Python flask.request.csrf_valid() Examples

The following are code examples for showing how to use *flask.request.csrf_valid()*. They are from open source Python projects. You can vote up the examples you like or vote down the ones you don't like.

Example 1

Project: *jbox* Author: *jpush* File: *csrf.py* MIT License 6 vo

```
def protect(self):
        if request.method not in self._app.config['WTF_CSRF_METHODS']:
            return

        if not validate_csrf(self._get_csrf_token()):
            reason = 'CSRF token missing or incorrect.'
            return self._error_response(reason)

        if request.is_secure and self._app.config['WTF_CSRF_SSL_STRICT']:
            if not request.referrer:
                reason = 'Referrer checking failed - no Referrer.'
                return self._error_response(reason)

            good_referrer = 'https://%s/' % request.host
            if not same_origin(request.referrer, good_referrer):
                reason = 'Referrer checking failed - origin does not match.'
                return self._error_response(reason)

        request.csrf_valid = True  # mark this request is csrf valid
```

Example 2

Project: *chihu* Author: *yelongyu* File: *csrf.py* GNU General Public License v3.0 6 vo

```
def protect(self):
        if request.method not in self._app.config['WTF_CSRF_METHODS']:
            return

        if not validate_csrf(self._get_csrf_token()):
            reason = 'CSRF token missing or incorrect.'
            return self._error_response(reason)

        if request.is_secure and self._app.config['WTF_CSRF_SSL_STRICT']:
            if not request.referrer:
                reason = 'Referrer checking failed - no Referrer.'
                return self._error_response(reason)

            good_referrer = 'https://%s/' % request.host
            if not same_origin(request.referrer, good_referrer):
                reason = 'Referrer checking failed - origin does not match.'
                return self._error_response(reason)

        request.csrf_valid = True  # mark this request is csrf valid
```

Example 3

Project: *WRGameVideos-Server* Author: *thundernet8* File: *csrf.py* GNU General Public License v2.0 6 vo

```
def protect(self):
        if request.method not in self._app.config['WTF_CSRF_METHODS']:
            return

        if not validate_csrf(self._get_csrf_token()):
            reason = 'CSRF token missing or incorrect.'
            return self._error_response(reason)

        if request.is_secure and self._app.config['WTF_CSRF_SSL_STRICT']:
            if not request.referrer:
                reason = 'Referrer checking failed - no Referrer.'
                return self._error_response(reason)

            good_referrer = 'https://%s/' % request.host
            if not same_origin(request.referrer, good_referrer):
                reason = 'Referrer checking failed - origin does not match.'
                return self._error_response(reason)

        request.csrf_valid = True  # mark this request is csrf valid
```

Example 4

Project: *flasky*   Author: *RoseOu*   File: *form.py*   MIT License   5 vc

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 5

Project: *jbox*   Author: *jpush*   File: *form.py*   MIT License   5 vc

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 6

Project: *oa_qian*   Author: *sunqb*   File: *form.py*   Apache License 2.0   5 vc

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 7

Project: *chihu* Author: *yelongyu* File: *form.py* GNU General Public License v3.0 5 vo

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 8

Project: *Tellal* Author: *mehtapgundogan* File: *form.py* GNU General Public License v2.0 5 vo

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 9

Project: *WRGameVideos-API* Author: *thundernet8* File: *form.py* GNU General Public License v2.0 5 vo

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 10

Project: *pipa-pay-server* Author: *davidvon* File: *form.py* Apache License 2.0 5 vo

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 11

Project: *WRGameVideos-Server* Author: *thundernet8* File: *form.py* GNU General Public License v2.0 5 vo

```
def validate_csrf_token(self, field):
        if not self.csrf_enabled:
            return True
        if hasattr(request, 'csrf_valid') and request.csrf_valid:
            # this is validated by CsrfProtect
            return True
        if not validate_csrf(field.data, self.SECRET_KEY, self.TIME_LIMIT):
            raise ValidationError(field.gettext('CSRF token missing'))
```

Example 12

Project: *flasky* Author: *RoseOu* File: *csrf.py* MIT License 4 vo

```
def init_app(self, app):
        app.jinja_env.globals['csrf_token'] = generate_csrf
        strict = app.config.get('WTF_CSRF_SSL_STRICT', True)
        csrf_enabled = app.config.get('WTF_CSRF_ENABLED', True)

        @app.before_request
        def _csrf_protect():
            # many things come from django.middleware.csrf
            if not csrf_enabled:
                return

            if request.method in ('GET', 'HEAD', 'OPTIONS', 'TRACE'):
                return

            if self._exempt_views:
                if not request.endpoint:
                    return

                view = app.view_functions.get(request.endpoint)
                if not view:
                    return

                dest = '%s.%s' % (view.__module__, view.__name__)
                if dest in self._exempt_views:
                    return

            csrf_token = None
            if request.method in ('POST', 'PUT', 'PATCH'):
                # find the ``csrf_token`` field in the subitted form
                # if the form had a prefix, the name will be ``{prefix}-csrf_toker
                for key in request.form:
                    if key.endswith('csrf_token'):
                        csrf_token = request.form[key]
            if not csrf_token:
                # You can get csrf token from header
                # The header name is the same as Django
                csrf_token = request.headers.get('X-CSRFToken')
            if not csrf_token:
                # The header name is the same as Rails
                csrf_token = request.headers.get('X-CSRF-Token')
            if not validate_csrf(csrf_token):
                reason = 'CSRF token missing or incorrect.'
                return self._error_response(reason)

            if request.is_secure and strict:
                if not request.referrer:
                    reason = 'Referrer checking failed - no Referrer.'
                    return self._error_response(reason)

                good_referrer = 'https://%s/' % request.host
                if not same_origin(request.referrer, good_referrer):
                    reason = 'Referrer checking failed - origin not match.'
                    return self._error_response(reason)
            request.csrf_valid = True  # mark this request is csrf valid
```

Example 13

Project: *WRGameVideos-API* Author: *thundernet8* File: *csrf.py* GNU General Public License v2.0 4 vo

```
def init_app(self, app):
        app.jinja_env.globals['csrf_token'] = generate_csrf
        strict = app.config.get('WTF_CSRF_SSL_STRICT', True)
        csrf_enabled = app.config.get('WTF_CSRF_ENABLED', True)

        @app.before_request
        def _csrf_protect():
            # many things come from django.middleware.csrf
            if not csrf_enabled:
                return

            if request.method in ('GET', 'HEAD', 'OPTIONS', 'TRACE'):
                return

            if self._exempt_views:
                if not request.endpoint:
                    return

                view = app.view_functions.get(request.endpoint)
                if not view:
                    return

                dest = '%s.%s' % (view.__module__, view.__name__)
                if dest in self._exempt_views:
                    return

            csrf_token = None
            if request.method in ('POST', 'PUT', 'PATCH'):
                # find the ``csrf_token`` field in the subitted form
                # if the form had a prefix, the name will be ``{prefix}-csrf_toker
                for key in request.form:
                    if key.endswith('csrf_token'):
                        csrf_token = request.form[key]
            if not csrf_token:
                # You can get csrf token from header
                # The header name is the same as Django
                csrf_token = request.headers.get('X-CSRFToken')
            if not csrf_token:
                # The header name is the same as Rails
                csrf_token = request.headers.get('X-CSRF-Token')
            if not validate_csrf(csrf_token):
                reason = 'CSRF token missing or incorrect.'
                return self._error_response(reason)

            if request.is_secure and strict:
                if not request.referrer:
                    reason = 'Referrer checking failed - no Referrer.'
                    return self._error_response(reason)

                good_referrer = 'https://%s/' % request.host
                if not same_origin(request.referrer, good_referrer):
                    reason = 'Referrer checking failed - origin not match.'
                    return self._error_response(reason)

            request.csrf_valid = True  # mark this request is csrf valid
```

Example 14

Project: *pipa-pay-server* Author: *davidvon* File: *csrf.py* Apache License 2.0 4 vo

```
def init_app(self, app):
        app.jinja_env.globals['csrf_token'] = generate_csrf
        strict = app.config.get('WTF_CSRF_SSL_STRICT', True)
        csrf_enabled = app.config.get('WTF_CSRF_ENABLED', True)

        @app.before_request
        def _csrf_protect():
            # many things come from django.middleware.csrf
            if not csrf_enabled:
                return

            if request.method in ('GET', 'HEAD', 'OPTIONS', 'TRACE'):
                return

            if self._exempt_views:
                if not request.endpoint:
                    return

                view = app.view_functions.get(request.endpoint)
                if not view:
                    return

                dest = '%s.%s' % (view.__module__, view.__name__)
                if dest in self._exempt_views:
                    return

            csrf_token = None
            if request.method in ('POST', 'PUT', 'PATCH'):
                # find the ``csrf_token`` field in the subitted form
                # if the form had a prefix, the name will be ``{prefix}-csrf_toker
                for key in request.form:
                    if key.endswith('csrf_token'):
                        csrf_token = request.form[key]
            if not csrf_token:
                # You can get csrf token from header
                # The header name is the same as Django
                csrf_token = request.headers.get('X-CSRFToken')
            if not csrf_token:
                # The header name is the same as Rails
                csrf_token = request.headers.get('X-CSRF-Token')
            if not validate_csrf(csrf_token):
                reason = 'CSRF token missing or incorrect.'
                return self._error_response(reason)

            if request.is_secure and strict:
                if not request.referrer:
                    reason = 'Referrer checking failed - no Referrer.'
                    return self._error_response(reason)

                good_referrer = 'https://%s/' % request.host
                if not same_origin(request.referrer, good_referrer):
                    reason = 'Referrer checking failed - origin not match.'
                    return self._error_response(reason)

            request.csrf_valid = True  # mark this request is csrf valid
```