



# CorreLog<sup>®</sup>

## **Syslog Windows Tool Set (WTS)** Configuration File Directives And Help

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.

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 via the Cconfig.exe program (available in the Installer's "Start Menu".)

### **Required Parameters**

#### **DestinationAddress**

This directive should be followed by a 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.

#### **Destination Port**

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.

#### **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.

## **Optional Parameters**

### **AuxAddress**

The user can include zero or more "AuxAddress" parameters as part of the header. This directive specifies an auxiliary address that will receive syslog messages generated by the agent. The "AuxAddress" value does allow remote configuration, and does not encrypt the syslog messages. To disable this function, remove the "AuxAddress" directive, or set the value to a non-valid address, such as -1.

### **MessagePrefix**

This is a single word 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. (See later section for more info.)

### **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 100 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 Msecs.

### **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. (See later section for more info.)

### **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 additional for more info.)

### **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.) 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.)

## **Event Log Directives**

### **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.

### **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.

# Log File Monitoring Directives

## 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".

## 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;"

## 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 amount 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.

## 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.

**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 it 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.

## 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\2011\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 one or more "\*" (astrisk) or "?" (question mark) wildcards as part of a "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 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 target directory.

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.

Note that, if the site is using CorreLog Server to manage the log file configuration of agents, this particular feature should be used only if the Correlog Server is version 3.5.1 or higher. Otherwise, this feature should not be 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.

## **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.

## Message Prefix

As a special function, the agent can be configured to prefix all messages with a special keyword or identifier. The "MessagePrefix" directive (which may or may not be included with the standard default configuration) can be added or modified to the configuration file to specify this keyword. If the "MessagePrefix" is omitted, messages are not prefixed. The "MessagePrefix" is typically short, but can be up to 256 characters in length.

The "MessagePrefix" can identify the platform, or can identify other characteristics of all messages sent by the agent, useful for correlating the message content.

In particular the "MarkerPrefix" content can contain environmental variable references, and date and time specifications. For example, to prefix the local time of the machine and the machine name, use the following as the value of the "MessagePrefix":

```
MessagePrefix %b %m %H:%M:%S %COMPUTERNAME%
```

The above prefix causes the agent message to conform to the "timestamp" and "hostname" header specifications suggested by RFC-3164 (Section 4.1.2)

Note that the %COMPUTERNAME% value corresponds to an environmental variable (*including the leading and trailing % characters.*) The %b %m %H:%M:%S values (*without a trailing % characters*) corresponds to the time at the current locale, represented as the abbreviated month, day, and hour:minute:second value (See earlier section on special log file names for a list of supported time specifications.)

## 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 serves as 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.

As with the "MessagePrefix" value, described above, the "MarkerMessage" content 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":

```
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 earlier section on special log file names for a list of supported time specifications.)

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.

## Other Notes

As shown 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 exist many thousands 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. This directive can be used only with the "LogFile" directive.

Finally, note that there is a special and optional "MaxSizeChange" directive associated with log file monitoring. If the log file 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 prevents a situation where the administrator truncates the file by hand, or copies some other file on top of the monitored file (as may often occur.)

## **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>