliance Checks	69512
ENABLED	Database Compliance Checks	33814
DISABLED	FireEye Compliance Checks	70469
DISABLED	Fortigate FortiOS Compliance Checks	70272
DISABLED	HP ProCurve Compliance Checks	70271

If you're doing windows compliance checks, you'll also need to scroll down further and enable that. Though we are doing windows file contents compliance checks, with Nessus v2 audit scripts, they can all be captured by the standard “Windows Compliance Checks” option (Windows File Contents Compliance need not be checked):

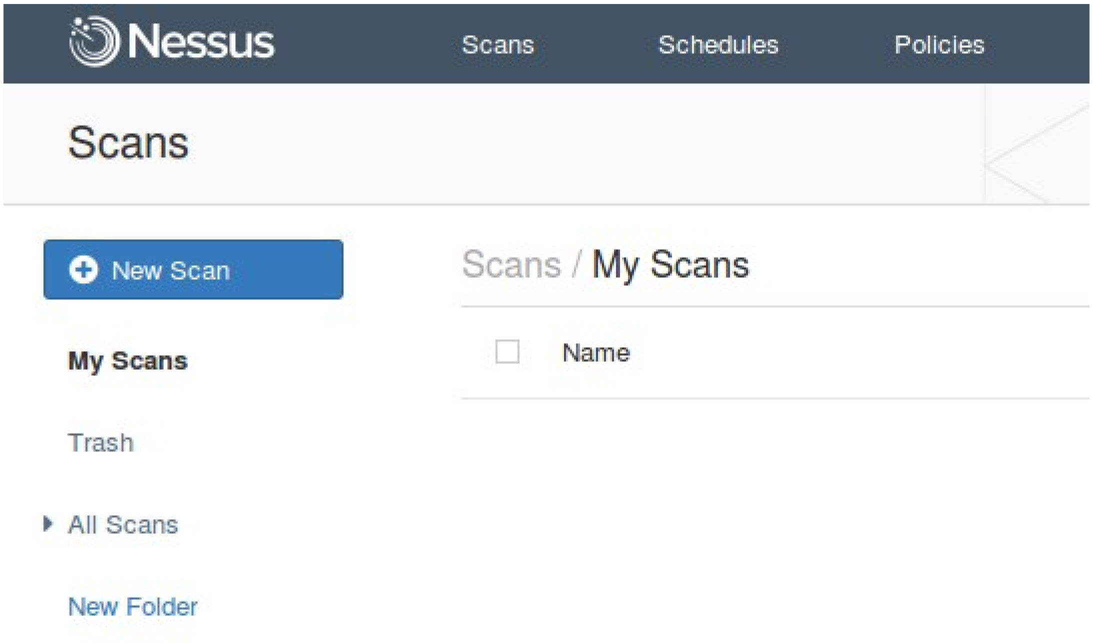


NCG Audit Demo / Plugins

DISABLED	Misc.	692
DISABLED	Mobile Devices	26
DISABLED	Netware	14
DISABLED	Oracle Linux Local Security Checks	1592
DISABLED	Peer-To-Peer File Sharing	69
MIXED	Policy Compliance	26
DISABLED	Red Hat Local Security Checks	2820
DISABLED	RPC	36

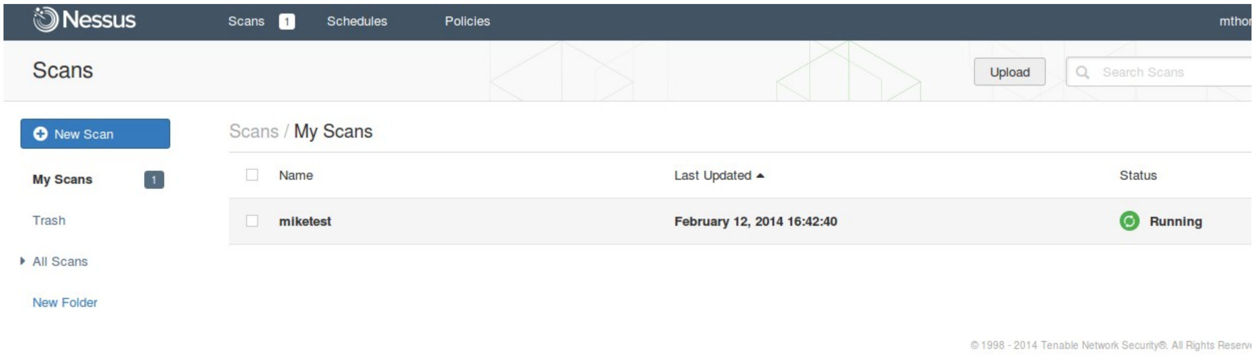
DISABLED	SCAP Information	66759
DISABLED	SCAP Linux Compliance Checks	66757
DISABLED	SCAP Windows Compliance Checks	66756
DISABLED	SCAP XML Results	66758
DISABLED	Unix Compliance Checks	21157
DISABLED	VMware vCenter/vSphere Compliance Checks	64455
ENABLED	Windows Compliance Checks	21156
DISABLED	Windows File Contents Compliance Checks	24760

Now, if you selected “Shared” earlier, anyone who has access to this Nessus instance should be able to run your audit. To run a test, click “Scans” at the top, you’ll see the screen below:



Click “+ New Scan,” Choose a name for this scan (I called mine “miketest”), . Under Policy, choose the policy name you gave earlier. I named my policy “CHEMS SQL Audit Test.” Add a target or a list of targets and press “Launch.”

You can enter your scan targets as IP Addresses or host names, or upload a space separated list of targets. After clicking “Launch” you should see something like the following:

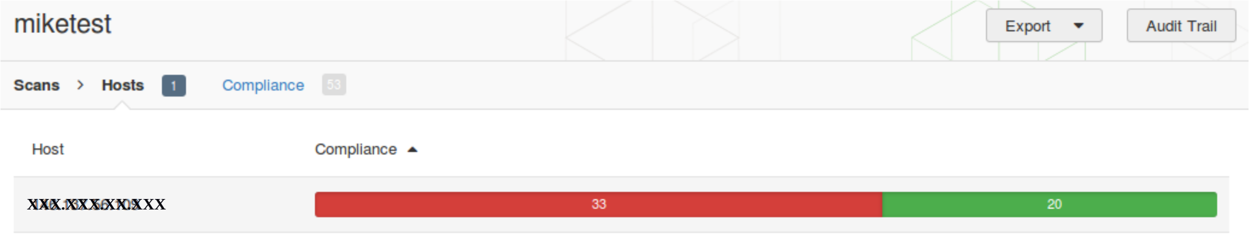


□	Name	Last Updated ▲	Status
□	miketest	February 12, 2014 16:42:40	Running

© 1998 - 2014 Tenable Network Security®. All Rights Reserv

Wait for”Running” to change to “Completed”, then click on the test name. You should see a summary of the results as below…

VI. Examining the Results.



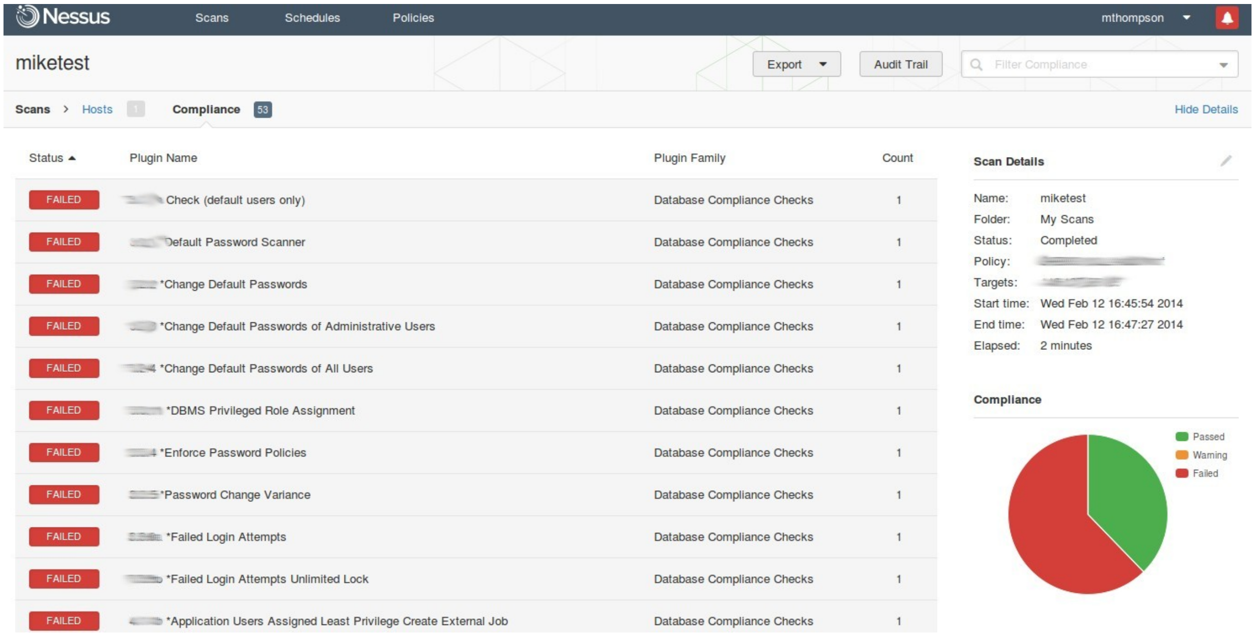
miketest

Export ▾ | Audit Trail

Scans › Hosts 1 | Compliance 53

Host	Compliance ▲
XXX.XXX.XXX.XXX	33 | 20

Clicking compliance will show us more detail:



Nessus | Scans | Schedules | Policies | mthompson ▾

miketest

Export ▾ | Audit Trail | Filter Compliance

Scans › Hosts 1 | Compliance 53 | Hide Details

Status ▲	Plugin Name	Plugin Family	Count
FAILED	[redacted] Check (default users only)	Database Compliance Checks	1
FAILED	[redacted]efault Password Scanner	Database Compliance Checks	1
FAILED	[redacted] *Change Default Passwords	Database Compliance Checks	1
FAILED	[redacted] *Change Default Passwords of Administrative Users	Database Compliance Checks	1
FAILED	[redacted] *Change Default Passwords of All Users	Database Compliance Checks	1
FAILED	[redacted] *DBMS Privileged Role Assignment	Database Compliance Checks	1
FAILED	[redacted] *Enforce Password Policies	Database Compliance Checks	1
FAILED	[redacted] *Password Change Variance	Database Compliance Checks	1
FAILED	[redacted] *Failed Login Attempts	Database Compliance Checks	1
FAILED	[redacted] *Failed Login Attempts Unlimited Lock	Database Compliance Checks	1
FAILED	[redacted] *Application Users Assigned Least Privilege Create External Job	Database Compliance Checks	1

Scan Details

Name: miketest
Folder: My Scans
Status: Completed
Policy: [redacted]
Targets: [redacted]
Start time: Wed Feb 12 16:45:54 2014
End time: Wed Feb 12 16:47:27 2014
Elapsed: 2 minutes

Compliance

Many of our checks show as failed. Failed checks can be used to positive ends by analyzing ambiguous responses to SQL queries. Many qudit procedures require a certain query output that cannot be programmatically checked, but needs to be looked at by an auditor. This is why its useful to think of a “Failed” check here as a “Failed” automated check. All that is required is for an auditor to click through each item and verify that the SQL output (which is shown) complies with what he/she expects--- Clicking on an individual audit item brings you to a description and the output of that audit

In this way, even audits that require manual checks can at least be automated to a large extent. The results can also be exported to a PDF file with full SQL output of all failed checks for ease of auditing.