Windows 7 Auditing: An Introduction

Todd Heberlein
14 June 2010

Windows 7's auditing system can provide a rich source of information to detect and analyze a wide range of threats against computer systems. Unfortunately few people know this auditing system exists much less how to turn it on and configure it. This paper provides step-by-step instructions to configure a simple audit policy useful for understanding how data was exfiltrated from the computer.

1 Introduction

Microsoft's Windows 7 operating system supports a powerful auditing system. When turned on and configured correctly it can serve as an important tool for detecting and understanding a wide range of threats against a computer system including insiders abusing their privileges and malicious software exfiltrating user data. In many ways the audit system can provide vital information that network monitoring, antivirus software, and post-mortem disk analysis simply cannot provide.

Unfortunately, few people are aware that the auditing system exists, what it is capable of, or how to turn it on and configure it. This paper walks you through the steps to turn on a simple auditing configuration. This simple configuration covers process creation, network connections, and reading user files. Although this is a relatively simple configuration, it requires many steps in numerous windows to set it up. All these steps are required in part because Windows 7 lets you fine tune the auditing system to meet very specific needs, but all this complexity can make it difficult to get started and sometimes easy to get it wrong. Thus Windows 7's highly configurable auditing system can be both a blessing and a curse. We hope this paper gives you enough information to get you started tailoring and configuring an audit policy meeting your specific needs.

Section 2 shows how to configure Windows 7 to use the new advanced audit policy. Section 3 shows how to use the Windows 7 advanced audit policy to collect the audit records of interest. For this paper we are looking for process creation and termination, file accesses, and network connections. Section 4 shows several audit records giving us confidence that we have configured the audit policy correctly. We use a simple exfiltration example (a sensitive file is sent out of the network), and show that the audit trail can show which file was exfiltrated, the program used to transfer the file, how that program was started, and what specific network activity was associated with this act. Finally, Section 5 summarizes the paper.

2 Use Advanced Audit Policy

In this section we show how to direct Windows 7 to use the advanced audit policy to collect information important to us. To configure auditing for Windows 7, use the *Local Security Policy* tool. To find this tool, open the *Control Panel*, select *System and Security* and then select *Administrative Tools*. In the main window (see Figure 1) select the *Local Security Policy* tool (label 1).

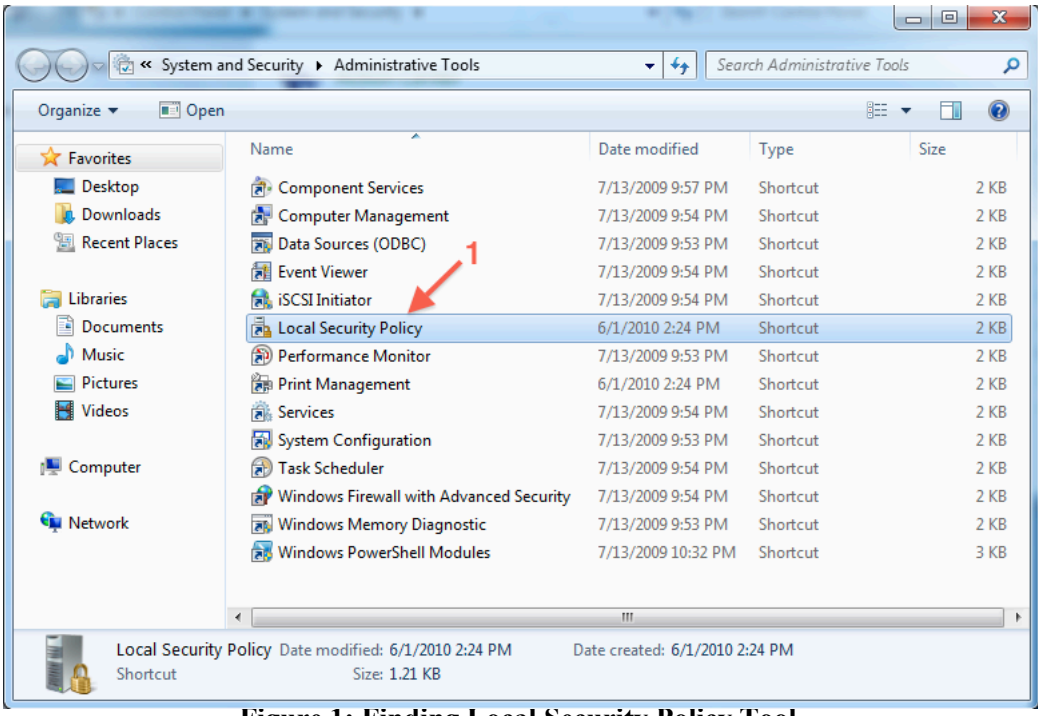

Figure 1: Finding Local Security Policy Tool

The *Local Security Policy* tool lets you configure numerous security elements, but we are only interested in auditing. In the left column, expand the *Local Policies* folder (see Figure 2). Three more folders are shown. *The Audit Policy* folder (label 1) is the old way to configure auditing in Vista. We recommend that you don't use this one for Windows 7. Windows 7 gives you a more powerful audit configuration capability generally referred to as the "advanced audit policy". To force Windows to use the "advanced audit policy" instead of the older audit policy configuration, select the *Security Options* folder (label 2), which displays a list of policy configurations in the larger column on the right. Double click *Audit: force audit policy subcategory settings* (label 3).

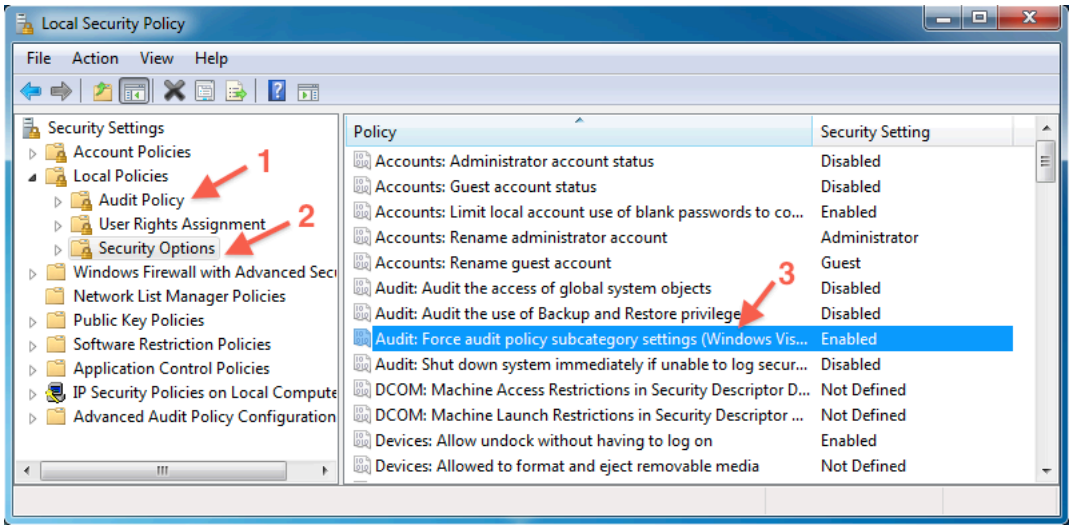

Figure 2: Force The Use of Advanced Audit Policy

The double click will bring up the window shown in Figure 3. Enabling "subcategory" will cause Windows to ignore the settings from the older Audit Policy and use the Advanced

Audit Policy Configuration. This gives us much finer control over what is and is not audited. After selecting the "Enabled" radio button, the operating system will use the advanced audit policy.

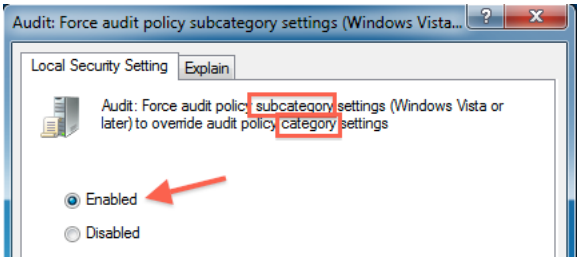

Figure 3: Enabling Advanced Audit Policy

3 Turn On Auditing For Specific Events

Now that we are using Windows 7's advanced audit policy configuration, we need to set this policy to collect audit records of interest. Returning to *the Local Security Policy's* main window, expand the *Advanced Audit Policy Configuration* folder (label 1 in Figure 4) and then expand the *System Audit Policies* folder. This shows several groups of auditing features. We can zoom into individual groups to turn on auditing for just the events we are interested in.

3.1 Auditing Process Creation and Termination

We start by telling the auditing system to tell us which programs were run and how they were started. Select the *Detailed Tracking* group (label 2 in Figure 4), and the window's second column will show several subcategories of auditing events that can be turned on or off. Double click on *Audit Process Creation* (label 3), and then enable Success and Failure events (Figure 5). Repeat these steps for *Audit Process Termination* (label 4). These are the two auditing features that let us track process creation and termination. For example, if you run the command shell and enter the command "ftp" to transfer a file across the network, Windows 7 will generate a record stating that you ran the ftp program from the command shell and at what time. When you quit the ftp program, Windows 7 will generate a record saying the program has terminated.

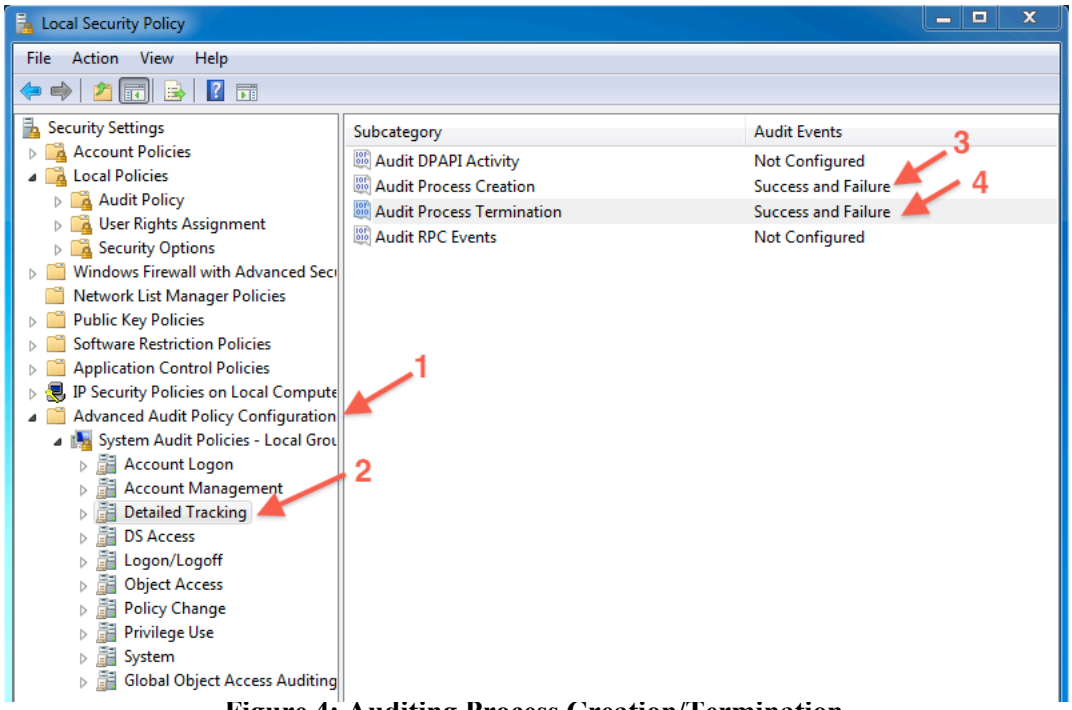

Figure 4: Auditing Process Creation/Termination

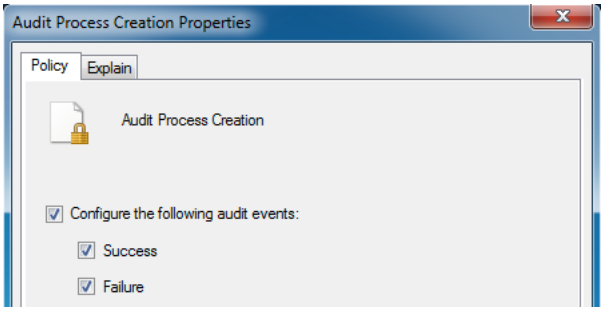

Figure 5: Turn On Events

3.2 Auditing Network Connections

Next we turn on auditing for network connections. Returning to *Local Security Policy's* main window, select the *Object Access* group in the left column (label 1 in Figure 6). The audit category lets us activate auditing for a rather wide range of events shown in the right column. Double click on *Audit Filtering Platform Connection* (label 3) and it will bring up a window similar to Figure 5. Enable successful and failed events. With this configuration when an application makes an outbound connection, Windows 7 will generate an audit record stating which application launched the connection, when it occurred, and provide identifying information about the connection (e.g., source and destination address and ports).

3.3 Auditing File Accesses

Next we turn on auditing for accessing files on the computer. Returning to *Local Security Policy's* main window and the *Object Access* group, double click *Audit File System* in the right column (label 2 in Figure 6). In the window that pops up, enable successful and failed events. With this setting in theory when an application reads (or writes) a file on the computer, Windows 7 will generate an audit record describing the program that is reading the file, what file is actually

being read, and when the program initially opened the file. We say "in theory" because additional steps are needed to make this work. These steps are described in Section 3.4.

To know when an application is finished with a file, return to the *Local Security Policy's Object Access* group, double click on *Audit Handle Manipulation* (label 4 in Figure 6), and enable successful and failed events for this subcategory. With this subcategory enabled, Windows 7 will generate an audit record with the program closes the file.

After closing the *Audit Handle Manipulation* audit event window we are finished with the *Local Security Policy* program, so close that window.

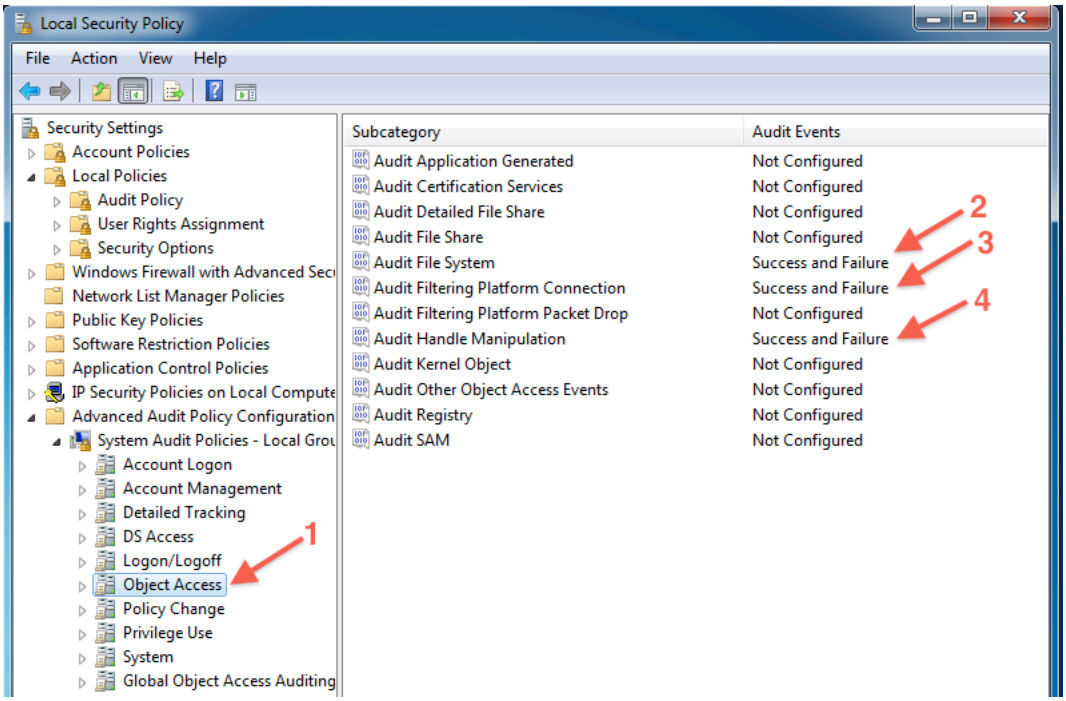

Figure 6: Configure File and Network Activity

3.4 Finishing File Access Auditing

At this point, in theory, we are auditing process creation and termination, connection attempts, and file opens and closes. However, auditing file accesses requires two things: (1) you saying you want to audit file accesses (what we just did) and (2) specifying which folders and files you want to audit. Enabling auditing on the file system will not actually audit folder and file accesses without doing another second step, and that is what is covered in this section.

For Windows 7 to generate and audit record when you access to a folder or file, the folder or file needs to have a *system access control list* (SACL). A SACL is a type of access control list that is attached to a folder or file, and entries in the SACL specify which users and actions by those users should generate audit records. Two files can generate different sets of audit records when accessed in the exact same way if they have different SACLs. On one hand, this is nice because it can let you really fine tune what is audited, but on the other hand, the additional complexity means it is more likely for you to screw up configuring your system and miss important things you would like to have audited. For our purposes here, we show how to turn on auditing for a single user accessing a specific folder as well as files and sub-folders below that folder.

We will use the folder "Experiments" in the Documents folder, see Figure 7, label 1. Right click the folder and select Properties. In the Properties window select the *Security* tab (label 2) and click the *Advanced* button (label 3).

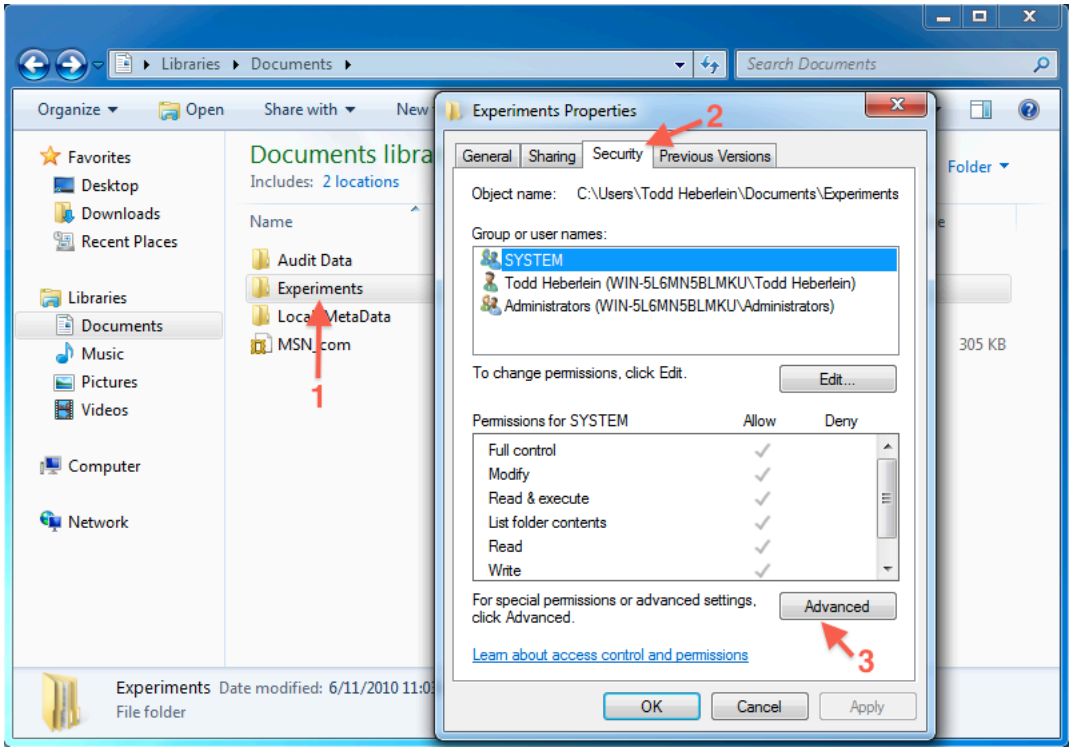

Figure 7: Directory Properties

After clicking the *Advanced* button the *Advanced Security Settings* window pops up (see Figure 8). Select the *Auditing* tab and click the *Continue* button (label 2).

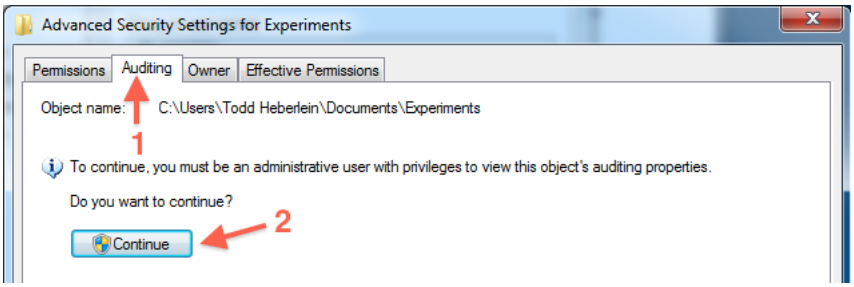

Figure 8: Security Settings for Folder

After clicking the *Continue* button the *Auditing* window pops up (see Figure 9). The list labeled 1 is where each item for the SACL is shown; the list is currently empty. Click the *Add...* button (label 2) to add an entry to the SACL for this folder.

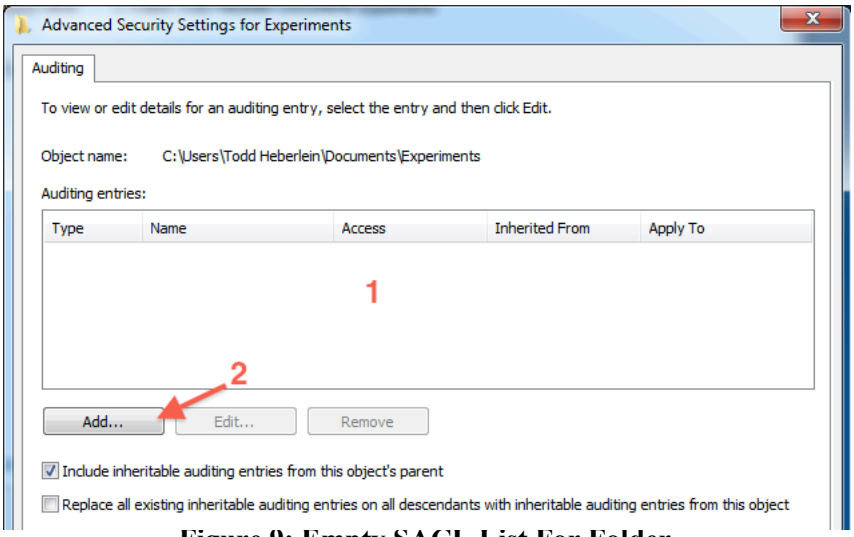

Figure 9: Empty SACL List For Folder

After clicking the *Add*... button another window pops up that lets you select the subject (e.g., a user or group, see Figure 10, label 1) for which you want to generate audit records. In the text field at the bottom (label 2) enter a user name; I used my name "Todd Heberlein", and then click the *Advanced*... button.

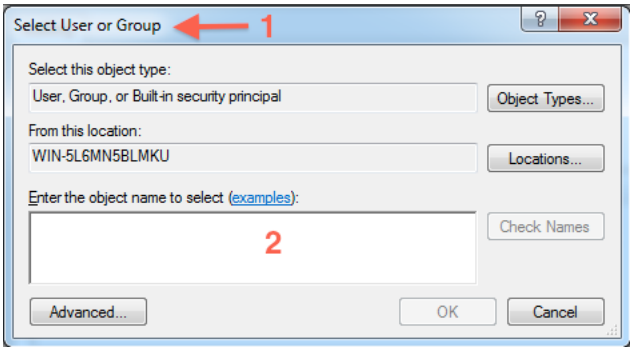

Figure 10: Select User For The SACL Item

Another window pops up (see Figure 11) letting you select the actions that will create audit records. Label 1 shows the subject, in this case "Todd Heberlein". Label 2 shows that we want these actions to generate audit records for not just this folder but all sub-folders and files inside this folder. There are lots of options to determine exactly what actions you want to generate an audit record for, but for this example we just selected the *Full control* boxes at the top (label 3) which causes all the rest of the boxes to be checked. Finally, click *OK*.

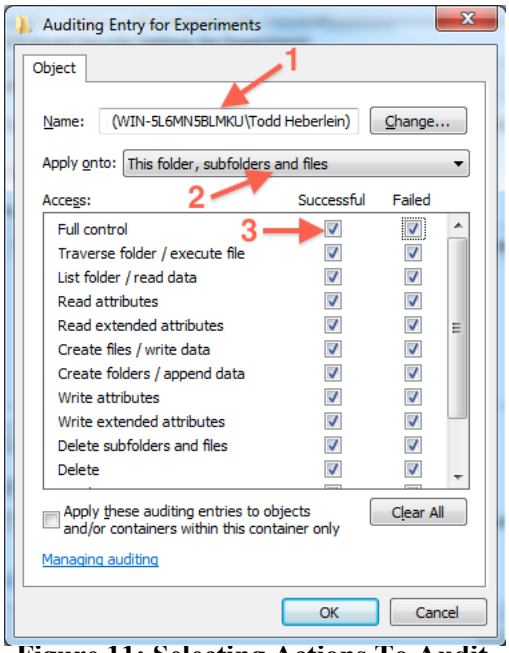

Figure 11: Selecting Actions To Audit

Now when we return to the Auditing window for the Experiments folder we see there is a single entry in the SACL (Figure 12, label 1).

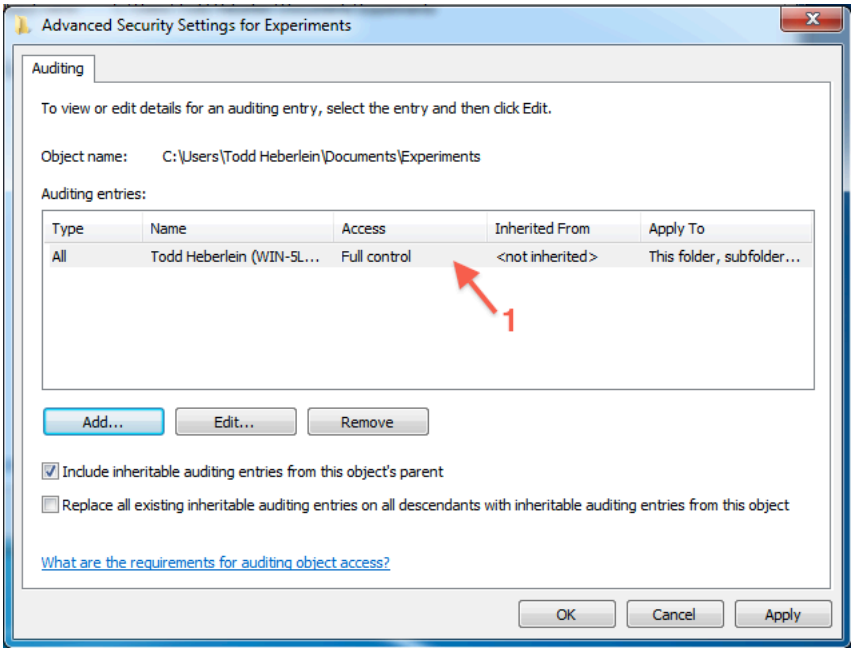

Figure 12: Entry For A SACL

To verify that the audit configuration is recursively applied to all sub-folders and files below the "Experiments" folder, we have pulled up the SACL list for the sub-folder "Test1" (Figure 13, label 1). In the SACL list we see an entry and that it has been inherited from a parent folder (label 2). This lets us know actions on involving this folder, including reading files contained in this folder, will generate audit records.

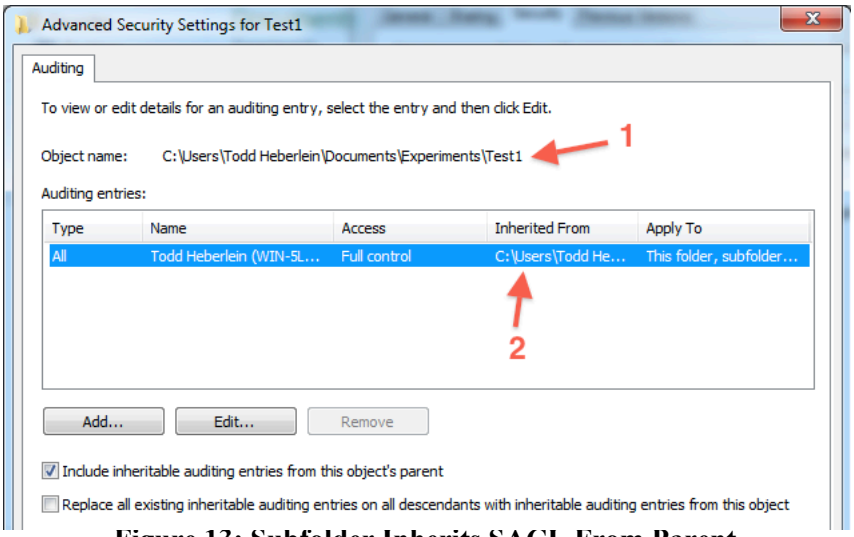

Figure 13: Subfolder Inherits SACL From Parent

4 Verify Audit Records Are Being Generated

To verify that Windows 7 is in fact generating the expected audit records, we conducted the following experiment. We started the Windows *command* program (a.k.a, the DOS prompt), from the command program we ran the *ftp* program connecting to a remote server, and we uploaded the file military_map.png. Next we ran the Window 7 *Event Viewer* program to review the audit records. We identified key audit records of interest showing important steps in the experiment. In Figure 14 – Figure 16 we show part of the details of those audit records.

Figure 14 shows the creation of the FTP process (label 1). The parent process, process ID 600 (displayed in hexadecimal as 0x258, see label 2), is the command shell cmd.exe. It creates the process 2604 (0xa2c, label 3) running the program ftp.exe (label 4).

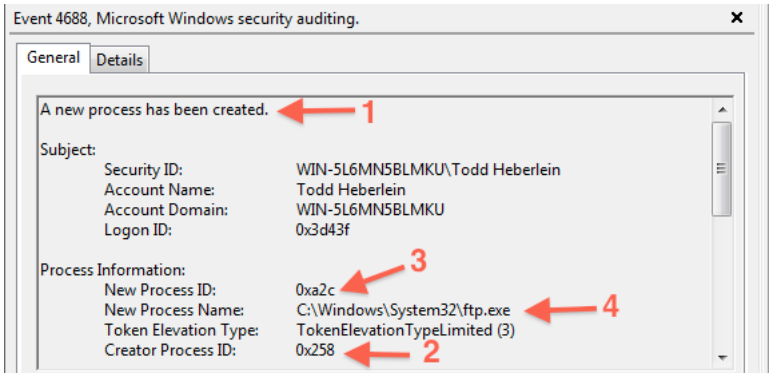

Figure 14: FTP Program Launched

Figure 15 shows the FTP process, process ID 2604 (label 1), making an outbound TCP/IP connection to port 21 on the server 192.168.10.69 (label 2). The TCP/IP information could later be used to correlate this activity with analysis performed by a network monitor or firewall.

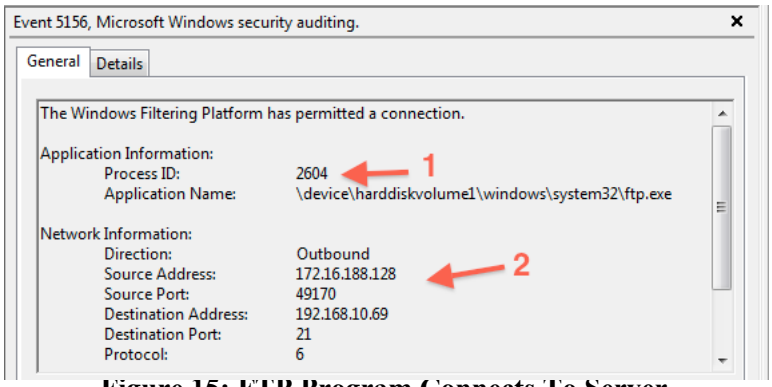

Figure 15: FTP Program Connects To Server

Figure 16 shows the FTP process, process ID 2604 (label 1), performing a read operation (label 2) on the file military_map.png (label 3). This is the file we uploaded to the remote FTP server. So now we know that the user "Todd Heberlein" ran the FTP program from the command shell and uploaded the file "military_map.png" to the remote server 192.168.10.69.

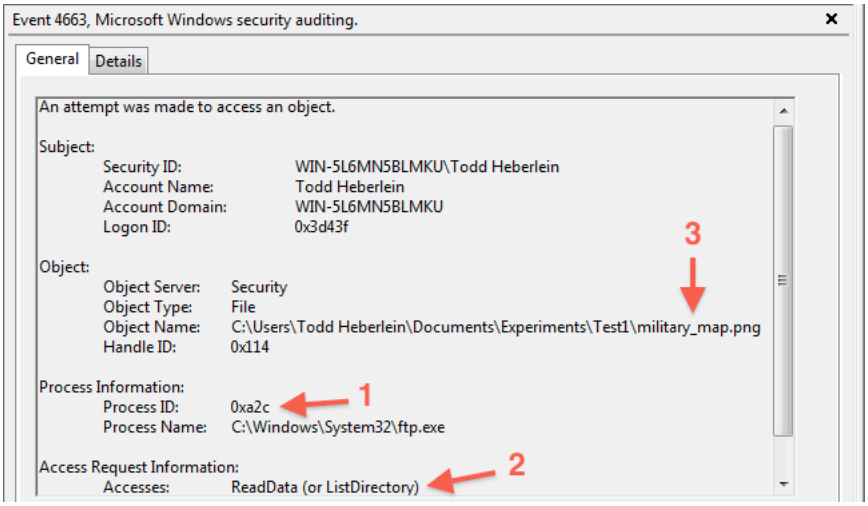

Figure 16: FTP Uploads The File military_map.png

5 Conclusions

Windows 7 supports a powerful and highly configurable auditing system. With this auditing system turned on and properly configured, attacks on and misuse of the computer can be detected and analyzed in a way that cannot be done by network monitors, antivirus tools, and post mortem disk forensics tools. However, few people are aware of the auditing system, its capabilities, or how to configure it.

This paper showed how to direct Windows 7 to use the advanced audit policy and then how to configure the policy to generate records for several interesting events – namely, process creation, network activity, and file accesses. Configuring Windows 7's auditing system is, unfortunately, non-trivial. In the example shown in this paper, just to configure the simple audit policy we accessed 13 different windows.

Finally, with the audit configuration in place, we conducted a simple experiment where an insider uploaded a sensitive file across the network, and we showed how the audit trail contained the information to reconstruct his activities.