

INBOX OPEN ID - Техническая спецификация

Для обеспечения Open ID авторизации на своей страничке необходимо проинсталлировать библиотеку Open ID, которая доступна по след адресу <https://github.com/openid>

Далее будет описано, как сконфигурировать тестовую аппликацию для взаимодействия с Inbox OpenID.

После инсталляции Open ID клиента в папке проекта создайте или отредактируйте **index.php**
Пример файла **index.php** (Данный пример доступен по адресу <http://github.com/openid/php-openid>)

```
<?php
require_once "common.php";

global $pape policy uris;
?>
<html>
<head>
<title>PHP OpenID Authentication Example</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8" />
</head>
<style type="text/css">
* {
font-family: verdana,sans-serif;
}
body {
width: 60em;
margin: 1em;
}
div {
padding: .5em;
}
table {
margin: none;
padding: none;
}
.alert {
border: 1px solid #e7dc2b;
background: #fff888;
}
.success {
border: 1px solid #669966;
background: #88ff88;
}
.error {
border: 1px solid #ff0000;
background: #ffa000;
}
#verify-form {
border: 1px solid #777777;
background: #d0d0d0;
margin-top: 1em;
padding-bottom: 0em;
}
</style>
<body>
<h1>PHP OpenID Authentication Example</h1>
<p>
This example consumer uses the <a
href="http://github.com/openid/php-openid">PHP
OpenID</a> library. It just verifies that the URL that you enter
is your identity URL.
</p>

<?php if (isset($msg)) { print "<div class=\"alert\">$msg</div>"; } ?>
<?php if (isset($error)) { print "<div class=\"error\">$error</div>"; } ?>
<?php if (isset($success)) { print "<div class=\"success\">$success</div>"; } ?>

<div id="verify-form">
<form method="get" action="try_auth.php" id="auth_form">
Autorizacion button:
<a href="#" onclick="document.getElementById('auth_form').submit();return false;"></a>
<!--Identity&nbsp;URL:-->
<input type="hidden" name="action" value="verify" />

```

```

<input type="hidden" name="openid identifier" value="http://login.inbox.lv/openid/INBOX OPEN ID/" />
<!--
<p>Optionally, request these PAPE policies:</p>
<p>
<?php foreach ($pape_policy_uris as $i => $uri) {
    print "<input type=\"checkbox\" name=\"policies[]\" value=\"\$uri\" />";
    print "$uri<br/>";
} ?>
</p>

<input type="submit" value="Verify" />-->
</form>
</div>
</body>
</html>

```

В данном файле замените INBOX_OPEN_ID на параметр (public key), который будет Вам предоставлен после регистрации администрацией портала Inbox. В качестве параметра INBOX_OPEN_ID_IMG используйте иконки из библиотеки графических материалов OPEN ID. Измените параметры и обновите файл на сервере.

Далее приведен пример файла для инициирования авторизации **try_auth.php** (Файл **common.php** идет в поставке клиентской части OPEN ID).

```

<?php

require once "common.php";
session_start();

function getOpenIDURL() {
    // Render a default page if we got a submission without an openid
    if (empty($ GET['openid identifier'])) {
        $error = "Expected an OpenID URL.";
        include 'index.php';
        exit(0);
    }

    return $ GET['openid identifier'];
}

function run() {
    $openid = getOpenIDURL();
    $consumer = getConsumer();

```

Начало процесса аутентификации OpenID:

```
$auth_request = $consumer->begin($openid);
```

Запрос означает, что нет аутентификации и мы не можем начать реализацию Open ID:

```

if (!$auth_request) {
    displayError("Authentication error; not a valid OpenID.");
}

$sreg_request = Auth_OpenID_SRegRequest::build(
    // Required
    array('nickname'),
    // Optional
    array('fullname', 'email'));

if ($sreg_request) {
    $auth_request->addExtension($sreg_request);
}

$pape_request = null;
if (isset($ GET['policies'])) {
    $policy_uris = $ GET['policies'];
}

$pape_request = new Auth_OpenID_PAPE_Request($policy_uris);
if ($pape_request) {
    $auth_request->addExtension($pape_request);
}

```

Перенаправление пользователей на сервер OpenID для аутентификации:

(Store the token for this authentication so we can verify the response. For OpenID 1, send a redirect. For OpenID 2, use a Javascript form to send a POST request to the server.)

```
if ($auth_request->shouldSendRedirect()) {
    $redirect_url = $auth_request->redirectURL(getTrustRoot(),
                                              getReturnTo());
```

Отображение сообщения об ошибке, если URL для перенаправления не может быть найден.

```
if (Auth_OpenID::isFailure($redirect_url)) {
    displayError("Could not redirect to server: " . $redirect_url->message);
} else {
```

Перенаправление:

```
    header("Location: ".$redirect_url);
}
} else {
```

Generate form markup and render it.

```
$form_id = 'openid message';
$form_html = $auth_request->htmlMarkup(getTrustRoot(), getReturnTo(),
                                       false, array('id' => $form_id));
```

Display an error if the form markup couldn't be generated otherwise, render the HTML.

```
if (Auth_OpenID::isFailure($form_html)) {
    displayError("Could not redirect to server: " . $form_html->message);
} else {
    print $form_html;
}
}
run();
?>
```

Далее приведен пример файла *finish_auth.php*, отвечающий за обработку данных после авторизации (Файл *common.php* идет в поставке клиентской части OPEN ID).

```
<?php
require_once "common.php";
session_start();

function escape($thing) {
    return htmlspecialchars($thing);
}

function run() {
    $consumer = getConsumer();
```

Завершение процесса аутентификации, вместе с ответом с сервера:

```
$return_to = getReturnTo();
$response = $consumer->complete($return_to);
// print_r($response);
// exit;
```

Проверка статуса об ответе:

```
if ($response->status == Auth_OpenID_CANCEL) {
```

Данное содержание означает, что аутентификация была отменена:

```
    $msg = 'Verification cancelled.';
} else if ($response->status == Auth_OpenID_FAILURE) {
```

Отображение сообщения об ошибке, если аутентификация не удалась:

```
$msg = "OpenID authentication failed: " . $response->message;
} else if ($response->status == Auth_OpenID_SUCCESS) {
```

Данное содержание означает, что аутентификация произошла успешно, был идентифицирован URL и регистрационные данные (если они были возвращены):

```
$openid = $response->getDisplayIdentifier();
$esc_identity = escape($openid);

$success = sprintf('You have successfully verified ' .
    '<a href="%s">%s</a> as your identity.',
    $esc_identity, $esc_identity);

if ($response->endpoint->canonicalID) {
    $escaped_canonicalID = escape($response->endpoint->canonicalID);
    $success .= ' (XRI CanonicalID: '.$escaped_canonicalID.') ';
}

$sreg_resp = Auth_OpenID_SRegResponse::fromSuccessResponse($response);

$sreg = $sreg_resp->contents();

if (@$sreg['email']) {
```

Вы также возвращаете email, nickname и fullname

```
    $success .= " You also returned '".escape($sreg['email']).
        "' as your email.";
}

if (@$sreg['nickname']) {
    $success .= " Your nickname is '".escape($sreg['nickname']).
        "'.";
}

if (@$sreg['fullname']) {
    $success .= " Your fullname is '".escape($sreg['fullname']).
        "'.";
}

$pape_resp = Auth_OpenID_PAPE_Response::fromSuccessResponse($response);

if ($pape_resp) {
    if ($pape_resp->auth_policies) {
        $success .= "<p>The following PAPE policies affected the authentication:</p><ul>";

        foreach ($pape_resp->auth_policies as $uri) {
            $escaped_uri = escape($uri);
            $success .= "<li><tt>$escaped_uri</tt></li>";
        }

        $success .= "</ul>";
    } else {
        $success .= "<p>No PAPE policies affected the authentication.</p>";
    }

    if ($pape_resp->auth_age) {
        $age = escape($pape_resp->auth_age);
        $success .= "<p>The authentication age returned by the " .
            "server is: <tt>".$age."</tt></p>";
    }

    if ($pape_resp->nist_auth_level) {
        $auth_level = escape($pape_resp->nist_auth_level);
        $success .= "<p>The NIST auth level returned by the " .
            "server is: <tt>".$auth_level."</tt></p>";
    }
}

} else {
    $success .= "<p>No PAPE response was sent by the provider.</p>";
}

include 'index.php';
}

header('Content-type: text/html; charset=UTF-8');
run();
?>
```

Скопируйте или обновите index.php файл на Вашем веб сервере. Откройте страницу ***index.php*** в браузере и проверьте правильность работы OPEN ID клиента с серверами Inbox.

Важно! У вашего веб сервера должно быть разрешено HTTP/HTTPS соединение с Open ID серверами Inbox - LOGIN.INBOX.LT.

Контакты для решения технических вопросов:

mailto: info@co.inbox.lt

тел.+371 67359626