Python flask.request.remote_addr() Examples

The following are code examples for showing how to use *flask.request.remote_addr()*. 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: *zmirror* Author: *aploium* File: *zmirror.py* MIT License 6 vo

```
def zmirror_status():
    """返回服务器的一些状态信息"""
    if request.remote_addr and request.remote_addr != '127.0.0.1':
        return generate_simple_resp_page(b'Only 127.0.0.1 are allowed', 403)
    output = ""
    output += strx('extract_real_url_from_embedded_url', extract_real_url_from_emb
    output += strx('\nis_content_type_streamed', is_mime_streamed.cache_info())
    output += strx('\nembed_real_url_to_embedded_url', embed_real_url_to_embedded_
    output += strx('\ncheck_global_ua_pass', check_global_ua_pass.cache_info())
    output += strx('\nextract_mime_from_content_type', extract_mime_from_content_t
    output += strx('\nis_content_type_using_cdn', is_content_type_using_cdn.cache_
    output += strx('\nis_ua_in_whitelist', is_content_type_using_cdn.cache_info())
    output += strx('\nis_mime_represents_text', is_mime_represents_text.cache_info
    output += strx('\nis_domain_match_glob_whitelist', is_domain_match_glob_whitel
    output += strx('\nverify_ip_hash_cookie', verify_ip_hash_cookie.cache_info())
    output += strx('\nis_denied_because_of_spider', is_denied_because_of_spider.ca
    output += strx('\nis_ip_not_in_allow_range', is_ip_not_in_allow_range.cache_in
    output += strx('\n\ncurrent_threads_number', threading.active_count())
    # output += strx('\nclient_requests_text_rewrite', client_requests_text_rewrit
    # output += strx('\nextract_url_path_and_query', extract_url_path_and_query.ca

    output += strx('\n----------------\n')
    output += strx('\ndomain_alias_to_target_set', domain_alias_to_target_set)

    return "<pre>" + output + "</pre>\n"
```

Example 2

Project: *flasky* Author: *RoseOu* File: *validators.py* MIT License 6 vo

```
def __call__(self, form, field):
        if current_app.testing:
            return True

        if request.json:
            challenge = request.json.get('recaptcha_challenge_field', '')
            response = request.json.get('recaptcha_response_field', '')
        else:
            challenge = request.form.get('recaptcha_challenge_field', '')
            response = request.form.get('recaptcha_response_field', '')
        remote_ip = request.remote_addr

        if not challenge or not response:
            raise ValidationError(field.gettext(self.message))

        if not self._validate_recaptcha(challenge, response, remote_ip):
            field.recaptcha_error = 'incorrect-captcha-sol'
            raise ValidationError(field.gettext(self.message))
```

Example 3

Project: *myweb* Author: *Busui* File: *__init__.py* MIT License 6 vo

```
def register_logging(app):
    class RequestFormatter(logging.Formatter):

        def format(self, record):
            record.url = request.url
            record.remote_addr = request.remote_addr
            return super(RequestFormatter, self).format(record)

    request_formatter = RequestFormatter(
        '[%(asctime)s] %(remote_addr)s requested %(url)s\n'
        '%(levelname)s in %(module)s: %(message)s'
    )

    formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(mess

    file_handler = RotatingFileHandler(os.path.join(basedir, 'logs/love.log'),
                                       maxBytes=10 * 1024 * 1024, backupCount=10)
    file_handler.setFormatter(formatter)
    file_handler.setLevel(logging.INFO)

    if not app.debug:
        app.logger.addHandler(file_handler)
```

Example 4

Project: *social-relay* Author: *jaywink* File: *views.py* GNU Affero General Public License v3.0 6 vo

```
def receive_public():
    if not request.data:
        return abort(404)

    # Queue to rq for processing
    public_queue.enqueue("workers.receive.process", request.data, timeout=app.conf

    # Log statistics
    log_receive_statistics(request.remote_addr)

    # return 200 whatever
    data = {
        'result': 'ok',
    }
    js = json.dumps(data)
    return Response(js, status=200, mimetype='application/json')
```

Example 5

Project: *Akeso* Author: *ameserole* File: *utils.py* MIT License 6 vo

```
def get_ip():
    """ Returns the IP address of the currently in scope request. The approach is
     (in this case the local network), and only trust the most recently defined ur
     Taken from http://stackoverflow.com/a/22936947/4285524 but the generator ther
     The trusted_proxies regexes is taken from Ruby on Rails.

     This has issues if the clients are also on the local network so you can remov

     CTFd does not use IP address for anything besides cursory tracking of teams a
```

```
     more than that if you do not know what you're doing.
    """
    trusted_proxies = app.config['TRUSTED_PROXIES']
    combined = "(" + ")|(".join(trusted_proxies) + ")"
    route = request.access_route + [request.remote_addr]
    for addr in reversed(route):
        if not re.match(combined, addr): # IP is not trusted but we trust the prox
            remote_addr = addr
            break
    else:
        remote_addr = request.remote_addr
    return remote_addr
```

Example 6

Project: *Arsenal-C2* Author: *KCarretto* File: *handlers.py* GNU General Public License v3.0 6 vo

```
def existing_agent(client, data):
    """
    This handler is called when an agent with a session id checks in.
    """
    session_id = data["session_id"]

    resp = {"session_id": session_id}
    remote_ip = request.headers.get("X-Forwarded-For", request.remote_addr)

    try:
        resp = client.session_checkin(
            session_id, data.get("responses"), data.get("config"), data.get("facts
        )
        resp["actions"] = [action.raw_json for action in resp["actions"]]
    except ResourceNotFound:
        # If the session does not exist on the teamserver, reset the session
        resp["actions"] = [{"action_id": "0", "action_type": 999}]

    return resp
```

Example 7

Project: *dudulu* Author: *MashiMaroLjc* File: *dudulu.py* MIT License 6 vo

```
def mood():
    """
    情绪分析
    :return:
    """
    ip = request.remote_addr
    sentence = request.args.get("sentence")
    if not sentence:
        return Response(FAILED, None, info="Miss Params").to_json()
    if len(sentence) > MAX_WORD or len(sentence) < MIN_WORD:
        return Response(FAILED, None, info="The Sentence "
                                           "is too long.It should be %s to %s." %
    result = get_mood(sentence, key_word=KEY_WORD, model_name=MODEL_NAME)
    print("ip: %s | sentence: %s | positive: %s | negative: %s | neutral: %s" % (i
                                                                                  r
                                                                                  r

    SENTENCE_FILE.flush()
    return Response(SUCCEED, result).to_json()
```

Example 8

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

```
def __call__(self, form, field):
        if current_app.testing:
            return True

        if request.json:
            response = request.json.get('g-recaptcha-response', '')
        else:
            response = request.form.get('g-recaptcha-response', '')
        remote_ip = request.remote_addr

        if not response:
            raise ValidationError(field.gettext(self.message))

        if not self._validate_recaptcha(response, remote_ip):
            field.recaptcha_error = 'incorrect-captcha-sol'
            raise ValidationError(field.gettext(self.message))
```

Example 9

Project: *picoCTF* Author: *picoCTF* File: *user.py* MIT License 6 vo

```
def _validate_captcha(data):
    """
    Validate a captcha with google's reCAPTCHA.

    Args:
        data: the posted form data
    """
    settings = api.config.get_settings()["captcha"]

    post_data = urllib.parse.urlencode(
        {
            "secret": settings["reCAPTCHA_private_key"],
            "response": data["g-recaptcha-response"],
            "remoteip": flask.request.remote_addr,
        }
    ).encode("utf-8")

    request = urllib.request.Request(settings["captcha_url"], post_data, method="P
    response = urllib.request.urlopen(request).read().decode("utf-8")
    parsed_response = json.loads(response)
    return parsed_response["success"] is True
```

Example 10

Project: *qis* Author: *quru* File: *views_util.py* GNU Affero General Public License v3.0 6 vo

```
def log_security_error(error, request):
    """
    Creates an error log entry and returns true if 'error' is a SecurityError,
    otherwise performs no action and returns false.
    """
    if error and isinstance(error, SecurityError):
        ip = request.remote_addr if request.remote_addr else '<unknown>'
        user = get_session_user()
        logger.error(
            'Security error for %s URL %s for user %s from IP %s : %s' % (
                request.method.upper(),
                request.url,
```

```
                user.username if user else '<anonymous>',
                ip,
                unicode_to_utf8(str(error))
            )
        )
        return True
    else:
        return False


# Cache of find/replace strings for safe_error_str()
```

Example 11

Project: *oa_qian* Author: *sunqb* File: *validators.py* Apache License 2.0 6 vo

```
def __call__(self, form, field):
        if current_app.testing:
            return True

        if request.json:
            challenge = request.json.get('recaptcha_challenge_field', '')
            response = request.json.get('recaptcha_response_field', '')
        else:
            challenge = request.form.get('recaptcha_challenge_field', '')
            response = request.form.get('recaptcha_response_field', '')
        remote_ip = request.remote_addr

        if not challenge or not response:
            raise ValidationError(field.gettext(self.message))

        if not self._validate_recaptcha(challenge, response, remote_ip):
            field.recaptcha_error = 'incorrect-captcha-sol'
            raise ValidationError(field.gettext(self.message))
```

Example 12

Project: *PyTaskManager* Author: *PersonalHealthTrain* File: *TaskMaster.py* Apache License 2.0 6 vo

```
def addClient():
    try:
        data = request.get_json()
    except:
        return Response(json.dumps({"success": False, 'message': "Could not parse

    try:
        clientId = dbDao.addClient(data["name"], data["email"], data["institute"],
        data = {
            'success': True,
            'clientId': clientId
        }
    except:
        data = {
            'success': False,
            'message': "Could not insert data in database"
        }

    return Response(json.dumps(data), mimetype="application/json")
```

Example 13

Project: *pyhackeriet* Author: *hackeriet* File: *__init__.py* Apache License 2.0 6 vo

```
def hello():
    addr = request.remote_addr
    if addr == "::1" or addr == "localhost" or addr == "127.0.0.1" and 'X-Forwarde
        addr = request.headers['X-Forwarded-For']

    if request.method == 'POST':
        if request.form['person']:
            person = request.form['person']
        else:
            person = ''

        mqtt("hackeriet/ding", "%s <%s>" % (person, encrypt(bytes(addr,"ascii"))))

        return render_template('knocked.html')
    else:
        return render_template('index.html', humla=humla)
```

Example 14

Project: *Python-IRC-Bot* Author: *wolfy1339* File: *web.py* MIT License 6 vo

```
def main():
    iplow = ip2long('192.30.252.0')
    iphigh = ip2long('192.30.255.255')
    if request.remote_addr in range(iplow, iphigh):
        payload = request.get_json()
        if payload["repository"]["name"] == "Python-IRC-Bot":
            try:
                subprocess.check_call(["git", "pull"])
            except subprocess.CalledProcessError:
                irc.privmsg("##wolfy1339", "git pull failed!")
            else:
                if "handlers.py" in payload['head_commit']['modified']:
                    reload_handlers(bot)
            return flask.Response("Thanks.", mimetype="text/plain")
        return flask.Response("Wrong repo.", mimetype="text/plain")
    else:
        flask.abort(403)
```

Example 15

Project: *webinspect* Author: *cxmcc* File: *app.py* MIT License 6 vo

```
def inspect(path):
    txt = ''
    txt += '=== path ===\n'
    txt += '/{}\n'.format(request.path)
    txt += '=== method===\n'
    txt += request.method + '\n'
    txt += '=== remote_address ===\n'
    txt += '{}\n'.format(request.remote_addr)
    txt += '=== headers ===\n'
    for k, v in request.headers:
        txt += '{}: {}\n'.format(k, v)
    txt += '=== cookies ===\n'
    for k, v in request.cookies.items():
        txt += '{}: {}\n'.format(k, v)
    txt += '=== data ===\n'
    txt += '{}\n'.format(request.data)
```

```
    txt += '=== curl ===\n'
    txt += gen_curl_command(path)
    headers = {'Server': 'github.com/cxmcc/webinspect'}
    return Response(txt, headers=headers, mimetype='text/plain')
```

Example 16

Project: *flask-geoip2* Author: *mattharley* File: *app.py* MIT License 6 vo

```
def geoip(ip_address=None):
    ip = ip_address if ip_address else request.remote_addr
    try:
        app.logger.info("looking up IP address: {}".format(ip))
        geoip_reader = get_db_reader()
        result = geoip_reader.city(ip)
        response = {}
        for key, value in JSON_MAPPING.items():
            try:
                response[key] = reduce(getattr, value.split('.'), result)
            except AttributeError:
                response[key] = ''
        response['ip'] = ip
        response['metro_code'] = METRO_CODE
        response['code'] = CODE
        app.logger.info("returning response: \n{}".format(json.dumps(response,inde
        return jsonify(**response)
    except geoip2.errors.AddressNotFoundError as e:
        app.logger.warning("Unable find ip address: {}".format(e))
        return jsonify({'error': {'message': e.message}})
```

Example 17

Project: *zmirror* Author: *aploium* File: *zmirror.py* MIT License 5 vo

```
def filter_client_request():
    """过滤用户请求, 视情况拒绝用户的访问
    :rtype: Union[Response, None]
    """
    dbgprint('Client Request Url: ', request.url)

    # crossdomain.xml
    if os.path.basename(request.path) == 'crossdomain.xml':
        dbgprint('crossdomain.xml hit from', request.url)
        return crossdomain_xml()

    # Global whitelist ua
    if check_global_ua_pass(str(request.user_agent)):
        return None

    if is_deny_spiders_by_403 and is_denied_because_of_spider(str(request.user_age
        return generate_simple_resp_page(b'Spiders Are Not Allowed To This Site',

    if human_ip_verification_enabled and (
                ((human_ip_verification_whitelist_from_cookies or enable_custom_ac
                 and must_verify_cookies)
            or is_ip_not_in_allow_range(request.remote_addr)
    ):
        dbgprint('ip', request.remote_addr, 'is verifying cookies')
        if 'zmirror_verify' in request.cookies and \
                ((human_ip_verification_whitelist_from_cookies and verify_ip_hash_
                 or (enable_custom_access_cookie_generate_and_verify and custom_ve
```

```
                        request.cookies.get('zmirror_verify'), request))):
            ip_whitelist_add(request.remote_addr, info_record_dict=request.cookie
            dbgprint('add to ip_whitelist because cookies:', request.remote_addr)
        else:
            return redirect(
                "/ip_ban_verify_page?origin=" + base64.urlsafe_b64encode(str(reque
                    encoding='utf-8'),
                code=302)

    return None
```

Example 18

Project: *chowk* Author: *fortyplustwo* File: *chowk.py* Apache License 2.0 5 vo

```
def receivesms():
    '''Handles and processes all messages coming from Kannel and going towards the

       NOTE: See the enclosed sample configuration file in kannel/ for knowing wha
       and the name of the arguments
    '''
    try: #TODO: Better exception handling!
        app.logger.debug("Received data %s", request.args)
        #TODO: Support GET as well as POST requests equally well

        msg = {}
        msg['from'] = request.args['from']
        msg['text'] = request.args['text']
        msg['args'] = request.args

        #get the ip address of the kannel server so that we can identify it and us
        #if request.remote_addr
        msg['host']   = get_kannel_server(request)

        app.logger.debug("Identified! This message came from %s Kannel server", ms

        if msg['host'] is False: #if we can't get the IP of the origin of request,
            raise Exception("Cannot retrieve IP from the request to recognize the

        send_to_rapidpro.apply_async(kwargs = {'msg': msg}, serializer = 'json')
        #we will NOT return any text because whatever is returned will be sent as
        #we return in the format (response, status, headers) so that Kannel knows
        return ('',200,[])

    except Exception as e:
        #TODO: Send an email when unrecoverable exceptions occur, instead of just
        app.logger.debug("Exception %s occurred", e)
        raise e
```

Example 19

Project: *flasky* Author: *RoseOu* File: *validators.py* MIT License 5 vo

```
def _validate_recaptcha(self, challenge, response, remote_addr):
        """Performs the actual validation."""
        try:
            private_key = current_app.config['RECAPTCHA_PRIVATE_KEY']
        except KeyError:
            raise RuntimeError("No RECAPTCHA_PRIVATE_KEY config set")

        data = url_encode({
```

```
            'privatekey': private_key,
            'remoteip':   remote_addr,
            'challenge':  challenge,
            'response':   response
        })

        response = http.urlopen(RECAPTCHA_VERIFY_SERVER, to_bytes(data))

        if response.code != 200:
            return False

        rv = [l.strip() for l in response.readlines()]

        if rv and rv[0] == to_bytes('true'):
            return True

        if len(rv) > 1:
            error = rv[1]
            if error in self._error_codes:
                raise RuntimeError(self._error_codes[error])

        return False
```

Example 20

Project: *PythonMicroservicesDevelopment_Code* Author: *mtianyan* File: *flask_middleware.py* Apache License 2.0 5 vo

```
def my_microservice():
        if "X-Forwarded-For" in request.headers:
                ips = [ip.strip() for ip in
                       request.headers['X-Forwarded-For'].split(',')]
                ip = ips[1]
        else:
                ip = request.remote_addr

        return jsonify({'Hello': ip})
```

Example 21

Project: *flask-request-logger* Author: *BbsonLin* File: *request_logger.py* MIT License 5 vo

```
def _logging_req_resp(self, response):
        req_log = RequestLog(request.method, request.url, request.content_length,
        self.db.add(req_log)
        self.db.commit()
        res_log = ResponseLog(response.status_code, response.content_length, req_l
        self.db.add(res_log)
        self.db.commit()

        return response
```

Example 22

Project: *rate.sx* Author: *chubin* File: *srv.py* MIT License 5 vo

```
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 23

Project: *ns-notifications* Author: *aquatix* File: *server.py* MIT License 5 vo

```
def disable_notifier(location=None):
    location_prefix = '[{0}][location: {1}]'.format(request.remote_addr, location
    try:
        should_run = mc.get('nsapi_run')
        logger.info('%s nsapi_run was %s, disabling' % (location_prefix, should_ru
    except KeyError:
        logger.info('%s no nsapi_run tuple in memcache, creating with value False'
    mc.set('nsapi_run', False, MEMCACHE_DISABLING_TTL)
    return 'Disabling notifications'
```

Example 24

Project: *ns-notifications* Author: *aquatix* File: *server.py* MIT License 5 vo

```
def enable_notifier(location=None):
    location_prefix = '[{0}][location: {1}]'.format(request.remote_addr, location
    try:
        should_run = mc.get('nsapi_run')
        logger.info('%s nsapi_run was %s, enabling' % (location_prefix, should_run
    except KeyError:
```

```
        logger.info('%s no nsapi_run tuple in memcache, creating with value True'
    mc.set('nsapi_run', True, MEMCACHE_DISABLING_TTL)
    return 'Enabling notifications'
```

Example 25

Project: *hooks* Author: *ddevault* File: *hooks.py* MIT License 5 vo

```
def hook_publish():
    raw = request.data.decode("utf-8")
    try:
        event = json.loads(raw)
    except:
        return "Hook rejected: invalid JSON", 400
    repository = "{}/{}".format(event["repository"]["owner"]["name"], event["repos
    matches = [h for h in hooks if h.repository == repository]
    if len(matches) == 0:
        return "Hook rejected: unknown repository {}".format(repository)
    hook = matches[0]

    allow = False
    remote = request.remote_addr
    if remote == "127.0.0.1" and "X-Real-IP" in request.headers:
        remote = request.headers.get("X-Real-IP")
    for ip in hook.valid_ips.split(","):
        parts = ip.split("/")
        range = 32
        if len(parts) != 1:
            range = int(parts[1])
        addr = networkMask(parts[0], range)
        if addressInNetwork(dottedQuadToNum(remote), addr):
            allow = True
    if not allow:
        return "Hook rejected: unauthorized IP", 403

    if any("[noupdate]" in c["message"] for c in event["commits"]):
        return "Hook ignored: commit specifies [noupdate]"

    if "refs/heads/" + hook.branch == event["ref"]:
        print("Executing hook for " + hook.name)
        p=Popen(hook.command.split(), stdin=PIPE)
        p.communicate(input=raw.encode())
        return "Hook accepted"

    return "Hook ignored: wrong branch"
```

Example 26

Project: *LDAP-RestAPI-Gateway* Author: *ziozzang* File: *server.py* MIT License 5 vo

```
def get_real_ip():
        ipaddr = request.remote_addr
        if "X-Forwarded-For" in request.headers.keys():
                if ipaddr != request.headers["X-Forwarded-For"]:
                        ipaddr = request.headers["X-Forwarded-For"].strip()
        if "X-Real-Ip" in request.headers.keys():
                if ipaddr != request.headers["X-Real-Ip"]:
                        ipaddr = request.headers["X-Real-Ip"].strip()
        return ipaddr

# Check if IP is restricted
```

Example 27

Project: *flask-monitor* Author: *fraoustin* File: *main.py* GNU General Public License v2.0 5 vo

```
def _dict(self):
        mydict = {}
        # manage timing
        mydict['timing'] = {}
        mydict['timing']['delta'] = self.timing
        mydict['timing']['start'] = self.request._stats_start_event
        mydict['timing']['asctime'] = asctime(gmtime(self.request._stats_start_eve
        # manage flask
        mydict['flask'] = {}
        mydict['flask']['secret_key'] = current_app.config['SECRET_KEY']
        mydict['flask']['server_name'] = current_app.config['SERVER_NAME']
        mydict['flask']['session_cookie_name'] = current_app.config['SESSION_COOKI
        mydict['flask']['session_cookie_domain'] = current_app.config['SESSION_COO
        mydict['flask']['session_cookie_path'] = current_app.config['SESSION_COOKI
        mydict['flask']['session_cookie_httponly'] = current_app.config['SESSION_C
        mydict['flask']['session_cookie_secure'] = current_app.config['SESSION_COO
        mydict['flask']['session_refresh_each_request'] = current_app.config['SESS
        # manage request
        mydict['request'] = {}
        mydict['request']['url'] = request.url
        mydict['request']['args'] = {arg: request.args.get(arg) for arg in request
        mydict['request']['view_args'] = request.view_args
        mydict['request']['path'] = request.path
        mydict['request']['method'] = request.method
        mydict['request']['remote_addr'] = request.remote_addr
        try:
            mydict['request']['rule'] = request.url_rule.rule
        except:
            mydict['request']['rule'] = ''
        #manage response
        mydict['response'] = {}
        mydict['response']['status_code'] = self.response.status_code
        mydict['response']['headers'] = { i:j for i,j in self.response.headers}
        return mydict
```

Example 28

Project: *gym* Author: *intrig-unicamp* File: *main.py* Apache License 2.0 5 vo

```
def post(self, path=None):
        method = 'post'
        prefix, call = self.parse_path(path)
        data = request.data
        address = request.remote_addr
        handler = self.handlers[method]
        ack, reply = handler((address, prefix, call, data))
        code = 200 if ack else 500
        resp = make_response(reply, code)
        resp.headers['Content-Type'] = self.content_type
        return resp
```

Example 29

Project: *gym* Author: *intrig-unicamp* File: *main.py* Apache License 2.0 5 vo

```
def get(self, path=None):
        method = 'get'
```

```
        prefix, call = self.parse_path(path)
        data = request.data
        address = request.remote_addr
        handler = self.handlers[method]
        ack, reply = handler((address, prefix, call, data))
        code = 200 if ack else 500
        resp = make_response(reply, code)
        resp.headers['Content-Type'] = self.content_type
        return resp
```

Example 30

Project: *gym* Author: *intrig-unicamp* File: *main.py* Apache License 2.0 5 vo

```
def put(self, path=None):
        method = 'put'
        prefix, call = self.parse_path(path)
        data = request.data
        address = request.remote_addr
        handler = self.handlers[method]
        ack, reply = handler((address, prefix, call, data))
        code = 200 if ack else 500
        resp = make_response(reply, code)
        resp.headers['Content-Type'] = self.content_type
        return resp
```

Example 31

Project: *gym* Author: *intrig-unicamp* File: *main.py* Apache License 2.0 5 vo

```
def delete(self, path=None):
        method = 'delete'
        prefix, call = self.parse_path(path)
        data = request.data
        address = request.remote_addr
        handler = self.handlers[method]
        ack, reply = handler((address, prefix, call, data))
        code = 200 if ack else 500
        resp = make_response(reply, code)
        resp.headers['Content-Type'] = self.content_type
        return resp
```

Example 32

Project: *track-scanner* Author: *skyderby* File: *logging.py* GNU Affero General Public License v3.0 5 vo

```
def after_request(response):
    # This IF avoids the duplication of registry in the log,
    # since that 500 is already logged via @app.errorhandler.
    if response.status_code != 500:
        logger.error(
            '%s %s %s %s %s %s',
            strftime('[%Y-%m-%d %H:%M:%S %z]'),
            request.remote_addr,
            request.method,
            request.scheme,
            request.full_path,
            response.status
        )
    return response
```

Example 33

Project: *karp-backend* Author: *spraakbanken* File: *__init__.py* MIT License 5 vo

```
def format(self, record):
        record.req_url = request.url
        record.req_remote_addr = request.remote_addr
        record.req_method = request.method
        return logging.Formatter.format(self, record)
```

Example 34

Project: *jbox* Author: *jpush* File: *validators.py* MIT License 5 vo

```
def _validate_recaptcha(self, response, remote_addr):
        """Performs the actual validation."""
        try:
            private_key = current_app.config['RECAPTCHA_PRIVATE_KEY']
        except KeyError:
            raise RuntimeError("No RECAPTCHA_PRIVATE_KEY config set")

        data = url_encode({
            'secret':     private_key,
            'remoteip':   remote_addr,
            'response':   response
        })

        http_response = http.urlopen(RECAPTCHA_VERIFY_SERVER, to_bytes(data))

        if http_response.code != 200:
            return False

        json_resp = json.loads(to_unicode(http_response.read()))

        if json_resp["success"]:
            return True

        for error in json_resp.get("error-codes", []):
            if error in RECAPTCHA_ERROR_CODES:
                raise ValidationError(RECAPTCHA_ERROR_CODES[error])

        return False
```

Example 35

Project: *PyOne* Author: *abbeyokgo* File: *views.py* Mozilla Public License 2.0 5 vo

```
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 36

Project: *flask-boilerplate* Author: *tko22* File: *__init__.py* MIT License 5 vo

```
def format(self, record):
        record.url = request.url
        record.remote_addr = request.remote_addr
        return super().format(record)


# why we use application factories http://flask.pocoo.org/docs/1.0/patterns/appfac
```

Example 37

Project: *macro_pack* Author: *sevagas* File: *listen_server.py* Apache License 2.0 5 vo

```
def hello():
    """ called by client when signalling itself"""
    # Add bot to network if necessary
    clientId = request.form['id']
    ip = request.remote_addr
    logging.info("   [-] Hello from %s. - IP: %s" % (clientId, ip))
    return make_response("OK")
```

Example 38

Project: *rucio* Author: *rucio* File: *trace.py* Apache License 2.0 5 vo

```
def post(self):
        """
        Trace endpoint used by the pilot and CLI clients to post data access infor

        .. :quickref: Trace; Send trace.

        :<json dict payload: Dictionary contain the trace information.
        :status 201: Created.
        :status 400: Cannot decode json data.
        :status 500: Internal Error.
        """
        try:
            payload = json.loads(request.data)

            # generate entry timestamp
            payload['traceTimeentry'] = datetime.datetime.utcnow()
            payload['traceTimeentryUnix'] = calendar.timegm(payload['traceTimeentr

            # guess client IP
            payload['ip'] = request.environ.get('HTTP_X_FORWARDED_FOR')
            if payload['ip'] is None:
                payload['ip'] = request.remote_addr

            # generate unique ID
            payload['traceId'] = str(uuid.uuid4()).replace('-', '').lower()

            trace(payload=payload)

        except ValueError:
```

```
            return generate_http_error_flask(400, 'ValueError', 'Cannot decode jso
        except Exception as error:
            print(traceback.format_exc())
            return error, 500

        return "Created", 201
```

Example 39

Project: *rucio* Author: *rucio* File: *nongrid_trace.py* Apache License 2.0 5 vo

```
def post(self):
        """
        Trace endpoint used by the XAOD framework to post data access information.

        .. :quickref: XAODTrace; Send XAOD trace.

        :<json dict payload: Dictionary contain the trace information.
        :status 201: Created.
        :status 400: Cannot decode json data.
        :status 500: Internal Error.
        """
        try:
            payload = json.loads(request.data)

            # generate entry timestamp
            payload['timeentry'] = int(time.time())

            # guess client IP
            payload['ip'] = request.environ.get('HTTP_X_FORWARDED_FOR')
            if payload['ip'] is None:
                payload['ip'] = request.remote_addr

            trace(payload=payload)

        except ValueError:
            return generate_http_error_flask(400, 'ValueError', 'Cannot decode jso
        except Exception as error:
            print(traceback.format_exc())
            return error, 500

        return "Created", 201
```

Example 40

Project: *picoCTF* Author: *picoCTF* File: *logger.py* MIT License 5 vo

```
def get_request_information():
    """
    Return a dictionary of information about the user at the time of logging.

    Returns:
        The dictionary.

    """
    information = {}

    if has_request_context():
        information["request"] = {
            "api_endpoint_method": request.method,
            "api_endpoint": request.path,
```

```
            "ip": request.remote_addr,
            "platform": request.user_agent.platform,
            "browser": request.user_agent.browser,
            "browser_version": request.user_agent.version,
            "user_agent": request.user_agent.string,
        }

        if api.user.is_logged_in():
            user = api.user.get_user()
            team = api.user.get_team()
            groups = api.team.get_groups(user["tid"])

            information["user"] = {
                "username": user["username"],
                "email": user["email"],
                "team_name": team["team_name"],
                "groups": [group["name"] for group in groups],
            }
    return information
```

Example 41

Project: *sysu-ctf* Author: *ssst0n3* File: *views.py* Apache License 2.0 5 vo

```
def tracker():
    if authed():
        if not Tracking.query.filter_by(ip=ip2long(request.remote_addr)).first():
            visit = Tracking(request.remote_addr, session['id'])
            db.session.add(visit)
            db.session.commit()
            db.session.close()
```

Example 42

Project: *FXTest* Author: *liwanlei* File: *views.py* MIT License 5 vo

```
def post(self):
        data = request.get_json()
        ip = request.remote_addr
        username = data['username']
        password = data['password']
        if username is None:
            return jsonify({'msg': login_username_not_message, 'code': 33, 'data':
        if password is None:
            return jsonify({'msg': login_password_not_message, 'code': 34, 'data':
        user = User.query.filter_by(username=username).first()
        if user:
            if user.status is True:
                return jsonify({'msg': login_user_free_message, 'code': 35, 'data'
            if user.check_password(password):
                user.is_login = True
                userlog = UserLoginlog(user=user.id, ip=ip, datatime=datetime.date
                db.session.add_all([user, userlog])
                db.session.commit()
                login_user(user)
                session['username'] = username
                return jsonify({'msg': login_user_sucess_message, 'code': 200, 'da
            else:
                try:
                    num=int(self.conris.getset(user.username))
                    if (user.is_free == True and num > 5):
```

```
                    return jsonify({'msg': login_user_fremm, 'code': 200, 'dat
                else:
                    self.conris.sethase(username,num+1,1000*60*10)
                    return jsonify({'msg': login_password_error_message, 'code
            except Exception as  e:
                self.conris.sethase(username, 1, 1000 * 60 * 10)
                return jsonify({'msg': login_password_error_message, 'code': 3
        return jsonify({'msg': login_user_not_exict_message, 'code': 37, 'data': '
```

Example 43

Project: *oa_qian* Author: *sunqb* File: *validators.py* Apache License 2.0 5 vo

```
def _validate_recaptcha(self, challenge, response, remote_addr):
        """Performs the actual validation."""
        try:
            private_key = current_app.config['RECAPTCHA_PRIVATE_KEY']
        except KeyError:
            raise RuntimeError("No RECAPTCHA_PRIVATE_KEY config set")

        data = url_encode({
            'privatekey': private_key,
            'remoteip':   remote_addr,
            'challenge':  challenge,
            'response':   response
        })

        response = http.urlopen(RECAPTCHA_VERIFY_SERVER, to_bytes(data))

        if response.code != 200:
            return False

        rv = [l.strip() for l in response.readlines()]

        if rv and rv[0] == to_bytes('true'):
            return True

        if len(rv) > 1:
            error = rv[1]
            if error in self._error_codes:
                raise RuntimeError(self._error_codes[error])

        return False
```

Example 44

Project: *PyChunkedGraph* Author: *seung-lab* File: *common.py* Mozilla Public License 2.0 5 vo

```
def unhandled_exception(e):
    status_code = 500
    response_time = (time.time() - current_app.request_start_time) * 1000
    user_ip = str(request.remote_addr)
    tb = traceback.format_exception(etype=type(e), value=e, tb=e.__traceback__)

    current_app.logger.error(
        {
            "message": str(e),
            "user_id": user_ip,
            "user_ip": user_ip,
            "request_time": current_app.request_start_date,
            "request_url": request.url,
```

```
            "request_data": request.data,
            "response_time": response_time,
            "response_code": status_code,
            "traceback": tb,
        }
    )

    resp = {
        "timestamp": current_app.request_start_date,
        "duration": response_time,
        "code": status_code,
        "message": str(e),
        "traceback": tb,
    }

    return jsonify(resp), status_code
```

Example 45

Project: *PyChunkedGraph* Author: *seung-lab* File: *common.py* Mozilla Public License 2.0 5 vo

```
def api_exception(e):
    response_time = (time.time() - current_app.request_start_time) * 1000
    user_ip = str(request.remote_addr)
    tb = traceback.format_exception(etype=type(e), value=e, tb=e.__traceback__)

    current_app.logger.error(
        {
            "message": str(e),
            "user_id": user_ip,
            "user_ip": user_ip,
            "request_time": current_app.request_start_date,
            "request_url": request.url,
            "request_data": request.data,
            "response_time": response_time,
            "response_code": e.status_code.value,
            "traceback": tb,
        }
    )

    resp = {
        "timestamp": current_app.request_start_date,
        "duration": response_time,
        "code": e.status_code.value,
        "message": str(e),
    }

    return jsonify(resp), e.status_code.value


# --------------------
# ------ Applications
# --------------------
```

Example 46

Project: *starctf2018* Author: *sixstars* File: *serve.py* MIT License 5 vo

```
def FLAG():
        flag = valid_url_prefixs[request.user_prefix]#+session['genesis_block_hash
        try:
```

```
                with open('flag.log', 'ab') as f:
                        f.write(request.remote_addr + '\n')
                try:
                        with open('blockchain.log', 'ab') as f:
                                f.write(json.dumps(session['blocks']) + '\n')
                except:
                        with open('blockchain.log', 'ab') as f:
                                f.write('FAILED ' + flag + '\n')
        except:
                return 'Something went ERROR, please contact admin of *CTF to get
        return 'Here is your flag: '+flag
```

Example 47

Project: *dbot-server* Author: *ATNIO* File: *decorates.py* MIT License 5 vo

```
def api_metric(f):
    @wraps(f)
    def decorated(*args, **kwargs):
        metric = dbot.get_server().metric
        endpoint = request.url
        caller = request.remote_addr
        apiinfo = metric.CallBegin(endpoint, caller)
        response = f(*args, **kwargs)
        metric.CallEnd(apiinfo, response.status_code)
        return response
    return decorated
```

Example 48

Project: *schort* Author: *sqozz* File: *schort.py* Creative Commons Zero v1.0 Universal 5 vo

```
def insertIdUnique(longUrl, idToCheck=None):
        hashUrl = hashlib.sha256(longUrl.encode()).digest()
        base64Url = base64.urlsafe_b64encode(hashUrl).decode()
        if idToCheck == None or idToCheck == "":
                idToCheck = base64Url[:4]

        conn = sqlite3.connect("data/links.sqlite")
        c = conn.cursor()
        try:
                c.execute('INSERT INTO links VALUES (?, ?, ?, ?, ?)', (idToCheck,
                databaseId = idToCheck
                conn.commit()
                conn.close()
        except sqlite3.IntegrityError as e:
                print("Hash already exists, does the long URL matches?")
                longUrlDb = c.execute('SELECT * FROM links WHERE shortLink=?', (id
                if longUrl == longUrlDb[1]:
                        print(longUrl + " is already in database with id " + idToC
                        databaseId = idToCheck
                else:
                        print("Found real hash collision for " + longUrl + " and '
                        conn.commit()
                        conn.close()
                        if len(base64Url) - 1 >= len(idToCheck) + 1:
                                databaseId = insertIdUnique(longUrl, idToCheck=bas
                        else:
                                print("Can't produce a long enough hash from the r
                                print("Bailing out, you are on your own. Good luck
                                print("==========================================
```

```
                                abort(500)

        return databaseId
```

Example 49

Project: *SempoBlockchain* Author: *teamsempo* File: *auth_api.py* GNU General Public License v3.0 5 vo

```
def get(self):

        print("process started")

        challenges = [
            ('Why don't they play poker in the jungle?', 'Too many cheetahs.'),
            ('What did the Buddhist say to the hot dog vendor?', 'Make me one with
            ('What does a zombie vegetarian eat?', 'Graaaaaaaains!'),
            ('My new thesaurus is terrible.', 'Not only that, but it's also terrib
            ('Why didn't the astronaut come home to his wife?', 'He needed his spa
            ('I got fired from my job at the bank today.',
             'An old lady came in and asked me to check her balance, so I pushed h
            ('I like to spend every day as if it's my last',
             'Staying in bed and calling for a nurse to bring me more pudding.')
        ]

        challenge = random.choice(challenges)

        # time.sleep(int(request.args.get('delay', 0)))
        # from functools import reduce
        # reduce(lambda x, y: x + y, range(0, int(request.args.get('count', 1))))

        # memory_to_consume = int(request.args.get('MB', 0)) * 1000000
        # bytearray(memory_to_consume)

        ip_address = request.environ.get('HTTP_X_REAL_IP', request.remote_addr)
        user_agent = request.environ["HTTP_USER_AGENT"]
        ip = request.environ["REMOTE_ADDR"]
        # proxies = request.headers.getlist("X-Forwarded-For")
        # http://esd.io/blog/flask-apps-heroku-real-ip-spoofing.html

        response_object = {
            'status': 'success',
            'who_allows_a_get_request_to_their_auth_endpoint': 'We do.',
            challenge[0]: challenge[1],
            # 'metadata': {'user_agent': user_agent, 'ip': ip_address, 'otherip':
        }
        return make_response(jsonify(response_object)), 200

    # @limiter.limit("20 per day")
```

Example 50

Project: *docker* Author: *skywind3000* File: *start.py* MIT License 5 vo

```
def handle(action):
        if request.method == 'GET':
                params = request.args.to_dict()
        elif request.method == 'POST':
                params = request.values.to_dict()
        else:
                params = {}
```

```
    params["ip"] = request.headers.get('X-Real-Ip', request.remote_addr)

    result = {}
    real_action = params.get("action")

    if real_action == "reg":
            ok, msg = do_reg(params)
    elif real_action == "change":
            ok, msg = do_change(params)
    elif real_action == 'reset':
            ok, msg = do_reset(params)
    else:
            ok = True
            msg = ""

    if ok and msg:
            result["title"] = msg
            result["username"] = params["username"]
            result["ip"] = params["ip"]
            result["opttime"] = time.strftime("%Y-%m-%d %H:%M:%S")
            result["oldurl"] = action
            templatefile = "result.html"
    else:
            result["msg"] = msg
            result["action"] = action
            titlemap = {"reset": u"重置SVN账号密码", "change":u"修改SVN账号密码",
            result["title"] = titlemap.get(action, "")
            templatefile = "index.html"

    if msg:
            app.logger.info("action:%r ip:%r username:%r ok:%r"\
                    %(action, params["ip"], params.get("username", ""), ok) )
    return Response(response=render_template(templatefile, **result))
```