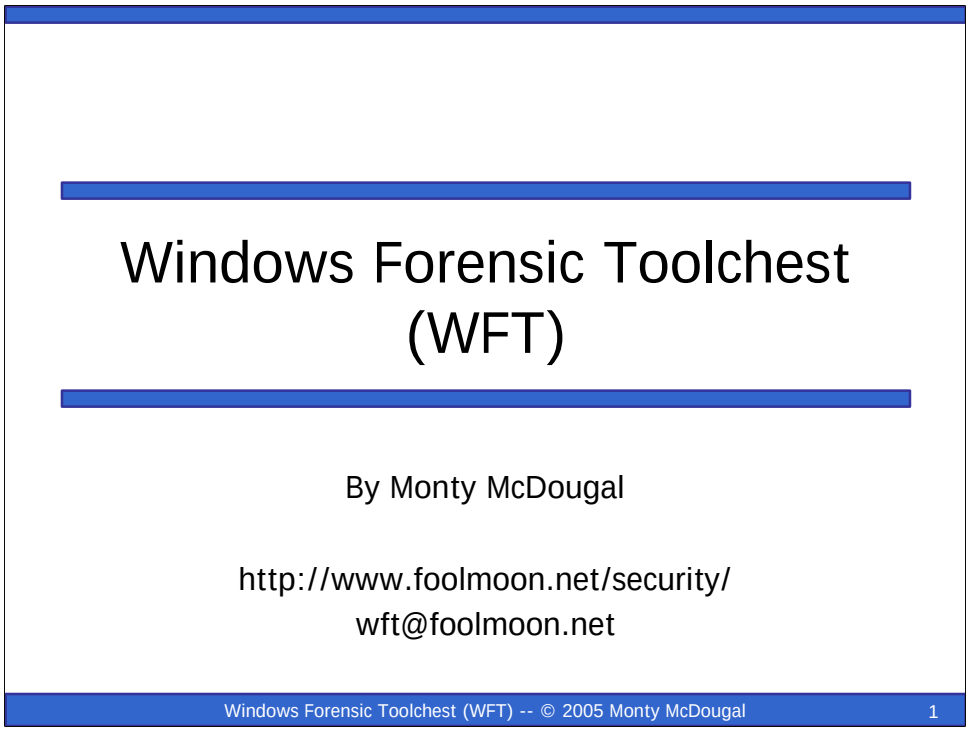
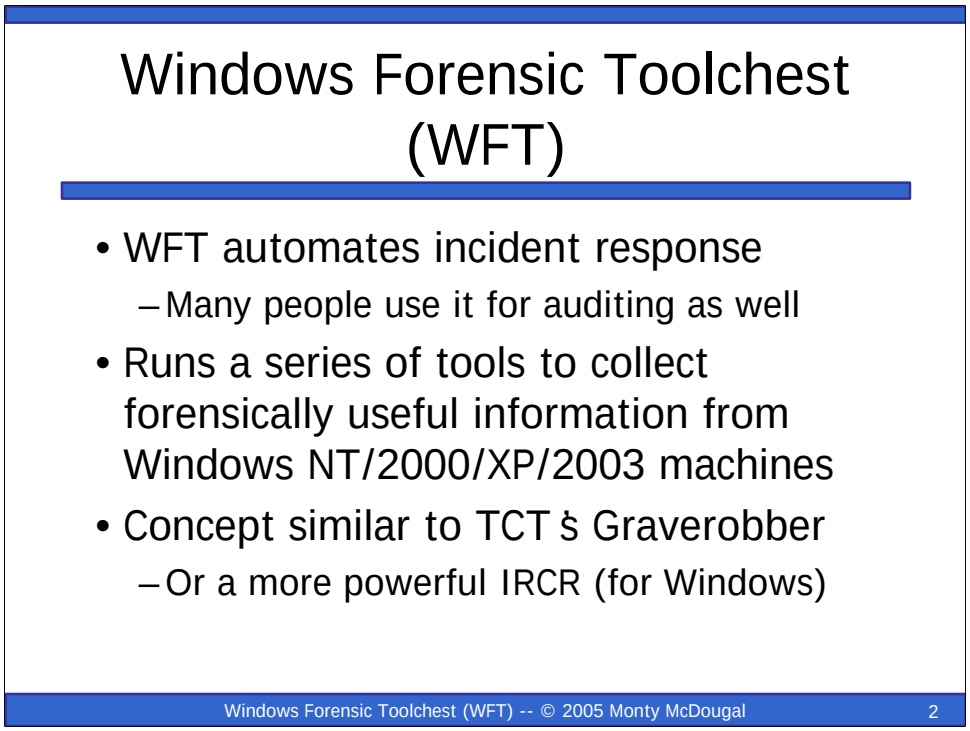


Hello and welcome to this Birds Of a Feather (BOF) presentation on the Windows Forensic Toolchest (WFT). For those of you that don't know me, I am Monty McDougal and am the author of this tool. The goal of this presentation is briefly describe WFT and how it may be useful for some of your Windows incident response and/or auditing needs. For those people already familiar with WFT, this presentation also includes information on some of the new features that have been included in the upcoming WFT 2.0 release (which is now in Beta 2). Additionally, I would like to use this as a forum to talk to users of this tool and solicit any feedback or comments they may have regarding WFT. I am particularly interested in hearing how people are using this tool in their environments and how it is working for them.

Windows Forensic Toolchest (WFT) and this presentation are
Copyright © 2003-2005 Monty McDougal. All rights reserved.



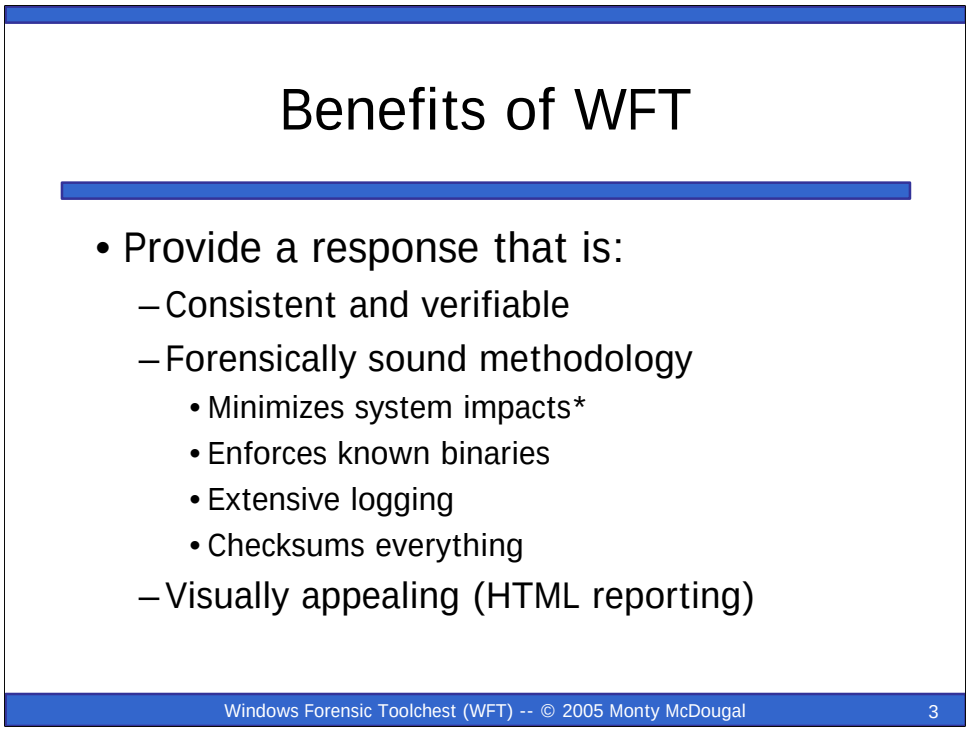
The Windows Forensic Toolchest (WFT) was written to provide an automated incident response [or even an audit] on a Windows system and collect security-relevant information from the system. It is essentially a forensically enhanced batch processing shell capable of running other security tools and producing HTML based reports in a forensically sound manner. A knowledgeable security person can use it to help look for signs of an incident (when used in conjunction with the appropriate tools). WFT is designed to produce output that is useful to the user, but is also appropriate for use in court proceedings. It provides extensive logging of all its actions along with computing the MD5 checksums along the way to ensure that its output is verifiable. The primary benefit of using WFT to perform incident responses is that it provides a simplified way of scripting such responses using a sound methodology for data collection.

The author of this tool is open for suggestions, criticisms of this tool, or offers to help improve the tool's config file and/or its documentation. Comments relating to WFT can be sent to the author at wft@foolmoon.net.

WFT and the GCFA practical paper which discuss it are available from:

http://www.foolmoon.net/security/

Windows Forensic Toolchest (WFT) and this presentation are
Copyright © 2003-2005 Monty McDougal. All rights reserved.

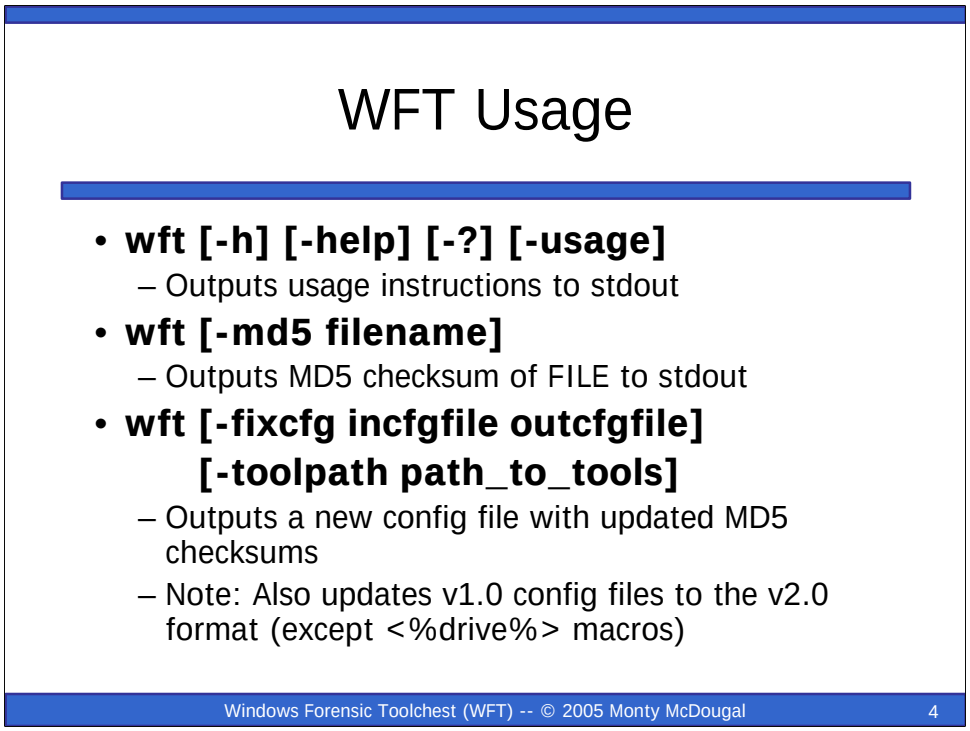


* Windows Forensic Toolchest (WFT) treads very, very lightly on the system it is being run on (i.e. uses running memory and reads a couple registry entries because it is compiled with Visual C++, but not much else). The tools WFT is invoking do not always exhibit such constraint. The tools included in the default configuration file do not make any "significant" alterations of the system they are being run on. This is described in more detail in the author's GCFA practical

==========

WFT was designed with forensic principles in mind. As such it is carefully coded, statically compiled, and written to ensure it provides extensive enough logging to be useful even in a court of law (complete with visually appealing reporting).

WFT v2.0 is a complete from the ground rewrite of WFT v1.0. In addition to several code optimizations, version 2.0 adds an enhanced config file format including "macros". This overcomes previous limitations regarding chaining WFT commands together that were written to a dynamically generated path. Version 2.0 also includes a number of new command line options, which support features added with this update. Additionally, version 2.0 includes a re-vamped config file that has been better optimized for forensic collection (including more tools). Previous restrictions on verifying cmd.exe before using the tool have been removed to better support people who are using WFT for auditing purposes. While not one of the original design goals, WFT has proven itself quite useful for the auditor and well as the incident responder.



wft [-h] [-help] [-?] [-usage]

Outputs usage instructions to stdout

wft [-md5 filename]

Outputs MD5 checksum of FILE to stdout

wft [-fixcfg incfgfile outcfgfile] [-toolpath path_to_tools]

Outputs a new config file with updated MD5 checksums

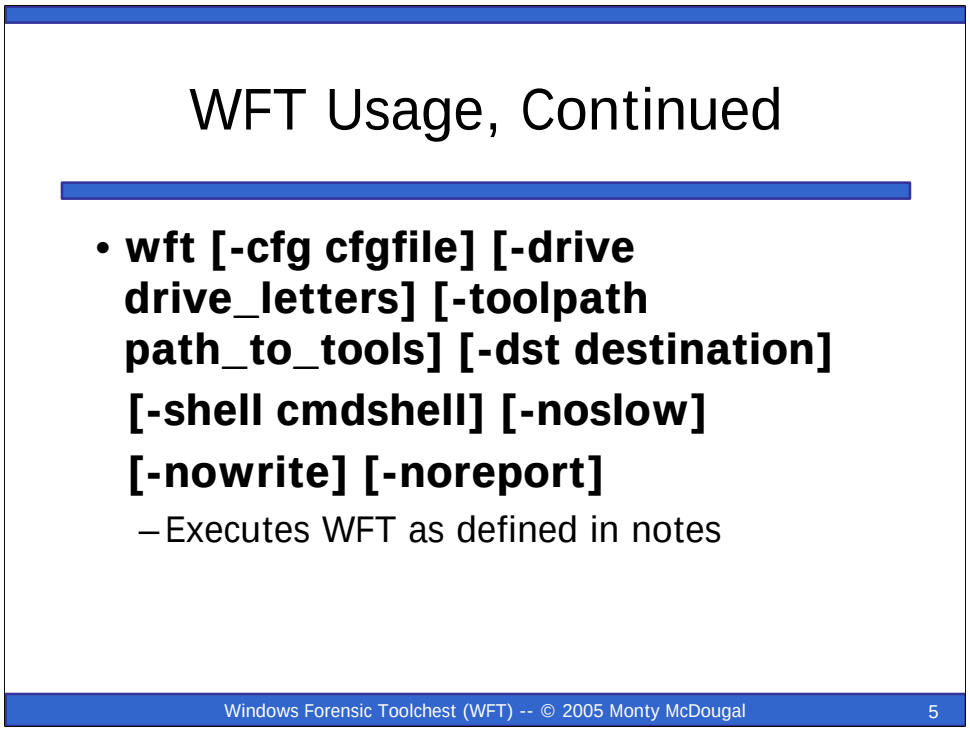
Note: Also updates v1.0 config files to the v2.0 format (except <%drive%> macros)

The last example of WFT usage '-fixcfg' was added in version 2.0. This option is designed to fulfill two needs:

1) Updating the MD5 checksums of all tools listed in the config file

2) Update previous config files from v1.0 format to v2.0 format.

Note: While I have made every effort to perform config file updates in an accurate manner, it is impossible for me to account for all possible variants of v1.0 config files. You need to verify that things work as intended in the new file.



-cfg cfgfile

Uses cfgfile to determine which tools to run (defaults to 'wft.cfg')

-drive drive_letters

Specifies the drives to be used by wft (defaults to 'C')

-toolpath path_to_tools

Defines the path where wft tools are stored (defaults to '.\')

-dst destination

Defines the path that reports will be written to (defaults to '.\')

Note: Destination can include macros **$magic$, $systemname$, $date$,** or **$time$**

-shell cmdshell

Redefines shell references from cmd.exe to cmdshell

-noslow

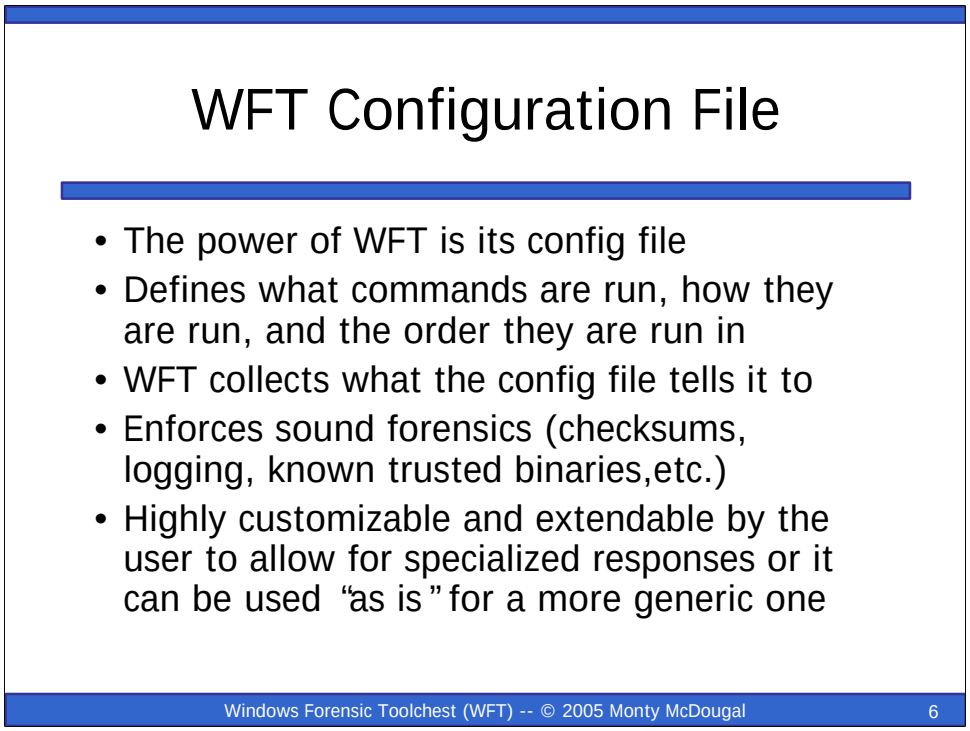
Causes WFT not to run slow (S) executables in cfgfile

-nowrite

Causes WFT not to run executables that write (W) to source machine

-noreport

Causes WFT not to create HTML (H) reports



This is the config file format used by WFT 2.0:

ACTION EXECUTABLE MD5CHECKSUM COMMAND OUTPUT MENU DESCRIPTION

Note: Each of these items is separated by a TAB (white space will not work).

Note: Lines beginning with # are treated as comments.

ACTION tells Windows Forensic Toolchest (WFT) how to process each line:

- **V** Perform MD5 verification of EXECUTABLE.
- **E** Build a COMMAND to execute.
- **N** COMMAND produces NO output to md5.
- **H** Build a HTML report.
- **M** Add a menu heading.
- **S** Skip COMMAND if -noslow option is used.
- **W** Skip COMMAND if -nowrite option is used.

Note: Multiple ACTIONS can be combined on a line

EXECUTABLE tells WFT what Executable this line will be using.

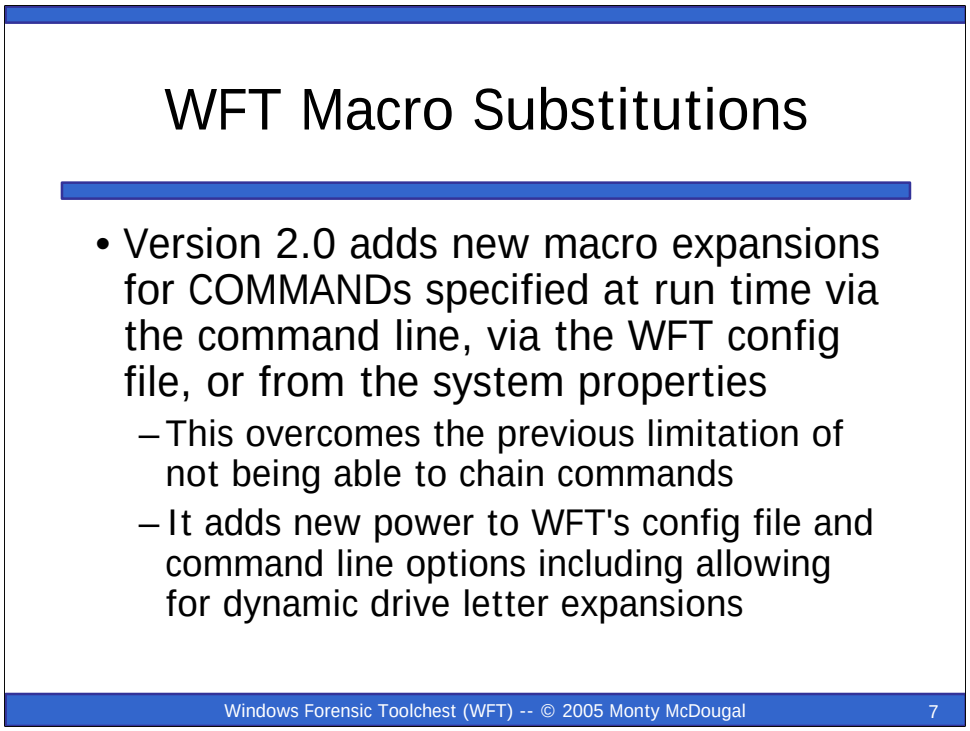
MD5CHECKSUM is the MD5 checksum of EXECUTABLE.

COMMAND tells WFT how to build the command line to be invoked.

OUTPUT is the filename (no extension) to be used for the raw report.

MENU sets the text to be used in the Report link or Menu header.

DESCRIPTION describes the EXECUTABLE and its purpose.



WFT 2.0 Macros:

<%executable%> -- the EXECUTABLE specified in the config file

<%output%> -- the value OUTPUT + '.txt' as specified in the config file

<%toolpath%> -- the -toolpath directory specified at run time (defaults to '.\')

<%dst%> -- the -dst directory specified at run time (defaults to '.\')

<%cfg%> -- the -cfg config file specified at run time (defaults to '.\wft.cfg')

<%shell%> -- the -shell specified at run time (defaults to 'cmd.exe')

<%drive%> -- which is an expanding macro and requires further explanation below

In addition to COMMANDs, these macros also work on the -dst arguments using '$' notation to replace the '<%' and '%>' such as **$magic$, $systemname$, $date$,** or **$time$**

<%magic%> -- expands to '<%systemname%>\<%date%>\<%time%>' and is done first

<%systemname%> -- system name of the computer for the current run

<%date%> -- date of the current run in the format 'MM_DD_YY'

<%time%> -- time of the current run in the format 'HH_MM_SS'

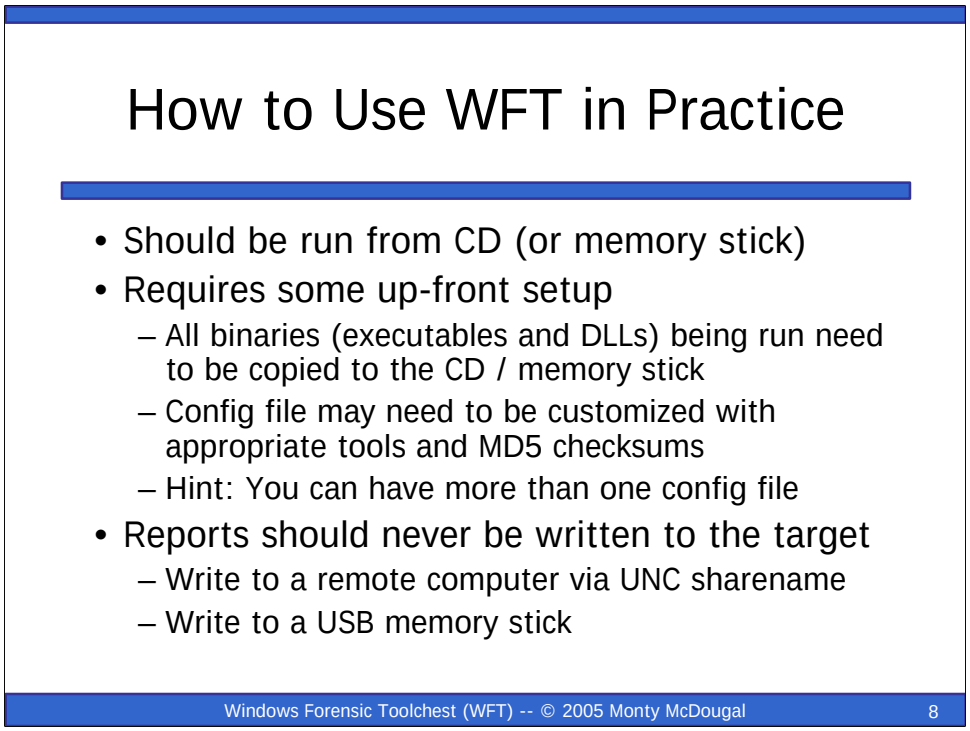
WFT 2.0 adds a new "macro expansion" option when the <%drive%> tag is used on a line

The -drive argument should be a list of drive letters to iterate through on commands

Note: -drives defaults to 'C' unless specified at run time

Each line that has a <%drive%> tag will iterate for each drive in the -drives argument

Note: COMMAND, OUTPUT, and MENU must all have this tag if it is being used or else output may be overwritten (this is enforced via WFT for safety)

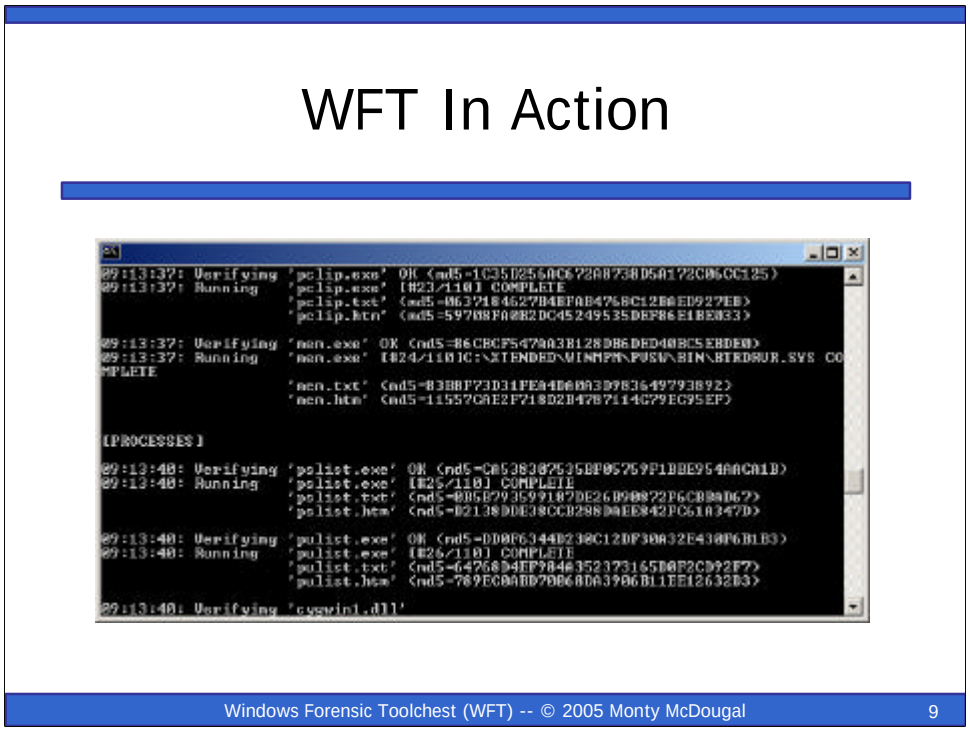


WFT should be run from a CD (or USB memory stick) to ensure the forensic integrity of the evidence it collects. In addition to the WFT binary, users will also need to copy any external programs it will be invoking to the CD / memory stick. The CD or memory stick media should also include a trusted cmd.exe matching the version of the one on the target system to ensure that WFT is being used in a forensically sound manner. WFT version 2.0 removes the requirement for this validation in order to make the tool more useable in an auditing environment.

The config file that is being used to invoke WFT should contain the MD5 checksums of not only all the tools being accessed, but also any external files they require (i.e. any DLLs, config files, etc.). Each of these files should be verified (using the V action in the config file) at least once during WFT execution to ensure that the MD5 is valid. All verifications are logged as part of WFT's execution.

Hint: It is quite possible that as a user of WFT, you may want to build multiple config files for use depending on the type of response desired. Config file can be selected at run-time via the –cfg argument.

Output of WFT should usually not be written to the target machine's fixed disk as this would alter the system during the data collection process.



This shows a capture of Windows Forensic Toolchest (WFT) in action. In this case, the output. Version 2.0 makes changes to the way this output is displayed making it more compact and easier to read. Additionally this screen capture demonstrates one of the new features in 2.0 where WFT now displays the number of current command being executed along with the total count to be executed.

Note: The number of commands may be greater than the number of lines in the file as the <%drive%> macro is expanded for each of the system drives.

For example, if [-drive argument] was 'CEF' and the COMMAND, OUTPUT, and MENU were:

<%toolpath%><%executable%> /C dir <%drive%>:*.* <%drive>_dir DIR <%drive%>

Then this would expand to three normal entries as:

<%toolpath%><%executable%> /C dir C:*.* C_dir DIR C

<%toolpath%><%executable%> /C dir E:*.* E_dir DIR E

<%toolpath%><%executable%> /C dir F:*.* F_dir DIR F

Example WFT Reports

Windows Forensic Toolchest (WFT) -- © 2005 Monty McDougal 10

Windows Forensic Toolchest (WFT) provides output in two data formats:

HTML Output: Opening the index.htm file produced by WFT provides an easy to read and easy to navigate interface to the output of the various tools invoked via WFT. Each of the reports produced under WFT includes the MD5 checksum for the binary being run, the exact command line issued to generate the output, a description of the tool, and the output produced by the tool along with the MD5 checksum associated with the output. The HTML reports are designed to be self-documenting via the text provided in the configuration file.

Raw Text Output: This format allows the viewer to see the output of the individual command exactly as it was produced. It is generally a bad idea to, in any way, manipulate data being used as evidence in a court of law. WFT seeks to preserve the original data while providing a user-friendlier HTML version for viewing. The MD5 checksums produced for each of the output files during collection provides a safeguard to ensure the output can be verified at a later date.

WFT Version 2.0 adds two subdirectories for this output – “html” and “txt”.

Additionally it supports system/date/time paths with auto directory creation to better facilitate historical comparisons between WFT runs or systems.

The author of this tool is open for suggestions, criticisms of this tool, or offers to help improve the tool's config file and/or its documentation. Comments relating to WFT can be sent to the author at wft@foolmoon.net.

About the author: Monty holds the following major degrees and certifications: BBA in Computer Science / Management (double major) from Angelo State University, MS in Network Security from Capitol College, CISSP, ISSEP, ISSAP, GIAC Certified Incident Handler (GCIH), GIAC Certified Forensic Analyst (GCFA), and serves on the SANS GCIH and GCFA Advisory Boards.