NAME

sudo_logsrvd.conf - configuration for sudo_logsrvd

DESCRIPTION

The **sudo_logsrvd.conf** file is used to configure the **sudo_logsrvd** log server. It uses an INI-style format made up of sections in square brackets and "key = value" pairs specific to each section below the section name. Depending on the key, values may be integers, booleans or strings. Section and key names are not case sensitive, but values are.

The pound sign (‘#’) is used to indicate a comment. Both the comment character and any text after it, up to the end of the line, are ignored. Lines beginning with a semi-colon (‘;’) are also ignored.

Long lines can be continued with a backslash (‘\’) as the last character on the line. Note that leading white space is removed from the beginning of lines even when the continuation character is used.

The *EXAMPLES* section contains a copy of the default **sudo_logsrvd.conf** file.

The following configuration sections are recognized:

- server
- iolog
- eventlog
- syslog
- logfile

Each section is described in detail below.

server

The *server* section configures the address and port the server will listen on. The following keys are recognized:

listen_address = host[:port][(tls)]

The host name or IP address, optional port to listen on and an optional Transport Layer Security (TLS) flag in parentheses.

The host may be a host name, an IPv4 address, an IPv6 address in square brackets or the wild card entry ‘*’. A host setting of ‘*’ will cause **sudo_logsrvd** to listen on all configured network interfaces.

If the optional tls flag is present, **sudo_logsrvd** will secure the connection with TLS version

1.2 or 1.3. Versions of TLS prior to 1.2 are not supported. See sudo_logsrvd(8) for details on generating TLS keys and certificates.

If a port is specified, it may either be a port number or a known service name as defined by the system service name database. If no port is specified, port 30343 will be used for plaintext connections and port 30344 will be used for TLS connections.

The default value is:

listen_address = *:30343
listen_address = *:30344(tls)

which will listen on all configured network interfaces for both plaintext and TLS connections. Multiple *listen_address* lines may be specified to listen on more than one port or interface.

pid_file = path

The path to the file containing the process ID of the running **sudo_logsrvd**. If set to an empty value, or if **sudo_logsrvd** is run with the **-n** option, no *pid_file* will be created. If *pid_file* refers to a symbolic link, it will be ignored. The default value is */var/run/sudo/sudo_logsrvd.pid*.

tcp_keepalive = boolean

If true, **sudo_logsrvd** will enable the TCP keepalive socket option on the client connection. This enables the periodic transmission of keepalive messages to the client. If the client does not respond to a message, the connection will be closed.

timeout = number

The amount of time, in seconds, **sudo_logsrvd** will wait for the client to respond. A value of 0 will disable the timeout. The default value is 30.

tls_cacert = path

The path to a certificate authority bundle file, in PEM format, to use instead of the system's default certificate authority database when authenticating clients. The default is to use */etc/ssl/sudo/cacert.pem* if it exists, otherwise the system's default certificate authority database is used.

tls_cert = path

The path to the server's certificate file, in PEM format. The default value is */etc/ssl/sudo/certs/logsrvd_cert.pem*.

tls_checkpeer = bool

If true, client certificates will be validated by the server; clients without a valid certificate will be unable to connect. If false, no validation of client certificates will be performed. It true and client certificates are created using a private certificate authority, the *tls_cacert* setting must be set to a CA bundle that contains the CA certificate used to generate the client certificate. The default value is false.

tls_ciphers_v12 = string

A list of ciphers to use for connections secured by TLS version 1.2 only, separated by a colon ':'. See the *CIPHER LIST FORMAT* section in openssl-ciphers(1) for full details. The default value is HIGH:!aNULL which consists of encryption cipher suites with key lengths larger than 128 bits, and some cipher suites with 128-bit keys. Cipher suites that offer no authentication are excluded.

tls_ciphers_v13 = string

A list of ciphers to use for connections secured by TLS version 1.3 only, separated by a colon ':'. Supported cipher suites depend on the version of OpenSSL used, but should include the following:

TLS_AES_128_GCM_SHA256
TLS_AES_256_GCM_SHA384
TLS_CHACHA20_POLY1305_SHA256
TLS_AES_128_CCM_SHA256
TLS_AES_128_CCM_8_SHA256

The default cipher suite is TLS_AES_256_GCM_SHA384.

tls_dhparams = path

The path to a file containing custom Diffie-Hellman parameters in PEM format. This file can be created with the following command:

openssl dhparam -out /etc/sudo_logsrvd_dhparams.pem 2048

By default, **sudo_logsrvd** will use the OpenSSL defaults for Diffie-Hellman key generation.

tls_key = path

The path to the server's private key file, in PEM format. The default value is */etc/ssl/sudo/private/logsrvd_key.pem*.

tls_verify = bool

If true, the server certificate will be verified at startup and clients will authenticate the server

by verifying its certificate and identity. If false, no verification is performed of the server certificate by the server or the client. When using self-signed certificates without a certificate authority, this setting should be set to false. The default value is true.

iolog

The *iolog* section configures I/O log parameters. These settings are identical to the I/O configuration in sudoers(5). The following keys are recognized:

iolog_compress = boolean

If set, I/O logs will be compressed using **zlib**. Enabling compression can make it harder to view the logs in real-time as the program is executing due to buffering. The default value is false.

iolog_dir = path

The top-level directory to use when constructing the path name for the I/O log directory. The session sequence number, if any, is stored in the directory. The default value is */var/log/sudo-io*.

The following percent (‘%’) escape sequences are supported:

%{seq}

expanded to a monotonically increasing base-36 sequence number, such as 0100A5, where every two digits are used to form a new directory, e.g., *01/00/A5*

%{user}

expanded to the invoking user’s login name

%{group}

expanded to the name of the invoking user’s real group-ID

%{runas_user}

expanded to the login name of the user the command will be run as (e.g., root)

%{runas_group}

expanded to the group name of the user the command will be run as (e.g., wheel)

%{hostname}

expanded to the local host name without the domain name

%{command}

expanded to the base name of the command being run

In addition, any escape sequences supported by the system's strftime(3) function will be expanded.

To include a literal '%' character, the string '%%' should be used.

iolog_file = path

The path name, relative to *iolog_dir*, in which to store I/O logs. Note that *iolog_file* may contain directory components. The default value is %{seq}.

See the *iolog_dir* setting above for a list of supported percent ('%') escape sequences.

In addition to the escape sequences, path names that end in six or more Xs will have the Xs replaced with a unique combination of digits and letters, similar to the mktemp(3) function.

If the path created by concatenating *iolog_dir* and *iolog_file* already exists, the existing I/O log file will be truncated and overwritten unless *iolog_file* ends in six or more Xs.

iolog_flush = boolean

If set, I/O log data is flushed to disk after each write instead of buffering it. This makes it possible to view the logs in real-time as the program is executing but may significantly reduce the effectiveness of I/O log compression. The default value is true.

iolog_group = name

The group name to look up when setting the group-ID on new I/O log files and directories. If *iolog_group* is not set, the primary group-ID of the user specified by *iolog_user is used.* If neither *iolog_group* nor *iolog_user* are set, I/O log files and directories are created with group-ID 0.

iolog_mode = mode

The file mode to use when creating I/O log files. Mode bits for read and write permissions for owner, group or other are honored, everything else is ignored. The file permissions will always include the owner read and write bits, even if they are not present in the specified mode. When creating I/O log directories, search (execute) bits are added to match the read and write bits specified by *iolog_mode*. The default value is 0600.

iolog_user = name

The user name to look up when setting the owner of new I/O log files and directories. If *iolog_group* is set, it will be used instead of the user's primary group-ID. By default, I/O log

files and directories are created with user and group-ID 0.

maxseq = number
: The maximum sequence number that will be substituted for the "%{seq}" escape in the I/O log file (see the *iolog_dir* description above for more information). While the value substituted for "%{seq}" is in base 36, *maxseq* itself should be expressed in decimal. Values larger than 2176782336 (which corresponds to the base 36 sequence number "ZZZZZZ") will be silently truncated to 2176782336. The default value is 2176782336.

eventlog

The *eventlog* section configures how (and if) security policy events are logged.

log_type = string
: Where to log accept, reject and alert events reported by the policy. Supported values are *syslog*, *logfile*, and *none*. The default value is *syslog*.

log_format = string
: The event log format. Supported log formats are "sudo" for traditional sudo-style logs and "json" for JSON-format logs. The JSON log entries contain the full contents of the accept, reject and alert messages. The default value is *sudo*.

syslog

The *syslog* section configures how events are logged via syslog(3).

facility = string
: Syslog facility if syslog is being used for logging. Defaults to auth.

  The following syslog facilities are supported: **authpriv** (if your OS supports it), **auth**, **daemon**, **user**, **local0**, **local1**, **local2**, **local3**, **local4**, **local5**, **local6**, and **local7**.

accept_priority = string
: Syslog priority to use when the user is allowed to run a command and authentication is successful. Defaults to notice.

  The following syslog priorities are supported: **alert**, **crit**, **debug**, **emerg**, **err**, **info**, **notice**, **warning**, and **none**. Setting it to a value of **none** will disable logging of successful commands.

reject_priority = string
: Syslog priority to use when the user is not allowed to run a command or when authentication is unsuccessful. Defaults to alert.

See *accept_priority* for the list of supported syslog priorities.

alert_priority = string
Syslog priority to use for event log alert messages received from the client. Defaults to alert.

See *accept_priority* for the list of supported syslog priorities.

maxlen = number
On many systems, syslog(3) has a relatively small log buffer. IETF RFC 5424 states that syslog servers must support messages of at least 480 bytes and should support messages up to 2048 bytes. By default, **sudo_logsrvd** creates log messages up to 960 bytes which corresponds to the historic BSD syslog implementation which used a 1024 byte buffer to store the message, date, hostname and program name.

To prevent syslog messages from being truncated, **sudo_logsrvd** will split up sudo-style log messages that are larger than *maxlen* bytes. When a message is split, additional parts will include the string "(command continued)" after the user name and before the continued command line arguments. JSON-format log entries are never split and are not affected by *maxlen*.

logfile
The *logfile* section consists of settings related to logging to a plain file (not syslog).

path = string
The path to the file-based event log. This path must be fully-qualified and start with a ‘/’ character. The default value is */var/log/sudo.log*.

time_format = string
The string used when formatting the date and time for file-based event logs. Formatting is performed via the system’s strftime(3) function so any escape sequences supported by that function will be expanded. The default value is "%h %e %T" which produces dates like "Oct 3 07:15:24" in the C locale.

FILES

/etc/sudo_logsrvd.conf Sudo log server configuration file

EXAMPLES

```
#
# sudo logsrv configuration
#
```

```
[server]
# The host name or IP address and port to listen on with an optional TLS
# flag.  If no port is specified, port 30343 will be used for plaintext
# connections and port 30344 will be used to TLS connections.
# The following forms are accepted:
#   listen_address = hostname(tls)
#   listen_address = hostname:port(tls)
#   listen_address = IPv4_address(tls)
#   listen_address = IPv4_address:port(tls)
#   listen_address = [IPv6_address](tls)
#   listen_address = [IPv6_address]:port(tls)
#
# The (tls) suffix should be omitted for plaintext connections.
#
# Multiple listen_address settings may be specified.
# The default is to listen on all addresses.
#listen_address = *:30343
#listen_address = *:30344(tls)

# The file containing the ID of the running sudo_logsrvd process.
#pid_file = /var/run/sudo/sudo_logsrvd.pid

# If set, enable the SO_KEEPALIVE socket option on the connected socket.
#tcp_keepalive = true

# The amount of time, in seconds, the server will wait for the client to
# respond.  A value of 0 will disable the timeout.  The default value is 30.
#timeout = 30

# If set, server certificate will be verified at server startup and
# also connecting clients will perform server authentication by
# verifying the server's certificate and identity.
#tls_verify = true

# Whether to verify client certificates for TLS connections.
# By default client certs are not checked.
#tls_checkpeer = false

# Path to the certificate authority bundle file in PEM format.
# Required if 'tls_verify' or 'tls_checkpeer' is set.
```

```
#tls_cacert = /etc/ssl/sudo/cacert.pem

# Path to the server's certificate file in PEM format.
# Required for TLS connections.
#tls_cert = /etc/ssl/sudo/certs/logsrvd_cert.pem

# Path to the server's private key file in PEM format.
# Required for TLS connections.
#tls_key = /etc/ssl/sudo/private/logsrvd_key.pem

# TLS cipher list (see "CIPHER LIST FORMAT" in the openssl-ciphers manual).
# NOTE that this setting is only effective if the negotiated protocol
# is TLS version 1.2.
# The default cipher list is HIGH:!aNULL.
#tls_ciphers_v12 = HIGH:!aNULL

# TLS cipher list if the negotiated protocol is TLS version 1.3.
# The default cipher list is TLS_AES_256_GCM_SHA384.
#tls_ciphers_v13 = TLS_AES_256_GCM_SHA384

# Path to the Diffie-Hellman parameter file in PEM format.
# If not set, the server will use the OpenSSL defaults.
#tls_dhparams = /etc/ssl/sudo/logsrvd_dhparams.pem

[iolog]
# The top-level directory to use when constructing the path name for the
# I/O log directory.  The session sequence number, if any, is stored here.
#iolog_dir = /var/log/sudo-io

# The path name, relative to iolog_dir, in which to store I/O logs.
# Note that iolog_file may contain directory components.
#iolog_file = %{seq}

# If set, I/O logs will be compressed using zlib.  Enabling compression can
# make it harder to view the logs in real-time as the program is executing.
#iolog_compress = false

# If set, I/O log data is flushed to disk after each write instead of
# buffering it.  This makes it possible to view the logs in real-time
# as the program is executing but reduces the effectiveness of compression.
```

```
    #iolog_flush = true

    # The group to use when creating new I/O log files and directories.
    # If iolog_group is not set, the primary group-ID of the user specified
    # by iolog_user is used.  If neither iolog_group nor iolog_user
    # are set, I/O log files and directories are created with group-ID 0.
    #iolog_group = wheel

    # The user to use when setting the user-ID and group-ID of new I/O
    # log files and directories.  If iolog_group is set, it will be used
    # instead of the user's primary group-ID.  By default, I/O log files
    # and directories are created with user and group-ID 0.
    #iolog_user = root

    # The file mode to use when creating I/O log files.  The file permissions
    # will always include the owner read and write bits, even if they are
    # not present in the specified mode.  When creating I/O log directories,
    # search (execute) bits are added to match the read and write bits
    # specified by iolog_mode.
    #iolog_mode = 0600

    # The maximum sequence number that will be substituted for the "%{seq}"
    # escape in the I/O log file.  While the value substituted for "%{seq}"
    # is in base 36, maxseq itself should be expressed in decimal.  Values
    # larger than 2176782336 (which corresponds to the base 36 sequence
    # number "ZZZZZZ") will be silently truncated to 2176782336.
    #maxseq = 2176782336

    [eventlog]
    # Where to log accept, reject and alert events.
    # Accepted values are syslog, logfile, or none.
    # Defaults to syslog
    #log_type = syslog

    # Event log format.
    # Currently only sudo-style event logs are supported.
    #log_format = sudo

    [syslog]
    # The maximum length of a syslog payload.
```

```
# On many systems, syslog(3) has a relatively small log buffer.
# IETF RFC 5424 states that syslog servers must support messages
# of at least 480 bytes and should support messages up to 2048 bytes.
# Messages larger than this value will be split into multiple messages.
#maxlen = 960

# The syslog facility to use for event log messages.
# The following syslog facilities are supported: authpriv (if your OS
# supports it), auth, daemon, user, local0, local1, local2, local3,
# local4, local5, local6, and local7.
#facility = authpriv

# Syslog priority to use for event log accept messages, when the command
# is allowed by the security policy.  The following syslog priorities are
# supported: alert, crit, debug, emerg, err, info, notice, warning, none.
#accept_priority = notice

# Syslog priority to use for event log reject messages, when the command
# is not allowed by the security policy.
#reject_priority = alert

# Syslog priority to use for event log alert messages reported by the
# client.
#alert_priority = alert

[logfile]
# The path to the file-based event log.
# This path must be fully-qualified and start with a '/' character.
#path = /var/log/sudo

# The format string used when formatting the date and time for
# file-based event logs.  Formatting is performed via strftime(3) so
# any format string supported by that function is allowed.
#time_format = %h %e %T
```

SEE ALSO

strftime(3), sudo.conf(5), sudoers(5), sudo(8), sudo_logsrvd(8)

HISTORY

See the HISTORY file in the **sudo** distribution (https://www.sudo.ws/history.html) for a brief history of

sudo.

AUTHORS

Many people have worked on **sudo** over the years; this version consists of code written primarily by:

Todd C. Miller

See the CONTRIBUTORS file in the **sudo** distribution (https://www.sudo.ws/contributors.html) for an exhaustive list of people who have contributed to **sudo**.

BUGS

If you feel you have found a bug in **sudo**, please submit a bug report at https://bugzilla.sudo.ws/

SUPPORT

Limited free support is available via the sudo-users mailing list, see https://www.sudo.ws/mailman/listinfo/sudo-users to subscribe or search the archives.

DISCLAIMER

sudo is provided "AS IS" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. See the LICENSE file distributed with **sudo** or https://www.sudo.ws/license.html for complete details.