

# Log Correlation Engine 4.4 Client Guide

September 13, 2016

(Revision 3)

# Table of Contents

<b>Introduction .....</b>	<b>4</b>
Standards and Conventions.....	4
<b>Log Correlation Engine Client Overview .....</b>	<b>4</b>
Running LCE Clients Directly on the LCE Server .....	5
Running Multiple LCE Clients on One Host.....	5
Maximum Number of LCE Clients.....	5
LCE Client Types and Platforms.....	6
<b>Quick Start Summary .....</b>	<b>6</b>
Diagnosing Connection Problems.....	7
<b>SecurityCenter Client Management .....</b>	<b>7</b>
<b>LCE Client Manager .....</b>	<b>9</b>
LCE Client Manager Interactive Mode.....	9
[g] Grant Authorization to a Client .....	10
[r] Revoke Authorization to a Client.....	10
[d] Display Clients by Policy Assignment .....	10
[p] Display Available Policies .....	10
[a] Add New Policy .....	10
[c] Copy a Policy.....	10
[m] Modify an Existing Policy.....	11
[s] Assign a Policy to a Client(s) .....	11
[v] Assign Client(s) to a New LCE Server.....	11
[i] Import a Policy File.....	11
[n] Assign a Sensor Name to Client(s).....	11
[x] Remove a Client.....	11
[q] Exit .....	11
LCE Client Manager Command Line Options.....	12
Usage Example (Interactive Mode).....	12
XML Policy Representation of Client Manager Parameters.....	16
LCE Conf Converter .....	17
<b>Log Correlation Engine Windows Client .....</b>	<b>18</b>
Installing the Windows Client.....	18
Installation Location .....	20
Service Location.....	20
Remote Installation/Configuration for Multiple Hosts .....	20
Removing the LCE Windows Client .....	21
Windows Client Configuration.....	24
Policy Parameters.....	27
<b>For More Information .....</b>	<b>31</b>
<b>About Tenable Network Security.....</b>	<b>32</b>

Appendix 1: Non-Tenable License Declarations ..... 33

Related 3<sup>rd</sup> Party and Open-Source Licenses ..... 33

## Introduction

This document describes various different clients that are available for Tenable Network Security's **Log Correlation Engine 4.4**. Please email any comments and suggestions to [support@tenable.com](mailto:support@tenable.com).

A working knowledge of Secure Shell (SSH), regular expressions, and SecurityCenter operation and architecture is assumed. Familiarity with general log formats from various operating systems, network devices and applications, as well as a basic understanding of Linux/Unix is also assumed.



This document describes the current LCE server (daemon) version of 4.4.x. The LCE Clients described are all version 4.4.x. Please refer to the Tenable Support Portal for the latest version of the LCE Client.



This document is intended to be used with LCE Clients 4.4 and greater as they become available.

## Standards and Conventions

Throughout the documentation, filenames, daemons, and executables are indicated with a **courier bold** font such as **gunzip**, **httpd**, and **/etc/passwd**.

Command line options and keywords are also indicated with the **courier bold** font. Command line examples may or may not include the command line prompt and output text from the results of the command. Command line examples will display the command being run in **courier bold** to indicate what the user typed while the sample output generated by the system will be indicated in **courier** (not bold). Following is an example running of the Linux/Unix **pwd** command:

```
# pwd
/opt/lce/
#
```



Important notes and considerations are highlighted with this symbol and grey text boxes.



Tips, examples, and best practices are highlighted with this symbol and white on blue text.

## Log Correlation Engine Client Overview



Throughout this document we will continually refer to three primary LCE components: the LCE Client (the end host that initially collects data and sends it on to the LCE server); the LCE server (or daemon), which is installed on Red Hat/CentOS and performs the bulk of the processing; and the LCE host (SecurityCenter), which provides a graphical user interface to view and report on the LCE data.

The Log Correlation Engine (LCE) Clients are agents that are installed on systems whose logs, network traffic, performance and other types of protocols and technologies are to be monitored by forwarding the data securely to the LCE server. Once an LCE is installed and configured, one or more LCE Clients can be used to send information back for normalization and correlation.

This document details available LCE 4.4 Clients along with their installation and configuration. As additional 4.4 clients become available, this document will be updated on the Tenable Support Portal to include instructions for them.

Various versions (current or previous) of LCE Clients can be configured to gather information and events from the following sources:

- Windows Event Logs (collected locally or remotely via WMIC)
- Windows/Linux/Unix system and application logs
- Check Point OPSEC events
- Cisco RDEP events
- Cisco SDEE events
- NetFlow
- Splunk
- Sniffed TCP and UDP network traffic (Tenable Network Monitor)
- Sniffed `syslog` messages in motion
- File monitoring (Linux, Unix, and Windows)

Many of these agents are required to take advantage of the LCE's power. For example, to perform "Blacklist" correlation, the LCE Clients that monitor network traffic via sniffing or NetFlow can be used to identify connections with known hostile IP addresses even if you do not have firewall or proxy logs.

## Running LCE Clients Directly on the LCE Server

Some LCE Clients can be run directly on the LCE server. They must be configured to connect to either the localhost (127.0.0.1) or the IP address of the LCE server. Multiple LCE Client types (such as the LCE Log Agent and the Tenable NetFlow Monitor) can be run at the same time as well. See the section titled "[LCE Client Types and Platforms](#)" for a list of available clients.



While using LCE Log Agents to watch LCE log files, be extremely careful to avoid feedback loops. For example, choosing to tail the `lce.log` file would cause any log saved by the `lced` process to be grabbed by the LCE Log Agent, sent back to `lced`, and repeated indefinitely.

## Running Multiple LCE Clients on One Host

Remote systems can run multiple LCE Clients. When using the LCE Client Manager and various LCE clients and versions, each client type is identified and managed appropriately upon connection to the LCE server.

## Maximum Number of LCE Clients

A maximum of 8,192 individual LCE Clients can be connected simultaneously to the LCE server. Once 8,192 clients have connected, the LCE server will stop accepting new connections.


## LCE Client Types and Platforms

There are a number of different LCE Client types available. All LCE Clients report performance statistics (memory, disk space, and CPU usage) on their host regardless of the platform.



The LCE Clients written for 32-bit platforms will run on 64-bit systems as long as the 32-bit libraries are installed. However, native 64-bit support is only available for certain platforms. See the table below for more details.

LCE Client	Platform	Architecture	Function
LCE Client (Log Agent)	MS Windows XP Professional, Server 2003	32-bit	<b>Windows Client:</b> <ul style="list-style-type: none"><li>• Malware Scanning</li><li>• Unknown Process Detection</li><li>• Events sent encrypted to the LCE</li><li>• Configurable Windows event log collection</li><li>• Remote collection of Windows event logs via WMI</li><li>• Collection of process execution through event log</li><li>• Directory and file tailing</li><li>• File integrity and directory change monitoring</li><li>• USB insert and remove events</li><li>• CD-ROM/DVD insert and remove events</li><li>• CPU, memory and disk statistics collection</li><li>• Heartbeats</li></ul>
	MS Windows Server 2008, 2012, Vista, Windows 7, and Windows 8	32/64-bit	



The LCE Clients are designed to send log data to the LCE server. Accepted log data is normally in ASCII text format and will not include binary files (with the exception of process accounting data). The LCE Log Agents will check all data before sending, specifically omitting binary files such as **.zip**, **.gz**, **.tar**, **.lzh**, **.bz2**, etc. If a binary file is sent to the LCE, it has the potential to corrupt the database. This filtering is automatically performed by the LCE Client software.

## Quick Start Summary

Use these steps to get your LCE Clients up and running quickly:

1. Install and configure the clients with the IP address or hostname and port of the LCE server as per the instructions in the [Installing the Windows Client](#) sections of this document. Make sure the client is started.
2. Using SecurityCenter CV or the LCE Client Manager on the LCE server, grant authorization and apply the appropriate configurations for the newly configured clients.
3. Exit the LCE Client Manager to save and apply the settings to the Policy Map.

## Diagnosing Connection Problems

If the LCE Client cannot connect:

- View the most recent LCE Client log file located in `/opt/lce_client/` (or appropriate directory for the client in question) to determine if any error messages exist. The log has a file name in the following format: “**YearMon.log**”.
- Check that the LCE server daemon is running and correctly licensed by running “**service lce status**”. If the process is running, output similar to the following is displayed:

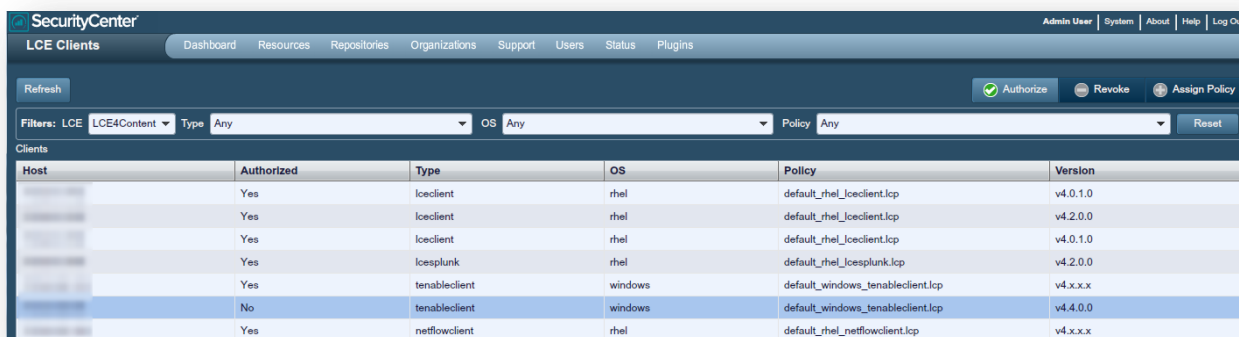
```
# service lce status
lced (pid 26868 26864) is running...
lce_queryd (pid 26876 26874) is running...
lce_indexerd (pid 26892) is running...
```

- Check to see if there is a local firewall, network firewall, or other network issue that would prevent connection from the LCE Client to the LCE server. To test this, run a sniffer on the LCE server monitoring TCP port 31300 (default port). If no connections are observed from the system running the LCE Client, something is blocking the connection. Running a sniffer on the system of the LCE Client may also help determine if something is blocking.
- If the LCE Client Manager is being used to manage the affected client(s), confirm that the server has authorized the client(s) to connect.
- Verify that the IP addresses of the LCE Client and LCE server are correct. The client will not connect to the LCE server if it has the wrong IP address or cannot correctly resolve the hostname, and the LCE server will not accept a random client unless it is specifically configured in the LCE Client Manager.

## SecurityCenter Client Management

Starting with SecurityCenter 4.8, authorization and revocation of client policies can be performed within the management GUI. Support for policy creation and change is planned for future releases.

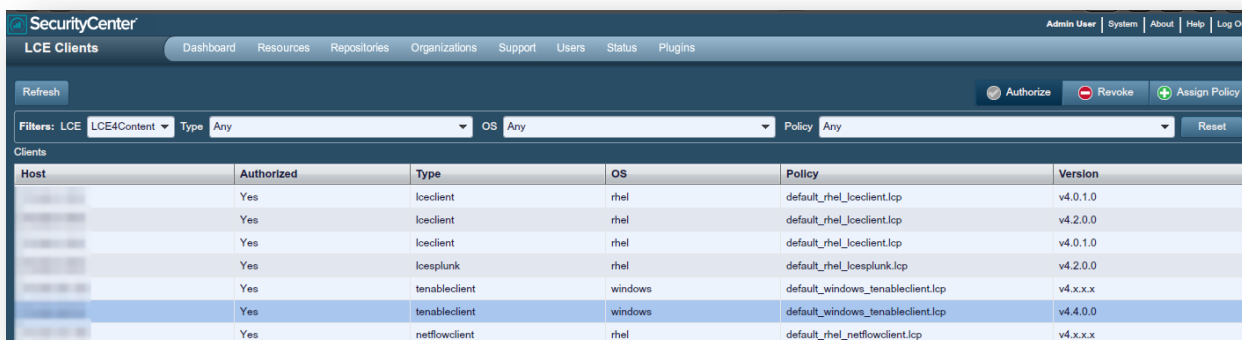
For example, if you have multiple LCE Clients installed on your network, the configuration files were set to point to one LCE Server when they were initially installed. The following screenshot shows that all LCE Clients have been authorized but one:



Host	Authorized	Type	OS	Policy	Version
	Yes	lceclient	rhel	default_rhel_lceclient.lcp	v4.0.1.0
	Yes	lceclient	rhel	default_rhel_lceclient.lcp	v4.2.0.0
	Yes	lceclient	rhel	default_rhel_lceclient.lcp	v4.0.1.0
	Yes	lcesplunk	rhel	default_rhel_lcesplunk.lcp	v4.2.0.0
	Yes	tenableclient	windows	default_windows_tenableclient.lcp	v4.x.x.x
	No	tenableclient	windows	default_windows_tenableclient.lcp	v4.4.0.0
	Yes	netflowclient	rhel	default_rhel_netflowclient.lcp	v4.x.x.x

Default LCE policies are included in the LCE content feed. For information on how to customize policies for use within your organization, see the [LCE Client Manager Command Line Options](#) section later in this document.

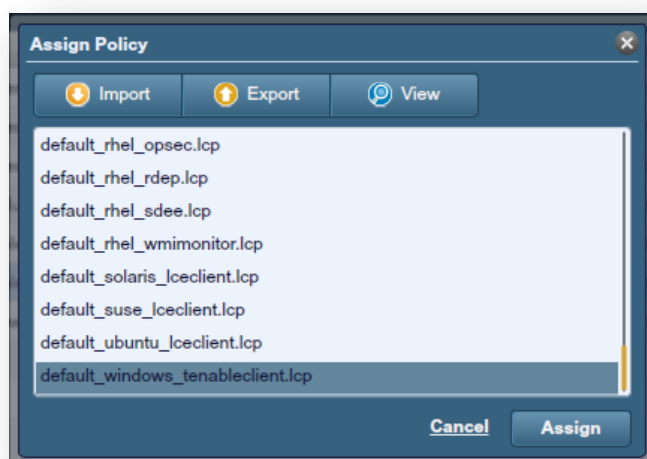
To authorize a new client, highlight the entry and then click “Authorize”. Once a client has been authorized, the LCE Server can accept data from that client using the policy listed.



Host	Authorized	Type	OS	Policy	Version
10.10.10.10	Yes	loecient	rhel	default_rhel_loecient.lcp	v4.0.1.0
10.10.10.11	Yes	loecient	rhel	default_rhel_loecient.lcp	v4.2.0.0
10.10.10.12	Yes	loecient	rhel	default_rhel_loecient.lcp	v4.0.1.0
10.10.10.13	Yes	loesplunk	rhel	default_rhel_loesplunk.lcp	v4.2.0.0
10.10.10.14	Yes	tenableclient	windows	default_windows_tenableclient.lcp	v4.x.x.x
10.10.10.15	Yes	tenableclient	windows	default_windows_tenableclient.lcp	v4.4.0.0
10.10.10.16	Yes	netflowclient	rhel	default_rhel_netflowclient.lcp	v4.x.x.x

To change a policy or revoke authorization, click on the appropriate buttons after highlighting the client(s) you wish to configure.

It is also possible to edit policies using the “Assign Policy” option. First select a client, and then select “Assign Policy”. Three options are available as shown below:



To change a policy selection, download the default policy for the client that requires an edited policy. For example, select **default\_windows\_tenableclient.lcp** and then select “Export”. Next, save the file and open it in a text editor to make changes to the policy. After the changes are made to the policy, “Import” the policy and assign the new policy to the designated client. A policy file cannot exceed 512KB in size.



The parameters that can be edited are located in the LCE Client section of this document along with a sample client policy.



## LCE Client Manager



The LCE Client Manager is used to manage clients of version 4.0 and higher to a LCE server 4.0 or higher.

The LCE Client Manager is a feature introduced in LCE 4.0 for use with all clients of version 4.0 and higher. This command line tool is used on the LCE server to manage the LCE Client access to the server and to manage the configuration files of the attached clients via policy files. The tool may be used in an interactive method or using command line options to facilitate scripting of some routine tasks that do not require a response.

The configurations are managed through the use of the Policy Map. The Policy Map contains a list of the managed clients and key information about them such as the IP address, assigned policy, client OS, and client type (log client, NetFlow, Network Monitor, WMI Monitor, etc.). When changes are made with the LCE Client Manager utility and saved, the Policy Map is reloaded with the new information and is ready for the LCE server to use without restarting the LCE server process.

All policy files (\*.lcp) are stored on the LCE server in XML format in the `/opt/lce/daemons/policies` directory.

Details for configuring policy files are included in their respective client type sections, described later in this document.

### LCE Client Manager Interactive Mode

The tool is launched by an authorized user on the LCE server by running `/opt/lce/daemons/lce_client_manager` from the server command line. When run without any options (interactive mode), a menu is presented to guide the user in managing the clients.

```
# /opt/lce/daemons/lce_client_manager

*****
* LCE Client Manager 4.4.0
* Please select an option from the menu below
*****
[g] Grant authorization to a client or clients to connect to LCE
[r] Revoke a client or clients access to connect to LCE
[d] Display clients by policy assignment
[p] Display available policies
[a] Add a new policy
[c] Copy a policy
[m] Modify an existing policy
[s] Assign a policy to a client or clients
[v] Assign a client or clients to a new LCE server
[i] Import a file as a policy
[n] Assign a custom sensor name to a client or clients
[x] Remove a client
[q] Exit

lce_client_manager >>
```

To select an option, enter the letter that corresponds with its description. As each option is selected, a submenu is offered that prompts for further information to complete the selected task.

## **[g] Grant Authorization to a Client**

After a LCE Client is initially installed on a machine, configured to direct traffic to the LCE server, and started, the LCE Client Manager must authorize the connection. This is done by selecting the “g” option from the menu.

After selecting the “g” option from the menu, the user is asked a yes or no question to authorize all clients or select the client to authorize from a list. Selecting “no” will display a list of all unauthorized clients attempting to make a connection. Entering the IP address or index number (ID number) of the client to authorize will write the information to the Policy Map file upon exiting the LCE Client Manager utility. Select “o” to return to the main menu. Selecting “yes” will cause all clients pending authorization to be written to the Policy Map upon exiting the utility. After a confirmation message is written to the terminal, the user is returned to the main menu.

Exit the utility with the “q” menu option to save the policy file to disk and activate the changes.

## **[r] Revoke Authorization to a Client**

There are situations where client access to the LCE server needs to be revoked. This is done by selecting the “r” option from the menu.

After selecting the “r” option from the menu, the user is asked a yes or no question to revoke access to all clients or select the client to revoke access from a list. Selecting “no” will display a list of all authorized clients. Entering the IP address or index number (ID number) of the client to revoke will write the information to the Policy Map file on exiting the LCE Client Manager utility. Select “o” to return to the main menu. Selecting “yes” will cause all clients to have their authorization revoked and to be written to the Policy Map on exiting the utility. After a confirmation message is written to the terminal, the user is returned to the main menu.

Exit the utility with the “q” menu option to save the policy file to disk and activate the changes.

## **[d] Display Clients by Policy Assignment**

To display a list of clients grouped by the policy assigned to the client, select the “d” option from the menu. A list of all clients in the Policy Map file will be displayed sorted by the client policy assigned. Once the list is complete, the user is returned to the main menu.

## **[p] Display Available Policies**

The “p” option displays the policies available to assign to clients, which can also be used as a base for developing new policies. A list is displayed with a column for the filename of the policy, client type, and OS.

## **[a] Add New Policy**

Selecting the “a” option from the main menu begins the process to add a new policy to the LCE Client Manager. During the creation of a new policy, the user is prompted for information including the policy name, the OS type, the client type, and add the elements and options for the policy file (elements are the valid options for a LCE Client). The policy file contents are displayed on screen during the creation process. When the addition and creation of the elements are completed, the changes may be saved to the new policy file and the user is returned to the main menu.

## **[c] Copy a Policy**

There are times when it is desirable to copy an existing policy, such as when a default policy needs to be modified for use in the environment. Select the “c” option from the main menu and a list of policy names will be displayed, ending with .lcp. Type the entire policy name at the prompt and press “Enter”. Enter the desired name of the policy to be created and press “Enter”. The policy will be copied and created under the new name, preserving the original policy file.

## **[m] Modify an Existing Policy**

Selecting the “m” option from the main menu allows the user to edit an existing policy. When selected, a list of editable policies is displayed ending with .lcp. Enter the filename to be modified followed by the “Enter” key. The policy is displayed, including its values. Select the appropriate option from the available menu to add, delete, or modify an existing key. Once the changes are satisfactorily made, select the “save and exit” option to preserve the changes and return to the main menu.

## **[s] Assign a Policy to a Client(s)**

The “s” option from the main menu allows the user to assign a policy to one or more clients. The first step is to select the clients from the presented list via the IP address or ID. The clients selected must be of the same OS and client type. Select “o” when all the clients have been selected. Enter the filename of the policy to apply to the clients from the available list and press “Enter”. The selected policy will be associated with the selected client(s) and applied on exiting the LCE Client Manager.

## **[v] Assign Client(s) to a New LCE Server**

As organizations grow or devices change location, LCE Clients may need to be modified to report to a different LCE server. In this case, select “v” from the main menu. Select the desired clients to change from the list of available clients by IP address or ID and select “o” when complete. Enter the IP address and the port of the new LCE server. Once applied, the designated client will have the new server information applied to it. The new LCE server must be configured to authorize the client and configure its policy information.

## **[i] Import a Policy File**

When a policy file has been created outside of the LCE Client Manager, it may be imported to the configuration via the “i” option of the main menu. After selecting the option, enter the full path and file name of the policy file to import. Once entered, answer the questions for the OS type, client type, and descriptive name for the policy. Once that information is entered, it will be imported for use.

## **[n] Assign a Sensor Name to Client(s)**

The “n” option allows the user to assign custom sensor names to clients. Sensor names are displayed in SecurityCenter to identify LCE Client sensors with names identifiable in the organization. By default, the sensor name is set to the DNS hostname if identified from the LCE server, otherwise it is listed as “unknown”. This option allows for customization of one or more sensor names to something meaningful for users within the organization.

When selected, a list of available clients is displayed. Select the IP address or ID of the client(s) followed by “o”. Then enter the sensor name to use for the selected client(s). Once the sensor name is entered, the user is returned to the main menu and the changes will be applied on exit.

## **[x] Remove a Client**

Selecting “x” from the main menu begins the process to remove a client. When selected, a list of all available clients is listed. Enter the IP address or ID of the client(s) to remove. Once completed, select “o” to save the changes. On exiting the LCE Client Manager, the selected clients will be removed from the Policy Map and no longer be accepted by the LCE server as valid clients.

## **[q] Exit**

The “q” command will cleanly exit the LCE Client Manager, apply pending changes to the Policy Map file, and reload the Policy Map to apply the new changes to the running file.

## LCE Client Manager Command Line Options

The options for the LCE Client Manager can also be invoked on the command line as in, for example:

“`/opt/lce/daemons/lce_client_manager --remove-client <client ID>`” (to remove a client). The command `/opt/lce/daemons/lce_client_manager -h` will display all the available options that can be invoked from the command line.

## Usage Example (Interactive Mode)

Shown below is an example of how to copy a default policy, customize it, and use it for LCE Client installations. The RHEL LCE Client policy will be copied and customized for use on RHEL systems running the Apache Web server, where it will monitor any file changes (recursively) in the configuration directory (`/etc/https`).

```
# /opt/lce/daemons/lce_client_manager

*****
* LCE Client Manager 4.4.0
* Please select an option from the menu below
*****
[g] Grant authorization to a client or clients to connect to LCE
[r] Revoke a client or clients access to connect to LCE
[d] Display clients by policy assignment
[p] Display available policies
[a] Add a new policy
[c] Copy a policy
[m] Modify an existing policy
[s] Assign a policy to a client or clients
[v] Assign a client or clients to a new LCE server
[i] Import a file as a policy
[n] Assign a custom sensor name to a client or clients
[x] Remove a client
[q] Exit

lce_client_manager >> c

Policy Filename                                Client Type      OS
TNS-MSEXchangeServer_windows_tenableclient.lcp  tenableclient    windows
TNS-MSSQLServer_windows_tenableclient.lcp       tenableclient    windows
TNS-MalwareDetectionOnly_osx_lceclient.lcp       lceclient        osx
TNS-MalwareDetectionOnly_rhel_lceclient.lcp      lceclient        rhel
TNS-MalwareDetectionOnly_windows_tenableclient.lcp tenableclient
windows
TNS-NTEvents-FileSysMon_windows_tenableclient.lcp tenableclient
windows
TNS-NTEvents_windows_tenableclient.lcp          tenableclient    windows
TNS-ProcessExecutionOnly_osx_lceclient.lcp       lceclient        osx
TNS-ProcessExecutionOnly_rhel_lceclient.lcp      lceclient        rhel
TNS-ProcessExecutionOnly_windows_tenableclient.lcp tenableclient
windows
TNS-TenableProducts-LCE_rhel_lceclient.lcp       lceclient        rhel
TNS-TenableProducts-Nessus_rhel_lceclient.lcp    lceclient        rhel
TNS-TenableProducts-Nessus_windows_tenableclient.lcp tenableclient
windows
TNS-TenableProducts-PVS_rhel_lceclient.lcp       lceclient        rhel
```

```

TNS-TenableProducts-SC_rhel_lceclient.lcp      lceclient      rhel
TNS-TenableProducts_rhel_lceclient.lcp lceclient      rhel
TNS-WinDesktop_windows_tenableclient.lcp      tenableclient  windows
apache_rhel_lceclient.lcp      lceclient      rhel
default_aix_lceclient.lcp      lceclient      aix
default_appliance_lceclient.lcp      lceclient      appliance
default_appliance_netflowclient.lcp      netflowclient  appliance
default_appliance_networkmonitor.lcp      networkmonitor  appliance
default_debian_lceclient.lcp      lceclient      debian
default_dragon_lceclient.lcp      lceclient      dragon
default_fedora_lceclient.lcp      lceclient      fedora
default_freebsd_lceclient.lcp      lceclient      freebsd
default_hpx_lceclient.lcp      lceclient      hpx
default_osx_lceclient.lcp      lceclient      osx
default_rhel_lceclient.lcp      lceclient      rhel
default_rhel_lcesplunk.lcp      lcesplunk      rhel
default_rhel_netflowclient.lcp      netflowclient  rhel
default_rhel_networkmonitor.lcp      networkmonitor  rhel
default_rhel_opsec.lcp      opsec          rhel
default_rhel_rdep.lcp      rdep          rhel
default_rhel_sdee.lcp      sdee          rhel
default_rhel_wmimonitor.lcp      wmimonitor     rhel
default_solaris_lceclient.lcp      lceclient      solaris
default_suse_lceclient.lcp      lceclient      suse
default_ubuntu_lceclient.lcp      lceclient      ubuntu
default_windows_tenableclient.lcp      tenableclient  windows
Enter the name of the policy to copy, or 0 to cancel.
lce_client_manager >> default_rhel_lceclient.lcp

```

Enter a descriptive name for the new policy.  
 LCE will append the client name and operating system type.  
 It may not start with "default" or "TNS" and should contain only a-z,A-Z,0-9, and -.  
 It may be, at most, 50 characters long.  
 Valid examples include: corp-desktops-1  
                           web-servers-2  
                           lab-machines-3

Enter 0 to cancel.

```
lce_client_manager >> apache2
```

```

Full name : /opt/lce/daemons/policies/apache2_rhel_lceclient.lcp
Successfully copied policy /opt/lce/daemons/policies/default_rhel_lceclient.lcp to
/opt/lce/daemons/policies/apache2_rhel_lceclient.lcp.
Successfully signaled LCE to reload the policy map.

```

```

*****
* LCE Client Manager 4.4.0
* Please select an option from the menu below
*****

```

```

[g] Grant authorization to a client or clients to connect to LCE
[r] Revoke a client or clients access to connect to LCE
[d] Display clients by policy assignment
[p] Display available policies

```

```

[a] Add a new policy
[c] Copy a policy
[m] Modify an existing policy
[s] Assign a policy to a client or clients
[v] Assign a client or clients to a new LCE server
[i] Import a file as a policy
[n] Assign a custom sensor name to a client or clients
[x] Remove a client
[q] Exit

```

```
lce_client_manager >> m
```

Policy Filename	Client Type	OS
apache_rhel_lceclient.lcp	lceclient	rhel

Enter the file name of the policy to modify (0 to cancel):

```
lce_client_manager >> apache_rhel_lceclient.lcp
```

Tip: Policies are lists of key-value pairs called elements. Elements can be nested, as follows:

```

[key0] -> [value0]
[key1]
    [subkey1] -> [value1]
    [subkey2] -> [value2]
[key2] -> [value4]

```

To reach value2, first ask to modify key1, then modify subkey2.

The current policy key-values being modified:

```

----- BEGIN POLICY -----
[log-directory] -> [/opt/lce_client/]
[tail-file] -> [/var/log/messages]
[tail-file] -> [/var/log/secure]
[tail-dir] -> [/var/log/*.log]
[scan-frequency] -> [60]
[monitor-file-changes] -> [/etc/passwd]
[report-ownership-changes] -> [yes]
[report-permissions-changes] -> [yes]
[modification-check-frequency] -> [30]
[heartbeat-frequency] -> [300]
[statistics-frequency] -> [60]
[compress-events] -> [1]

```

```
----- END POLICY -----
```

Select an option to modify your policy:

```

[a] Add new key (and values)
[d] Delete existing key/element (and values)
[m] Modify value for existing key
[s] Save policy to file and exit
[q] Exit WITHOUT saving changes

```

```
lce_client_manager >> a
```

```
Enter the new key to add to your policy:
lce_client_manager >> recursive-directory-changes
```

```
Current element being modified:
```

```
[recursive-directory-changes]
```

```
Select an option for this element:
```

```
[a] Add a nested element
[v] Add a new value
[d] Delete a value
[z] Modify a nested element
[m] Modify a value
[s] Save this element and proceed
```

```
lce_client_manager >> v
```

```
Enter the new value to add to your element:
```

```
lce_client_manager >> /etc/httpd
```

```
Current element being modified:
```

```
[recursive-directory-changes] -> [/etc/httpd]
```

```
Select an option for this element:
```

```
[a] Add a nested element
[v] Add a new value
[d] Delete a value
[z] Modify a nested element
[m] Modify a value
[s] Save this element and proceed
```

```
lce client manager >> s
```

```
Saving element...
```

```
The current policy key-values being modified:
```

```
----- BEGIN POLICY -----
```

```
[log-directory] -> [/opt/lce_client/]
[tail-file] -> [/var/log/messages]
[tail-file] -> [/var/log/secure]
[tail-dir] -> [/var/log/*.log]
[scan-frequency] -> [60]
[monitor-file-changes] -> [/etc/passwd]
[report-ownership-changes] -> [yes]
[report-permissions-changes] -> [yes]
[modification-check-frequency] -> [30]
[heartbeat-frequency] -> [300]
[statistics-frequency] -> [60]
[compress-events] -> [1]
```

```

[recursive-directory-changes] -> [/etc/httpd]

----- END POLICY -----

Select an option to modify your policy:
[a] Add new key (and values)
[d] Delete existing key/element (and values)
[m] Modify value for existing key
[s] Save policy to file and exit
[q] Exit WITHOUT saving changes

lce_client_manager >> s

Successfully saved the modified policy.
Successfully signaled LCE to reload the policy map.

*****
* LCE Client Manager 4.4.0
* Please select an option from the menu below
*****
[g] Grant authorization to a client or clients to connect to LCE
[r] Revoke a client or clients access to connect to LCE
[d] Display clients by policy assignment
[p] Display available policies
[a] Add a new policy
[c] Copy a policy
[m] Modify an existing policy
[s] Assign a policy to a client or clients
[v] Assign a client or clients to a new LCE server
[i] Import a file as a policy
[n] Assign a custom sensor name to a client or clients
[x] Remove a client
[q] Exit

lce_client_manager >>

```

The \*.lcp policy files located in `/opt/lce/daemons/policies` are in XML format, which can be edited manually.



Never edit the default policies, as they will be overwritten with future updates.



When using the LCE Client Manager to edit a policy, all comments and white space are removed from the file.

## XML Policy Representation of Client Manager Parameters

The following is an example of how the parameters entered using the Client Manager are stored when using the Client Manager:



```
[log-directory] -> [./]
[interface] -> [eth0]
[syslog-only] -> [no]
[include-networks]
  [filter] -> [192.168.20.5/32]
  [filter] -> [127.0.0.1]
  [filter] -> [172.0.0.0/8]
[exclude-networks]
[heartbeat-frequency] -> [300]
[statistics-frequency] -> [60]
[compress-events] -> [1]
[filter-expression] -> [udp or tcp or icmp]
```

This is what the XML policy file contains:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?> <options
  xmlns:xi="http://www.w3.org/2003/XInclude">
  <log-directory>./</log-directory>
  <interface>eth0</interface>
  <syslog-only>no</syslog-only>
  <include-networks>
    <filter>192.168.20.5/32</filter>
    <filter>127.0.0.1</filter>
    <filter>172.0.0.0/8</filter>
  </include-networks>
  <exclude-networks/>
  <heartbeat-frequency>300</heartbeat-frequency>
  <statistics-frequency>60</statistics-frequency>
  <compress-events>1</compress-events>
  <filter-expression>udp or tcp or icmp</filter-expression> </options>
```

## LCE Conf Converter

The LCE Conf Converter is a utility to convert LCE configuration files from versions of the LCE Clients prior to 4.0 to new policy files.

The following command run from the command line with no options will display the help file:

```
# /opt/lce/daemons/lce_conf_file_converter
```

There are four valid options to use as described in the table below:

Option	Description
<b>--input-conf-file</b> <b>-i</b>	The input configuration file (i.e., <b>lce_client.conf</b> )
<b>--output-policy-file</b> <b>-o</b>	The output policy file (e.g., <b>my-new-policy.lcp</b> )
<b>--help</b> <b>-h</b>	Display the help menu

<code>--version</code> <code>-v</code>	Display version information
---	-----------------------------

Once saved as a policy file, the converted file may be imported to the LCE Client Manager and assigned to the appropriate client(s).

The following is an example of how to convert an `lce_client.conf` to a policy file (for RHEL):

```
# /opt/lce/daemons/lce_conf_file_converter -i
    /opt/lce_client/lce_client.conf -o ~/lce_client_conf.lcp

Successfully converted /opt/lce_client/lce_client.conf to policy
/root/lce_client_conf.lcp.

# /opt/lce/daemons/lce_client_manager --import-policy
    ~/lce_client_conf.lcp --output-policy my-converted-conf
    --client-type lceclient --os-type rhel
    /opt/lce/daemons/policies/my-converted-conf_rhel_lceclient.lcp
```

If there is an error, a non-zero error code will be displayed.

## Log Correlation Engine Windows Client

The Log Correlation Engine Windows Client monitors events, as well as specific log files or directories, for new event data. Tenable currently has two LCE Windows Clients: one for Windows XP/2003 platforms and one for Windows Vista/2008/2012/7/8 platforms.

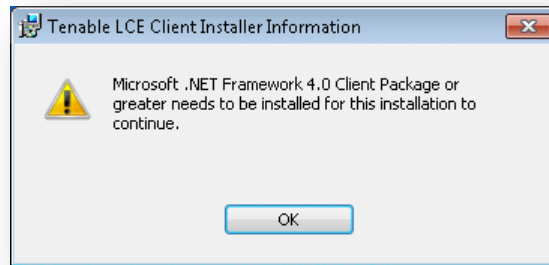
New in the LCE Windows Client 4.4.0 is the ability to scan for malware in monitored directories. Previously, the malware detection could only be used for analyzing running processes. If there is a `<malware-scan-frequency>` time already configured in your client policy, nothing further needs to be added to the client policy. A sample client policy and policy parameters are covered in more detail in this section of the documentation.

Platform	LCE Client Type	Install File Name and Utility
MS Windows XP Professional, Windows Server 2003	LCE Log Agent	<code>lce_client-4.x.x-windows_2003_x86.msi</code>
MS Windows Server 2008, 2012 Windows Vista, Windows 7, Windows 8	LCE Log Agent	<code>lce_client-4.x.x-windows_2008_x86.msi</code> <code>lce_client-4.x.x-windows_2008_x64.msi</code>

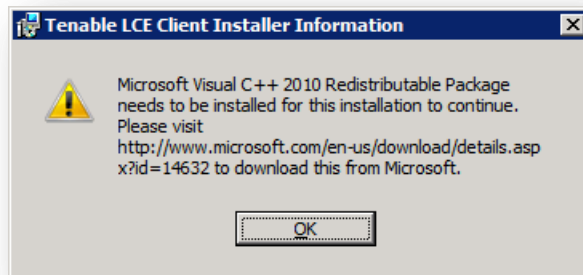
## Installing the Windows Client

The LCE Windows Log Agent client is installed by clicking on the `.msi` distribution file, which will launch the InstallShield Wizard. On machines where Universal Access Control (UAC) is enabled, the user must run the installer as an Administrator level user. Right click on the installer icon and select “**Run as Administrator**”.

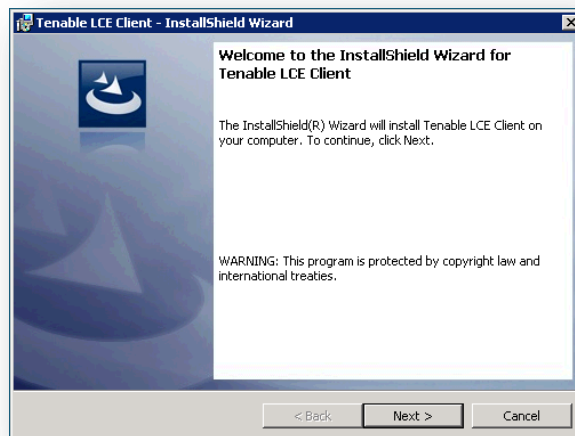
The LCE Windows Client requires .NET 4.0 to install successfully. If .NET 4.0 isn’t installed, a message similar to the one below will be displayed.



The LCE Windows Client also requires Microsoft Visual C++ 2010 SP1 Redistributable Package ([x86](#)) or ([x64](#)). If the Microsoft Visual C++ 2010 SP1 Redistributable Package is not installed, an error similar to the one below will be displayed.

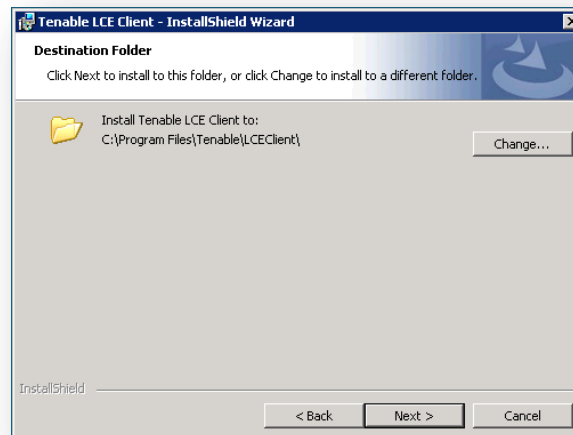


The initial screen of the installation is shown below. Clicking on “Next” displays the license agreement. After reading and agreeing to the license agreement, installation can continue.



## Installation Location

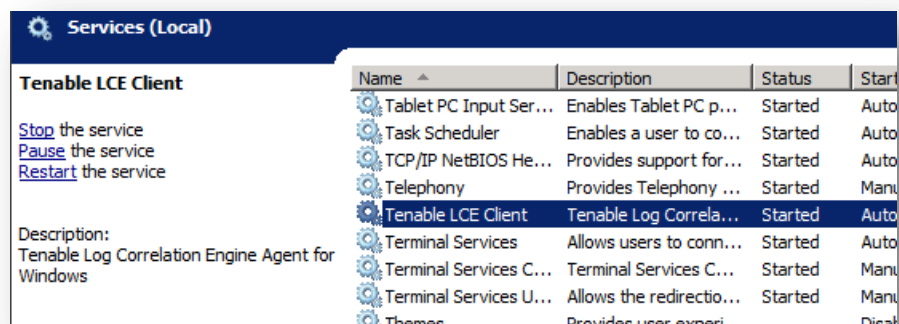
After the terms and condition are agreed to, the next screen that is displayed will allow the user to change the default installation location:



Click the “Change...” button and select a new location if the application is to be installed in an alternate location. To use the default location, simply click “Next” and a screen will be displayed to begin the installation by clicking “Install”. If this is an upgrade and a previous version of LCE Windows Client is running, you may be prompted to either automatically close and restart the application or not close the running application and restart it at a later time for the new version to be applied. After a short period, the InstallShield Wizard will display a screen indicating that the installation is complete. Once installation is complete, you may be prompted to restart the system for the configuration changes to take effect.

## Service Location

Once installation is complete, a new Windows Service named the “**Tenable LCE Client**” will be added that can be viewed through Control Panel -> Administrative Tools -> Services window as shown below:



## Remote Installation/Configuration for Multiple Hosts

The installation of the LCE Windows Client can be accomplished from a command line or script via the execution of the “msiexec.exe” program. This makes it possible to perform remote installations of LCE Windows Clients for multiple hosts.

To facilitate this process, the option exists to set the client's initial configuration settings at the time of the installation from the same command.

The following table contains a list of PUBLIC properties for the Tenable LCE Windows Client MSI install package. Because all parameters (except LCE server IP address and port) are set using policies on the server, there are only the two options available.

Property	Description
<b>SERVERIP</b>	The IP address or hostname (maximum hostname length of 46 characters) of the LCE server. Defaults to: "203.0.113.250".
<b>SERVERPORT</b>	This setting denotes the port used to communicate with the LCE server. The default port is 31300.

The following demonstrates an installation of the LCE Windows Client from the command line:

```
msiexec.exe /qn /i "lce_client-4.4.0-windows_2008_x64.msi" SERVERIP="127.0.0.2"  
SERVER_PORT=31300
```

The `/qn` on the above line instructs the installer to run with no user feedback. When performing an installation from the command line, the `/qn` option can be used to keep the installation program from stopping the process to ask if previous settings should be applied. The `/i` is the operative parameter that specifies the name of the file to be installed. If desired, the `/passive` option can be used in place of `/q`, which will display the progress of the installation, but it doesn't allow for user interaction. If a log file of the installation is desired, `/l` can be used followed by the path to the log file. An example is shown below:

```
msiexec.exe /l C:\Users\Administrator\Documents\lce_client_install.txt /passive /i  
"lce_client-4.4.0-windows_2008_x64.msi" SERVERIP="127.0.0.2" SERVER_PORT=31300
```



The display, restart, logging, and repair options for Msiexec are also supported. More information on Msiexec can be found at <http://technet.microsoft.com/en-us/library/cc759262%28v=WS.10%29.aspx>.

## Removing the LCE Windows Client

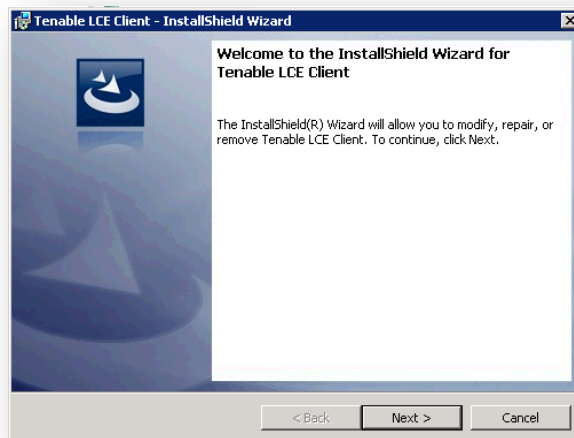
To remove the LCE Windows Client, under Control Panel, open **"Add or Remove Programs"** or **"Programs and Features"** depending on the version of Windows. Select **"Tenable LCE Client"** and then click the **"Change/Remove"** button. This will open the InstallShield Wizard. Follow the directions in this wizard to completely remove the LCE Windows Client.

The LCE Windows Client can also be uninstalled via the command line. The example command below will uninstall the LCE Windows Client quietly from the system.

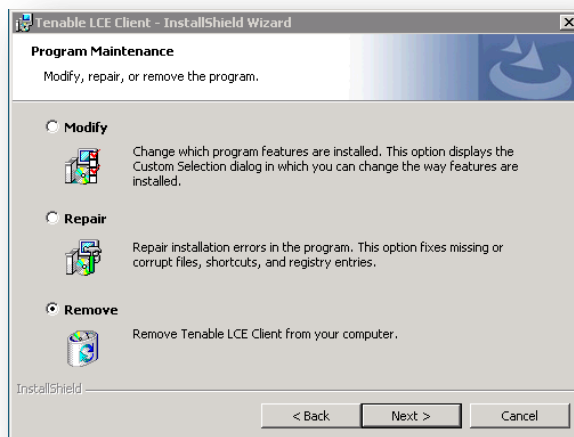
```
msiexec.exe /qn /uninstall "lce_client-4.4.0-windows_2008_x64.msi"
```

Removal of the client can also be performed using the Tenable LCE Client installer as shown below.

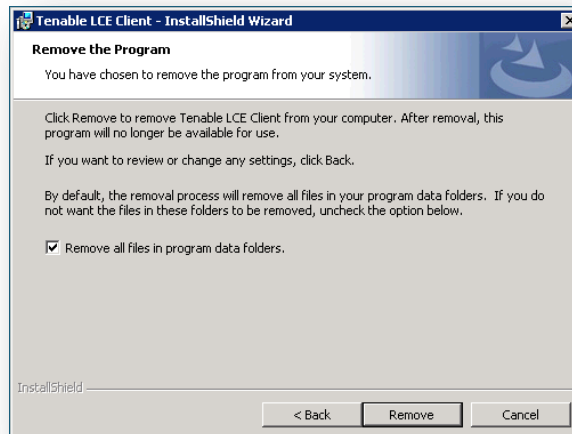
First launch the client installer, and choose "Next".



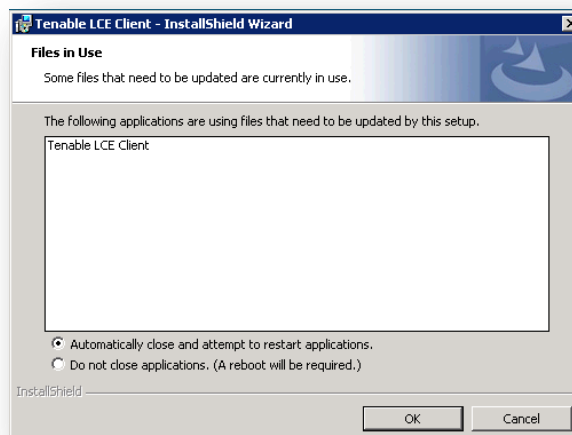
Then select **“Remove”** from the **“Program Maintenance”** options and select **“Next”**.



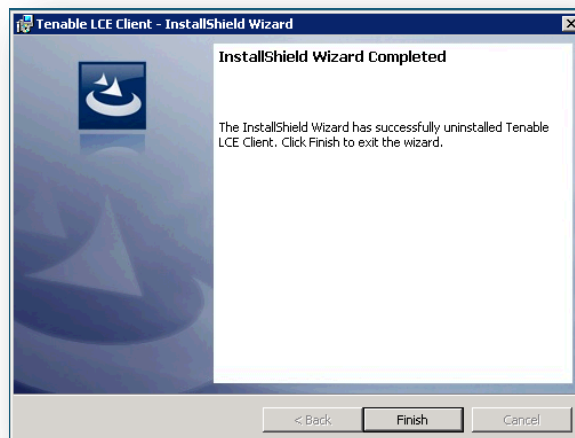
The next option allows for the retention or removal of data found in **“C:\Program Data\Tenable”**. Selecting **“Remove all files in program data folders.”** will remove the existing data from the current client install, which includes the current configuration, server assignment, state, and administrative log. The option to remove the files is selected in the example below.



The next step requires either that the “Tenable LCE Client” service is stopped, or the system will require a reboot after the uninstall is complete to finish the removal of the Tenable LCE Client. After the choice is made, select “OK” to continue. The default is chosen in the example below.



After the Tenable LCE Client removal is complete, select “Finish” to complete the wizard.

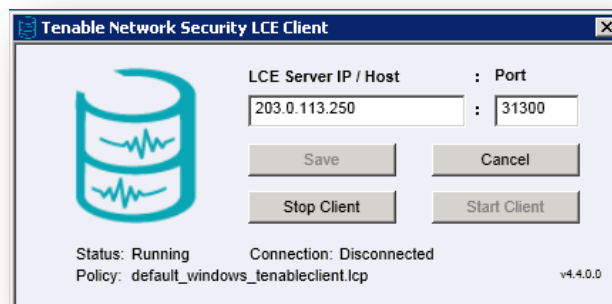


## Windows Client Configuration

To configure the LCE Windows Client, launch the LCE Configuration tool located at “**C:\Program Files\Tenable\LCEClient\LCE\_Server\_Assignment.exe**”. The “LCE Client Configuration” utility can be found by selecting the “**Start**” menu, and then navigating to “**Tenable Network Security**”.

When connecting to a LCE 4.x server, the only configuration required is the LCE server IP address or DNS name and the port (if the server is configured for one other than the default of 31300). All other configuration options will be managed by the LCE Client Manager upon connection.

An example screen for the LCE Client Configuration tool is shown below:



By default, the LCE Windows Client is configured using a non-routable documentation IP address (203.0.113.250) and LCE Server Port 31300. These settings must be changed to the IP address or hostname and listening port of the actual LCE server. No further local configuration is required. Once set, select the “**Save**” button, which will save the configuration and restart the LCE client.

Once the client connects to the LCE server and is authorized by the LCE Client Manager, the appropriate policy file will be pushed to the client. The `default_windows_tenableclient.lcp` policy that is shown below will be assigned as the default policy, but another policy can be selected in SecurityCenter CV.

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```



```

<options xmlns:xi="http://www.w3.org/2003/XInclude">

  <!-- The LCE Client monitors Windows event logs specified in the event-logs
        element. Each event that appears in the Windows event log will be forwarded to
        the LCE server. List each event log under its own tag. -->
  <event-log>Application</event-log>
  <event-log>Security</event-log>
  <event-log>System</event-log>

  <!-- The LCE Client can monitor other text files as well.
        List each file under its own tag.
        Subtags allow include/exclude filtering with wildcards
        as well as deleting log files after a size threshold. -->
  <!-- flat-file>C:\\log.txt</flat-file -->
  <!-- flat-file>
    <location>C:\\logs</location>
    <include>*.txt</include>
    <include>*.log</include>
    <exclude>*.exe</exclude>
    <delete-on-size-bytes>20M</delete-on-size-bytes>
  </flat-file -->

  <!-- When monitoring event-logs, flat-files, and remote-files, the LCE Client
        checks these files periodically. This specifies the number of seconds to wait
        before rechecking the files. -->
  <interval-log-seconds>60</interval-log-seconds>

  <!-- When sending event log data, the LCE Client can send all of the existing
        data, or only send new events. It is recommended to send only new events. -->
  <send-new-events-only>1</send-new-events-only>

  <!-- The LCE Client can also monitor binary files for changes. List those here.
        List each file under its own tag.
        Subtags allow include/exclude filtering with wildcards. -->
  <monitor-file>C:\\MSDOS.SYS</monitor-file>
  <monitor-file>C:\\IO.SYS</monitor-file>
  <monitor-file>C:\\config.sys</monitor-file>
  <monitor-file>C:\\BOOTSECT.BAK</monitor-file>
  <monitor-file>C:\\autoexec.bat</monitor-file>
  <monitor-file>C:\\Program Files\\Tenable\\LCEClient</monitor-file>
  <monitor-file>C:\\Windows</monitor-file>
  <monitor-file>C:\\Windows\\System32</monitor-file>
  <monitor-file>C:\\Windows\\system</monitor-file>
  <!-- monitor-file>
    <location>C:\\Windows\\System32</location>
    <include>*.dll</include>
    <include>*.exe</include>
    <exclude>*.txt</exclude>
  </monitor-file -->

  <!-- When monitoring binary files via monitor-file, the LCE Client may recurse
        into subdirectories if desired, to monitor all of the files underneath this
        directory. For system folders such as C:\\Windows\\System32\\ this can be taxing,

```

```

    so exercise caution when enabling this feature. -->
<monitor-subdirectories>0</monitor-subdirectories>

<!-- When monitoring binary files via monitor-file, the LCE Client checks these
    files periodically. This specifies the number of seconds to wait before
    rechecking the files -->
<interval-monitor-seconds>1800</interval-monitor-seconds>

<!-- Beginning with LCE Client 4.2.0, the LCE Client can perform checks of running
    processes to determine if they match known malware. The period between checks
    (in seconds) can be configured with malware-scan-frequency. This scanning is
    enabled by default, but can be disabled if the value is set to 0. -->
<!-- malware-scan-frequency>600</malware-scan-frequency -->

<!-- If scanning is enabled, processes can be whitelisted to prevent them from
    being reported. To whitelist processes, place their MD5 hash within the
    following element. Each hash specified can be listed in a separate element or
    multiple hashes may be separated by whitespace within a single element. -->
<!-- whitelist-hashes>da13a4d8ba3e6171ac60e32c0ac39e6e</whitelist-hashes -->

<!-- Beginning with LCE Client 4.2.0, the LCE Client can flag running processes
    with an MD5 hash that matches a user-provided hash. Each hash specified can be
    listed in a separate element or multiple hashes may be separated by whitespace
    within a single element. -->
<!-- custom-malware-hashes>da13a4d8ba3e6171ac60e32c0ac39e6e</custom-malware-hashes
    -->

<!-- Beginning with LCE Client 4.2.1, the LCE Client can flag processes that are
    not in a known malware or a known whitelist database of processes. Values are
    defined as:
        0 - disable unknown process reporting [default]
        1 - enable unknown process reporting, and report each process once, ever
        2 - enable unknown process reporting, and report each process after all
        client restarts and policy changes -->
<!-- report-unknown-processes>1</report-unknown-processes -->

<!-- This tells the LCE Client to write debugging information to its log file.
    Use this setting under the direction of Tenable Network Security Support
    only. -->
<debug>0</debug>

<!-- This tells the LCE Client to write info-level logging information to its log
    file. -->
<info>0</info>

<!-- This tells the LCE Client to write verbose-level logging information to its
    log file. -->
<verbose>0</verbose>

<!-- This tells the LCE Client to monitor changes to its own configuration. -->
<monitor-config>1</monitor-config>

<!-- The LCE Client can also monitor logs on another machine remotely.

```

```

        Each host is contained within its own element. A host consists of:
        ip, namespace, domain, user(name), password (unsafe!), and a set of log files
        to watch. -->
<!-- Host>
  <ip>203.0.113.250</ip>
  <namespace></namespace>
  <domain></domain>
  <user>Administrator</user>
  <password>password</password>
  <logfilename></logfilename>
  <logfilename></logfilename>
</Host -->

<!-- The heartbeat-frequency option defines the number of seconds between each
pair of client heartbeat messages that are sent to the server. -->
<heartbeat-frequency>300</heartbeat-frequency>

<!-- The LCE client provides the option of periodically sending a log file
containing performance statistics to the LCE server. The following option
determines the number of minutes between each performance statistics
report. -->
<statistics-frequency>60</statistics-frequency>

<!-- LCE clients can compress log data prior to sending it to the LCE server,
saving bandwidth. For debugging purposes, event packet compression may be
disabled, but this will increase the bandwidth required to send data from LCE
clients to the LCE server. Setting the following option to 0 will disable
compression only during transmission. -->
<compress-events>1</compress-events>


</options>


```



## Policy Parameters

The following is a list of all valid “keys” available for use in with the Windows policies:

Key Name	Description	Valid Values
<b>event-log</b>	The name of a Windows NT Event log to monitor. Each event is sent to LCE as a new log.	Any NT event log name, or “all” will monitor all NT event logs at the time the client is started. The “all” option will also capture application specific event logs, like Task Scheduler.
<b>flat-file</b>	The full path and name of a text file to monitor. Each new line is sent to LCE as a new log.	Any fully qualified path and file name, with the file extension. It is best practice to escape folder separators with a backslash.
<b>flat-file</b>	<b>Sub Key</b>	<b>Description</b>
	<b>location</b>	The full path of which to monitor text files. Each new line in each file is sent to LCE as a new log.

	<b>include</b>	Optional sub key. Files at “location” will only be monitored if they match this pattern. Wildcards are allowed.	
	<b>exclude</b>	Optional sub key. Files at “location” will NOT be monitored if they match this pattern. Wildcards are allowed.	
	<b>delete-on-size-bytes</b>	<p>Optional sub key. Files at “location” will be deleted once they reach the size specified in this key (in bytes). Optional letters can be post-fixed to change the multiplier (K for kilobytes, M for megabytes, or G for gigabytes). This option was added specifically for Exchange log files, which can grow unbounded.</p> <p>EXERCISE CAUTION AND DISCRETION with this option - the LCE Client will attempt to delete log files above a certain size with this option.</p>	
	If “flat-file” holds sub-keys, then “location” is the fully qualified path and file name. The other sub keys apply ONLY to the files monitored at this specified location.		
<b>interval-log-seconds</b>	The number of seconds between scanning logs watched with “flat-file” and “event-log”.		A non-zero integer
<b>tail-subdirectories</b>	Whether or not to follow subdirectories given in “flat-file” and “flat-file” “location” values. Setting this to “1” when watching large directories with no include/exclude filters (like C:\\Windows) may impact performance.		0 or 1 (0=off,1=on)
<b>monitor-file</b>	<p>The full path and name of a file to monitor. If the file changes, the old and new MD5 checksums are sent immediately in an event to the LCE server. The maximum number of files that can be specified is 63. If multiple files are being monitored in the same directory location, monitoring the parent directory is suggested.</p> <div><p>Recursively monitoring large directories such as C:\\Windows will use more CPU and memory resources, and impact the performance of any LCE Client features configured to operate at faster-than-default intervals.</p></div>		Any fully qualified path and file name, with the file extension. It is best practice to escape folder separators with a backslash.
<b>monitor-file</b>	<b>Sub Key</b>	<b>Description</b>	<b>Valid Values</b>
	<b>location</b>	The full path at which to monitor binary files. For each file that changes, the old and new MD5 checksums are sent in an event to the LCE server.	Any fully qualified path.
	<b>include</b>	Optional sub key. Files at “location” will only be monitored if they match this pattern. Wildcards are allowed.	Optional sub key. Files at “location” will only be monitored if they match this pattern. Wildcards are allowed.

	<b>exclude</b>	Optional sub key. Files at “location” will NOT be monitored if they match this pattern. Wildcards are allowed.	Optional sub key. Files at “location” will NOT be monitored if they match this pattern. Wildcards are allowed.
<b>interval-monitor-seconds</b>	<div><div>This option is deprecated in the LCE Windows 4.4 client version and ignored, but will be enforced on 4.2 client versions.</div></div> <p>The number of seconds between scanning files watched with “monitor-file”.</p>	A non-zero integer.	
<b>monitor-subdirectories</b>	Whether or not to follow subdirectories given in “monitor-file” and “monitor-file” “location” values. Setting this to “1” when watching large directories with no include/exclude filters (like C:\\Windows) may impact performance.	0 or 1 (0=off,1=on)	
<b>send-new-events-only</b>	Whether to only send new events encountered. Setting this to “0” results in sending all data in all logs every time the client is restarted or when the policy changes.	0 or 1 (0=off,1=on)	
<b>heartbeat-frequency</b>	The number of seconds between each client heartbeat message to the LCE server. If “0”, it will not send heartbeats.	A positive integer.	
<b>statistics-frequency</b>	The number of minutes between each client host performance statistics report (CPU, Disk Space, and Physical Memory) sent to the LCE server. If “0”, it will not send stats.	A positive integer.	
<b>compress-events</b>	Whether or not to compress events before transmitting them to the LCE server. Marginally saves bandwidth, marginally increases CPU usage.	0 or 1 (0=off,1=on)	
<b>compression-level</b>	This can be used to further define “compress-events”. The value can be 0-9 with the highest value offering the most amount of compression. The more compression that is used, the more impact it will have on CPU usage.	0-9	
<b>minimum-compression-ratio</b>	The minimum compression ratio is the minimum required ratio of the original data size to the compressed data size. If the ratio satisfies the minimum ratio, compression is used for that subset of events in transit to the LCE server. Otherwise, compression is not used for that subset of events. Lower this value to compress packets more often.	1.0-10.0	
<b>minimum-compression-input-size</b>	This defines the size of event packets that will be compressed in bytes. The lower the value, the more often compression takes place. For example, a lower value of 100 can be used if the goal is to compress	0-1500	

	more packets.	
<b>info</b>	Enable or disable info-level logging in lce_client.log (the LCE client debugging log).	0 or 1 (0=off,1=on)
<b>verbose</b>	<div>  <p>This option is deprecated in the LCE Windows 4.4 client version and ignored, but will be enforced on 4.2 client versions.</p> </div> <p>Enable or disable verbose logging in lce_client.log (the LCE client debugging log).</p>	0 or 1 (0=off,1=on)
<b>debug</b>	Enable or disable debugging messages in lce_client.log (the LCE client debugging log). This is NOT recommended to be set to 1 unless specifically directed by Tenable Network Security.	0 or 1 (0=off,1=on)
<b>report-unknown-processes</b>	<p>Including this option in the LCE Windows client policy will report a new event of “<b>LCE_Client_Detected_Unknown_Process</b>” for each unknown process encountered by the LCE server. These events will be sent only once for each unknown process detected.</p> <p>An example of how this is configured in the LCE client policy is shown below.</p> <pre>&lt;report-unknown-processes&gt;1&lt;/report-unknown-processes&gt;</pre>	<p>0=off</p> <p>1= A list of LCE_Client_Detected_Unknown_Process events will be sent only once, and subsequently only newly encountered DLLs and EXEs will be reported.</p> <p>2= The list of reported unknown processes will be cleared every time the client is restarted or a new policy is received. Either of these will cause all of the existing unknown dlls and exes to be re-sent to the LCE server.</p>
<b>malware-scan-frequency</b> (LCE Client 4.2.X)	<p>This option specifies the interval (in seconds) that the LCE client will scan running processes, and monitored directories. The following configuration items can be used to tune the scanning, but should not be specified in LCE Client 4.0 policies. This configuration can only be used in LCE Client 4.2 policies and later. An example of the &lt;malware-scan-frequency&gt; tag is shown below:</p> <pre>&lt;malware-scan-frequency&gt;3600&lt;/malware-scan-frequency&gt;</pre> <div>  <p>Note: Each time a malware scan is run there will be an increased amount of DNS network traffic to *.l2.nessus.org.</p> </div>	
<b>custom-malware-hashes</b>	<p>Custom malware hashes can also be specified, and will be flagged as malware if they are detected by the LCE client, via the &lt;custom-malware-hashes&gt; tag. More than one hash can be entered, but it does require that each hash have the “&lt;custom-malware-hashes&gt;” open and closed tag This configuration can only be used in LCE Client 4.2.X policies. An example is shown below.</p> <pre>&lt;custom-malware-hashes&gt;0e17d427520db98aa72f5c509f015f5e1866eda15efde7fc1d4360da92b315e3&lt;/custom-malware-hashes&gt;</pre>	

<b>whitelist-hashes</b> (LCE Client 4.2.X)	<p>This option is made available for MD5 file hashes that can be ignored by LCE that may otherwise be considered malware. More than one hash can be entered, but it does require that each hash have the “&lt;white-list-hashes&gt;” open and closed tag. This configuration can only be used in LCE Client 4.2.X policies. An example of how this option can be used is shown below.</p> <pre>&lt;whitelist-hashes&gt; 0e17d427520db98aa72f5c509f015f5e 1866eda15efde7fc1d4360da92b315e3 &lt;/whitelist-hashes&gt;</pre>																							
<b>host</b>	<p>Host contains sub keys describing a remote machine on which this LCE Client will perform monitoring via the WMI interface. The maximum character length of each “sub key” is 63 characters.</p> <table><tr><th>Sub Key</th><th>Description</th><th>Valid Values</th></tr><tr><td><b>ip</b></td><td>The IP address of the remote machine to monitor</td><td>A valid IPv4 address.</td></tr><tr><td><b>namespace</b></td><td>The namespace of the WMI classes being monitored (almost always root\cimv2)</td><td>A valid WMI namespace.</td></tr><tr><td><b>domain</b></td><td>The domain of the remote machine to monitor</td><td>A valid domain name.</td></tr><tr><td><b>user</b></td><td>The username of the account on the remote machine that should be used for monitoring</td><td>A valid user account.</td></tr><tr><td><b>password</b></td><td>The password to use to login to the user account</td><td>A valid password. Be sure to escape special XML characters.</td></tr><tr><td><b>logfilename</b></td><td>A remote NT Log file to monitor</td><td>The name of a remote NT Log file.</td></tr></table>			Sub Key	Description	Valid Values	<b>ip</b>	The IP address of the remote machine to monitor	A valid IPv4 address.	<b>namespace</b>	The namespace of the WMI classes being monitored (almost always root\cimv2)	A valid WMI namespace.	<b>domain</b>	The domain of the remote machine to monitor	A valid domain name.	<b>user</b>	The username of the account on the remote machine that should be used for monitoring	A valid user account.	<b>password</b>	The password to use to login to the user account	A valid password. Be sure to escape special XML characters.	<b>logfilename</b>	A remote NT Log file to monitor	The name of a remote NT Log file.
Sub Key	Description	Valid Values																						
<b>ip</b>	The IP address of the remote machine to monitor	A valid IPv4 address.																						
<b>namespace</b>	The namespace of the WMI classes being monitored (almost always root\cimv2)	A valid WMI namespace.																						
<b>domain</b>	The domain of the remote machine to monitor	A valid domain name.																						
<b>user</b>	The username of the account on the remote machine that should be used for monitoring	A valid user account.																						
<b>password</b>	The password to use to login to the user account	A valid password. Be sure to escape special XML characters.																						
<b>logfilename</b>	A remote NT Log file to monitor	The name of a remote NT Log file.																						

## For More Information

Tenable has produced a variety of documents detailing the LCE's deployment, configuration, user operation, and overall testing. These documents are listed here:

- [Log Correlation Engine 4.2 Architecture Guide](#) – provides a high-level view of LCE architecture and supported platforms/environments.
- [Log Correlation Engine 4.4 Administrator and User Guide](#) – describes installation, configuration, and operation of the LCE.
- [Log Correlation Engine 4.4 Quick Start Guide](#) – provides basic instructions to quickly install and configure an LCE server. A more detailed description of configuration and management of an LCE server is provided in the "LCE Administration and User Guide" document.
- [Log Correlation Engine 4.4 Client Guide](#) – how to configure, operate, and manage the various Linux, Unix, Windows, NetFlow, and other clients.

- [Log Correlation Engine 4.4 OPSEC Client Guide](#) – how to configure, operate, and manage the OPSEC Client.
- [LCE 4.4 High Availability Large Scale Deployment Guide](#) – details various configuration methods, architecture examples, and hardware specifications for performance and high availability of large scale deployments of Tenable's Log Correlation Engine (LCE).
- [LCE Best Practices](#) – Learn how to best leverage the Log Correlation Engine in your enterprise.
- [Tenable Event Correlation](#) – outlines various methods of event correlation provided by Tenable products and describes the type of information leveraged by the correlation, and how this can be used to monitor security and compliance on enterprise networks.
- [Tenable Products Plugin Families](#) – provides a description and summary of the plugin families for Nessus, Log Correlation Engine, and the Passive Vulnerability Scanner.
- [Log Correlation Engine Log Normalization Guide](#) – explanation of the LCE's log parsing syntax with extensive examples of log parsing and manipulating the LCE's `.prml` libraries.
- [Log Correlation Engine TASL Reference Guide](#) – explanation of the Tenable Application Scripting Language with extensive examples of a variety of correlation rules.
- [Log Correlation Engine 4.0 Statistics Daemon Guide](#) – configuration, operation, and theory of the LCE's statistic daemon used to discover behavioral anomalies.
- [Log Correlation Engine 3.6 Large Disk Array Install Guide](#) – configuration, operation, and theory for using the LCE in large disk array environments.
- [Example Custom LCE Log Parsing - Minecraft Server Logs](#) – describes how to create a custom log parser using Minecraft as an example.

Documentation is also available for Nessus, the Passive Vulnerability Scanner, and SecurityCenter through the Tenable Support Portal located at <https://support.tenable.com/>.

There are also some relevant postings at Tenable's blog located at <http://www.tenable.com/blog> and at the Tenable Discussion Forums located at <https://discussions.nessus.org/community/lce>.

For further information, please contact Tenable at [support@tenable.com](mailto:support@tenable.com), [sales@tenable.com](mailto:sales@tenable.com), or visit our web site at <http://www.tenable.com/>.

## About Tenable Network Security

Tenable Network Security provides continuous network monitoring to identify vulnerabilities, reduce risk and ensure compliance. Our family of products includes SecurityCenter Continuous View™, which provides the most comprehensive and integrated view of network health, and Nessus®, the global standard in detecting and assessing network data. Tenable is relied upon by more than 20,000 organizations, including the entire U.S. Department of Defense and many of the world's largest companies and governments. For more information, visit [tenable.com](http://tenable.com).



## Appendix 1: Non-Tenable License Declarations

Below you will find third party software packages that Tenable provides for use with the Log Correlation Engine.

Section 1 (b) (ii) of the Log Correlation Engine License Agreement reads:

(ii) The Software may include code or other intellectual property provided to Tenable by third parties (collectively, “Third Party Components”). Any Third Party Component that is not marked as copyrighted by Tenable is subject to other license terms that are specified in the Documentation. By using the Software, you hereby agree to be bound by such other license terms as specified in the Documentation.

The Log Correlation Engine’s Software License Agreement can be found on the machine in the top-level directory for the LCE application, `/opt/lce`.

### Related 3<sup>rd</sup> Party and Open-Source Licenses

#### blowfish.h

This product includes cryptographic software written by Eric Young ([eay@mincom.oz.au](mailto:eay@mincom.oz.au)).

This product includes software written by Tim Hudson ([tjh@mincom.oz.au](mailto:tjh@mincom.oz.au)).

crypto/bf/blowfish.h

Copyright (C) 1995-1998 Eric Young ([eay@mincom.oz.au](mailto:eay@mincom.oz.au))

All rights reserved.

This package is an SSL implementation written by Eric Young ([eay@mincom.oz.au](mailto:eay@mincom.oz.au)).

The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson ([tjh@mincom.oz.au](mailto:tjh@mincom.oz.au)).

Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

“This product includes cryptographic software written by Eric Young ([eay@mincom.oz.au](mailto:eay@mincom.oz.au))”

The word ‘cryptographic’ can be left out if the routines from the library being used are not cryptographic related :-).

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement:

“This product includes software written by Tim Hudson ([tjh@mincom.oz.au](mailto:tjh@mincom.oz.au))”

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

## **libCURL**

### **COPYRIGHT AND PERMISSION NOTICE**

Copyright (c) 1996 - 2011, Daniel Stenberg, <daniel@haxx.se>.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

## **OpenSSL**

Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:  
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [openssl-core@openssl.org](mailto:openssl-core@openssl.org).

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment:  
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

## **zlib**

(C) 1995-2010 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly     Mark Adler  
jloup@gzip.org     madler@alumni.caltech.edu

## **Hash functions**

'[Hash functions](#)' is Copyright 2004-2008 by Paul Hsieh, and distributed under the [LGPL 2.1 license](#).

## **OpenBSM**

[OpenBSM](#) is covered by a number of copyrights, with licenses being either two or three clause BSD licenses. Individual file headers should be consulted for specific copyrights on specific components.

### **libpcap**

License: BSD

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

### **libmcrypt**

[libmcrypt](#) (part of the mcrypt project) is distributed under the [LGPL 2.1 license](#).

### **libxml2**

[Libxml2](#) is the XML C parser and toolkit developed for the Gnome project (but usable outside of the Gnome platform), it is free software available under the [MIT License](#).