#

# From oslo.policy

#

# This option controls whether or not to enforce scope when evaluating

# policies. If ``True``, the scope of the token used in the request is

# compared to the ``scope\_types`` of the policy being enforced. If the

# scopes do not match, an ``InvalidScope`` exception will be raised.

# If ``False``, a message will be logged informing operators that

# policies are being invoked with mismatching scope. (boolean value)

#enforce\_scope = false

# The file that defines policies. (string value)

#policy\_file = policy.json

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

policy\_file = {{ \_data.policy\_file }}

{%- endif %}

# Default rule. Enforced when a requested rule is not found. (string

# value)

#policy\_default\_rule = default

# Directories where policy configuration files are stored. They can be

# relative to any directory in the search path defined by the

# config\_dir option, or absolute paths. The file defined by

# policy\_file must exist for these directories to be searched.

# Missing or empty directories are ignored. (multi valued)

#policy\_dirs = policy.d

# Content Type to send and receive data for REST based policy check

# (string value)

# Possible values:

# application/x-www-form-urlencoded - <No description provided>

# application/json - <No description provided>

#remote\_content\_type = application/x-www-form-urlencoded

# server identity verification for REST based policy check (boolean

# value)

#remote\_ssl\_verify\_server\_crt = false

# Absolute path to ca cert file for REST based policy check (string

# value)

#remote\_ssl\_ca\_crt\_file = <None>

# Absolute path to client cert for REST based policy check (string

# value)

#remote\_ssl\_client\_crt\_file = <None>

# Absolute path client key file REST based policy check (string value)

#remote\_ssl\_client\_key\_file = <None>