#

# From oslo.log

#

# If set to true, the logging level will be set to DEBUG instead of the default

# INFO level. (boolean value)

# Note: This option can be changed without restarting.

#debug = false

{%- if \_data.debug is defined %}

debug = {{ \_data.debug }}

{%- endif %}

# The name of a logging configuration file. This file is appended to any

# existing logging configuration files. For details about logging configuration

# files, see the Python logging module documentation. Note that when logging

# configuration files are used then all logging configuration is set in the

# configuration file and other logging configuration options are ignored (for

# example, logging\_context\_format\_string). (string value)

# Note: This option can be changed without restarting.

# Deprecated group/name - [DEFAULT]/log\_config

#log\_config\_append = <None>

{%- if \_data.log\_appender %}

log\_config\_append=/etc/{{ \_data.app\_name }}/logging.conf

{%- endif %}

# Defines the format string for %%(asctime)s in log records. Default:

# %(default)s . This option is ignored if log\_config\_append is set. (string

# value)

#log\_date\_format = %Y-%m-%d %H:%M:%S

# (Optional) Name of log file to send logging output to. If no default is set,

# logging will go to stderr as defined by use\_stderr. This option is ignored if

# log\_config\_append is set. (string value)

# Deprecated group/name - [DEFAULT]/logfile

#log\_file = <None>

{%- if \_data.log\_file is defined %}

log\_file = {{ \_data.log\_file }}

{%- endif %}

# (Optional) The base directory used for relative log\_file paths. This option

# is ignored if log\_config\_append is set. (string value)

# Deprecated group/name - [DEFAULT]/logdir

#log\_dir = <None>

{%- if \_data.log\_dir is defined %}

log\_dir = {{ \_data.log\_dir }}

{%- endif %}

# Uses logging handler designed to watch file system. When log file is moved or

# removed this handler will open a new log file with specified path

# instantaneously. It makes sense only if log\_file option is specified and

# Linux platform is used. This option is ignored if log\_config\_append is set.

# (boolean value)

#watch\_log\_file = false

# Use syslog for logging. Existing syslog format is DEPRECATED and will be

# changed later to honor RFC5424. This option is ignored if log\_config\_append

# is set. (boolean value)

#use\_syslog = false

{%- if \_data.use\_syslog is defined %}

use\_syslog = {{ \_data.use\_syslog }}

{%- endif %}

# Enable journald for logging. If running in a systemd environment you may wish

# to enable journal support. Doing so will use the journal native protocol

# which includes structured metadata in addition to log messages.This option is

# ignored if log\_config\_append is set. (boolean value)

#use\_journal = false

# Syslog facility to receive log lines. This option is ignored if

# log\_config\_append is set. (string value)

#syslog\_log\_facility = LOG\_USER

{%- if \_data.syslog\_log\_facility is defined %}

syslog\_log\_facility = {{ \_data.syslog\_log\_facility }}

{%- endif %}

# Use JSON formatting for logging. This option is ignored if log\_config\_append

# is set. (boolean value)

#use\_json = false

# Log output to standard error. This option is ignored if log\_config\_append is

# set. (boolean value)

#use\_stderr = false

# Format string to use for log messages with context. (string value)

#logging\_context\_format\_string = %(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [%(request\_id)s %(user\_identity)s] %(instance)s%(message)s

# Format string to use for log messages when context is undefined. (string

# value)

#logging\_default\_format\_string = %(asctime)s.%(msecs)03d %(process)d %(levelname)s %(name)s [-] %(instance)s%(message)s

# Additional data to append to log message when logging level for the message

# is DEBUG. (string value)

#logging\_debug\_format\_suffix = %(funcName)s %(pathname)s:%(lineno)d

# Prefix each line of exception output with this format. (string value)

#logging\_exception\_prefix = %(asctime)s.%(msecs)03d %(process)d ERROR %(name)s %(instance)s

# Defines the format string for %(user\_identity)s that is used in

# logging\_context\_format\_string. (string value)

#logging\_user\_identity\_format = %(user)s %(tenant)s %(domain)s %(user\_domain)s %(project\_domain)s

# List of package logging levels in logger=LEVEL pairs. This option is ignored

# if log\_config\_append is set. (list value)

#default\_log\_levels = amqp=WARN,amqplib=WARN,boto=WARN,qpid=WARN,sqlalchemy=WARN,suds=INFO,oslo.messaging=INFO,oslo\_messaging=INFO,iso8601=WARN,requests.packages.urllib3.connectionpool=WARN,urllib3.connectionpool=WARN,websocket=WARN,requests.packages.urllib3.util.retry=WARN,urllib3.util.retry=WARN,keystonemiddleware=WARN,routes.middleware=WARN,stevedore=WARN,taskflow=WARN,keystoneauth=WARN,oslo.cache=INFO,dogpile.core.dogpile=INFO

# Enables or disables publication of error events. (boolean value)

#publish\_errors = false

# The format for an instance that is passed with the log message. (string

# value)

#instance\_format = "[instance: %(uuid)s] "

# The format for an instance UUID that is passed with the log message. (string

# value)

#instance\_uuid\_format = "[instance: %(uuid)s] "

# Interval, number of seconds, of log rate limiting. (integer value)

#rate\_limit\_interval = 0

# Maximum number of logged messages per rate\_limit\_interval. (integer value)

#rate\_limit\_burst = 0

# Log level name used by rate limiting: CRITICAL, ERROR, INFO, WARNING, DEBUG

# or empty string. Logs with level greater or equal to rate\_limit\_except\_level

# are not filtered. An empty string means that all levels are filtered. (string

# value)

#rate\_limit\_except\_level = CRITICAL

# Enables or disables fatal status of deprecations. (boolean value)

#fatal\_deprecations = false