**Monitoring Checklist of both Linux and Windows:**

**\*All windows will be done later, focus is strictly on linux for now\***

1. Update Machine (**IF INTERNET IS AVAILABLE**)
   1. Debian, Ubuntu, Kali
      1. Apt-get update **OR** apt-get upgrade
   2. Redhat, YellowDog, CentOS, Scientific Linux, Fedora
      1. Yum update **OR** yum list updates
   3. Windows
      1. Need to find what to do for updating windows
2. Change Password/minimum length of password
   1. Linux
      1. Change lifetime in “vi /etc/pam.d/common-password
         1. Change “password requisite pam\_unix.so nullok obscure sha512” to
         2. “password requisite pam\_unix.so nullok obscure sha512 min=(minimum character length)”
   2. Windows
3. Examine startup scripts (Not sure if this is useful)
   1. Linux
      1. rc.\*(as rc.1-6 **OR** rc1-6.d):
   2. Windows
4. Ensure path is secure
   1. linux
      1. Make sure current directory is not in the PATH
      2. Ensure there are no empty strings in the PATH (this allows attackers to take advantage of ./ commands to attack a machine)
   2. Windows
5. Ensure only scripts running are NEEDED for the service to be functional
   1. Linux
      1. /sbin/chkconfig --list |grep ‘3:on’
         1. To remove unwanted scripts use “chkconfig **SERVICENAME** off”
6. Deny use of Cron (may or may not be useful)
   1. Linux
      1. Echo ALL >>/etc/cron.deny
         1. Prevents all users from using or tampering with the cron application
7. Ensure all users have active passwords
   1. Linux
      1. Cat /etc/shadow | awk -F: ‘($2==””) {print $1}’
      2. Not 100% sure how this command works
8. Add timeout for users not active
   1. Linux
      1. Typeset -r TMOUT=900 (15 minutes =900 seconds)
   2. Windows
9. Ensure permissions are set for certain files and services
   1. Linux
      1. Chmod **number** (file name)
         1. NUMBER= the combination and permissions each set of users have within the system. (**4 allows reading, 2 allows writing and 1 allows for executing**)
            1. Each section of number separates different user levels, the first digit represents the root directory, the second is specified users, and the last is the general public.
            2. 744 would represent root being able to read, write and execute, but both users and public would only allow to read the file
   2. Windows
10. Deny write capabilities in /boot processes (not sure if this would present problems for system as a whole)
    1. Linux
       1. Open vi /etc/fstab
       2. Then add “LABEL=/boot /boot ext2 defaults,ro 1 2” to bottom of file

**Monitoring Software after updating and hardening**

1. Monitor running processes
   1. Linux
      1. System processes (top / htop)
      2. Apache Monitoring (apachetop)
      3. FTP monitoring (ftptop)
      4. SQL monitoring (mytop)
   2. Windows
2. Monitor Desktop processes
   1. Linux
      1. **If Internet is Available** (ntopng)
         1. Provides windowed server to analyze ip address connections, usage by each one, and where they originated from.
      2. Built in service similar to wireshark (iftop/jnettop)
         1. Watches packets being sent and received from the computer
      3. Monitor Connections (netstat)
         1. For focusing on one specific tcp packet (justsniffer/tcpdump)
      4. Ultimately if possible download and use wireshark because we have all previously used it and provides the most information for packet and transport layer communications
   2. Windows
3. Monitor entire system
   1. Linux
      1. Monitorix
         1. Provides default http page with information about the current system it is downloaded on, reports on all mechanisms within the machine
      2. Uptime
         1. Shows the information on run time and how many users are currently logged on. (can be helpful to ensure only the users you know are on the system)
      3. acct / psacct
         1. Acct (for when the system uses apt-get)
            1. Monitors the commands that are entered on the machine (can be used to ensure commands that are malicious are not entered and if they are quickly seen)
         2. Psacct (for when the system uses yum)
            1. Monitors the commands that are entered on the machine (can be used to ensure commands that are malicious are not entered and if they are quickly seen)
   2. Windows
4. Common log files within **linux** and where to find them
   1. /var/log/message – Where whole system logs or current activity logs are available.
   2. /var/log/auth.log – Authentication logs.
   3. /var/log/kern.log – Kernel logs.
   4. /var/log/cron.log – Crond logs (cron job).
   5. /var/log/maillog – Mail server logs.
   6. /var/log/boot.log – System boot log.
   7. /var/log/mysqld.log – MySQL database server log file.
   8. /var/log/secure – Authentication log.
   9. /var/log/utmp or /var/log/wtmp : Login records file.
   10. /var/log/yum.log: Yum log files

**What to do if System is compromised (quick steps to lockdown service without shutting it down)** (idk if this will be useful)

1. Lock users from access to any particular system
   1. Passwd -l **USERNAME** (locks a user from access to the system)
   2. Passwd -u **USERNAME** (unlocks the user and gives them access to system again)

**Windows**

**Event Logging API (Windows 2003, XP, 2000)**

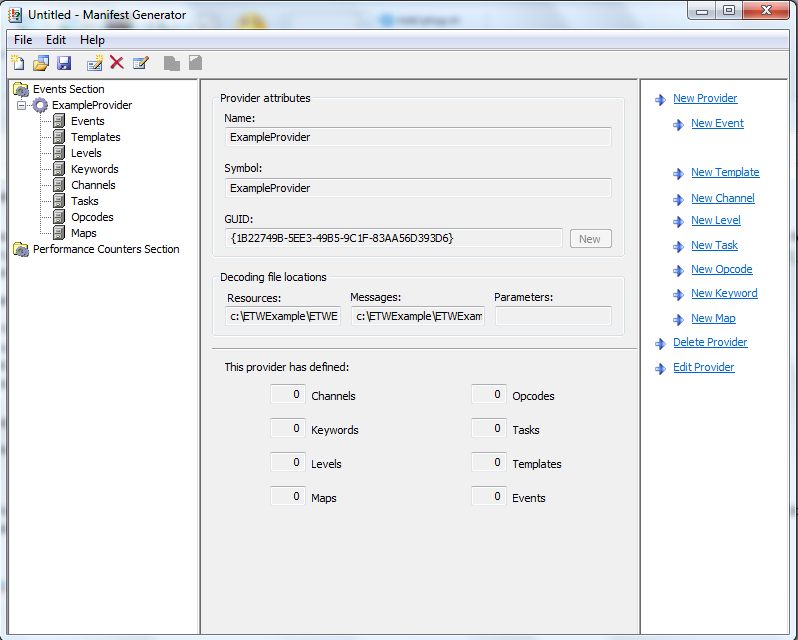
* Use the following C++ code to query for events logged in API
  + <https://msdn.microsoft.com/en-us/library/windows/desktop/bb427356(v=vs.85).aspx>
  + Puts all logged events on a buffer and prints the contents of the buffer
* Use the following C++ code to receive notifications of event logs
  + <https://msdn.microsoft.com/en-us/library/windows/desktop/aa363677(v=vs.85).aspx>
  + Loops through all event logs and if a new event appears, displays a notification

**Windows Event Log (Vista-Win10)**

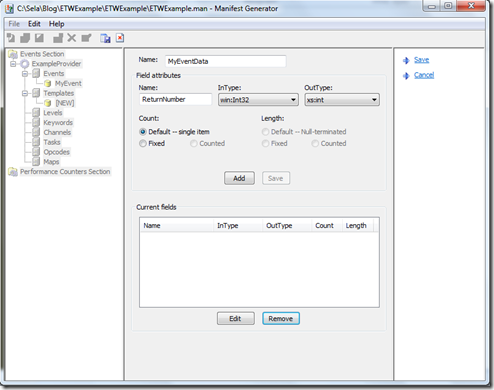
* Download Windows SDK
  + <https://www.microsoft.com/en-us/download/details.aspx?id=8279>
  + Windows tool kit that allows users to create applications. We’ll use this to create an application for log monitoring
* Choose which events your event manifest will record
  + Recommended:
    - Authentication successes/ failures
    - Authorization failures
    - Application and related systems start-ups and shut-downs, and logging initialization (starting, stopping or pausing)
    - Use of higher-risk functionality e.g. network connections, addition or deletion of users, changes to privileges, assigning users to tokens, adding or deleting tokens, use of systems administrative privileges, etc.
* Create an event manifest
  + Run ECManGen.exe from Windows SDK \bin directory

1. **Create a new event provider.** This step requires two actions:

* First, select the “Events Section” node in the far left pane and then “New Provider” in the far right pane.
* Second, give the provider a name, symbol, and GUID. Set the Resources and Messages boxes to the full path where the application will be installed. Click “Save” in the far right pane.
  + Use the example from the example section for each of the events you are going to be putting in your event manifest

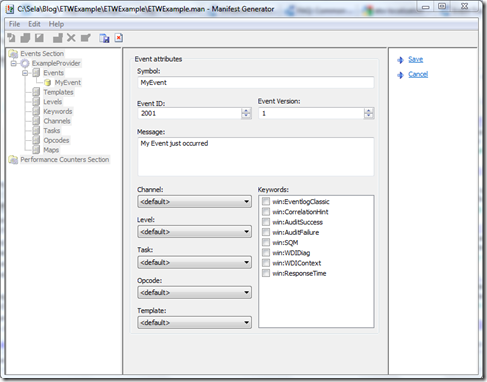


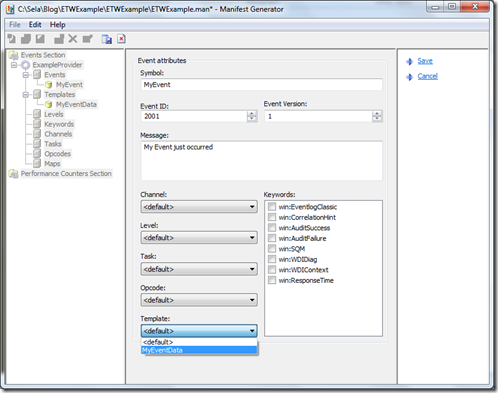
2. **Create a new template to define parameter types for each message that requires input parameters.**



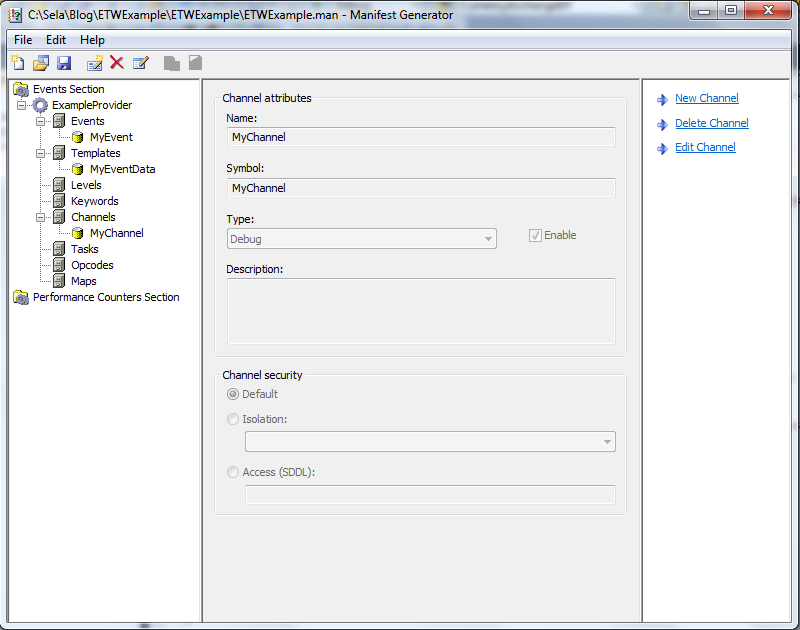
3. **Define metadata used to tag events.**

4. **Create events by giving each a symbol, event ID, message text and template.** When you write the message text, use %1, %2, %3, …, %12 to specify the positions of each parameter. Make sure the event template matches the format of the message text with parameters. Assign any channels, levels, tasks, keywords, and opcodes which apply to the event. Save the event. Do the same for all remaining event messages your application will produce.





5. **Save the event manifest to your project directory.**



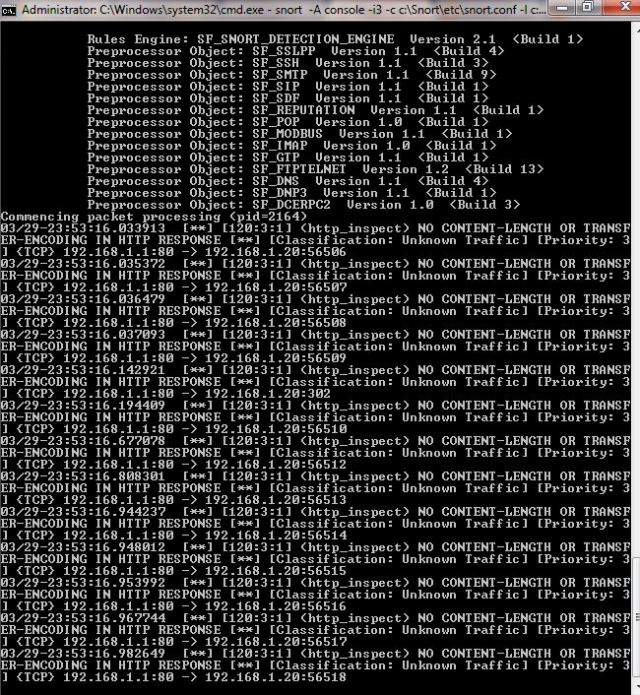
**Windows Defender**

* Download most recent version
  + Note:this will not run on anything earlier than Windows XP SP2
  + https://www.microsoft.com/en-us/windows/comprehensive-security
* Set up a more frequent scan to search for malware
  + Search for and open schedule tasks
  + In the left pane, expand **Task Scheduler Library** > **Microsoft** > **Windows** and then scroll down and double click the **Windows Defender** folder
  + In the top center pane, double click **Windows Defender Scheduled Scan**
  + Select the **Triggers** tab, and then select **New**
  + Set your time and frequency (every half hour?) and then select **OK**

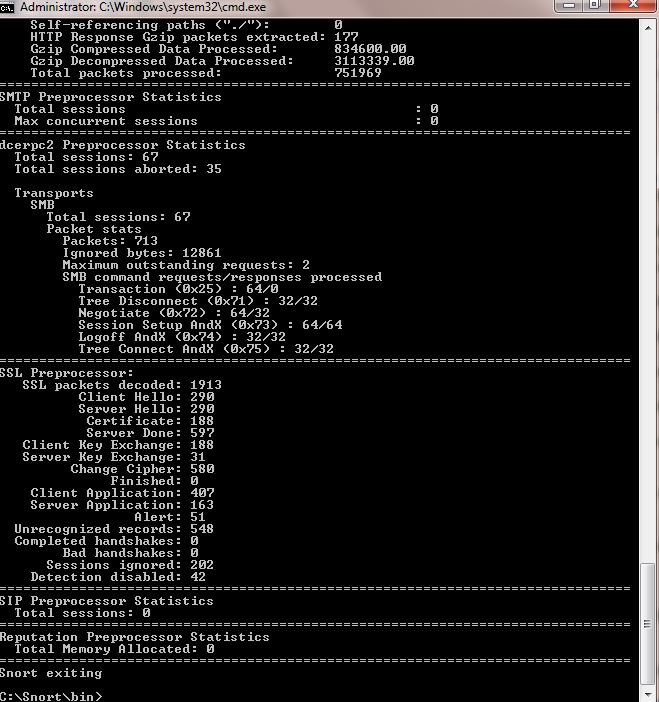
**Snort (IDS mode)**

* Ensure you have WinPcap installed
  + https://www.winpcap.org/install/
* Download most recent version from snort.org/snort-downloads
* Download Rules
  + <https://www.snort.org/snort-rules>
  + Extract Rules file
  + Copy and paste extracted files to C:\Snort\rules
  + Copy the “snort.conf” file from the “etc” folder of the extracted folder into “C:\Snort\etc”
* Edit the conf file
  + var HOME\_NET 192.168.1.0/24
    - Change the address on this line to your network address
  + Repeat for DNS servers
* Open a command prompt and navigate to “C:\Snort\bin”
* Change the RULE\_PATH variable to the path of rules folder.
  + It should now read “var RULE\_PATH c:\snort\rules”
* Change the path of all library files with the name and path on your system. and you must change the path of snort\_dynamicpreprocessorvariable.
* C:\Snort\lib\snort\_dynamiccpreprocessor
  + Do this to all library files in the “C:\Snort\lib” folder. The old path might be: “/usr/local/lib/…”. you will need to replace that path with your system path. Using C:\Snort\lib
* Change the path of the “dynamicengine” variable value in the “snort.conf” file
* Add the paths for “include classification.config” and “include reference.config” files.
  + include c:\snort\etc\classification.config
  + include c:\snort\etc\reference.config
* Remove the comment (#) on the line to allow ICMP rules, if it is commented with a #.
  + include $RULE\_PATH/icmp.rules
* You can also remove the comment of ICMP-info rules comment, if it is commented.
  + include $RULE\_PATH/icmp-info.rules
* To add log files to store alerts generated by snort, search for the “output log” test in snort.conf and add the following line:
  + output alert\_fast: snort-alerts.ids
* Comment (add a #) the whitelist $WHITE\_LIST\_PATH/white\_list.rules and the blacklist
* Change the nested\_ip inner , \ to nested\_ip inner #, \
* Comment out (#) following lines:
  + #preprocessor normalize\_ip4
  + #preprocessor normalize\_tcp: ips ecn stream
  + #preprocessor normalize\_icmp4
  + #preprocessor normalize\_ip6
  + #preprocessor normalize\_icmp6
* Save snort.conf file
* Run on command line
* snort -c c:\snort\etc\snort.conf -l c:\snort\log -i 3

Example output:



After done scanning:



**(Windows Hardening)**

* Change all default passwords
* Have firewall filter on port 445
  + Defends against ms08\_067\_netpai
* In the Security Policy Editor under “Local Policies->User Rights Assignment” there is a policy called “Deny access to the computer from this network”
  + Ensure this is enabled to defend against remote access to local accounts, specifically admin
* Set the RunAsPPL registry key for the LSA
  + Prevents code injection that can compromise credentials
    - Prevents code injection that can compromise credentials
      * Open the registry editor (regEdit.exe) and navigate to the registry key that is located at: HKEY\_LOCAL\_MACHINES\SYSTEM\CurrentControlSet\Control\Lsa
      * Set value of the registry key to “RunAsPPL”=dword:00000001.