CorreLog®

Syslog Windows Tool Set (WTS)

User Reference Manual

http://www.CorreLog.com mailto:support@CorreLog.com

Microsoft SILVER CERTIFIED Partner

CorreLog Syslog WTS, User Reference Manual

Table of Contents

Section 1: Introduction	……..	5
Section 2: CorreLog WTS Installation	……..	9
Section 3: CorreLog WTS Usage	……..	15
Section 4: CO-sysmsg Config File	……..	23
Section 5: Remote Configuration	……..	45
Section 6: CO-tsend Tunnel Sender	……..	51
Section 7: Automatic Deployment	……..	61
Section 8: CorreLog WTS FAQs	……..	67
Appendix A: The CO-sysmsg.cnf File	……..	73
Appendix B: Syslog Protocol	……..	79
Appendix C: Severity Mappings	……..	85

Section 1: Introduction

This document contains installation and application notes regarding the CorreLog Windows Agent and Windows Tool Set (CorreLog WTS) which is a compact set of software tools that instrument a Windows Vista, XP, or 200X operating system with Syslog capability. This permits the CorreLog Security Correlation Server to effectively manage Windows platforms, in addition to managing Unix platforms, routers, and mainframe computers.

The CorreLog WTS is very lightweight and easy to install. The program should be installed on each Windows platform of interest in an organization or enterprise.

If you are unfamiliar with Syslog protocol as a management technique, refer to the CorreLog User Reference manual, which contains a comprehensive description of Syslog functionality. If you wish to get started immediately with the installation of the CorreLog WTS, see notes at the bottom of this current section.

This manual should be of interest to network managers and administrators, responsible for installing and maintaining CorreLog. This manual will also be useful to any developers who are interested in using this software as a basis for larger enterprise management strategies.

CorreLog Windows Tool Set (WTS) Overview

The CorreLog Syslog Windows Tool Set (WTS) is a collection of executables and files that add Syslog capability to Windows platforms. In particular, this tool set includes a non-intrusive Windows Agent program that relays Syslog messages to a Syslog receiver, permitting easy integration of the CorreLog Security Correlation Server with Microsoft Windows platforms.

Note that, with no special software installation, the CorreLog system works with Unix platforms, Routers, and various application programs. However, the native Windows platform does not support Syslog protocol. This is easily fixed by installing the CorreLog WTS on the platform, as described herein. After installation, event log messages will immediately begin forwarding over to the CorreLog Server, permitting data aggregations and correlation.

The CorreLog WTS consists of the following programs.

- **CorreLog Syslog Message Service.** This is a compact but powerful Windows service, which listens for new events in the event log, and then converts these events to Syslog messages. The process uses minimal CPU and memory, and runs as a normal Windows service on XP, Vista, and 200X servers.

- **CorreLog Logfile Monitor Service.** This is a compact but powerful utility program (actually incorporated in the CorreLog Syslog Message Service above, but separately enabled), which allows arbitrary log files to be instrumented with match patterns. When specific match patterns are detected in streaming log files, Syslog messages of the appropriate severity and facility are sent to the Syslog server program.

- **CorreLog Sendlog Utility**. This is a simple utility that can be used in batch files and launched by application programs to send Syslog messages to a Syslog server host. The utility is a completely stand-alone executable that relies on no other files or DLLs in the system, hence is easily adapted to user specific applications.

- **CorreLog Remote Configuration Utility**. This is a stand-alone program that permits the user to download and upload the configuration file of the CorreLog Syslog Message and Logfile Monitor Service, so that batch configuration of these programs can be remotely performed.

- **CorreLog Tunnel Sender Service.** This is a program that can be optionally installed to send both Syslog messages and SNMP Trap messages to the CorreLog system using reliable and encrypted TCP. This program provides the user with various important security options related to firewalls, routing, and extra encryption.

The above programs are documented in this manual, including installation and configuration, along with extra application notes that describe how to perform advanced configuration of the system.

CorreLog Windows Tool Set Interoperability

Note that, although these programs contain special features (such as data encryption) that work with the CorreLog server, the programs also work with any standard Syslog receiver. Although this manual will assume that these programs are being installed to support the CorreLog server, all information in this manual (unless otherwise specifically noted) can be generalized to apply to any Syslog receiver program running on the network.

Because the Syslog is highly interoperable, the installer can leverage this standards based protocol to develop new management techniques that interoperate with a variety of other software.

Using CorreLog WTS with CorreLog Server

The CorreLog WTS is already an embedded part of the main CorreLog installation package, and does not need to be installed on the same platform as the CorreLog Server. (The exception to this might be if the user wishes to direct the event log messages to another Syslog server.) By default, the CO-sysmsg.exe program, which monitors the event logs, is installed as part of the base CorreLog system, and directs messages to “localhost”, i.e. to the locally running copy of the CorreLog Server.

The CorreLog WTS software contains one special feature that can be used ONLY with the CorreLog, and that is the data encryption functions of the software. The user can enable data encryption of Syslog messages between the CorreLog WTS and CorreLog Server for those situations that might warrant this (such as sending messages across a public internet.) More information on the data encryption function is found in the section of this manual dealing with the Syslog Message Server Configuration file.

The CorreLog Windows Tool Set, Fast Start

The remainder of this manual will deal with the various detailed aspects of the CorreLog WTS software in detail. For those users wishing a quick start, the following information will get the CorreLog WTS software up and running as quickly as possible on a Windows platform, permitting you to immediately begin using the program.

- The CorreLog WTS is obtained as a single self-extracting WinZip file from various locations. In particular, the user can download this package with a web browser directly from the CorreLog Server “Home” screen.

- The user executes the self-extracting WinZip file, and extracts files to the desired location, by default the C:\CorreLog directory of the platform.

- When the files are extracted, the installation dialog automatically starts. The user needs to provide only one argument to the installation dialog, which is the destination hostname or IP address of the Syslog server running on the network. (This will generally be the hostname or IP address of the platform running the CorreLog Server software.)

- When the dialog finishes, the CorreLog Syslog Message Server is installed and started. The user can check the Syslog file of the destination host to verify that a Syslog message was correctly sent and received. The platform does not need to be rebooted.

The entire installation steps, outlined above, will usually take about one minute or so to complete. An “Administrator” type login is required. If the installation fails (for example if the installer mistypes the destination hostname or IP address) the installation procedure can be run again without running the uninstall program.

Future sections will describe in detail the various other features, adaptations, customizations, and applications associated with the CorreLog WTS system. The reader is encouraged to experiment with the system. In particular, almost all of the information required to understand the essentials of the CorreLog WTS system has now been explained. You can begin exploring your enterprise Syslog information right now!

Section 2: CorreLog WTS Installation

The CorreLog WTS is usually delivered as a self-extracting WinZip file, and contains install and uninstall programs, residing in the “wintools” directory of the CorreLog root directory. The install program normally starts after files are extracted. To uninstall the system, the operator accesses the Windows “Add / Remove” programs application, and clicks on the “CorreLog Framework” entry.

CorreLog WTS is specifically designed not to scatter DLL or other files into system directories. All files within the CorreLog directory reside in the CorreLog root directory, by default the directory C:\CorreLog (although this directory may be specified differently when extracting files.)

CorreLog is uninstalled via the standard Windows “Add / Remove” programs screen (or "Program Features" screen on Vista platforms.). Additionally, if the user stops a CorreLog Syslog Message service, the entire CorreLog directory can be simply dragged and dropped into the Windows “Recycle Bin” and this will effectively discard the entire installation. (However, note that this will still leave the service entry for CorreLog, within the Windows Service Manager, which is normally cleaned up by the Uninstall procedure.)

The CorreLog install and uninstall programs are very simple, and require very little explanation. This section describes the detailed steps needed to install and uninstall the system, as well as containing application notes that may be useful to system developers and managers.

CorreLog WTS System Installation Requirements

The CorreLog WTS programs are non-invasive, and can be installed on a variety of Windows platforms and operating systems. An “Administrator” login is required to install the software. Specific system requirements of the CorreLog Server are described below.

- **Disk Space.** The CorreLog WTS has a small disk footprint of less than 0.5 Mbytes. The program should generally NOT be installed on a network drive. If possible, the program should be consistently installed in its default location of C:\CorreLog, which may assist technical support personnel.

- **CPU Requirements.** The CorreLog WTS makes minimal use of CPU, and can co-exist with other server components and applications. The actual CPU requirements will typically be much less than 1%, even under heavy load.

- **TCP Connectivity.** The CorreLog WTS cannot be installed on a platform that does not have TCP connectivity. (This will typically not be a problem, but may occur in certain evaluation and test scenarios.) The program works with any normal network interface card.

- **Service Ports.** The CorreLog WTS program requires access to the UDP 514 port of the configured destination host, which may require modifications to firewalls and port blockers. Additionally, the CorreLog WTS will listen at TCP 55514 for optional remote configuration requests.

To insure proper installation of the program, the user should close all windows, and temporarily disable any port blocking or Virus Scan software on the system. Reboot, after installation, is not required.

Basic Installation Steps

The precise steps needed to install the CorreLog WTS program on a target platform are detailed in Section 2 of the “CorreLog Web Framework Users Manual”. These steps are also outlined below.

1. Login to the target platform with Administrative permissions.

2. If the platform is running port blocking or Virus Scan software, this software should be disabled during the installation.

3. Download a copy of the CorreLog WTS files. One location to obtain these files is from the “Home” screen of the central CorreLog Server. The user runs a web browser on the target machine, connects to the CorreLog

server, and then downloads the CorreLog WTS files by clicking the "Download Tool Set Here" hyperlink on the "Home" screen. (Other methods of obtaining the executable program include mapping disk drives or using FTP.)

4. After downloading the CorreLog WTS package, execute the self-extracting WinZip file, and extract files to the target directory, by default the C:\CorreLog directory.

5. When the self-extracting WinZip file completes, the automatic installation procedure starts. The installation dialog for the program is depicted below. The user must view the license agreement and click the check box in order to proceed to the next screen.

6. Follow the prompts of the automatic installation procedure. A single value is required from the user, on the second screen of the dialog, which is the hostname or IP address of the machine running the Syslog server (normally the CorreLog Server.)

7. When the installation is complete, the CO-sysmsg.exe program is installed and running. On startup, this process sends a single Syslog message to the configured destination host. Check that host to verify a message was correctly sent and received.

The entire installation process will normally take only one minute or so. No other steps are needed to install and start the program.

Installation Checkout

The most likely problem with the installation that a user may experience will be that a firewall or port blocker prevents the CO-sysmsg.exe program from sending Syslog messages across the network. This can be tested and verified as described here.

Once the system is running, the user can test the installation using the “sendlog.exe” program, which resides in the “CorreLog/wintools” directory. Brief help on how this program is used can be acquired by running the program at a Windows command prompt, and typing the command “sendlog –help”, which will show the syntax of the command.

Send an initial Syslog message to verify the CorreLog Server is listening for messages. At a command prompt on the CorreLog platform, type:

```
sendlog (dest) “First Test Message.” 7 1
```

The value of (dest) is the destination hostname or IP address of the platform running the Syslog receiver software, or the CorreLog Server, entered into the installation dialog. (See step #6 above.) This should cause a message from the platform to appear in the “Syslog” section of the web interface. The facility will be user(1), and the severity will be debug(7).

Make sure that any port blocking or Virus Protection program is not interfering with the proper operation of the CO-sysmsg.exe program. This is a common problem, but easily fixed by accessing the “exclusions” section of your protection software, and permitting access to UDP port 514.

Re-installing the Software

The user can re-install the server by running the CorreLog\wintools\CO-install.exe program again. (The user does not have to uninstall and re-install the program software from the WinZip file.) By executing the CO-install.exe program, the service is stopped, removed, re-installed, and reconfigured with the user specified destination host.

The platform will generally not have to be rebooted during any installation. However, under certain rare circumstances, such as if the CO-sysmsg.exe program has some conflict with third-party software, it may be necessary to re-initialize the entire platform by rebooting before performing a re-install.

Uninstalling the CorreLog WTS

The CorreLog WTS is uninstalled via the “Add / Remove Programs” windows facility. The user navigates to this screen (via the Control Panel) and clicks on the “CorreLog Syslog Message Server” entry to execute the Uninstall program. The user follows the instructions of the dialog to uninstall the CorreLog Framework system.

Note that, unlike many uninstall programs, the CorreLog Framework files are left intact on the disk. Following the uninstall procedure, the user must physically remove these files, such as by dragging the CorreLog root directory to the Microsoft Windows “Recycle Bin”. This extra step safeguards any accidental removal of data on the system.

Installing WTS via an MSI Package

Most versions of the CorreLog Server system incorporate an MSI package that assists with automated or remote installations. This MSI file normally resides in the following location of the CorreLog Server site:

```
C:\CorreLog\s-doc\wt-agent.msi
```

The above package can be installed with utilities such as Microsoft SMS or SCOM, or via the "msiexec" utility. The basic package requires passing of a command line argument that specifies the destination address as follows:

```
Msiexec /i c:\CorreLog\s-doc\wt-agent.msi dest=10.1.2.3
```

The above example will install the CorreLog package, passing the destination address for syslog messages as "10.1.2.3" (The user will typically change the path to the MSI executable, and the destination address when executing the above command.)

More information on the MSI package, including options to create your own MSI package, can be found in Section 7 of this manual.

Additional Notes

The user should refer to the “CorreLog Sigma Web Framework Users Manual” Section 2 for additional important notes, including procedures on how to configure the system to work with IIS, how to change the location of the C:\CorreLog directory, and how to rerun the CorreLog Framework Installer in order to accomplish specific objectives. For notes on the basic system parameters, how to change system logins, and how to configure the program scheduler, see Section 3 of that document.

Section 3: CorreLog WTS Usage

At many sites, the entire usage of the CorreLog WTS will consist of installing the program (as discussed in the previous section) and then rarely if ever visiting that installation again. The CorreLog WTS does not require program maintenance, and will not interfere with other system processes. The system configuration file (discussed in the next section of this manual) is ready-to-run and does not require any customization, other than the destination Syslog host supplied by the installation dialog.

However, the WTS programs have various command line options and capabilities available for general users, documented in this section. Specifically, the CO-sysmsg.exe program has a comprehensive configuration file that is easy to understand. This service is designed to be easily modified and configured, such as to monitor streaming log files. The sendlog.exe program is specifically supplied to permit users the ability to send arbitrary Syslog messages, and support user scripting. For example, the sendlog.exe program can easily be incorporated in any Windows batch file to send notifications to the CorreLog (or other) Syslog server.

This section provides detailed notes on the WTS tool command line options and application notes suitable for use by administrators and developers wishing to extend the Windows Syslog monitoring capabilities of their organization. The section will be of interest to other users wishing to assess the capabilities of the WTS tools, and Syslog protocol in general.

The CO-sysmsg.exe Program

The CO-sysmsg.exe program, normally residing in the CorreLog\wintools directory, executes as a single persistent background process, and as a standard Windows service. The program can be seen in the Windows Service Manager, with the name “CorreLog Syslog Message Server”. The program can be started and stopped from that location, or can be started and stopped via the “net” command, with the short name “CorreLog Message”. The program is normally configured in the Windows Service Manager to start automatically when the host platform boots. Users can verify the program is running by finding the CO-sysmsg.exe program name in the Windows Task Manager.

The CO-sysmsg.exe program monitors the event logs including the “Application”, “System”, “Security”, and other third-party event logs. When a new event message is found, it sends the message with an appropriate facility and severity to the Syslog receiver. By default, all messages logged by the Windows Event Log facility are relayed to the destination host (but the configuration file can modify this behavior.)

The destination address for all messages is configured in the “CO-sysmsg.cnf” file, which is in the same directory as the “CO-sysmsg.exe” program. This file MUST exist in that location, and is read whenever the CO-sysmsg.exe program starts. A detailed explanation of this configuration file, including all directives that can be included in the file, is provided in the next section of this manual.

The CO-sysmsg.exe program creates the CO-sysmsg.log file in the same directory as the executable program and the configuration file. This log file can contain any errors, and can contain Syslog messages (if so configured). The file is overwritten each time the server starts, and typically has just a very few lines. This log file will not need any maintenance.

CO-sysmsg.exe Command Line Arguments

The CO-sysmsg.exe program contains various command line options that allow the program to execute at a command prompt. While these command line options are never required, it may facilitate certain user operations, especially in batch files. The various options of the program are as follows:

CO-sysmsg –install

The –install option causes the program to be installed as the “CorreLog Syslog Message Service” in the Windows Service manager. If the service is already installed, no action occurs. This is normally executed by the CO-install.exe program, but can be executed manually to re-install the service.

CO-sysmsg –remove

The –remove option removes the program from the Windows Service Manager, first stopping the service (as needed.) This is normally executed by the CO-install.exe program, but can be executed manually to uninstall the service.

CO-sysmsg –start

The –start option starts the “CorreLog Syslog Message Service”, identical to starting the service via the Windows Service Manager, or executing the “net start “CorreLog Message” command. If the service is already started, this option has no affect.

CO-sysmsg –stop

The –stop option stops the “CorreLog Syslog Message Service”, identical to ending the service via the Windows Service Manager, or executing the “net stop “CorreLog Message” command. If the service is already stopped, this option has no affect.

CO-sysmsg -mode auto | manual | disable

The –mode option must be followed by the keyword “auto”, “manual”, or “disable”, and modifies the “CorreLog Syslog Message Service” startup mode, identical to making this modification via the Windows Service Manager.

CO-sysmsg –permit

The –permit option tests the permissions of the user to access the Windows Service Manager. The program displays the status of the permissions, either “available” or not.

CO-sysmsg –foreground

The –foreground option executes the CO-sysmg.exe program as a foreground process, without the service manager. In addition to sending Syslog messages to the receiver, the program displays any internal error messages or warnings, and additionally displays message to standard output.

CO-sysmsg –check

The –check option executes the CO-sysmsg.exe program, checking the configuration file, and displaying any error messages. The program exits after the checks. This option is useful if any changes are made to the configuration data. (Error messages are always logged to the “CO-sysmsg.log” file for later inspection.)

CO-sysmsg –help

The –help option displays brief help on the above options.

Note that the CO-sysmsg.exe program MUST be executed with at least one command line argument. One of the above command line arguments is required for interactive usage. If the program is executed with no arguments, the program will either hang, or will exit with no message. (This is the mode of operation used by the Windows Service Manager.)

The Sendlog.exe Program

The Sendlog.exe program, normally residing in the CorreLog\wintools directory, is a simple but powerful utility that allows a user to send an arbitrary Syslog message of arbitrary severity and facility to a Syslog host. The program does not require the execution of the CorreLog Syslog Message Server, or any other DLL or program on the Windows platform. It is completely stand-alone, and can be copied or moved to any location on any system.

The command can be used at a command prompt, in a windows batch file, and can also be launched from other scripting languages such as Perl, PHP, Ruby, VB, and many others.

Sendlog.exe Command Line Arguments

The Sendlog.exe program requires three, four, or five arguments. The operator can execute the command with no arguments to display brief help on program usage. The basic syntax of the command is as follows:

```
Sendlog (destname) "message" [(sevnum) [ (facilnum) ] ]
```

Each argument is explained as follows:

(destname)

The first argument is required, and is the hostname or IP address of the device that is running the Syslog receiver, which must be listening to port 514 at the address. Either an IP address, or an official hostname or alias is required as the first argument.

(message)

The second argument is required, and is the message to send to the remote Syslog server. The message will require double quote marks if the message contains any spaces. The message can be up to 1024 characters long, and should generally not contain any strange characters, to promote readability of the message by end-users. The special value of "-stdin", when used as the message, causes the Sendlog.exe program to read standard input (useful for sending files of data to the Syslog server.)

(sevnum)

The third argument is optional, and is the severity number, ranging from 0=emergency, to 7=debug. If this option is not specified, then the default severity for sending a message is 7=debug. A list of severities is provided at the end of this section.

(facilnum)

The fourth argument is optional, and is the facility number, ranging from 0=kernel to 23=local7. If this option is not specified, then the default facility for sending a message is 1=user. A list of facilities is provided at the end of this section. NOTE that, if this option is specified, the user must also specify a severity number, described above.

Sendlog.exe Command Line Examples

To illustrate the operation, consider the following examples of Sendlog program usage that might be typical in an operations center. These messages might be incorporated into a batch file, or might be the result of some program check that is launched periodically by the Windows scheduler.

Sendlog 127.0.0.1 “This is a debug message”

The above command sends a message to the Syslog program running 127.0.0.1. The severity and facility is not specified. These items default to the values of “debug” and “user”.

Sendlog 192.168.1.1 “The system is restarting” 6

The above command sends a message to the Syslog program running 192.16.1.1. The severity is “info”, and the facility is not specified, so defaults to “user”.

Sendlog myhost “Error during file transfer” 3 11

The above command sends a message to the Syslog program running at the “myhost” platform. The severity is “error”, and the facility is “ftp”. The command is fully qualified and uses all the available arguments. Note that the user can specify either an IP address, or an official hostname, as shown here.

Sendlog myhost -stdin 6 11 < filename

The above command sends the contents of (filename) to the Syslog program running at the “myhost” platform. The severity is "info", and the facility is "ftp". The command illustrates the usage of the "-stdin" option, which causes the Sendlog.exe program to read standard input (in this case the contents of "filename"), terminating when the entire file is sent.

As shown above, the severities and facility codes are useful for categorizing the message at the Syslog receiver program. Although these values are expressed by their numeric values, they are easily referenced.

Executing the “sendlog” program with no command line arguments will display a list of all the numeric severity and facility numbers, along with brief notes on usage.

The WTS also includes the "Wsendlog.exe" utility, discussed later in this section, which provides an identical function to the Sendlog program, except provides a user-friendly Windows dialog to the user. (See below.)

Encrypting Data

The “SIGMA_ENCRYPT_DATA” environmental, if set to any value, will cause the “sendlog” program to encrypt data it sends. This environmental variable should be used ONLY if the destination address is the CorreLog Server. The environmental variable can be set to any value, can be set by the administrator, or can be set in a batch file which launches the sendlog.exe program.

Note that this environmental variable affects only the sendlog.exe program, and does not affect the CO-sysmsg.exe process discussed earlier. The CO-sysmsg.exe program can also encrypt data, but that control is located in the configuration file for the program, discussed in the next section.

Using an environmental variable rather than a command line option makes the usage of the sendlog.exe program easier (because the user does not have to remember another command line argument.) However, the more important reason for making this setting an environmental variable, rather than a command line option, is that it makes a naïve hacker less able to transfer information from the platform using the sendlog.exe program. Hence, setting this environmental variable is a more secure way of enabling encryption.

The Wsendlog.exe Program

In addition to the Sendlog utility program, the CorreLog WTS provides an alternate Windows based utility named "Wsendlog.exe", which employs a user-friendly Windows dialog to send messages.

The "Wsendlog.exe" utility operates in a fashion identical to the "Sendlog.exe" program. It permits the user to specify a destination IP address, facility, and severity, along with a command line for entering the message to send. The utility may be useful for testing correlation rules or other features of the CorreLog system.

The "Wsendlog.exe" program does not make use of encryption, hence will work with any Syslog receiver (in addition to CorreLog). The program is completely stand-alone, and does not require any DLL programs or support software. It may be copied to any Windows platform and used by itself.

The CO-tsend.exe Tunnel Sender Program

The CorreLog WTS provides one special utility to support secure TCP tunneling of messages through firewalls and routers. The "CO-tsend.exe" program is the "CorreLog Tunnel Sender Service" program, and works with the "CO-trecv.exe" program of the CorreLog system.

By default, neither the CO-tsend.exe nor the CO-trecv.exe program is installed or configured. The administrator must manually install the CO-tsend.exe service on one or more Windows platforms, and manually edit the configuration files for the program. The administrator should also install the CO-trecv.exe program at the CorreLog server.

Neither of these programs is required for proper operation of the CorreLog server. However, if a site as special firewall, routing or encryption requirements, these programs can enhance the over-all security of the CorreLog system.

Detailed notes on the CO-tsend.exe and CO-trecv.exe programs are provided in later sections of this manual, including the steps needed to install and configure the program on both a client windows platform, and also on the central CorreLog server.

The Rsmconf.exe Utility

The CorreLog WTS includes a utility that permits remote configuration changes of the CO-sysmsg.exe program. This utility is found in the "system\rsmconf.exe" location of the main CorreLog Server. The utility permits an administrator (with authentication and security) to remotely change the configuration of the CO-sysmg.exe program, assisting in the configuration and maintenance of the program. Note that the program executes only on a licensed copy of the main CorreLog server program.

The configuration of the CorreLog WTS is discussed in detail within the sections that follow. Although it may not be necessary to ever change the default settings of the CO-sysmsg.exe program, it may also be the case that match patterns, log file monitors, and other parameters will need to be maintained, especially during the initial setup and configuration of the system.

The Rsmconf.exe program permits the user to download and upload the configuration file from a CO-sysmsg.exe program. When uploading changes, the new configuration immediately takes affect within the CO-sysmsg.exe program

without requiring a restart of the service. Extensive checks and security features are incorporated into the system as detailed within later sections of this manual.

Note that the Rsmconf.exe program is agreeable to use within batch files, so that an administrator can easily apply large numbers of changes to an enterprise. Also note that the CorreLog server itself can effect these changes via the web interface. These remote configuration features are discussed in more detail within a separate section of this manual.

Section Summary And Additional Notes

1. The “CorreLog Syslog Message Service” monitors Windows event logs for changes, and sends Syslog messages to the destination host.

2. The destination host address, which receives messages, is configured in the CO-sysmsg.cnf file, which is in the same location as the CO-sysmsg.exe program, by default the directory “C:\CorreLog\wintools”.

3. The CO-sysmsg.cnf file MUST exist in the above directory, and specifies a variety of parameters and configuration items detailed in the next section.

4. The CO-sysmsg.exe program, which is the program corresponding to the “CorreLog Syslog Message Service” reads its configuration data only on startup.

5. The sendlog.exe program is a stand-alone executable that permits programmers to send Syslog messages to arbitrary hosts, using arbitrary messages, severities and facility codes.

6. The wsendlog.exe program provides a function identical to the sendlog.exe program, except provides a user-friendly user dialog (rather than being a command line program.)

7. Setting the SIGMA_ENCRYPT_DATA environmental variable to any value will cause any data sent by the sendlog.exe program to be automatically encrypted. This is useful ONLY if the target Syslog receiver is the CorreLog system.

8. The CO-tsend.exe program is the CorreLog Tunnel Sender Service, which can be manually installed on one or more Windows platform to provide secure TCP transmissions of Syslog and SNMP trap messages to the CorreLog system.

9. The CO-tsend.exe program is not required for proper operation of CorreLog, however may enhance the overall security of a site, and handle special firewall and routing issues.

Section 4: CO-sysmsg Config File

The CO-sysmsg.cnf file contains all the parameters and specifications related to the program's operation. This file is found in the same directory as the CO-sysmsg.exe program, by default the "C:\CorreLog\wintools\CO-sysmsg.cnf" file. An example of this file is found in Appendix A of this document.

There is no required editing of this file. The installation dialog creates a version of this file that will be adequate for many (and perhaps most) situations. However, if a user wishes to fine tune the parameters of the Syslog messages, or wishes to monitor streaming log files in addition to the Windows Event logs, or needs to change the location of the CorreLog Syslog destination, the file can be edited with a standard text editor, as explained here, or modified via the remote configuration functions as detailed in later sections of this manual.

If the configuration file changes via a manual edit, the user must stop the CO-sysmsg.exe service and restart the service. Any errors detected while reading the configuration file are logged to the CO-sysmsg.log file, in the same directory as the CO-sysmsg.exe program and CO-sysmsg.cnf file. If the configuration file is changed via a remote configuration operation, no restart of the CO-sysmsg.exe program is required.

Detailed notes on this file, possibly of interest to administrators or developers, are provided in this section. Note that the information herein is NOT REQUIRED to install and use the CO-sysmsg.exe program, but is provided only to support more advanced applications and requirements.

Configuration Items

In addition to specifying the destination address and port number, the configuration file contains a number of other settings that can be used to specify log files (in addition to the Win32 Event Log files), as well as match patterns that set the facility and severities of the various Syslog messages.

The file contains the following sections.

- **Destination Address And Port Number.** The top of the file contains the destination and port number for Syslog messages, both of which are required for all configurations of the agent. The destination address can be specified as a hostname or IP address. The port number is usually not modified (and is the default UDP port 514, appropriate for syslog messages.)

- **Remote Configuration Parameters.** The next section of the file contains information regarding the remote configuration capability of the program, including the type of authentication and optional passkey required to permit remote configuration.

- **Auxiliary Addresses.** The next section of the file contains optional auxiliary addresses. These addresses can be used to forward the syslog information for up to eight different auxiliary addresses.

- **Optional Parameters**. Following the above fields, the user can specify ancillary parameters, such as whether encryption is to be used. These optional parameters apply to all event log and log file monitors configured in later sections.

- **Event Log Specifications.** Following the “Optional Parameters” section are multiple entries that list all the Win32 event logs, and the facilities to use for each event log. The user can configure multiple match patterns for multiple facilities and severities using the “MatchKeyWord” directive.

- **Log File Monitors**. Following the Event Log specifications are multiple entries that allow the user to specify zero or more streaming log files, which can be continuously monitored by the program. The user can configure multiple log files, each with multiple patterns to control multiple facilities and severities, using the “MatchKeyWord” directive.

Each of the above items is explained in more detail within the pages that follow.

Destination Address And Port Number Directives

The CO-sysmsg.exe program requires only two directives, which must be configured in the CO-sysmsg.cnf file. The initial configuration of these two directives is performed by the installation procedure, however the user can modify the values after installation to change the destination of Syslog messages. These directives typically appear at the top of the file. Only the first occurrence of these directives is read. If these directives occur elsewhere in the file, these extraneous directives are ignored.

The following two directives are required.

DestinationAddress

This directive should be followed by an IP address, which corresponds to the location of the CorreLog Syslog receiver (typically the IP address of the CorreLog web server.) If this value is invalid, the CO-sysmsg program will not send Syslog messages.

DestinationPort

This directive should be an integer number of 514, which is the standard UDP port number used by Syslog. Generally, this value is provided mainly for reference and cannot be easily changed.

These directives are required at the very top of the file, and cannot be moved to some other section of the file. If there are multiple entries, the last entry recorded in the file for these parameters is used, and the other directives are ignored.

Optional Destination Protocol Parameter

Following the "DestinationPort" parameter, the administrator can optionally specify the "DestinationProtocol" parameter, to change the protocol from UDP to TCP (or possibly some other value, such as TLS, depending on the installed agent capabilities.) The "DestinationProtocol" value (if it is specified in the configuration file) should immediately follow the "DestinationPort" directive in the configuration file, and can be set to "tcp" (for example) to indicate messages will be sent via TCP rather than UDP. See a later section herein for further discussion about using TCP instead of the default UDP protocol.

Remote Configuration Parameters

The CO-sysmsg.exe program supports remote configuration directives by the main CorreLog server, or by the "rsmconf.exe" remote configuration utility. Following the Destination Address and Destination Port directives are a series of optional parameters to support this function.

The following three remote configuration directives are optional.

Note – these parameters can potentially affect the security of the remote agent, so caution is advised. Set the "ListenAuthMode" to 3, or delete these directives to promote the highest level of security for the agent program.

ListenAuthMode

This directive specifies the authentication mode used when processing remote requests. The directive is followed by an integer number between 0 and 3 as follows: 0=No authentication; 1=Authentication by source address; 2=Authentication by passkey; 3=Authentication by both source address and passkey. The default value is 3.

ListenPassKey

This directive is the passkey used with remote configuration when the ListenAuthMode is 2 or 3. The value serves as a simple password. (The corresponding password in the CorreLog Server is found in the System > Parameters tab of the web interface.)

ListenPort

This directive should be the integer number of 55514, which is the TCP port at which the CO-sysmsg.exe program listens for remote requests. Generally, this value is provided mainly for reference and cannot be easily changed.

ListenAdmin

This directive, if included in the configuration, may be either "True" or "False", and controls whether the agent will support the special "rsmconf.exe -admin" options, useful for special system configurations. (Contact support@correlog.com before using this directive.)

Alternate and Auxiliary Syslog Destinations

The Primary Destination Address and Destination Port directives, discussed previously, have special properties. Both of these values must be at the top of the file, and cannot be reconfigured remotely. (The user must edit these values by hand, such as with the Setup Wizard, or via the "Agent Configuration" tool.)

An administrator can add "AuxAddress" directives to the file to send the syslog message information to other locations. Zero to eight of these directives (with IP address values) can be added to the configuration file, and any messages generated by the agent program will be sent to these addresses.

The "AuxAddress" value does allow remote configuration, and does not encrypt the syslog messages. The value extends the role of the agent to include other syslog collectors (including other CorreLog Servers) as part of the collection

strategy. For example, the "AuxAddress" value can be used to send messages to a failover device, a redundant collector, or a different department within the enterprise.

To disable this function, remove the "AuxAddress" directive, or set the value to a non-valid address, such as –1.

Optional Parameters Section

Following the Remote Configuration directives (and Auxiliary Syslog Destinations, if any) are a series of optional parameters. These can be commented out or deleted. The Optional Directives that can appear in the file are as follows:

MessagePrefix

This is a phrase that will prefix any messages sent by the system. If the directive is omitted, the message is not prefixed by any special text. This can be used to distinguish the messages, such as by placing a keyword, or the device name, or the organization, or some other keyword at the very start of any message. The message prefix must be under 256 characters, and can contain static text, environmental variables (delimited by "%" characters), and / or a date and time specification.

MsgDelayMsecs

This is an integer number ranging from 10 to 5000, indicating the number of milliseconds to wait after sending a message. This is a way of throttling the number of messages that can be sent, ranging from 10 per second to only 12 per minute. This prevents any single Syslog process from flooding a Syslog message receiver. The default value, if this directive is omitted is 10 milliseconds.

MaxMessageSize

This is an integer number ranging from 200 to 5000 bytes, indicating the maximum message size to be sent by the agent, including message prefix, facility and severity codes, etc. If the value is omitted, the maximum message size is 500 bytes, which is sufficient to capture virtually all message information without taxing disk resources. (This value is particularly important for Windows 2008 platforms, which can generate highly verbose and non-pertinent messages that waste disk space and correlation CPU time.)

LogLocal

This value is set to either “True” or “False”. If the value is “True”, then all Syslog messages sent by the CO-sysmsg.exe program are also logged in the CO-sysmsg.log file (along with any error messages encountered by the program.) Additionally, setting this value to "True" creates the system

"CO-sysmsg.dbg" file, useful for diagnosing agent system errors. This provides a simple way to verify whether UDP messages are being dropped. Note that the CO-sysmsg.log file is restarted each time the service is started, hence the file does not grow without bounds. If this directive is omitted, it is interpreted to be “False”.

EncryptData

This value is set to either “True” or “False”. If the value is “True”, then the message data is encrypted before it is transmitted, which is the default. This setting will make the CO-sysmsg.exe program usable ONLY with the CorreLog Server. The program WILL NOT operate with any other Syslog server if this value is set to “True”. If this directive is omitted, it is interpreted to be “False”. (See later notes in this section for more details.)

MarkerMessage

This value is the content of a syslog message that is issued periodically to the receiving program, useful for generating a "heartbeat". The value is ignored unless the value of "MarkerMinutes" (described below) is set to a positive integer value. The message must be under 256 characters, and can contain static text, environmental variables (delimited by "%" characters), and / or a date and time specification. By default, no periodic syslog message is sent. This directive should be used only with CorreLog Version 3.5.1 or higher.

MarkerMinutes

This value is an integer number ranging from 1 to 65535 minutes, indicating how fast the "MarkerMessage" heartbeat is sent, if any. A value of zero or less disables the marker message (the default condition.) In order to send a marker message, this directive must exist in the configuration file, and the value must be a positive integer value, and the "MarkerMessage" must be defined. By default, no periodic syslog message is sent. Remote configuration of this directive requires CorreLog Version 3.5.1 or higher. (See later notes in this section for more details.)

Deduplicate

This value is an integer number ranging from 0 to 5 seconds, indicating the deduplication seconds applied to messages generated by the agent. If this value does not exist, the agent will deduplicate messages that occur within 3 seconds. The user can turn off the deduplication filter by setting this value to zero, or increase the value to 5 seconds.

MessageFormat

This value is a phrase or keyword flag that can be used to control the message format of all messages output by the agent program. Specific values include "normal", "RFC" (or RFC3169), "CEF", and "LEEF", but other values may exist to support special message formats requested by

end users. (Contact support for other values and variants.) The "MessageFormat" directive is usually omitted from the configuration file.

Event Message Encryption

As a special facility, the CO-sysmsg.exe program encrypts messages sent to the CorreLog Server system. The administrator edits the “EncryptData” directive, and sets the value to “"False” in order to disable this function. The encryption prevents casual snooping of the data by using a block rotating, time-based cipher that is built into both the CorreLog Server, and the CO-sysmsg.exe program. There will be no apparent change to the data displayed. However, if the destination address is made some other Syslog server, it will be apparent that the data is actually encrypted.

The encryption provides a fair degree of protection against network sniffers. However, since a single 1024 bit private key is used for all the transmissions, this encryption does not protect against man-in-the-middle type attacks, or replay attacks. This encryption is mainly useful for sending Syslog messages across a public Internet, where casual observers might intercept and observe the message content.

Note that encryption is available only with the "DestinationAddress" directive. Any configured "AuxAddress" directives will receive unencrypted information, and are unaffected by any message encryption settings.

Marker Message

As a special function, the agent can be configured to generate a periodic "Marker" message at a user-defined interval, such as a marker message each hour. To enable this function, the configuration file must contain both the "MarkerMessage" and "MarkerMinutes" directives, and the "MarkerMinutes" value must be greater than zero.

The Marker Message can be used to indicate that the agent is still alive on the network, and can provide a "heartbeat" function for the system. The message content, defined by the user via the "MarkerMessage" directive, must be under 256 characters, and is always send with the syslog "clock" facility and "debug" severity.

The message can contain environmental variable references, and date and time specifications. For example, to send a message once each hour that identifies the local time of the machine and the machine name, use the following as the value of the "MarkerMessage":

```
Hostname: %COMPUTERNAME% - Time: %H:%M:%S
```

Note that the %COMPUTERNAME% value corresponds to an environmental variable (*including the leading and trailing % characters*.) The %H:%M:%S values (*without a trailing % characters*) corresponds to the time at the current locale. (See later section on special log file names for a list of supported time specifications.)

Environmental variables in the marker message are expanded first, and each environmental variable must include leading and trailing % characters. After any environmental variables are expanded, time specifications are substituted. The user should be aware of possible conflicts should an environmental variable have the exact same name as a time specification.

Finally, note that the "MarkerMinutes" value indicates an interval since system startup or reconfiguration of the agent, and is not a fixed time of day. Hence, the marker message should not be used to schedule events that need to occur at a particular fixed time (such as at noon or midnight.) Although the marker message can be used to schedule events that occur at periodic intervals, the message is not tightly synchronized to the platforms internal clock, hence should not be used to drive real-time processes or perform time-of-day scheduling.

Syslog Message Formats

The default message format for messages sent by the agent is a simplified basic syslog message that contains the facility and severity code, followed by an optional message prefix, and then the content of the event log message in a reduced format (where extra spaces and tabs have been eliminated from the message content.)

This default message format can be modified by the "MessageFormat" directive. If this directive is included in the configuration file, the directive can take on the value of "default" (which is the default format described above), as well as "RFC" (or "RFC3164"), "LEEF", or "CEF". These directives change the message content as follows:

- **RFC (or "RFC3164").** This applies a standard message header containing the date and time that a message was generated, as detected from the Windows event log, followed by the hostname, followed by the message content. This is the format prescribed in RFC3164, section 4.1.2. The value can be either RFC or RFC3164 (both values are exactly equivalent.)

- **LEEF.** This reformats the message to use QRadar "LEEF" format, compliant with IBM specifications. This format works with LEEF compatible systems, but is less easily read and parsed by humans.

- **CEF.** This reformats the message to use ArcSight "CEF" format, compliant with HP specifications. In particular, the event signature portion of the CEF

message is consistently derived from the message content, and contains the Windows event identifier. This format works with CEF compatible systems, but is even harder for humans to parse.

In the absence of any other requirements, the "MessageFormat" directive should not be included in the configuration file, and the standard format described at the start of this section should be used. This facilitates interoperability with other programs, and permits easier configuration and inspection of syslog content by end users of the CorreLog Server system.

Note that the "MessageFormat" directive works best with Windows Event Log messages (which have well defined and known structures.) The setting can also affect Log File Monitor messages, but since this data is not necessarily structured, the message format may have limited use in this particular application.

Configuring TCP Transmission

By default, the agent program sends messages via UDP, which is generally best practices for interoperability and security. (UDP message reception is generally regarded as *more secure* that TCP transmission, especially when coupled with encryption, since UDP is immune to port probing and scanning.) However, the operator can configure messages to be sent via TCP as per RFC 6587 using the "DestinationProtocol" directive described earlier. This directive, if it exists in the configuration file, accepts the following values:

- **TCP.** This value indicates the agent is to send data using a TCP connection. The data encoding uses "octet counting". The message is terminated with a null character. This is the most commonly supported TCP transmission type for syslog, and is generally adequate for most TCP syslog receivers.

- **TCP CRLF.** This value indicates that the message will be terminated by a CRLF character combination. There will be no "octet count" in the message.

- **TCP CR.** This value indicates that the message will be terminated by a single CR character. There will be no "octet count" in the message.

- **TCP LF.** This value indicates that the message will be terminated by a single LF character. There will be no "octet count" in the message.

- **TCP NULL.** This value indicates that the message will be terminated by a single null character. There will be no "octet count" in the message.

The above flags permit a high degree of flexibility in configuring the TCP transmission of messages. Additional options may be available from the vendor.

Event Log Specifications

Following the optional parameters section are the event log specifications. Each event log on the system is identified, along with the default facility used for any message associated with the particular log. Additionally, each log can have a series of “UseFacility” and “UseSeverity” statements, each associated with MatchKeyWord values. This permits the user to fine-tune the Facilities and Severities of messages. The following directives are supported.

A maximum of fifty different event log and log file monitor specifications can be configured per agent. Note that these fifty specifications can be distributed in any way between the "Event Log" specifications in this section and the "Log File" specifications of the next section, but the total number of specifications cannot exceed fifty specifications total.

EventLog

This directive is followed by the name of a Windows Event Log, either “Application”, “System”, “Security”, or some other event log name that appears in the Microsoft Local Event Viewer Program. All the directives that follow, delineated by the next “EventLog” or “LogFile” directive, apply to the specified Event Log. Note that, on Windows 2008 and later systems, the user may also specify a "Windows Application Log", in addition to the standard event log. (See additional notes below.)

Formatter

This directive (if it is present) can be used to specify or change the formatting of event log messages for the particular event log. (Generally, this setting should not be specified, and is available for system level debug. Contact CorreLog support for specific information on this topic.)

DefaultFacility

This directive is must be preceded by the EventLog directive. The value specifies a facility name (or an official facility number between 0 and 23), which identifies the default facility code used in all messages that are logged to the specified EventLog.

DefaultSeverity

This directive is must be preceded by the EventLog directive. The value specifies a severity name, which identifies the default severity code used in all messages that are logged to the specified EventLog. This directive can be a number between 0=emergency and 7=debug, or can be an official severity name, or can be one of the special values of "auto" or "disabled". The value of "auto" indicates that the severity is automatically

set according to the built-in type of event message. The value of "disabled" indicates that no messages are sent unless the message specifically matches a "MatchKeyWord" directive.

UseFacility

This directive may follow the “DefaultFacility” directive, and is followed by one or more “MatchKeyWord” directives. This directive starts a series of match patterns, any of which will cause the “UseFacility” value to be specified as the message facility. This provides a way of using a facility based upon the content of a message. The value must specify a facility name (or an official facility number between 0 and 23), which identifies the facility to use if any of the match patterns that follow are satisfied. This directive is not meaningful unless immediately followed by one or more “MatchKeyWord” directives, described below. Multiple “UseFacility” directives, each followed by multiple “MatchKeyWord” directives, can be configured.

UseSeverity

This directive is similar to the “UseFacility” directive above, but affects the message severity instead of the facility code. This directive starts a series of match patterns, any of which will cause the “UseSeverity” value to be specified as the message severity. The value must specify a severity name (or an official facility number between 0 and 7, or the special “disabled” severity, or a “-1” value), which identifies the severity to use if any of the match patterns that follow are satisfied. This directive is not meaningful unless immediately followed by one or more “MatchKeyWord” directives, described below. Multiple “UseSeverity” directives, each followed by multiple “MatchKeyWord” directives, can be configured.

MatchKeyWord

This directive is nested within a “UseFacility” or “UseSeverity” directive, and specifies a single match keyword, with possible ‘*” or ‘?” wildcards. If the message content contains the match pattern, then the related severity or facility is used. Multiple patterns can be specified, without limit. The “MatchKeyWord” list is ended by any other directive, so the “MatchKeyWord” directives must all be contiguous within a single “UseFacility” or “UseSeverity” block.

Monitoring Application And Service Event Logs

On Windows 2008 and Windows 2012 systems, in addition to the standard event logs (such as "Security", "System", and "Application") The operator can add an "Application Event Log" to the system via the "EventLog" field by specifying the official name of the event log. This name is available on Windows 2008, 2012, and other post Vista systems using the "wevtutil.exe" program at a command prompt as follows:

```
C:> wevtutil.exe el
```

The above command displays an enumerated list of all application logs on the system. Any name can be added as an Event log specification (without the "Microsoft-Windows-" prefix). When entered as an "EventLog" specification, the application log will be polled for changes approximately once every 30 seconds, detecting a maximum of 100 new messages per poll cycle.

These EventLog strings include (but are not limited to) text strings such as any of the following:

```
DriverFrameworks-UserMode/Operational
PrintService/Admin
PrintService/Operational
SystemHealthAgent/Diagnostic
TaskScheduler/Debug
TaskScheduler/Operational
Windows Defender/Operational
Windows Firewall With Advanced Security/Firewall
```

Note that this feature requires the "wevtutil" library, hence is not applicable on 2003 or XP (or potentially other) Windows OS configurations. These logs MUST contain a '/' forward slash in their name, such as "/Admin", "/Operational", "/Debug", "/Diagnostic", etc. (A "Microsoft-Windows" prefix can be specified with the log name, but will be ignored by the agent.) Finally, note that this can cause certain performance problems if overused.

Notes On Event Log Specifications

As shown above, each event log has a “DefaultFacility”, followed by multiple optional “UseFacility” and “UseSeverity” statements. Each “UseFacility” and “UseSeverity” statement can have multiple “MatchKeyWord” statements. This provides a simple way to configure facilities and severities for any particular message.

If the “DefaultSeverity” directive is set to "auto", then the default severity of messages depends upon the Windows Event Log “Message Type” field, as follows:

- **Event Log Error Type.** Any event log message of this type is by default assigned a Syslog severity of “error”.

- **Event Log Warning Type.** Any event log message of this type is by default assigned a Syslog severity of “warning”

- **Event Log Info Type.** A message logged to the “System” log of this type is assigned a Syslog severity of “notice”. A message logged to any other log is assigned a Syslog severity of “info”.

- **Event Log Audit Success.** A message logged to the “Security” log of this type is assigned a Syslog severity of “notice”.

- **Event Log Audit Failure.** A message logged to the “Security” log of this type is assigned a Syslog severity of “error”.

The above default severities can be overridden by the “UseSeverity” statement discussed above. Experience shows that the above mapping will be entirely satisfactory for the vast majority, or perhaps all, of the event log messages generated by a Windows platform, for most applications.

It is quite possible (and even likely) that a message's content might match multiple “UseFacility” or “UseSeverity” statements. In that case, the following rules apply:

If a message matches multiple “UseSeverity” statements, then the severity that is actually used will be the highest severity (actually the lowest number) of any severity matched. For example, if a message matches two “UseSeverity” statements, one of which is “info”, and one of which is “critical”, then the “critical” severity is used in the transmitted message.

Likewise, if a message matches multiple “UseFacility” statements, then the facility with the highest number facility code is used as the facility in the transmitted message. If no facility is matched, but a severity is matched, then the DefaultFacility” is used.

The Special “disabled” Severity

There is no explicit “ExcludeKeyWord” type of statement. However, the user can easily exclude any message with a particular content by specifying a “UseSeverity” statement with a severity of “disabled”. This special severity is the

highest rank (actually the lowest number) and permits the user to filter or exclude any keyword that matches one of the “MatchKeyWord” directives.

For example, the user can configure a directive such as “UseSeverity disabled” (or “UseSeverity –1”) and then follow this directive with a series of MatchKeyWord values, any of which will exclude a message from the event log, regardless if a match is found elsewhere in the event log specification. The “disabled” keyword can be used only in the configuration file, is given a rank of –1 (below “emergency” = 0) and is taken as the highest severity of the system.

If the "DefaultSeverity" value is set to "disabled", then a message must specifically match one of the "MatchKeyWord" values for it to be sent. This is a way of sending messages by exception, useful for targeting only those messages of interest on the system. By default, the "Security" log, uses this technique to reduce the number of security messages sent to the CorreLog server.

Filtering Messages At the Agent

Using the "disabled" severity described above, the administrator can filter out particular messages from the system so that this data does not reach the SIEM collector. (If the collector is the CorreLog Server program, you can filter data out via the "Messages > Config > Filters" screen, but filtering at the agent furnishes a good alternative, and lowers network congestion by preventing the data from reaching the network.)

There are two main ways to filter specific messages.

- **Filtering Messages That Match A Pattern.** The operator can filter out specific messages by configuring a "MatchKeyword" and setting the severity to "disabled" as described above. If the message matches the keyword, and the severity is "disabled", then the message is not placed onto the network.

- **Filtering Messages That Do Not Match A Pattern.** The operator can filter out all messages UNLESS they match a pattern by setting the "DefaultSeverity" to "disabled", and then configuring messages that match a specific pattern. In this case, the message is "disabled" and not placed on the network UNLESS the message has been identified to be some other severity with a match pattern. (Note that this configuration is the default configuration used by the "Security" event log, with the out-of-box configuration for the agent.

Log File Monitoring

The CO-sysmsg.exe program, in addition to monitoring the Windows Event Logs, can be used to monitor multiple and arbitrary streaming log files on the system. This function can be used independent of the Windows Event Log, and permits an administrator to instrument special log files, such as the Apache HTTP server logs, Oracle database error logs, and many other logs on the system.

Only “streaming text” type log files can be monitored. That is, the log file must be normally appended with text information, with new information tacked on to the end of the file. The program cannot monitor files that continuously change size, or are written in reverse chronological order, or are not mainly ASCII text. Fortunately, this type of log file is uncommon; the vast majority of error logs, transfer logs, and transaction logs are “streaming text”, growing in size, and reset only occasionally. These log files can be monitored quite easily.

The Log File Monitoring capability is an integral part of the CO-sysmsg.exe program, and is quite powerful. This function, by itself, may very well justify the installation of the CO-sysmsg.exe program irrespective of whether an administrator wishes to monitor the native Windows event logs.

Log File Monitor Specifications

The Log File Monitor directives are similar to those used in Event Log Monitoring, discussed previously. Following the Event Log Specifications, the user can configure one or more log file monitors, default facilities and severities, and match patterns that overwrite these defaults. The following directives are supported.

A maximum of fifty different event log and log file monitor specifications can be configured per agent. Note that these fifty specifications can be distributed in any way between the "Event Log" specifications in the previous section and the "Log File" specifications of this section, but the total number of specifications cannot exceed fifty specifications total.

LogFile

This directive indicates the pathname to a streaming text log file on the system. The user can specify the pathname as a relative pathname, with respect to the location of the CO-sysmsg.exe program, or absolute pathname, using either forward or backward slashes. All the directives that follow, until the next “LogFile” directive, apply to the specified log file. This directive can contain Time Format values, such as "%y", "%m, "%d", to respectively match the two-digit year, two-digit month, or the two-digit day. For example, the file specification "C:/windows/logfiles/f-%y%m%d.log" can be used to monitor a file with a name such as "f-091231.log". Additionally, the root filename can contain a "*" wildcard to match the file

that was most recently modified. For example, the file specification "C:/windows/logfiles/*.log" matches the file in the "logfiles" directory that was most recently modified.

LogName

This optional directive, if it exists, must follow LogFile directive. It is the name of the log file (or subsystem) that is displayed in the Syslog message. If this value is not provided, the event message is not identified with the log file other than in the message content. The value can be any arbitrary text string. It is commonly punctuated with a trailing semicolon, supplied by the user, such as “Oracle Data:” or “HTTP Log File:”

Encoding

This optional directive, if it exists, describes the encoding for messages in the file. The values can be "Western", "UTF-8", "GB2312", "Big5", or "KR-EUC". If this directive does not exist (or the value of this setting is "Default") then the device type settings determine the message encoding. This switch works only for CorreLog implementations, and is not applicable to third party syslog receivers that may receive messages from the agent. The setting is mainly useful for those systems where a log file may be written using an encoding scheme that is different from the rest of the encoding used on the platform, such as when a UTF-8 encoded file is being written on a GB2312 type system.

Formatter

This optional directive, if it exists, is an interface to a custom formatter. This custom formatter consists of a DLL (supplied by CorreLog or some other vendor) that is installed at the agent, and used to process data to format syslog records. If this directive is specified, its value can be "user01", "user02", "user03", or user04", based upon information supplied by CorreLog and / or the DLL designer. (See additional notes below.)

MaxSizeChange

This optional directive, if it exists, must follow the LogFile directive. It is an integer size, in bytes. If the file increases by this number of bytes or more during a single 500 msec interval, it will trigger a special message indicating that the file has increased rapidly in size. If this value is not furnished, the value of 10,000 bytes is used. (The value may be increased to 1 Mbyte.) This parameter helps prevent excessive Syslog messages from being generated should a file undergo extremely rapid updates, such as a new file being copied into place. Each log file has its own MaxSizeChange value.

LogStatType

This optional directive, if it exists, must follow the LogFile directive, and must have a value of "Active" or "Passive". The default "Active" will cause

the relevant log file monitor to actively open log files to determine their sizes, whereas the "Passive" (default) setting uses the operating system to determine log file sizes. The "Active" setting may be useful for files that are flushed to the disk only occasionally, but this mode of operation is much more intrusive, and may cause unexpected crashes of the log processes. If the directive is not specified, log files are checked passively, i.e. a log file is not opened unless a change to the file exists.

LogStatChange

This optional directive, if it exists, must follow the LogFile directive, and have a value of "disable", "enable". The directive indicates that the monitor agent is not to read the file, but only send a Syslog message (with the "DefaultFacilty" and "DefaultSeverity") should the file modification time change. This is useful for monitoring file objects that are not necessarily log files. The file object specified by "LogFile" can be a directory or any file, including an executable file or configuration file. Note that this directive cannot be used with any "MatchKeyword" expressions. If no "LogStatChange" directive exists, then changes to the file modification times are not monitored. If the value of the associated "LogFile" directive contains an asterisk wildcard, all files that match are monitored for changes (providing a simple way to monitor changes to a directory of files.)

RequireChangeSecs

This optional directive, it exists, should follow the LogFile directive (usually after the MaxSizeChange directive, above.) The value specifies the number of seconds that may pass without the log file changing. This directive is useful for monitoring files that are expected to change within a certain period of time, such as every 300 seconds. If the file does not change in the specified period of time, then an IDLE message is generated. Note that at least one write must have occurred to the specified log file, and the directive does not apply to a log file that has not changed since agent startup.

DefaultFacility

This optional directive may follow the LogFile directive. The value specifies a facility name (or an official facility number between 0 and 23), which identifies the default facility code used in all messages that are logged to the specified file. If this directive is omitted, the default facility is assumed to be “user”.

DefaultSeverity

This optional directive may follow the LogFile directive. The value specifies a severity name (or an official facility number between 0 and y), which identifies the severity code used in all messages that are logged to the specified. The value of this directive is commonly “disabled” or “-1”,

indicating that no message is processed unless in matches one of the “UseSeverity” patterns (described below). If this directive is omitted, the default severity is assumed to be “disabled”.

UseFacility

This directive may follow the “DefaultFacility” directive, and is followed by one or more “MatchKeyWord” directives. This operates identically to the Event Log monitor directive, described previously. The directive is followed by a series of match patterns, any of which will cause the “UseFacility” value to be specified as the message facility. Multiple “UseFacility” directives, each followed by multiple “MatchKeyWord” directives, can be configured.

UseSeverity

This directive is similar to the “UseFacility” directive above, but affects the message severity instead of the facility code. This operates identically to the Event Log monitor directive, described previously. The directive is followed by a series of match patterns, any of which will cause the “UseSeverity” value to be specified as the message facility. Multiple “UseSeverity” directives, each followed by multiple “MatchKeyWord” directives, can be configured.

MatchKeyWord

This directive operates identically to the Event Log monitor directive, discussed previously. The directive nested within a “UseFacility” or “UseSeverity” directive, and specifies a single match keyword, with possible ‘*” or ‘?” wildcards. If a new log file entry matches the specified pattern, then the related severity or facility is used. Multiple patterns can be specified, without limit. The “MatchKeyWord” list is ended by any other directive, so the “MatchKeyWord” directives must all be contiguous within a single “UseFacility” or “UseSeverity” block.

Custom Formatter DLL

The special "Formatter" directive, documented above, extends the range of messages that the log file monitor can read, to include binary data, or highly unstructured data. This option requires a custom DLL to be installed at the agent site. The DLL reads data from the log file and provides special formatting.

The formatting function, provided by the DLL, is completely arbitrary, and may consist of converting binary data to text, mapping field values to new values, joining lines, deduplicating lines, or other arbitrary processing that may be necessary to convert the log data into a syslog message. Applications of this DLL include Exchange monitoring, custom monitoring of SAP files, and monitoring of pseudo-binary log files. For more information, contact CorreLog support.

Special Log File Names

The "LogFile" specification permits the user to incorporate a Time Format specification into the file name. This allows CorreLog to monitor log files whose names change each day. CorreLog employs standard UNIX type time formatting of file names, where the following symbols have special significance in a file name:

%a	Abbreviated weekday name
%A	Full weekday name
%b	Abbreviated month name
%B	Full month name
%d	Day of month as decimal number (01 – 31)
%H	Hour in 24-hour format (00 – 23)
%I	Hour in 12-hour format (01 – 12)
%j	Day of year as decimal number (001 – 366)
%m	Month as decimal number (01 – 12)
%M	Minute as decimal number (00 – 59)
%U	Week of, with Sunday as first day (00 – 51)
%w	Weekday as decimal number (0 – 6; Sunday is 0)
%W	Week of year, with Monday as first day (00 – 51)
%y	Year without century, as decimal number (00 – 99)
%Y	Year with century, as decimal number
%z	Time-zone name or abbreviation.
%%	Percent sign

For example, consider the case where a log is created each night with the month and date, and placed in a folder each night with the name of the specified year. Such a file might be named: Z:\logs\2018\ex0620.log". The user can specify this file in the "LogFile" directive as "Z:\logs\%Y\ex%m%d.log", which will correctly resolve to the correct name without any further adjustments.

Log Name Wildcards

In addition to including a date and time specification, the user can also incorporate an asterisk "*" character as part of the root filename used in the "LogFile" directive. In this special case, a list of files matching the wildcard is gathered, and the file within that list, which was most recently updated on the system, is used as the operant file. This provides a mechanism for monitoring log files that do not follow an easy naming convention, such as files that have numeric prefix or suffix values.

For example, consider the case of a system of log files, where a monotonically increasing integer number is added as a suffix to each file as it is created. (This might occur if the size of each file was self-limiting, so that when a file reaches a

certain size it is closed and a new file is started.) In this case, the "LogFile" directive might be something such as "Z:\logs\Logfile_*.log", which would match the most recently updated log file in the "logs" target directory that begins with the prefix "Logfile" and ends with the suffix ".log".

Using Log Name Wildcards can be CPU intensive if the number of files in the match list is high. In particular, the wildcard should match a short list of files (such as only a dozen files or so) and the directory containing these files should not contain large numbers of files (such as only a hundred files or so.) Otherwise, performance of the system may be significantly degraded. Generally, the user will not experience any problem if a log name wildcard is used reasonably.

Note that "Wildcard" paths do not match multiple files, but match the most recently updated file. The "LogFile" type directive cannot be used to match multiple files. To match multiple files, the special "MultiLog" directive (described below) can be used.

Matching Multiple Log Files

It may be the case (in certain applications) where no part of the log file name is known, or an entire folder of log files needs to be monitored. The Windows Agent program permits this through the use of a special "MultiLog" directive, which is similar to the "LogFile" directive (described above) except that this single directive can monitor up to fifty different log files.

The "MultiLog" directive arguments are identical to the "LogFile" arguments, except that the wildcard (furnished with the "MultiLog" directive) can match up to 50 different log files. The various parameters of the operation (such as "UseFacility", "UseSeverity", "MaxSizeChange", etc.) apply to all matched files.

Only one "MultiLog" directive can be used per configuration file. Note that this directive can potentially reduce the performance of a system that contains many log files that are rapidly updated.

When using the "MultiLog" directive, a new directory listing of the target folder is generated when the agent is re-initialized, and subsequently every 30 seconds of operation. (If a file is added to the target folder, it will be monitored within 30 seconds.)

This special directive may not be supported by all of the "Remote Configuration" options of the CorreLog Server web interface, depending on the version of the CorreLog Server program being used. Contact program support for more information.

Other Notes On Log File Monitor Specifications

As discussed above, each log file has a “DefaultFacility”, and a “DefaultSeverity” value, followed by multiple optional “UseFacility” and “UseSeverity” statements. Each “UseFacility” and “UseSeverity” statement can have multiple “MatchKeyWord” statements. This provides a simple way to configure facilities and severities for any particular message.

Unlike the EventLog specifications discussed earlier, there is "auto" value available for the “DefaultSeverity” statement in the Log File Specification. This is because, unlike the Event Logs, there is no obvious severity assigned to arbitrary text strings in a file. The operator must define these severities.

One useful technique, to filter out data that is not important, is to make the “DefaultSeverity” for each log file “disabled”. The default severity is applied ONLY if no other severity specification is found. In this way, only those messages that have assigned severities will be sent as Syslog messages. This reduces the load on the Syslog server, especially if there are many hundreds of log file monitors. The administrator can specifically target a key set of messages using this technique.

The "LogStatChange" directive permits the user to monitor for the existence or modification of any file system object. This directive should not be used with any "UseFacility" or "UseSeverity" values, or any "MatchKeywords". The directive permits the operator to watch for changes to critical file system objects, such as password files, configuration files, or directories.

Finally, note that there is a special “MaxSizeChange” directive associated with log file monitoring. If the log file size jumps to a very large value, rather than sending out many Syslog messages, the program sends out a single “File Size Changed” Syslog message to indicate this condition. This handles file truncation situations, or the case where a file is copied some on top of the monitored file.

Section Summary And Additional Notes

1. The CorreLog Syslog Message Service configuration file resides in the same directory as the CO-sysmsg.exe executable, and is the CO-sysmsg.cnf file. By default, this file is located in the C:\CorreLog\wintools directory.

2. This file is read on Service Startup, and contains the name of the destination host, as well as other directives.

3. The file does not need to be modified, and comes ready-to-run. However, a user can tailor the file with match patterns that filter and set the severities and facilities associated with Event Log messages.

4. The file has an additional section (which does not require or interact with the event log monitors) that allows the user to specify up to 50 different streaming log files to be monitored. This works independent of the event log monitor, and is extremely useful for instrumenting arbitrary system log files, such as those associated with Oracle and Apache.

5. Log file paths (used by the log file monitor) can be dynamically derived to contain dates and times using standard time specifications incorporated as part of the pathname.

6. Log file paths (used by the log file monitor) can contain wildcards (either an asterisk (*) to match multiple characters, or a question mark (?) to match a single character.) In this case, the file that matches the wildcard and which has been most recently modified on the system is used as the operant file.

7. The agent can contain a marker function that permits a "heartbeat" type indication to be sent at periodic intervals. This heartbeat can be used to schedule periodic activities, but cannot be coordinated to a particular time of day such as midnight or noon.

8. The agent can send data to multiple IP Addresses using the "AuxAddress" directive. Zero to eight auxiliary addresses can be defined. These addresses are not authorized to remotely reconfigure the agent, and cannot be encrypted, but otherwise serve as destinations for all syslog messages generated by the program.

9. The "CO-sysmsg.log" file contains a transcript of actions (and errors) encountered during the execution of the agent. This file is useful for diagnosing system level problems. The file is created in the same folder as the CO-sysmsg.exe program.

10. The "LogLocal" directive within the CO-sysmsg.cnf configuration file can be used to locally log all messages sent in the CO-sysmsg.log file (described above). This directive also creates the "CO-sysmsg.dbg" file, useful for system level diagnosis.

The best way to learn about the configuration items is to experiment with the file, adding directives, and then possibly running the CO-sysmsg.exe program in foreground (using the “-foreground” option.) With this technique, a user can quickly target specific messages on the system. To check the configuration file without executing the CO-sysmsg.exe program, use the “CO-sysmsg –check” option.

Section 5: Remote Configuration

The behavior and operation of the CO-sysmsg.exe program is completely driven by its single configuration file, residing in the same directory as the program with a ".cnf" suffix. This configuration file does not necessarily have to be modified or adapted for the enterprise. However, depending on the organizational requirements, it may be necessary to make changes to this file in order to receive particular messages of interest.

The user can manually edit this file, and restart the CO-sysmsg.exe program (via the Windows "CorreLog Syslog Message Service" entry of the service manager.) This requires administrative access to the Windows platform that is hosting the CO-sysmsg.exe program, and is the most secure way of implementing this service.

As a special facility, configuration files can also be remotely downloaded and uploaded to effect changes in an automated way. This requires various permissions and adaptations described in this section. Specifically, remote configuration capability is limited by the value of the "ListenAuthMode" directive within CO-sysmsg.cnf file, which controls and limits remote request via the source address of the client, or via passkey, or both. The default "ListenAuthMode" setting is 3, which requires both a valid passkey, and also the client to be at the same IP address as the destination address.

Remote configuration capabilities of the CO-sysmsg.exe program permit a high degree of flexibility, security, and maintainability of this program. This section will be of interest to system installers, administrators, and operations personnel.

Authentication Of Remote Configuration Requests

The CO-sysmsg.exe program listens for remote configuration requests at TCP port 55514. This port number is included in the configuration file for the program, but cannot be easily changed. If this port is busy when the CO-sysmsg.exe program starts, or if the "ListenPort" directive is commented out of the file, then no remote configuration is possible and this particular capability of the CO-sysmsg.exe program is disabled.

Three different modes of operation are possible, as determined by the "ListenAuthMode" setting of the configuration file:

- **Auth Mode 0.** Setting the ListenAuthMode to a value of "0" disables authentication of requests. This value should probably never be used except in those very special circumstances where the CO-sysmsg.exe program is executing on a detached network where security is not a concern.

- **Auth Mode 1.** Setting the ListenAuthMode to a value of "1" causes authentication of remote configuration requests based upon the IP address of the requesting platform. If this auth mode is used, then any remote configuration request to the CO-sysmsg.exe program that originates on a platform other than the localhost "127.0.0.1" address, or the value of the DestinationAddress directive, is rejected. The requesting program must be at the location that receives the Syslog messages, or on the localhost.

- **Auth Mode 2.** Setting the ListenAuthMode to a value of "2" causes authentication of remote configuration requests solely based upon the configured passkey. The value of the "ListenPassKey" value must agree precisely with the value passed to the "rsmconf.exe" program (discussed below) or the value configured at the CorreLog Server platform on the "System > Parms" screen. Initially, both of these passkey values are set to the key string "Default", so no special configuration is required out-of-box.

- **Auth Mode 3.** Setting the ListenAuthMode to a value of "3" causes authentication of remote configuration requests to occur based both on the passkey (used in Auth Mode 2) and the source IP address (used in Auth Mode 1). This is the most secure way of managing the remote configuration process, and is the default out-of-box setting for the CO-sysmsg.exe program.

The values of "DestinationAddress", "DestinationPort", "ListenAuthMode", "ListenPassKey" and "ListenPort" cannot be change by the remote configuration process. Each of these values can be changed only by manually editing the CO-sysmsg.exe configuration file. Attempts to modify any of these values are silently

bypassed. This enhances security by ensuring that these values can be changed only by remotely logging into the host platform with an administrative login, editing the configuration file manually, and then restarting the CO-sysmsg.exe program.

PassKey Configuration

In some circumstances, the best (or only) type of authentication available will be with Auth Mode 2, which is "passkey" authentication. In particular, using a passkey as the sole authentication will be necessary on networks that are using NAT (Network Address Translation) or if the CorreLog server is multi-homed, or if tunneling software is being used. In these cases, the destination address for Syslog messages may not be the same as the location of remote configuration requests, making the use of Auth Mode 1 or Auth Mode 3 difficult or impossible.

The passkey is simply a text string of 40 characters or less. The value is case-sensitive, but can contain any printable characters, including spaces. The value is passed as an argument to the "rsmconf.exe" program (discussed below) and also is configured in the CorreLog server via the "System > Parms" tab. Because this value is "well-known", it is important to change this value across the enterprise when relying solely on passkey authentication.

In general, for extra security, the "passkey" should be used to supplement the source IP address authentication. There is no "downside" to using passkey authentication other than making firewall issues slightly more complex to troubleshoot.

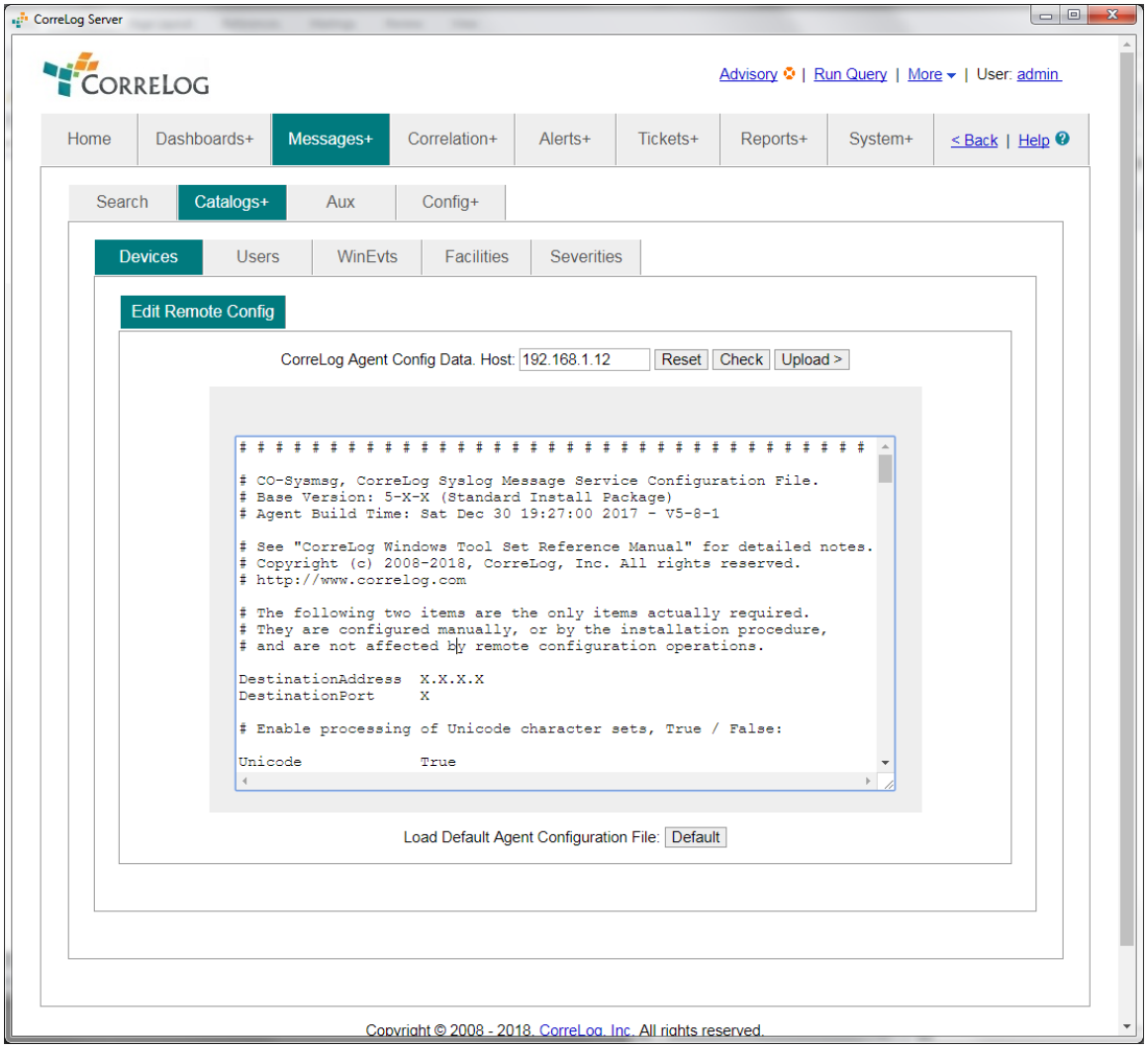
The passkey is not transmitted across the network in clear text. The value is encrypted, hence is secure from attack by network sniffers. However, the value is in clear text within the CO-sysmsg.cnf file, hence this file should be protected from unauthorized access (such as by limiting access to the host machine.)

Remote Config Via The CorreLog Web Interface

To execute a remote configuration operation at the CorreLog Web interface, the user may first need to enable remote configuration by accessing the "Device Information" screen for the host device. This is accomplished by clicking on the target device hyperlink (found in various locations within CorreLog, in particular in the "Messages > Device" tab.)

To enable remote configuration, on the CorreLog "Device Information" screen, the user clicks the "Edit Device Info" hyperlink, and sets the value of "Enable Remote Config Editor" to be "Yes", and commits the data. This causes a new "Edit Remote Config" link to appear on the Device Information screen whenever it is accessed. Clicking the "Edit Remote Config" hyperlink will download the remote configuration, permit editing, and uploading of this data.

When the user clicks on the "Edit Remote Config" hyperlink, the CorreLog "Remote Configuration Editor" screen downloads the remote configuration and displays an edit dialog that permits the user to modify and upload the data back to the CO-sysmsg.exe process. This Remote Configuration Editor screen is depicted below:



As shown above, the user makes changes to the remote configuration, can check the information, or upload the information back to the remotely executing CO-sysmsg.exe program.

Clicking the "Upload" button automatically initiates a check of the data. The data cannot be uploaded if any errors are encountered. The user can also manually check the data before uploading via the "Check" button.

Note that it is possible to download the CO-sysmsg.cnf file from one platform, and upload the file to another platform, affording a simple method of copying the

configuration of one CO-sysmsg.cnf to another. The user simply changes the target IP address to be that of the receiving platform before clicking the "Upload" button. The specified platform must be executing the CO-sysmsg.exe program, and must be configured to permit remote configuration.

Remote Configuration Via The Rsmconf.exe Utility

An alternative to remotely configuring the CO-sysmsg.exe program via the CorreLog web interface is to use the "rsmconf.exe" program, which is included both in the main CorreLog server (within the "system" directory), and also the CorreLog WTS software (within the "wintools" directory).

This utility permits the user to perform remote configuration at a command line, possibly within a batch file. The "rsmconf.exe" program accepts the following command line arguments:

rsmconf -download ipaddr filename

These command options download the remote configuration file from the specified IP address, using the specified passkey. The resulting configuration data is placed in the specified filename.

rsmconf -upload ipaddr filename

These command options upload the remote configuration file to the specified IP address, using the specified passkey. The data residing in the specified filename is uploaded.

rsmconf -info ipaddr

These command options get information on the specified agent, using the specified passkey. The agent information is displayed to standard output, and includes the agent version, up time, activity counters, and any licensing info that may be required for one or more agent adapters. (This option works only on agents of Version 5.6.3 or higher. Contact support for latest version info.)

rsmconf -check filename

These command options check the specified filename for errors. (This check is also always performed when using the "-upload" option.) This permits the user to check configuration data that has been previously downloaded. No passkey or IP address is specified or required.

Note that, if the ListenAuthMode of the CO-sysmsg.exe process is set to a value of 1 or 3, the rsmconf.exe program can only be executed on the platform specified by the DestinationAddress. If the ListenAuthMode is set to 2 or 3, then the passkey must be correctly specified, otherwise it is ignored.

The "rsmconf.exe" program is especially useful in performing batch configure operations, where the command is repeated multiple times within a Windows ".bat" file, needed to effect reconfiguration on many different platforms.

Other Security Features

Any attempt made to access the CO-sysmsg.exe program without proper authenticated credentials causes the CO-sysmsg.exe program to transmit a Syslog message to the destination address. The occurrence is also logged in the program error log (which resides in the same location as the executable, with a ".log" suffix.) The resulting Syslog message indicates the time of the error, and the client IP address. This can be used to monitor unauthorized access.

Additionally, successful remote reconfiguration is also logged, providing an audit trail of changes to the remote configuration data. These messages cannot be disabled.

Section Summary And Additional Notes

1. The remote configuration capability of the CO-sysmsg.exe program increases program maintainability by permitting administrators to access, modify, and upload configuration changes.

2. The CO-sysmsg.exe program authentications remote configuration requests by IP address, passkey or both. These values cannot be changed by the remote configuration process, but must be manually set in the configuration file.

3. The user can perform the remote configuration via the CorreLog web interface by first enabling remote configuration on the "Device Information" screen.

4. The user can execute remote configuration via the "rsmconf.exe program, which is a command line utility program that can download, upload, and check remote configuration data.

5. The remote configuration process provides a secure method of accessing and maintaining the remote configurations of multiple CO-sysmsg.exe programs on the network.

Section 6: CO-tsend Tunnel Sender

The CO-tsend.exe "CorreLog Tunnel Sender Service" is an optionally installed program, included with the WTS, which will relay messages to the CorreLog system server program using an encrypted, reliable TCP channel, including possible buffering of this data if a connection cannot be immediately established.

This type of program is often called a "Tunnel", since it provides a way to tunnel through firewalls and blocking routers in a highly secure manner. The program service is installed as part of the setup dialog, but must be manually enabled and configured, as described in this section. Using this function, a network administrator can limit the number of firewall holes to a single point-to-point connection between the CO-tsend.exe and the CorreLog program. The program also provides additional encryption, and reliable TCP connections.

In order to use the CO-tsend.exe program, the user must also configure the CO-trecv.exe "CorreLog Tunnel Receiver Service" program on the main CorreLog server platform. This receiver program accepts messages from the CO-tsend.exe program, decrypts and authenticates these messages, and relays these messages to the appropriate processes running at the central CorreLog Server. The configuration of both the CO-tsend.exe and CO-trecv.exe programs are detailed in this section.

Note that the information herein is NOT REQUIRED to install and use the CO-sysmsg.exe program, but is provided only to support more advanced applications and requirements, with special emphasis on security and reliability of message delivery.

Basic Installation, CorreLog Server Platform

To configure the CorreLog Tunnel Sender and Receiver programs, the user must have access to both the Windows client program and the central CorreLog server program. The user must perform a one-time installation and configuration of the "CO-trecv.exe" program at the main CorreLog platform as follows:

1. Log into main CorreLog server platform as an Administrator. (Note: This is the platform running the CorreLog HTTP server and CO-syslog.exe process, and not the Windows client program.)

2. At the main CorreLog server platform, edit the "system\CO-trecv.cnf" file using Windows "Notepad" or some other text editor. Provide a value for the "EncryptKey" setting. Remember the value for this encryption key, because it will also be entered at each platform. (The user can elect to use the "Default" encryption key, if no special encryption requirements exist.) The encryption key can be arbitrarily long, and consist of any alphanumeric characters, including spaces.

3. Log into the CorreLog server web interface, with an "admin" type logon.

4. Click on the "System > Schedule" tab of the web interface. This displays the "CorreLog Scheduler" screen, which describes controls the startup and periodic execution of CorreLog processes. Click "AddNew" to add a new entry.

5. On the "AddNew Entry" screen, select "start" as the Directive (selected via the pull down menu) and enter the precise name "CO-trecv.exe" as the scheduled command. Then click "Save" to add the entry.

6. Stop and restart the "CorreLog Framework Service" on the main platform. You may use the Windows Service Manager (in the Windows Control Panel "Administrative Tools") or the "Stop and Start Services" dialog in the Windows Start menu, or you can simply reboot the platform.

After completing the above steps, optionally verify that the "CO-trecv.exe" program has correctly started on the platform using the Windows task manager. Any errors during program startup will appear in the "system\CO-trecv.log" file.

The "EncryptKey" value may be arbitrarily long, and can include any alphanumeric characters. The key may also include spaces, but the spaces are automatically removed from the encryption key, and are not significant. No other changes to the "CO-trecv.cnf" file are needed.

Basic Installation, CorreLog Windows Client Platform

After installing and configuring the CorreLog Server platform, the following procedure can be used to configure the Windows client platform:

1. Login to the Windows Client program, where the WTS software is installed. (Note: This is the client platform, and not the platform that is running the CorreLog HTTP server.)

2. Edit the "wintools\CO-tsend.cnf" file using Windows "Notepad" or some other text editor. Edit the value for "DestinationAddress" to be the IP address of the CorreLog server platform (configured earlier.) Also, set the value for "EncryptKey" to be the same value used in the "CO-trecv.cnf" file, as described earlier.

3. Go to the Windows Service Manager (in Windows "Control Panel / Administrative Tools") and click on the "CorreLog Tunnel Sender Service" entry. Set the "Startup Type" to "automatic" and start the service.

The only two required settings in the "CO-tsend.cnf" file are the "EncryptKey" value, which must agree with the "CO-trecv.cnf" file, and the "DestinationAddress" value, which must be the IP address where the "CO-trecv.exe" program is executing.

After completing the above steps, verify that the "CO-tsend.exe" program has correctly started on the platform using the Windows task manager. Any errors during program startup will appear in the "wintools\CO-tsend.log" file.

Optionally, test the service using the "sendlog.exe" program. You can send a Syslog message to the platform where "CO-tsend.exe" is executing and the log message should appear correctly in the main CorreLog server as if it was sent to the server directly.

Manual, Alternate Client Installation Procedure

The CO-tsend.exe program is a stand-alone utility. Although it is installed and partially configured by the WTS installation dialog, it can also be copied to any Windows platform, to any folder (along with its configuration file) and installed using the command line options. The precise steps needed to manually install the CO-tsend.exe program as a service are as follows:

1. Copy the "CO-tsend.exe" and "CO-tsend.cnf" file to the target Windows platform, to any folder on that platform. Both files must appear in the same folder. No other files are required.

2. Edit the "CO-tsend.cnf" file, copied above, using Windows "Notepad" or some other text editor, as described in step #2 of the previous procedure. (Supply values for the "DestinationAddress" and "EncryptKey" parameters, as described above.)

3. Install the CorreLog Tunnel Sender service by issuing the following commands at a command prompt:

   CO-tsend –install
   CO-tsend –mode auto
   CO-tsend –start

No other steps are needed to configure the CO-tsend.exe program. The above commands, entered at a Windows command prompt, will install the service, set the startup type to be automatic, and start the service. The "CorreLog Tunnel Sender Service" entry will appear in the Windows Service Manager screens, and the "CO-tsend.exe" program will appear in the Windows task manager.

Note that the CO-tsend.exe program provides command line arguments that are the same as the CO-sysmsg.exe program. The user can install, remove, start, stop, and check the service via arguments. Type "CO-tsend.exe –help" for brief help on program usage.

Any errors during program startup or execution will appear in the " CO-tsend.log" file, in the same directory as the "CO-tsend.exe" and "CO-tsend.cnf" file. You may test the service's operation by sending a Syslog message to the platform where the "CO-tsend.exe" is executing. The log message should appear correctly in the main CorreLog server as if it was sent to the CorreLog server directly.

Firewall Configuration

A main application of the CO-tsend.exe program is to permit tunneling through firewalls. In this case, the administrator configures a single copy of the CO-

tsend.exe program on a Windows platform, and then directs all SNMP traps and Syslog messages to that platform. The administrator can then open up a single port in the firewall to permit traffic between the CO-tsend.exe and CO-trecv.exe program.

By default, the CO-tsend.exe and CO-trecv.exe programs communicate across the TCP port 51462. This can be changed in the configuration files of the programs. (Both programs must have the same port number specified in their configuration files.) Generally, there is not much need to specify a different port number, since this number is quite obscure. However, if the port number is changed, the configuration files of the CO-trecv.exe and CO-tsend.exe programs must both be changed to permit proper communications.

Note that the CO-trecv.exe program, running at the CorreLog server platform, can service any number of CO-tsend.exe programs. Generally, there will be only one copy of the CO-trecv.exe program executing anywhere on the network. However, while it is quite common to have a single CO-tsend.exe program executing, there can be many such programs. For example, each Windows client may have its own copy of the CO-tsend.exe program running, each directing encrypted messages to the CO-trecv.exe program on the CorreLog server.

The CO-tsend.cnf Configuration File

The CO-tsend.cnf file resides in the same directory as the CO-tsend.exe program, and provides the following directives.

DestinationAddress

This is the address where the CO-trecv.exe program is listening for messages, and will always be the location of the CorreLog server IP address. The value is configured by the WTS installation dialog, and can be changed thereafter.

DestinationPort

This is the standard TCP port number of 51462, which must agree with the "ListenPort" directive of the CO-trecv.cnf file (discussed later). For the CO-tsend.exe program to communicate with the CO-trecv.exe program, this port number must be the same for both programs, and any intermediate firewall or filtering router must have this TCP port number open. There is generally no need to change this value.

TrapListenPort

This is the UDP port number where the CO-tsend.exe program listens for SNMP traps, by default the standard port number of UDP 162. No other program can be listening to this port while the CO-tsend.exe program executes, and this port must be free when the CO-tsend.exe program starts. If any conflict exists, a message is logged in the "CO-tsend.log" file,

found in the same directory as the "CO-tsend.exe" program. There is generally no need to change this value.

SyslogListenPort

This is the UDP port number where the CO-tsend.exe program listens for Syslog messages, by default the standard port number of UDP 514. No other program can be listening to this port while the CO-tsend.exe program executes, and this port must be free when the CO-tsend.exe program starts. If any conflict exists, a message is logged in the "CO-tsend.log" file, found in the same directory as the "CO-tsend.exe" program. There is generally no need to change this value.

MatchAddress

This directive is an IP address or wildcard in the form "*.*.*.*", which can limit the devices that can send messages to the CO-tsend.exe program. The user can leave the default value (to match all IP addresses) or specify a subnet such as "10.30.*.*" or "10.15.8.*", or can specify a precise IP address. This provides extra security by limiting the range of devices that can access the TCP tunnel.

ErrorSeverity

This directive indicates the severity of any Syslog message generated by the program. If the CO-tsend.exe program encounters an error, it attempts to send an unencrypted UDP message to the CorreLog server (running at the "DestinationPort" IP address.) The value can be set to "disabled" to prevent any Syslog messages. All errors are also logged in the "CO-tsend.log" file, residing the in the same directory as the ""CO-tsend.exe" program.

LogLocal

This value is set to either “True” or “False”. If the value is “True”, then all messages sent by the CO-tsend.exe program are also logged in the CO-tsend.log file (along with any error messages encountered by the program.) This provides a simple way to verify whether UDP messages are being dropped. Note that the CO-tsend.log file is restarted each time the service is started; hence the file does not grow without bounds. If this directive is omitted, it is interpreted to be “False”.

EncryptKey

This value is the encryption key used to encrypt message. It can contain any alphanumeric characters, including spaces (but all spaces are ignored.) The value must agree with the "EncryptKey" value specified in the "CO-trecv.exe" program. The value of "Default" is the initial key, configured in both "CO-tsend.exe" and "CO-trecv.exe". If the value is modified in one location, it must be modified in all locations.

BufferData

This optional value can be used to enable data buffering. If set to true, then the CO-tsend.exe program will buffer up to 10,000 lines of data should the connection between the program and CorreLog become unavailable. CorreLog will try each "BufferDelaySecs" seconds (by default one second) to resend the data. This value must exist, and must be set to "True" to enable data buffering. (See additional notes below.)

BufferDelaySecs

This optional value is valid only if "BufferData" is set to "True", and can be used to control the delay before any data is recent. The default value is 1 second indicating that the queued messages will retry to send the data each second. Careful selection of this value can improve performance if the connection is known to be intermittent for a period of time, such as 300 seconds.

MessagePrefix

This is an optional prefix that is applied to all syslog messages sent by the tunnel sender. If the directive is omitted, no messages are prefixed. The value can contain environmental variables, or can be static text (such as text reflecting the identity of the remote site.)

The CO-trecv.cnf Configuration File

The "CO-trecv.cnf" file resides in the same directory as the "CO-trecv.exe" program, and provides the following directives.

ListenPort

This is the standard TCP port number of 51462, which must agree with the "DestinationPort" directive of the CO-tsend.cnf file (discussed earlier). For the CO-tsend.exe program to communicate with the CO-trecv.exe program, this port number must be the same for both programs, and any intermediate firewall or filtering router must have this TCP port number open. There is generally no need to change this value.

TrapDestPort

This is the UDP port number that the "CO-trecv.exe" program relays SNMP traps to on the local platform. By default, this is the standard port number of UDP 162. The value must agree with the "ListenPort" value of the "CO-systrap.cnf" file in the system directory of the CorreLog server. There is generally no need to change this value.

SyslDestPort

This is the UDP port number that the "CO-trecv.exe" program relays Syslog messages to on the local platform. By default, this is the standard port number of UDP 514. The value must agree with the

"udp_port_number" value of the "syslog.cnf" file in the system directory of the CorreLog server. There is generally no need to change this value.

MatchAddress

This directive is an IP address or wildcard in the form "*.*.*.*", which can limit the location of CO-tsend.exe programs which are send messages to the CO-trecv.exe program. The user can leave the default value (to match all IP addresses) or specify a subnet such as "10.30.*.*" or "10.15.8.*", or can specify a precise IP address. This provides extra security by limiting the range of devices that can access the TCP tunnel. Note that this is the IP address of the CO-tsend.exe program, and not the IP address of the devices actually sending messages.

ErrorSeverity

This directive indicates the severity of any authentication error generated by the program, due to a fault encryption key, or failure of a program to match the "MatchAddress" directive. The value can be set to "disabled" to prevent any Syslog messages from being generated by the "CO-trecv.exe" program, however authentication errors are extremely useful for detecting security breaches, and this value should normally be set to "Error" or above. The authentication message, generated by the "CO-trecv.exe" program will report the IP address of the "CO-tsend.exe" program, or other program attempting to send a message to the "CO-trecv.exe" program.

LogLocal

This value is set to either “True” or “False”. If the value is “True”, then all messages sent by the CO-trecv.exe program are also logged in the CO-tsend.log file (along with any error messages encountered by the program.) This provides a simple way to verify whether UDP messages are being dropped. Note that the CO-trecv.log file is restarted each time the service is started; hence the file does not grow without bounds. If this directive is omitted, it is interpreted to be “False”.

EncryptKey

This value is the encryption key used to encrypt message. It can contain any alphanumeric characters, including spaces (but all spaces are ignored.) The value must agree with the "EncryptKey" value specified in the "CO-tsend.exe" program. The value of "Default" is the initial key, configured in both "CO-tsend.exe" and "CO-trecv.exe". If the value is modified in one location, it must be modified in all locations.

Note that the "CO-trecv.cnf" file and "CO-trecv.exe" program are not found within the WTS package. Rather, these two files exist as standard part of the CorreLog server installation, within the "CorreLog\system" folder, at the main server platform. (The information on how to install and configure this program was

discussed in a previous section of this manual, and is not documented within the "CorreLog User Reference Manual".)

Buffering Data Functions

The CO-tsend.exe program provides the special function of buffering data, should the connection between the program and CorreLog become unavailable. This is useful if CorreLog or an intermediate router or switch is rebooted, or if a connection is otherwise unavailable. In this case, the CO-tsend.exe program will temporarily store up to 10,000 messages, and resend these messages when the connection is re-established.

Normally, the "BufferData" configuration value is set to "False". (On some versions of CorreLog, this directive may not appear in the default configuration file, in which case buffering is not enabled.) To enable buffering, the administrator must set this value to "True".

Buffering the data can interfere with any alerts configured in CorreLog, since buffering is not a "real-time" operation. When the connection between the CO-tsend.exe program and CorreLog is re-established, all messages that have been buffered are resent, and each message includes a flag that indicates the amount of time the message was delayed. This flag can be used to exclude retransmitted data from alerts, threads, and triggers. The CO-tsend.exe program additionally sends a message indicating the amount of time that the connection was unavailable, and the amount of messages lost (if the buffer size is exceeded.)

Finally, note that the "BufferDelaySecs" (if it exists) can control when the program attempts to retransmit data. Normally, the retransmission occurs approximately once each second. The value can be increased to marginally improve performance, especially if the intermittency is known to be a certain duration, such as 300 seconds or so.

Troubleshooting Information

The most common reasons for failure are as follows:

1. A firewall, port blocker, or filtering router does not permit communication between the CO-tsend.exe and CO-trecv.exe programs via the configured port number; by default the TCP port 51462. The user should check the firewall configuration, or use the "telnet" command to verify that the CO-trecv.exe program is reachable from the CO-tsend.exe program. The user should also verify that no port blocking software (such as found in many common Virus protection programs) is preventing communication between the two programs.

2. The CO-tsend.exe and CO-trecv.exe programs are not executing. The user should verify via the Windows Task Manager that these programs are running. If these programs fail to start, the user should check the respective error logs, either the "CO-tsend.log" or "CO-trecv.log" files (located in the same folder as the executable programs.)

3. The CO-tsend.exe program does not have the "DestinationAddress" correctly configured. This may be difficult to detect. Double-check the IP address by pinging the DestinationAddress node, or use the "telnet" command to access the "CO-trecv.exe" program.

4. The CO-tsend.exe program and CO-trecv.exe program do not have the same EncryptKey value. If the "CO-tsend.exe" program sends a message with the wrong encryption key, this will cause an Authentication error message to be logged at the main CorreLog console. Check the CorreLog messages screen to see if such authentication errors are occurring.

As a final way of verifying operation, set the "LogLocal" directives of both the "CO-tsend.cnf" and 'CO-trecv.cnf" files to be "True", and try issuing Syslog messages via the "sendlog.exe" utility. The user can check the log files of the programs to see if the message reaches the CO-tsend.exe program, the CO-trecv.exe program, or neither.

Section Summary And Additional Notes

1. The CO-tsend.exe program can be configured at any Windows platform to create an encrypted TCP tunnel between that platform and the CorreLog server. The CO-trecv.exe program runs at the CorreLog server, and must be enabled in order to receive messages from the CO-tsend.exe program.

2. Both the CO-tsend.exe and CO-trecv.exe programs make use of a configuration file, with the same name as the program, except with a ".cnf" (rather than an ".exe") suffix. These configuration files are located in the same folder as the program.

3. The default TCP port number, used for communication between the CO-tsend.exe and CO-trecv.exe programs, is TCP port 51462. This value can be changed. However, if the value is changed in one configuration file, it must be changed in all configuration files.

4. Exactly one copy of CO-trecv.exe should normally be installed, to accept messages from one or more CO-tsend.exe programs. The user can install one CO-tsend.exe program, and direct all traffic to that program, or can install multiple copies of the CO-tsend.exe program (such as one copy on each Windows platform.)

Section 7: Automatic Deployment

Manual installation of the CorreLog WTS can be done quickly within an enterprise, and provides a highly controlled rollout of the agent software. Typically, a manual installation of the software requires only a few minutes per target platform, including the time to log into the target platform, acquire the software from the CorreLog home page, and run the installation procedure.

Even though manual installation is fast and easy, some enterprises may find it desirable to perform an automated deployment of the CorreLog WTS across hundreds (or even thousands) of different system. This section provides procedures and methods to assist with automatically deploying the CorreLog WTS to multiple Windows platforms. The CorreLog Windows Agent can be installed using a silent and / or unintended installation procedure, using standard using deployment software such as Microsoft SMS.

The CorreLog WTS includes the "wt-silent.exe" program in the "s-doc" of the CorreLog distribution, which permits the system to be installed via a command line option in a fashion identical to the standard agent installer. The details of this program are discussed in this section

Automating the deployment of any software system incorporates certain risks. Before executing an automated deployment, the automated installation procedure should be carefully tested on a variety of different platforms to insure proper operation. While the Windows WTS does not pose any special hazards beyond that of normal software installations, experience shows that certain common problems (such as firewall and networking issues) should be considered as part of the software installation process.

Silent Agent Installation Program

The "wt-silent.exe" program is a standard feature of current CorreLog versions, residing in the "s-doc" directory of the CorreLog Server. The program allows the agent program to be installed at a command line without a GUI or console. The program accepts a single argument, which is the destination IP address to send messages to. Otherwise, the program operates in a fashion identical to the standard installer.

The "wt-silent.exe" program will install software in the folder where the package is executed (which may be different than the location of the "wt-silent.exe" package.) The package creates a subfolder called "wintools" which contains all the files necessary to run the agent.

Note that the location of the installed files will be the current working directory where the program is executed AND NOT the working directory or location of the wt-silent.exe package.

For example, consider the case where the "wt-silent.exe" program is in the user's path. The agent is to be installed in the "C:\Program Files (x86)\wintools" folder, and is to send messages to the CorreLog Server running at IP address "192.168.1.1"

In that case, the following commands can be executed at a command prompt:

```
cd "C:\program Files (x86)"
Wt-silent.exe 192.168.1.1
```

The above command creates the "wintools" folder within the "C:\program files (x86)" folder, then installs files in the "wintools" folder, installs the "CorreLog Syslog Message Service" service, and starts the service. This usually takes about 10 to 20 seconds to complete.

The "wintools" folder location is built into the "wt-silent.exe" program. This value cannot be changed by the user. If you have special considerations in this area, you can request this change from program support.

Note that "wt-silent.exe" can be executed with no command line arguments to display version and help information, including information on the location where the agent will be installed.

Silent Agent Installation Using PSEXEC

The "wt-silent.exe" program works with most remote software installation programs, including the "Sys Internals" program "psexec.exe". For example, to remotely install the program using the "psexec.exe" program, the operator can use a batch file such as the following:

```
REM: # Remote installation script for agent programs
REM: # Requires the "psexec.exe" Sys Internals command.
REM: # See: http://technet.microsoft.com/en-us/sysinternals/bb897553

REM: # Parameters are as follows:
REM: # D - The destination address of the CorreLog server.
REM: # R - The remote directory path to receive the install.
REM: # U - The administrative username for the install.
REM: # P - The password for the administrative username.
REM: # T - The target node to perform the remote installation on.
REM: # Modify parameters appropriately below.

SET D=127.0.0.1
SET R=C:\CorreLog
SET T=127.0.0.1
SET U=Administrator
SET P=clear-text-password

REM: # Perform the installation:
REM: # First, create the remote directory, then install the package.

psexec -h \\%T% -u %U% -p %P% mkdir %R%
psexec -h \\%T% -u %U% -p %P% -c -f -w %R% wt-silent.exe %D%
```

The above program can be cut and pasted to a file in the "s-doc" directory, to effect a remote installation of the "wt-silent.exe" package. Note that the "psexec.exe" program, used by the batch file, is not a part of the CorreLog installation, but a standard utility distributed by Microsoft, and should be in the user's path. Contact CorreLog support for details on how to acquire this utility.

Standard CorreLog "Wt-agent.msi" Package

The CorreLog Server site contains a "wt-agent.msi" package in the "s-doc" directory of the installation, suitable for use with remote agent deployment using a variety of deployment software, including Microsoft SCOM and SMS. This package contains the latest agent for the site, and directly supports remote installation.

The "wt-agent.msi" package is interoperable and standards-based, but usage of the program requires the end-user software deployment system to accept arguments to the MSI file. (If this restriction prevents the usage of the MSI file,

given end user resources, the end-user should consider creating their own site-specific MSI package as discussed in the next session.)

Specifically, when using the "wt-agent.msi" file, the destination address for syslog messages must be passed to the MSI installer, such as follows:

```
Msiexec /i wt-agent.msi dest=10.1.2.3
```

The above command installs the wt-agent.msi distribution, pointing to the destination address of 10.1.2.3. The actual IP address specified above is placed in the "CO-sysmsg.cnf" file on installation.

When installing the MSI package, the actual agent files are placed in the appropriate program files directory, usually the "C:\Program Files (x86)\CorreLog" directory of the managed platform (but possible some other location depending on the type of machine hardware and environment.) After performing an MSI installation, the software deployment tool should manage the uninstall (if any) such as with the following command:

```
Msiexec /x wt-agent.msi
```

Note that the MSI package is particularly lightweight, contains only core and essential files, and does not include the documentation found in the "wt-silent.exe" program, or any extra utilities found in the "wt-silent.exe" package other than the "sendlog.exe" program.

Creating A Custom MSI Package

If the user wishes to create a custom MSI package, rather than using the standard "wt-agent.msi" package documented above, the user can create an MSI package from the CorreLog WTS distribution files in a variety of ways using several public domain programs. This permits the CorreLog WTS to be remotely deployed using Microsoft SMS or some third party Windows software deployment system, including repackaging of the files and configuration data.

CorreLog professional services can furnish the CO-WiX program, which is one method of creating custom packages. Parties interested in creating their own MSI packages should contact program support for further information on this topic.

Additional Notes

1. The "wt-silent.exe" program performs a silent installation of the Windows agent, and can be used in remote software deployment.

2. The CSetup.exe program can be used to perform a command line installation of the Windows agent, and provides extra capabilities, such as in creation of self-extracting WinZip files and MSI files. In addition to using an MSI file, a self-extracting WinZip program, or other installation software can execute the CSetup.exe program, possibly within a CMD prompt or batch file.

3. Caution may be necessary to insure that the CO-sysmsg.exe program is not already executing when the installation takes place, because the CO-sysmsg.exe program cannot be overwritten by new installation software if it is in use by the operating system. (The "wt-silent.exe" program will attempt to uninstall any existing version of the program before installing the new one.)

4. The installation program will uninstall the WTS from the service manager, and then re-install itself, thereby permitting the user to change the location of the Windows Tool Set software to another pathname. This greatly simplifies the relocation of the WTS software, if that is needed.

Section 8: Trouble Shooting & FAQs

This section finishes this manual by discussing common problems encountered by CorreLog WTS administrators and developers, and how to troubleshoot and solve these problems.

The CorreLog WTS is designed to be flexible and open, and cover a variety of different application areas. It is common to experience problems, especially when getting started. The range of the typical problems experienced is quite small, and can largely be addressed by the notes in this section.

Prior to opening a trouble ticket, users may wish to review the list of issues in this section. In addition to providing assistance with solving problems, this section can be used to achieve greater insight into the workings of the CorreLog WTS. After reviewing this section, if you still have questions or issues, please contact customer support, or visit our website, or call us directly.

See also the CorreLog Framework User Reference, Section 8, which has a similar list of frequently asked questions and troubleshooting techniques, useful when working with the CorreLog and framework components.

Thanks for your interest in the CorreLog Framework. We are always interested in comments, suggestions, and construction criticism, as well as discussing your particular applications!

I am not any receiving Syslog messages

The most likely reason is that a firewall is preventing the transmission of the message to the destination host. You need to open a hole in the firewall to permit UDP port 514 to be accessed at the remote machine. This is the most common problem with the CorreLog WTS, since this port number is frequently shut off by many firewalls.

How do I verify that the CO-sysmsg.exe program is actually running?

You can run the CO-sysmsg.exe program at a command prompt using the "-foreground" option to the program. You can also check the CO-sysmsg.log file to see what errors have been encountered during normal execution. You can enable the logging of all Syslog messages to this file by setting the "LogLocal" directive to "True" in the CO-sysmg.cnf file. These techniques are all useful for assessing what messages are being sent and how the CO-sysmsg.exe program is running.

I want to use the sendlog.exe program, but not the CO-sysmsg.exe program.

The sendlog.exe program is a completely stand-alone utility. It does not require any special DLL files, and does not require the execution of the CO-sysmsg.exe program. Simply copy the sendlog.exe program to any location and it will run just fine. This is very handy for supporting batch files and scripting applications. (The program can be easily launched via "Perl", "PHP", and many other scripting languages.

My CO-sysmsg.exe program appears to ignore the Security or other Windows Event Logs

Assuming that you do not have the configuration file completely filtering all messages, this situation can occur if the user makes the event log too small, and the event log is rapidly being updated. For example, if the user is logging many security events, and the security log file is set to the minimum size, it is possible that an event message is actually being dropped before the CO-sysmsg.exe program can see it. Increase the size of the log files to 1024 Kbytes, which should be more than sufficient. (Use the Control Panel > Admin Tools > Event Viewer > Properties dialog.)

My CO-sysmsg.exe program appears to be ignoring certain events.

Make sure that the Event Viewer program is not filtering the events. (Use the Control Panel > Admin Tools > Event Viewer > Properties dialog, and click on the “Filters” tab.) The Microsoft Event Viewer can selectively shut off certain events at the source, so that they are never seen by the CO-sysmsg.exe program, and never relayed to the Syslog server.

Can I reduce the Microsoft Event Log sizes to their minimum size?

No. The system requires ample buffer space, in the form of non-trivial event log sizes. Some users may think that, because the CO-sysmsg.exe program is in place, the event logs can be made very small (because the CO-sysmsg.exe program will relay events to the Syslog server host.) In practice, this is partially true. However the system requires around 1024 Kbytes as the maximum size of any event log, or more.

How can I decrypt an encrypted Syslog message?

You can’t. The CorreLog encryption routine uses a secret key available only to the CorreLog Sigma Web Framework. The encryption is a one-use pad type encryption with a time-based rotating cipher. Contact the vendor for enhanced encryption techniques, including public key encryption schemes.

I enabled encryption in the configuration file, but I see no change.

The CorreLog Server is decrypting the messages correctly, and the user will not see any change. However, if the destination for these Syslog messages is some other program (such as a Unix “syslogd” program) it will be readily apparent by inspecting the Syslog that the message content is encrypted.

How do I run multiple copies of CO-sysmsg.exe?

Do not install the CO-sysmg service. Instead, use the Use the CO-svc.exe program, which is a regular component of the CorreLog Web Framework. Install this service, configure the “sched.cnf” file to launch the CO-sysmsg –foreground program, and install the CO-sysmsg program in several different directories, one directory for each destination. This provides the added benefit of providing a separate configuration file for each destination, which is flexibility that will probably be needed in any management scenario.

How do I interface to the Windows Performance Monitor?

The easiest way to send Syslog messages to monitor performance is to use the Windows performance monitor to send alerts. Although it is not commonly known, the Windows Performance Monitor contains a very sophisticated alerting facility, which permits event log messages (of arbitrary text content) to be generated whenever a performance counter is at a limit. The event log messages are subsequently converted into to Syslog messages, which arrive at the Syslog receiver. As an alternative, the user can actually launch external programs when Windows Performance Alerts occur. In this case, the user can configure the Performance Monitor to execute the “sendlog” program, passing as the Syslog message the counter name, counter values, and other items of interest. This is all available via the “Admin Tools > Performance > Performance Logs And Alerts” screen, which is a standard part of the Windows system.

How do I detect when a user logs on or off the system?

This usually requires no modifications to the system. However, it may be that the Windows Audit policies (in the Local Security Settings screen of the Windows Control Panel > Administrative Tools screen) have not been enabled. Launch this standard windows tool, drill down into the Security Settings > Local Policy > Audit Policy settings and make sure that the “Audit Logon Events” settings are set to send event messages on both Success and Failure type events.

I am generating a lot of messages that say “No data available”. What does that mean?

This message is actually a briefer message that is logged in the event log whenever the registry does not contain an event log key. This is probably the result of a third-parties sloppy uninstaller program, or may indicate a problem with the registry. You can filter out all of these messages via the CO-sysmsg.cnf file, or you can investigate what software components are missing from the system. This is a common problem of older machines, or machines that have migrated from older operating system versions to newer versions (such as Vista.) These keys can be inspected via the “regedit” program, by looking at the “SYSTEM\CurrentControlSet\Services\Eventlog” portion of the Windows registry.

I am receiving messages that begin “data://” followed by garbage.

This is due to the encryption mechanism of the CO-sysmsg.exe and “sendlog” programs. You are receiving this data at some server OTHER than the CorreLog Server. The data encryption works only between the CorreLog WTS and

CorreLog server. Turn off the data encryption in the CO-sysmsg.exe program via the “EncryptData” directive in the CO-sysmsg.cnf file. To turn off the encryption associated with the “sendlog.exe” program you need to unset the SIGMA_ENCRYPT_DATA environmental variable. This will take care of the problem.

The CorreLog server does not receive the Syslog message until several minutes after the message was generated.

The CO-sysmsg.exe program parses events in the order that they were received. The program contains specific logic to prevent messages from being sent faster than the “MsgDelayMsecs” directive. The minimum value for this is 100 msecs, regardless of what minimum value the user has configured. Therefore, the most messages that can be sent by the CO-sysmsg.exe program are 10 messages per second. You are probably in a situation where a flood of messages is being entered into the Windows event log, and the CO-sysmsg.exe program is pacing itself to report all of these messages. Note that this many event messages are anomalous. Fix the problem at the root cause by reducing the number of event messages being generated.

The CorreLog tunneling software is not working.

There are various configuration issues that can occur with the CorreLog tunneling utilities, ranging from firewall issues, to mismatched encryption keys. Check the "Troubleshooting" information, found in this manual. In particular, you can set the "LogLocal" configuration directive in the CO-tsend.cnf and CO-trecv.cnf file to "True" and restart these two processes. This will log all the connections and transfers of these two programs from the CO-tsend.exe to the CO-trecv.exe programs, so you can easily find the communication failure point by examining the CO-tsend.log and CO-trecv.log files.

Appendix A: The CO-sysmsg.cnf File

This appendix provides an example of the CO-sysmsg.cnf file, which is the central configuration file used by the CorreLog Syslog Message Service. An administrator or system developer can edit this file to specify the facility and severity codes used by the Event Log monitor. The file also allows users to monitor arbitrary streaming log files on the system (that is, any file which is continuously appended, such as Oracle error logs, HTTP server logs, and many other types of log files.)

The CO-sysmsg.cnf file is documented in detail within Section 4 of this manual. As stated in that section, the configuration file does not necessarily EVER have to be modified by a user. The default configuration, created by the installation utility, is adequate for many (perhaps most) environments. However, if the user wishes to create a highly customized installation, targeting specific types of event log messages, that capability readily exists through the directives in the file.

This file resides in the same directory as the CO-sysmsg.exe program (which corresponds to the CorreLog Syslog Message Windows Service.) The file provided here is the default configuration that comes with the system.

```
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# CO-Sysmsg, CorreLog Syslog Message Service Configuration File.

# See "CorreLog Windows Tool Set Reference Manual" for detailed notes.
# Copyright (c) 2008 - 2018, CorreLog, Inc. All rights reserved.
# http://www.correlog.com

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The following two items are the only items actually required.
# They are configured manually, or by the installation procedure,
# and are not affected by remote configuration operations.

DestinationAddress  127.0.0.1
DestinationPort     514

# Zero to eight alternate IP addresses can be listed below:

AuxAddress          -1

# Parameters used for remote configuration of this process via the
# CorreLog web interface. The user can comment these values out to
# disable remote configuration. The "ListenAuthMode" can take values
# 0=No Auth, 1=Source Address, 2=PassKey, 3=Address and Key. These
# values cannot be changed via remote configuration.

ListenAuthMode      3
ListenPassKey       Default
ListenPort          55514

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# These are optional. Consult the Win32 agent user's manual for info.

# MessagePrefix     CorreLog
# MsgDelayMsecs     10
# MaxMessageSize    500
# LogLocal          True
# EncryptData       True
# MarkerMessage     Mark - Host: %COMPUTERNAME% alive.
# MarkerMinutes     20
# Deduplicate       3

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The next section provides an optional list of event logs, the default
# facility for eventlog messages, and optional special keywords that
# change the default facility and severity of messages. Each event
# log has a separate specification.

# Including the three standard Event Logs, a maximum of 10 different
# Event Logs can be specified, each with a virtually unlimited number
# of UseFacility and UseSeverity combinations.

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
```

```
# The following parameters apply to the Windows "Application" log.

Eventlog            Application
DefaultFacility     user
DefaultSeverity     auto

# User can configure other "Application" severities and facilities
here.

# UseFacility       local7
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseFacility       local8
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       debug
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       critical
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The following parameters apply to the Windows "System" log.

Eventlog            System
DefaultFacility     system
DefaultSeverity     auto

# User can configure other "System" severities and facilities here.

# UseFacility       local7
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseFacility       local8
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       debug
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       critical
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The following parameters apply to the Windows "Security" log.

# This particular event log can be quite busy, so the default severity
# is set here to "disabled", which requires particular matches to be
```

```
# explicitly listed. The following are typical, and may be sufficient
# for many security applications.

Eventlog            Security
DefaultFacility     auth
DefaultSeverity     disabled

UseSeverity         info
MatchKeyWord        bad password
MatchKeyWord        successful logon: * type: 2

UseSeverity         notice
MatchKeyWord        change password attempt
MatchKeyWord        account deleted
MatchKeyWord        account disabled

UseSeverity         warning
MatchKeyWord        policy change
MatchKeyWord        account created
MatchKeyWord        account enabled

UseSeverity         error
MatchKeyWord        account locked out

# User can configure other "Security" severities and facilities
# here.

# UseFacility       local7
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseFacility       local8
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       debug
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       critical
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The next section provides a list of filenames, match keywords and
# the facility and severity of the resulting syslog message.

# NOTE: After starting the "LogFile" section, no further "EventLog"
# directives should appear in the file.

# LogFile           /program files/file1.log
# LogName           FILE1:
# Encoding          Default
# MaxSizeChange     10000
# DefaultFacility   user
# DefaultSeverity   disabled
```

```
# UseFacility       local1
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseFacility       local2
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       debug
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       critical
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# The above list can be repeated multiple times, to specify various
# streamling log files, match patterns, and severities. A maximum
# of 10 different log files can be specified, each with a virtually
# unlimited number of UseFacility and UseSeverity combinations.

# LogFile           /program files/file2.log
# LogName           FILE1:
# Encoding          Default
# MaxSizeChange     10000
# DefaultFacility   user
# DefaultSeverity   disabled

# UseFacility       local1
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseFacility       local2
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       debug
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# UseSeverity       critical
# MatchKeyWord      keyword
# MatchKeyWord      keyword

# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #

# END OF FILE
```

Appendix B: Facilities And Severities

This section provides a reference guide explaining the Syslog Facility and Severity codes used with the “sendlog.exe” program, and within the CO-sysmsg.cnf configuration file. Information on the Syslog protocol, as well as more information on the various facilities and severities, can be found in the CorreLog Server Users manual. The information provided in this appendix is briefer, suitable as a quick reference.

Syslog Facility Codes And Their Meaning

The basic facilities, defined by RFC 3164, are discussed below.

Kernel	0	These are messages related to the Unix kernel process, or generated by very low-level driver software and system programs.
User	1	These are typically user-defined messages. This facility is used (and over-used) as a central way of defining messages that have not been otherwise classified.
Mail	2	These are messages related to the SMTP system, Microsoft Exchange, as well as mail relay systems, and sometimes e-mail programs.
System	3	This is another catchall type of facility that is often over-used, but generally related to system services, Unix daemons not otherwise classified. It can also indicate
Security	4	These are messages related to security processing, such as login detection, virus protection, and intrusion detection systems. Other security related messages are found in the “Audit(13)” and “Alert(14)” facilities.
Internal	5	Originally, these were messages related only to the internal operations of the Syslog protocol process, but have evolved to include general internal processes, often related to performance monitoring, but occasionally simply the internal workings of the system.
Printer	6	These messages are related to the Unix “lpd” process, and can also indicate problems with printer hardware, printer queues, and other types of queues that are not particularly related to printers.
News	7	These are messages related to the Network News processes, which have been fairly deprecated, although are still found on many Unix and mainframe systems. This facility is sometimes used to indicate low severity news events, such as a system being brought down.
Network	8	These messages are related to the Unix “uucpd” process. (UUCP is an acronym for “Unix to Unix Copy”.) They may also refer to network events, such as interfaces being enabled or disabled.

Lock	9	This facility is listed in the RFC as “clock”, but is often renamed as “lock”, and used for locking mechanisms, such as file locking queues It is often substituted for the “Clock(15)” facility code. It may (on some systems) be identical to the Clock(15) facility.
Auth	19	This is similar to the “Security(4)” facility, but is generally reserved for authorization errors, such as invalid logins. It is somewhat synonymous with both “Security(4)” and “Audit(13)”. It represents one of the areas of the RFC that is not clearly delineated, hence is subject to interpretation.
FTP	11	These messages are related to the Unix “ftpd” process, and FTP program, which is somewhat deprecated but still in use. This facility is sometimes used for non-FTP protocol messages related to file transfers.
NTP	12	These messages are related to the Unix “ntpd” (News Transport Protocol) processes. This is somewhat deprecated, but can still be found on a variety of Unix platforms.
Audit	13	This is similar to the “Security(4) and “Auth(19)” facility codes, but mainly appropriate for audit processing, including performance monitoring. For example, a performance monitor might use this facility to periodically send the disk space and disk utilization statistics to the Syslog process for data collection. The messages that use this facility should be pertinent to performance reporting.
Alert	14	This is a general-purpose (hence heavily overused) facility to indicate an Alert condition. This may be somewhat confusing, because this is really a severity rather than a facility. Ideally, these messages would represent problems with the alerting process rather than actual alerts.
Clock	15	These messages are related to the Unix clock daemons, and other processes involved with time synchronization and maintenance. This facility is also sometimes used to mark event times, such as by issuing a Syslog message via the Unix “cron” or Windows “at” program. Some scheduler programs use this facility. Occasionally, due to ambiguities in the RFC, this facility is confused with, and substituted for the “Lock(9)” facility.

Local0	16	This is a user definable facility, used by Cisco and many other vendors. It is of the used in application software, and is an ideal candidate for being modified by the CorreLog Server to provide a more meaningful facility name, based upon the message content.
Local1	17	This is another user definable facility. See notes regarding the Local0(16) facility.
Local2	18	This is another user definable facility. See notes regarding the Local0(16) facility.
Local3	19	This is another user definable facility. See notes regarding the Local0(16) facility.
Local4	20	This is another user definable facility. In particular, this is commonly used by RedHat clustering software, and is used by the Cisco PIX software, and is used in some Perl scripts. See notes regarding the Local0(16) facility for more information.
Local5	21	This is another user definable facility. See notes regarding the Local0(16) facility.
Local6	22	This is another user definable facility. See notes regarding the Local0(16) facility.
Local7	23	This is another user definable facility. See notes regarding the Local0(16) facility.

Syslog Severity Codes And Their Meaning

The basic severities, defined by RFC 3164, are discussed below.

Debug	7	The lowest severity, reserved strictly for debugging the system. In practice, debug messages can be totally ignored by everyone.
Info	6	These are informational messages, which can be reviewed later (having some pertinence) but which can be operationally ignored because they have no effect on management activities.
Notice	5	These are messages that are sent with the intention of being noticed. They have a fairly significant level of importance. A filter should generally not remove arbitrarily remove all messages with this severity.
Warning	4	A significant message. It signifies a non-trivial degree of risk. There may not be any corrective action needed with this type of message.
Error	3	A highly significant message. The message indicates that corrective action, manual intervention, or operational change is necessary.
Critical	2	A critical situation exists that requires immediate attention. All other activities should be set aside and the problem addressed as soon as possible. Possible risk to security or data or infrastructure is eminent.
Alert	1	An extremely critical condition exists that will require immediate intervention by the highest levels of management, requiring whatever resources necessary to immediately fix. Data has been lost, security has been breached, or infrastructure has been damaged.
Emergency	0	This severity should NEVER be used, reserved for situations where human safety, or the over-all health of the organization has been compromised or is in extreme jeopardy.

Appendix C: Severity Mapping

This section provides a reference guide explaining how the Windows agent assigns severity codes when the "auto" type severity is configured for the default severity for an event log.

When the user configures a setting of "auto" for the default severity of the Windows agent (or the log file monitor) the severity of the syslog message, generated by the agent and sent to the syslog collector, is automatically derived based upon the native severity of the message, or by keywords in the message. The exact mapping depends upon the type of log file being monitored, as described in this appendix.

Note that the "default" severity of "auto" can be overridden by keywords configured at the agent, and can also be overridden at the CorreLog manager. So the severities described herein apply only to the case where no special handling of message keywords.

For further information on default severities and the "DefaultSeverity" keyword, see Section 4 of this document, and Appendix A.

Windows Application Event Log Severities

The table below provides the mapping between the Windows Application event log and syslog messages generated by the agent.

Windows Application Log Severity	Assigned Syslog Severity
Critical	Critical
Error	Error
Warning	Warning
Other	Info

Windows System Event Log Severities

The table below provides the mapping between the Windows System event log and the syslog messages generated by the agent. These are similar to the application log, except that messages other than the standard Critical, Error, and Warning categories are mapped as "Notice" severities (because system events are generally more important than application events.)

Windows System Log Severity	Assigned Syslog Severity
Critical	Critical
Error	Error
Warning	Warning
Other	Notice

Windows Security Event Log Severities

The table below provides the mapping between the Windows Security event log and syslog messages generated by the agent. Unlike the other event logs, the Windows security log provides only two classes of messages, either "Success" or "Failure". The syslog message severity associated with these message types is mapped as show below.

Windows Security Log Severity	Assigned Syslog Severity
Success	Notice
Failure	Error
Other	Debug

Windows Log File Monitor Severities

Unlike the windows event logs, the log file monitor portion of the agent relies strictly on keywords within the message to establish severities. (This permits the agent to monitor arbitrary files that may have no specific event information assigned to a message.) CorreLog uses a heuristic method of establishing severities based upon the content of keywords within the log files, as shown below:

Match Keyword	**Assigned Syslog Severity**
"emergency" or "extreme" or "danger", or "hazard"	Emergency
"alert" or "attention" or "caution"	Alert
"critical" or "important" or "severe" or "significant" or "urgent" or "immediate'	Critical
"error" or "fail" or "fault" or "crash" or "abnormal"	Error
"warning" or "caution" or "terminate	Warning
"notice" or "start" or "success"	Notice
"info"	Info
"debug" or "ignore"	Debug

For Additional Help And Information…

Detailed specifications regarding the CorreLog Server, add-on components, and resources are available from our corporate website. Test software may be downloaded for immediate evaluation. Additionally, CorreLog is pleased to support proof-of-concepts, and provide technology proposals and demonstrations on request.

CorreLog, Inc., a privately held corporation, has produced software and framework components used successfully by hundreds of government and private operations worldwide. We deliver security information and event management (SIEM) software, combined with deep correlation functions, and advanced security solutions. CorreLog markets its solutions directly and through partners.

We are committed to advancing and redefining the state-of-art of system management, using open and standards-based protocols and methods. Visit our website today for more information.

CorreLog, Inc.
http://www.CorreLog.com
mailto:support@CorreLog.com