Python flask.request.referrer() Examples

The following are code examples for showing how to use *flask.request.referrer()*. 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

```python
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: *splunk-google-drive* Author: *LukeMurphey* File: *flask_util.py*   Apache License 2.0   6 vo

```python
def authorize_view(self):
        """Flask view that starts the authorization flow.

        Starts flow by redirecting the user to the OAuth2 provider.
        """
        args = request.args.to_dict()

        # Scopes will be passed as mutliple args, and to_dict() will only
        # return one. So, we use getlist() to get all of the scopes.
        args['scopes'] = request.args.getlist('scopes')

        return_url = args.pop('return_url', None)
        if return_url is None:
            return_url = request.referrer or '/'

        flow = self._make_flow(return_url=return_url, **args)
        auth_url = flow.step1_get_authorize_url()

        return redirect(auth_url)
```

Example 3

Project: *invenio-oauthclient* Author: *inveniosoftware* File: *utils.py*   MIT License   6 vo

```python
def get_safe_redirect_target(arg='next'):
    """Get URL to redirect to and ensure that it is local.

    :param arg: URL argument.
    :returns: The redirect target or ``None``.
    """
    for target in request.args.get(arg), request.referrer:
        if target:
            redirect_uri = urisplit(target)
            allowed_hosts = current_app.config.get('APP_ALLOWED_HOSTS', [])
            if redirect_uri.host in allowed_hosts:
                return target
            elif redirect_uri.path:
                return uricompose(
                    path=redirect_uri.path,
                    query=redirect_uri.query,
                    fragment=redirect_uri.fragment
                )
    return None
```

Example 4

```python
def authorize_view(self):
    """Flask view that starts the authorization flow.

    Starts flow by redirecting the user to the OAuth2 provider.
    """
    args = request.args.to_dict()

    # Scopes will be passed as mutliple args, and to_dict() will only
    # return one. So, we use getlist() to get all of the scopes.
    args['scopes'] = request.args.getlist('scopes')

    return_url = args.pop('return_url', None)
    if return_url is None:
        return_url = request.referrer or '/'

    flow = self._make_flow(return_url=return_url, **args)
    auth_url = flow.step1_get_authorize_url()

    return redirect(auth_url)
```

Example 5

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

    try:
        validate_csrf(self._get_csrf_token())
    except ValidationError as e:
        logger.info(e.args[0])
        self._error_response(e.args[0])

    if request.is_secure and current_app.config['WTF_CSRF_SSL_STRICT']:
        if not request.referrer:
            self._error_response('The referrer header is missing.')
```

```
            good_referrer = 'https://{0}/'.format(request.host)

            if not same_origin(request.referrer, good_referrer):
                self._error_response('The referrer does not match the host.')

        g.csrf_valid = True  # mark this request as CSRF valid
```

Example 6

```
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 7

```
def delete(self):
        from airflow.api.common.experimental import delete_dag
        from airflow.exceptions import DagNotFound, DagFileExists

        dag_id = request.values.get('dag_id')
        origin = request.values.get('origin') or url_for('Airflow.index')

        try:
            delete_dag.delete_dag(dag_id)
        except DagNotFound:
            flash("DAG with id {} not found. Cannot delete".format(dag_id), 'error
            return redirect(request.referrer)
        except DagFileExists:
            flash("Dag id {} is still in DagBag. "
                  "Remove the DAG file first.".format(dag_id),
                  'error')
            return redirect(request.referrer)

        flash("Deleting DAG with id {}. May take a couple minutes to fully"
              " disappear.".format(dag_id))

        # Upon success return to origin.
        return redirect(origin)
```

Example 8

```python
def refresh(self, session=None):
        DagModel = models.DagModel
        dag_id = request.values.get('dag_id')
        orm_dag = session.query(
            DagModel).filter(DagModel.dag_id == dag_id).first()

        if orm_dag:
            orm_dag.last_expired = timezone.utcnow()
            session.merge(orm_dag)
        session.commit()

        dag = dagbag.get_dag(dag_id)
        # sync dag permission
        appbuilder.sm.sync_perm_for_dag(dag_id, dag.access_control)

        flash("DAG [{}] is now fresh as a daisy".format(dag_id))
        return redirect(request.referrer)
```

Example 9

```python
def version():
    if is_valid_origin(request.referrer):
        response = make_response(jsonify({
            "version": PROTOCOL_VERSION
        }))

        if request.referrer:
            response.headers['Access-Control-Allow-Origin'] = \
                request.referrer[:-1] if request.referrer.endswith("/") else \
                request.referrer[:-1]

        response.headers['Content-Type'] = 'application/json'

        return response
    else:
        return answers_with_404()
```

Example 10

```python
def font_files():
    if is_valid_origin(request.referrer):
        response = make_response(jsonify({
            "version": PROTOCOL_VERSION,
            "fontFiles": FONT_FILES
        }))

        if request.referrer:
            response.headers['Access-Control-Allow-Origin'] = \
                request.referrer[:-1] if request.referrer.endswith("/") else \
                request.referrer[:-1]

        response.headers['Content-Type'] = 'application/json'

        return response
```

```
    else:
        return answers_with_404()
```

Example 11

```python
def font_file():
    file_name = request.args.get("file")

    if file_name:
        if file_name in FONT_FILES:
            with open(file_name, 'rb') as bites:
                response = make_response(send_file(
                    io.BytesIO(bites.read()),
                    attachment_filename=os.path.basename(file_name),
                    mimetype='application/octet-stream'
                ))

                if request.referrer:
                    response.headers['Access-Control-Allow-Origin'] = \
                        request.referrer[:-1] if request.referrer.endswith("/") e
                        request.referrer[:-1]

                response.headers['Content-Type'] = 'application/json'

                return response

    return ('', 404)
```

Example 12

```python
def need_update():
    if is_valid_origin(request.referrer):
        response = make_response(jsonify({
            "version": PROTOCOL_VERSION
        }))

        if request.referrer:
            response.headers['Access-Control-Allow-Origin'] = \
                request.referrer[:-1] if request.referrer.endswith("/") else \
                request.referrer[:-1]

        response.headers['Content-Type'] = 'application/json'

        return response
    else:
        return answers_with_404()
```

Example 13

```python
def google_login():
    # Stores URL of page user was on last
    # This allows them to resume where they left off
    global USER_RETURN_URL
    if request.host_url in request.referrer:
```

```
        USER_RETURN_URL = request.referrer
    else:
        USER_RETURN_URL = request.host_url

    redirect_uri = url_for('google_authorized', _external=True)
    params = {
        'scope': 'https://www.googleapis.com/auth/userinfo.email',
        'response_type': 'code',
        'redirect_uri': redirect_uri
    }
    return redirect(google.get_authorize_url(**params))
```

Example 14

```
def facebook_login():
    # Stores URL of page user was on last
    # This allows them to resume where they left off
    global USER_RETURN_URL
    if request.host_url in request.referrer:
        USER_RETURN_URL = request.referrer
    else:
        USER_RETURN_URL = request.host_url

    redirect_uri = url_for('facebook_authorized', _external=True)
    params = {
        'client-id': app.config['FACEBOOK_CLIENT_ID'],
        'redirect_uri': redirect_uri,
        'scope': 'email'
    }
    return redirect(facebook.get_authorize_url(**params))
```

Example 15

```
def reset_api(self, id):
        """Reset a user's API key.

        .. :quickref: User; Reset API key

        When used on another user account, requires the `manage_users` permission.

        :param id: user id.

        :>json User user: modified user.
        """
        self.ensure_permission(id)

        user = User(get_or_404(User.get_collection(), _id=id))
        user.update_value('api_key', User.generate_api_key())

        return redirect({'user': clean_users(user)}, request.referrer)
```

Example 16

```
def add_comment(self, id):
    if comments_enabled():
        f = File(get_or_404(current_user.files, _id=id))

        if current_user.has_permission('add_probable_name'):
            probable_name = request.form.get('probable_name')
        else:
            probable_name = None

        comment = request.form.get('comment')
        analysis_id = request.form.get('analysis')
        notify = request.form.get('notify')

        if comment:
            # If there is an analysis ID, make sure it is accessible
            if analysis_id:
                get_or_404(current_user.analyses, _id=analysis_id)

            f.add_comment(current_user['_id'], comment, analysis_id, probable_
        else:
            flash('Comment should not be empty', 'danger')

    return redirect(request. referrer)
```

Example 17

```
def get_safe_redirect_target(arg='next'):
    """Get URL to redirect to and ensure that it is local.

    :param arg: (str) URL argument.
    :returns: (str|None) Redirect target or none.
    """
    for target in request.args.get(arg), request. referrer:
        if target:
            redirect_uri = uritools.urisplit(target)
            allowed_hosts = current_app.config.get('APP_ALLOWED_HOSTS', [])

            if redirect_uri.host in allowed_hosts:
                return target

            if redirect_uri.path:
                return uritools.uricompose(path=redirect_uri.path,
                                           query=redirect_uri.query)

    return None
```

Example 18

```
def getRedirectTarget():
    """ Extracts the Next target and checks its safety.

    Note:
        Relies on the flask.request object.

    Args:
        None
```

```
    Returns:
        str: URL if the target is safe.
    """
    for target in [request.values.get('next'), request.referrer]:
        if not target:
            continue
        if isSafeUrl(target):
            return target
```

Example 19

```
def deleteElementAttributeTemplate(elementAttributeTemplateId):
        elementAttributeTemplate = ElementAttributeTemplate.query.get_or_404(eleme
        tags = []
        for elementAttribute in elementAttributeTemplate.ElementAttributes:
                if elementAttribute.Element.isManaged():
                        tags.append(elementAttribute.Tag)

        elementAttributeTemplate.delete()
        db.session.commit()
        for tag in tags:
                if not tag.isReferenced():
                        tag.delete()

        db.session.commit()
        flash('You have successfully deleted the element attribute template "{}".'
        return redirect(request.referrer)
```

Example 20

```
def endEventFrame(eventFrameId):
        rootEventFrame = EventFrame.query.get_or_404(eventFrameId)
        endTimestamp = datetime.utcnow()
        for dictionary in rootEventFrame.lineage([], 0):
                eventFrame = dictionary["eventFrame"]
                if eventFrame.EndTimestamp is None:
                        eventFrame.EndTimestamp = endTimestamp
                        eventFrame.UserId = current_user.get_id()
                        for eventFrameAttributeTemplate in eventFrame.EventFrameTe
                                if eventFrameAttributeTemplate.DefaultEndValue is
                                        eventFrameAttribute = EventFrameAttribute.
                                                EventFrameAttribute.EventFrameAttr
                                        if eventFrameAttribute is not None:
                                                tagValue = TagValue(TagId = eventF
                                                        Value = eventFrameAttribut
                                                db.session.add(tagValue)

        db.session.commit()
        flash('You have successfully ended "{}" for event frame "{}".'.format(root
                "alert alert-success")
        return redirect(request.referrer)
```

Example 21

```
def addArea(siteId):
        operation = "Add"
        form = AreaForm()

        # Add a new area.
        if form.validate_on_submit():
                area = Area(Abbreviation = form.abbreviation.data, Description = f
                db.session.add(area)
                db.session.commit()
                flash("You have successfully added the new area \"{}\".".format(ar
                return redirect(form.requestReferrer.data)

        # Present a form to add a new area.
        form.siteId.data = siteId
        if form.requestReferrer.data is None:
                form.requestReferrer.data = request. referrer

        site = Site.query.get_or_404(siteId)
        breadcrumbs = [{"url" : url_for("physicalModels.selectPhysicalModel", sele
                "text" : "<span class = \"glyphicon glyphicon-home\"></span>"},
                {"url" : url_for("physicalModels.selectPhysicalModel", selectedCla
                        "text" : site.Enterprise.Name},
                {"url" : url_for("physicalModels.selectPhysicalModel", selectedCla
        return render_template("addEdit.html", breadcrumbs = breadcrumbs, form = f
```

Example 22

```
def deleteElement(elementId):
        element = Element.query.get_or_404(elementId)
        tags = []
        if element.isManaged():
                for elementAttribute in element.ElementAttributes:
                        tags.append(elementAttribute.Tag)
                for eventFrameAttribute in element.EventFrameAttributes:
                        tags.append(eventFrameAttribute.Tag)

        element.delete()
        db.session.commit()
        for tag in tags:
                if not tag.isReferenced():
                        tag.delete()

        db.session.commit()
        flash('You have successfully deleted the element "{}".'.format(element.Nam
        return redirect(request. referrer )
```

Example 23

```
def deleteEventFrameAttributeTemplate(eventFrameAttributeTemplateId):
        eventFrameAttributeTemplate = EventFrameAttributeTemplate.query.get_or_404
        tags = []
        for eventFrameAttribute in eventFrameAttributeTemplate.EventFrameAttribute
                if eventFrameAttribute.Element.isManaged():
                        tags.append(eventFrameAttribute.Tag)

        eventFrameAttributeTemplate.delete()
        db.session.commit()
```

```
        for tag in tags:
                if not tag.isReferenced():
                        tag.delete()

        db.session.commit()
        flash("You have successfully deleted the event frame attribute template \'
        return redirect(request. referrer)
```

Example 24

6 vo

```
def addLookup(enterpriseId):
        operation = "Add"
        form = LookupForm()

        # Add a new lookup.
        if form.validate_on_submit():
                lookup = Lookup(EnterpriseId = form.enterpriseId.data, Name = form
                db.session.add(lookup)
                db.session.commit()
                flash("You have successfully added the lookup \"{}\".".format(look
                return redirect(form.requestReferrer.data)

        # Present a form to add a new lookup.
        form.enterpriseId.data = enterpriseId
        if form.requestReferrer.data is None:
                form.requestReferrer.data = request. referrer

        enterprise = Enterprise.query.get_or_404(enterpriseId)
        breadcrumbs = [{"url" : url_for("lookups.selectLookup", selectedClass = "F
                {"url" : url_for("lookups.selectLookup", selectedClass = "Enterpri
        return render_template("addEdit.html", breadcrumbs = breadcrumbs, form = f
```

Example 25

6 vo

```
def editLookup(lookupId):
        operation = "Edit"
        lookup = Lookup.query.get_or_404(lookupId)
        form = LookupForm(obj = lookup)

        # Edit an existing lookup.
        if form.validate_on_submit():
                lookup.EnterpriseId = form.enterpriseId.data
                lookup.Name = form.name.data
                db.session.commit()
                flash("You have successfully edited the lookup \"{}\".".format(loo
                return redirect(form.requestReferrer.data)

        # Present a form to edit an existing lookup.
        form.lookupId.data = lookup.LookupId
        form.enterpriseId.data = lookup.EnterpriseId
        form.name.data = lookup.Name
        if form.requestReferrer.data is None:
                form.requestReferrer.data = request. referrer

        breadcrumbs = [{"url" : url_for("lookups.selectLookup", selectedClass = "F
                {"url" : url_for("lookups.selectLookup", selectedClass = "Enterpri
```

```
                {"url" : None, "text" : lookup.Name}]
        return render_template("addEdit.html", breadcrumbs = breadcrumbs, form = f
```

Example 26

```
def addSite(enterpriseId):
        operation = "Add"
        form = SiteForm()

        # Add a new site.
        if form.validate_on_submit():
                site = Site(Abbreviation = form.abbreviation.data, Description = f
                db.session.add(site)
                db.session.commit()
                flash("You have successfully added the new site \"{}\".".format(si
                return redirect(form.requestReferrer.data)

        # Present a form to add a new site.
        form.enterpriseId.data = enterpriseId
        if form.requestReferrer.data is None:
                form.requestReferrer.data = request. referrer

        enterprise = Enterprise.query.get_or_404(enterpriseId)
        breadcrumbs = [{"url" : url_for("physicalModels.selectPhysicalModel", sele
                "text" : "<span class = \"glyphicon glyphicon-home\"></span>"},
                {"url" : url_for("physicalModels.selectPhysicalModel", selectedCla
        return render_template("addEdit.html", breadcrumbs = breadcrumbs, form = f
```

Example 27

```
def login():
    data_info.last_path = request. referrer
    log.info("User is logging in")
    scope = ['identify', 'guilds','email'] #email is for discourse rank purpose.
    discord = utils.make_session(scope=scope)
    authorization_url, state = discord.authorization_url(
        data_info.AUTHORIZATION_BASE_URL,
        access_type="offline"
    )
    session['oauth2_state'] = state
    return redirect(authorization_url)
```

Example 28

```
def restart_page(pr_id, pr_sha1):
    worker.user_add(pr_id, pr_sha1)
    # if request.is_secure:
    #    if request. referrer:
    #        return redirect(request. referrer)
    return redirect(request. referrer)
```

Example 29

```python
def stop_page(pr_id, pr_sha1):
    worker.cancel(pr_id, pr_sha1)
    # if request.is_secure:
    #     if request.referrer:
    #         return redirect(request.referrer)
    return redirect(request.referrer)


# logs/jbech-linaro/
```

Example 30

5 vo

```python
def answer(topic = None):
    """
    Main rendering function, it processes incoming weather queries.
    Depending on user agent it returns output in HTML or ANSI format.

    Incoming data:
        request.args
        request.headers
        request.remote_addr
        request.referrer
        request.query_string
    """

    user_agent = request.headers.get('User-Agent', '').lower()
    html_needed = is_html_needed(user_agent)
    options = parse_query(request.args)
    hostname = request.headers['Host']

    if request.headers.getlist("X-Forwarded-For"):
        ip = request.headers.getlist("X-Forwarded-For")[0]
        if ip.startswith('::ffff:'):
            ip = ip[7:]
    else:
        ip = request.remote_addr
    if request.headers.getlist("X-Forwarded-For"):
        ip = request.headers.getlist("X-Forwarded-For")[0]
        if ip.startswith('::ffff:'):
            ip = ip[7:]
    else:
        ip = request.remote_addr


    if topic is None:
        topic = ":firstpage"

    answer = cmd_wrapper(topic, hostname=hostname, request_options=options, html=i

    if ip not in SKIP_LOGGING_FOR_THIS_IPS:
        log_query(ip, hostname, topic, user_agent)
    return answer
```

Example 31

5 vo

```python
def get_redirect_target():
    for target in request.form.get('next'), request.values.get('next'), request.re
        if is_safe_url(target):
            return target
    return ''
```

Example 32

```python
def redirect_to_referer(endpoint='general.home', **values):
    if is_safe_url(request.referrer):
        return redirect(request.referrer)
    return redirect(url_for(endpoint, **values))
```

Example 33

```python
def get_redirect_target():
    for target in request.form.get('next'), request.values.get('next'), request.re
        if is_safe_url(target):
            return target
    return ''
```

Example 34

```python
def redirect_to_referer(endpoint='general.home', **values):
    if is_safe_url(request.referrer):
        return redirect(request.referrer)
    return redirect(url_for(endpoint, **values))
```

Example 35

```python
def unlink_oauth(provider):
        if request.host_url + 'me' != request.referrer:
            pass
        query = ub.session.query(ub.OAuth).filter_by(
            provider=provider,
            user_id=current_user.id,
        )
        try:
            oauth = query.one()
            if current_user and current_user.is_authenticated:
                oauth.user = current_user
                try:
                    ub.session.delete(oauth)
                    ub.session.commit()
                    logout_oauth_user()
                    flash(_(u"Unlink to %(oauth)s success.", oauth=oauth_check[pro
                except Exception as e:
                    log.exception(e)
                    ub.session.rollback()
                    flash(_(u"Unlink to %(oauth)s failed.", oauth=oauth_check[prov
        except NoResultFound:
            log.warning("oauth %s for user %d not fount", provider, current_user.i
```

```
                    flash(_(u"Not linked to %(oauth)s.", oauth=oauth_check[provider]), cat
            return redirect(url_for('web.profile'))


    # notify on OAuth provider error
```

Example 36

```
def get_redirect_target():
    for target in request.values.get('next'), request.referrer:
        if not target:
            continue
        if is_safe_url(target):
            return target
```

Example 37

```
def before_request():
    bad_ua=['Googlebot-Image','FeedDemon ','BOT/0.1 (BOT for JCE)','CrawlDaddy ','
    global referrer
    try:
        ip = request.headers['X-Forwarded-For'].split(',')[0]
    except:
        ip = request.remote_addr
    try:
        ua = request.headers.get('User-Agent')
    except:
        ua="null"
    if sum([i.lower() in ua.lower() for i in bad_ua])>0:
        return redirect('http://www.baidu.com')
    # print '{}:{}:{}'.format(request.endpoint,ip,ua)
    referrer=request.referrer if request.referrer is not None else 'no-referrer'
```

Example 38

```
def get_next_url(self):
    """Returns the URL where we want to redirect to.  This will
    always return a valid URL.
    """
    return (
        self.check_safe_root(request.values.get('next')) or
        self.check_safe_root(request.referrer) or
        (self.fallback_endpoint and
         self.check_safe_root(url_for(self.fallback_endpoint))) or
        request.url_root
    )
```

Example 39

```
def delete_flashcardcollection(id):
    flashcardcollection = FlashcardCollection.query.get_or_404(id)
    db.session.delete(flashcardcollection)
```

```
        db.session.commit()
        flash('Flashcardcollection {0} has been deleted'.format(flashcardcollection.na
        return redirect(request. referrer )
```

Example 40

```python
def get_redirect_target(endpoint="content.index"):
    targets = (
        session.get("next_url"),
        request.args.get("next"),
        request. referrer ,
        url_for(endpoint),
    )
    for target in targets:
        if not target:
            continue
        if is_safe_url(target, allowed_hosts=None):
            return target
```

Example 41

```python
def redirect_url(default='index'):
    return request.args.get('next') or \
           request. referrer or \
           url_for(default)
```

Example 42

```python
def _local_access() -> bool:
    if not request. referrer :
        return False
    if not re.match(r"^http://127\.0\.0\.1(:\d+)?/", request. referrer ):
        return False
    return request.remote_addr == "127.0.0.1"
```

Example 43

```python
def _local_access() -> bool:
    if request. referrer is not None and not re.match(r"^http://127\.0\.0\.1(:\d+)
        return False
    return request.remote_addr == "127.0.0.1"
```

Example 44

```python
def anticsrf(f):
    @wraps(f)
    def wrapper(*args, **kwargs):
        try:
            if request. referrer and request. referrer .replace('http://', '').split
```

```
                return f(*args, **kwargs)
            else:
                return redirect(url_for('NotFound'))
        except Exception, e:
            print e
            return redirect(url_for('Error'))
```

Example 45

```
def get_next_url(self):
        """Returns the URL where we want to redirect to.  This will
        always return a valid URL.
        """
        return (
            self.check_safe_root(request.values.get('next')) or
            self.check_safe_root(request.referrer) or
            (self.fallback_endpoint and self.check_safe_root(url_for(self.fallback
            request.url_root
        )
```

Example 46

```
def get_redirect_target():
    for target in request.args.get('next'), request.referrer:
        if not target:
            continue
        if is_safe_url(target):
            return target
```

Example 47

```
def login():
    """ Login the User with a referral code from the github oauth process"""
    session['session_code'] = request.args.get('code')
    if _request_access_token() and _load_user_data():
        logger.info("User " + session['user_data']['login'] + " logged in")
        if request.referrer is not None and 'github' not in request.referrer:
            origin = origin_from_referrer(request.referrer)
            return redirect(origin + get_config()['frontend-redirect'])
        return redirect(get_config()['frontend-host'] + get_config()['frontend-red
```

Example 48

```
def origin_from_referrer(referrer):
    double_slash_index = referrer.find("//")
    return referrer[0:referrer.find("/", double_slash_index + 2)]
```

Example 49

```
def signout():
    """Signout"""
    signout_user()
    return redirect(request.referrer or url_for('site.index'))
```

Example 50

Project: *airflow*   Author: *apache*   File: *github_enterprise_auth.py*   Apache License 2.0   5 vo

```
def login(self, request):
        log.debug('Redirecting user to GHE login')
        return self.ghe_oauth.authorize(callback=url_for(
            'ghe_oauth_callback',
            _external=True),
            state=request.args.get('next') or request.referrer or None)
```