Настройка сервера OpenID Connect

Данный компонент предназначен для аутентификации пользователей в web-приложениях системы по протоколу OpenID Connect и является обязательным для инсталляций системы под управлением ОС Linux и дополнительным для инсталляций под управлением ОС Windows.

> **OpenID Connect** (OIDC) - это протокол аутентификации и авторизации, разработанный на основе OAuth 2.0, который добавляет слой идентификации к протоколу OAuth. Он позволяет приложениям проверять идентичность пользователя и получать информацию о нем от провайдера идентификации (Identity Provider, IdP).

Установка сервера OIDC

- На ОС Windows сервер OIDC устанавливается из дополнительного пакета **IndeedCM.Oidc.Server-<*номер версии*>.x64.ru-ru.msi**.
- На ОС Linux сервер OIDC устанавливается вместе с сервером системы, являясь частью дистрибутива.

Настройка сервера OIDC

Отредактируйте файл **appsettings.json**, который находится по следующему пути:

- Для ОС Windows: *C:\inetpub\wwwroot\cm\oidc\appsettings.json*
- Для ОС Linux: */opt/indeed/cm/oidc/appsettings.json*

Откройте файл **appsettings.json** в текстовом редакторе, запущенном от имени Администратора:

1. По умолчанию, после установки, подключение к базе данных сервера OIDC настроено на использование **SQLite**. В этом случае данные сервера OIDC будут храниться локально, в каталоге */opt/indeed/cm/oidc/data*. Если требуется использовать базу данных Microsoft SQL или PostgreSQL, заполните секции "**defaultConnection**" и "**provider**".
   - **SQLite**. Внесение изменений не требуется. Секции имеют следующие значения:
     - "**defaultConnection**": "Filename=./data/oidc-server.sqlite3"
     - "**provider**": "sqlite"
   - **Microsoft SQL**. Для использования Microsoft SQL, создайте базу данных и настройте подключение к ней (в примере, для подключения к базе данных используется SQL аутентификация):
     - "**defaultConnection**": "Data Source=172.17.0.10;Initial Catalog=oidcdb;Persist Security Info=True;User ID=servicesql;Password=p@ssw0rd"
     - "**provider**": "mssql"
   - **PostgreSQL**. Для использования PostgreSQL, создайте базу данных и настройте подключение к ней:
     - "**defaultConnection**": "Host=172.17.0.11;Port=5432;Database=oidcdb; Username=servicepg;Password=p@ssw0rd"
     - "**provider**": "pgsql"
2. Заполните секции "**clientSecret**", "**redirectUris**" и "**postLogoutRedirectUris**" в "**clients**". В качестве клиентов выступают приложения Indeed CM: ManagementConsole, Self-Service и WebApi:
   - "**clientSecret**" - генерируется Мастером настройки Indeed CM для каждого приложения. Значения доступны в файле **oidc_secrets.json**, который будет создан в директории конфигурационных файлов Мастера настройки.

     Пример файла oidc_secrets.json

     ```
     {
       "managementConsoleClientSecret":
     "9d5d705e1cf5c12b2a5432c5a40c711e6505e939ca2d7cf0df48fc505c022329",
       "selfServiceClientSecret":
     "319e8b577563b7c6f27653d72b49659d16f06e0a150fd3a224002c778432319d",
       "webApiClientSecret":
     "9a9c56e5e8090c7fbcdffcc13537fc60d7a2f8547cc92131893e88cf08a7d5f9"
     }
     ```

   - "**redirectUris**". Вместо *REDIRECT_URL* укажите FQDN сервера Indeed CM: "https://REDIRECT_URL/cm/mc/signin-oidc".
   - "**postLogoutRedirectUris**". Вместо *POST_LOGOUT_URL* укажите FQDN сервера Indeed CM: "https://POST_LOGOUT_URL/cm/mc/signout-callback-oidc".

3. Заполните секцию "**signingCertificateThumbprint**", указав Отпечаток сертификата подписи (Thumbprint). В качестве сертификата подписи можно использовать SSL /TLS-сертификат, используемый для работы веб-сервера Indeed CM.

> В ОС Linux для работы .Net Core необходимо предоставить файл .**pfx** без пароля, содержащий сертификат, Отпечаток (Thumbprint), которого указан в поле "**signingCertificateThumbprint**" и закрытый ключ.

Для создания такого файла потребуются сертификат и закрытый ключ, которые были созданы на этапе настройки веб-сервера NGINX или Apache, или создайте их заново с помощью следующих команд (необходимо подставить имя импортированного pfx файла вместо PFXFILE) :

```
openssl pkcs12 -in PFXFILE.pfx -chain -nokeys | sed -ne '/-BEGIN CERTIFICATE/,/END CERTIFICATE/p' > SSL.crt
openssl pkcs12 -in PFXFILE.pfx -nocerts -out SSLencrypted.key
openssl rsa -in SSLencrypted.key -out SSL.key
rm -f SSLencrypted.key
```

Затем создайте директорию в домашнем каталоге пользователя, из под которого происходит настройка сервера Indeed CM, и создайте в этом каталоге файл .**pfx** без пароля с помощью утилиты **openssl**:

```
mkdir -p ~/.dotnet/corefx/cryptography/x509stores/my/
openssl pkcs12 -export -out ~/.dotnet/corefx/cryptography/x509stores/my/SSL.pfx -inkey SSL.key -in SSL.crt
```

4. Заполните секцию "**authentication**", указав метод аутентификации пользователей, который будет использовать сервер OIDC. В зависимости от расположения сервера системы и ОС, может быть указан следующий метод: **Windows** или **WindowsCustom**.

   - Для инсталляций системы под управлением ОС Windows доступны два метода: **Windows** или **WindowsCustom**.
   - Для инсталляций системы под управлением ОС Linux доступен только метод **WindowsCustom**.

   > **Windows** - используется, если сервер системы развернут на доменной рабочей станции под управлением ОС Windows. Для данного метода не требуется заполнение раздела "**ldap**".
   > **WindowsCustom** - используется, если сервер развернут вне домена или если требуется аутентификация пользователей в Web-приложения системы из каталога пользователей домена Active Directory, расположенного за пределами того домена в котором развернут сервер системы или с которым нет трастовых отношений. Для данного метода необходимо заполнить секцию "**ldap**".

5. Заполните секцию "**ldap**", если в качестве метода аутентификации пользователей выбран **WindowsCustom**:
   - **server** - имя хоста или IP-адрес LDAP-сервера.
   - **port** - обычно LDAP-сервер принимает входящие соединения на порт 389 по протоколам TCP или UDP. Для LDAP-сеансов, инкапсулированных в SSL, обычно используется порт 636.
   - **secureSocketLayer** - опция для включения или отключения SSL (Secure Sockets Layer) для защищенного соединения.
   - **verifyServerCertificate** - опция для включения или отключения проверки сертификата сервера при использовании SSL.
   - **authType** - тип аутентификации, который будет использоваться при подключении к LDAP-серверу.
   - **userName** - имя сервисной учетной записи для работы с каталогов пользователем в формате: "Имя домена(NetBIOS)\имя учетной записи".
   - **password** - пароль от сервисной учетной записи.
   - **domainDnsName** - DNS-имя домена.
   - **domainNetbiosName** - NetBIOS-имя домена.

     Чтобы узнать DNS-имя домена и NetBIOS-имя домена, выполните в командной строке:

     set USERDNSDOMAIN - выводит DNS-имя домена
     set USERDOMAIN - выводит NetBIOS-имя домена

     Пример заполненной секции, с подключением к серверу dc.demo.local по LDAP (389 порт):

```
"ldap": {
      "directories": [
         {
               "server": "dc.demo.local",
               "port": 389,
               "secureSocketLayer": false,
               "verifyServerCertificate": false,
               "authType": "Basic",
               "userName": "DEMO\\servicecm",
               "password": "Q1w2e3r4",
               "domainDnsName": "demo.local",
               "domainNetbiosName": "DEMO"
         }
    ]
},
```

6. Сохраните внесенные изменения в файл конфигурации сервера OpenID Connect в файле **appsettings.json**.

```
{
  "pathBase": "/cm/oidc",
  "culture": "ru",
  "certHeaderName": "x-ssl-client-cert",
  "connectionStrings": {
    "defaultConnection": "Filename=./data/oidc-server.sqlite3"
  },
  "database": {
    "provider": "sqlite"
  },
  "oidc": {
    "clients": [
      {
        "clientId": "ManagementConsole",
        "clientSecret": "9d5d705e1cf5c12b2a5432c5a40c711e6505e939ca2d7cf0df48fc505c022329",

        "displayName": "Management console",
        "type": "confidential",
        "consentType": "implicit",
        "permissions": [ "ept:authorization", "ept:token", "ept:logout", "gt:authorization_code", "rst:
code", "scp:profile", "scp:roles" ],
        "requirements": [ "ft:pkce" ],
        "redirectUris": [ "https://cm-core.demo.local/cm/mc/signin-oidc" ],
        "postLogoutRedirectUris": [ "https://cm-core.demo.local/cm/mc/signout-callback-oidc" ]
      },
      {
        "clientId": "SelfService",
        "clientSecret": "319e8b577563b7c6f27653d72b49659d16f06e0a150fd3a224002c778432319
d",
        "displayName": "Self-service",
        "type": "confidential",
        "consentType": "implicit",
        "permissions": [ "ept:authorization", "ept:token", "ept:logout", "gt:authorization_code", "rst:
code", "scp:profile", "scp:roles" ],
        "requirements": [ "ft:pkce" ],
        "redirectUris": [ "https://cm-core.demo.local/cm/ss/signin-oidc" ],
        "postLogoutRedirectUris": [ "https://cm-core.demo.local/cm/ss/signout-callback-oidc" ]
      },
      {
        "clientId": "WebApi",
```

```
      "clientSecret": "9a9c56e5e8090c7fbcdffcc13537fc60d7a2f8547cc92131893e88cf08a7d5f9",
      "displayName": "Web api",
      "type": "confidential",
      "consentType": "implicit",
      "permissions": [ "ept:introspection" ],
      "requirements": [],
      "redirectUris": [],
      "postLogoutRedirectUris": []
    },
    {
      "clientId": "WebApiClient",
      "clientSecret": null,
      "displayName": "Web api client",
      "type": "public",
      "consentType": "implicit",
      "permissions": [ "ept:token", "gt:password", "scp:profile", "scp:roles", "scp:webapi" ],
      "requirements": [],
      "redirectUris": [],
      "postLogoutRedirectUris": []
    }
  ],
  "signingCertificateThumbprint": "fed6d86ce6caa079f80d1b6c089cddf109d19c2d",
  "useEphemeralSigningKey": false,
  "disableTransportSecurityRequirement": false,
  "accessTokenLifetime": 43200,
  "identityTokenLifetime": 43200,
  "authentication": "Windows",
  "allowRememberLogin": false
},
"ldap": {
  "directories": [
    {
      "server": "DC_SERVER",
       "port": 389,
       "secureSocketLayer": false,
       "verifyServerCertificate": false,
       "authType": "Basic",
       "userName": "ACCOUNT_NAME",
       "password": "ACCOUNT_PASSWORD",
       "domainDnsName": "DOMAIN_DNS_NAME",
       "domainNetbiosName": "DOMAIN_NETBIOS_NAME"
```

```
    }
   ]
  },
  "Logging": {
   "LogLevel": {
    "Default": "Information",
    "Microsoft": "Warning",
    "Microsoft.Hosting.Lifetime": "Information"
   }
  },
  "AllowedHosts": "*"
}
{
  "pathBase": "/cm/oidc",
  "culture": "ru",
  "certHeaderName": "x-ssl-client-cert",
  "connectionStrings": {
   "defaultConnection": "Host=172.17.0.11;Port=5432;Database=oidcdb;Username=servicepsql;
Password=Q1w2e3r4"
  },
  "database": {
   "provider": "pgsql"
  },
  "oidc": {
   "clients": [
    {
     "clientId": "ManagementConsole",
     "clientSecret": "6e08e2f151262ddd8db961a18a8d7f3bb2ecaf1ccdf6037b9292a358b56f2ff6",
     "displayName": "Management console",
     "type": "confidential",
     "consentType": "implicit",
     "permissions": [ "ept:authorization", "ept:token", "ept:logout", "gt:authorization_code", "rst:
code", "scp:profile", "scp:roles" ],
     "requirements": [ "ft:pkce" ],
     "redirectUris": [ "https://astra-174-srv/cm/mc/signin-oidc" ],
     "postLogoutRedirectUris": [ "https://astra-174-srv/cm/mc/signout-callback-oidc" ]
    },
    {
     "clientId": "SelfService",
     "clientSecret": "48f410e3268d418a89b3d21073fa4962c816adb19899365c3472eb24b1a876
af",
```

```
      "displayName": "Self-service",
      "type": "confidential",
      "consentType": "implicit",
      "permissions": [ "ept:authorization", "ept:token", "ept:logout", "gt:authorization_code", "rst:
code", "scp:profile", "scp:roles" ],
      "requirements": [ "ft:pkce" ],
      "redirectUris": [ "https://astra-174-srv/cm/ss/signin-oidc" ],
      "postLogoutRedirectUris": [ "https://astra-174-srv/cm/ss/signout-callback-oidc" ]
    },
    {
      "clientId": "WebApi",
      "clientSecret": "e79b81d198478ccb852e8dd7f8ea62750e9626722942ba7dbb57a17119a7d5f
0",
      "displayName": "Web api",
      "type": "confidential",
      "consentType": "implicit",
      "permissions": [ "ept:introspection" ],
      "requirements": [],
      "redirectUris": [],
      "postLogoutRedirectUris": []
    },
    {
      "clientId": "WebApiClient",
      "clientSecret": null,
      "displayName": "Web api client",
      "type": "public",
      "consentType": "implicit",
      "permissions": [ "ept:token", "gt:password", "scp:profile", "scp:roles", "scp:webapi" ],
      "requirements": [],
      "redirectUris": [],
      "postLogoutRedirectUris": []
    }
  ],
  "signingCertificateThumbprint": "A85869C270CB2BDB113A28ADF24522A6EC55FF02",
  "useEphemeralSigningKey": false,
  "disableTransportSecurityRequirement": false,
  "accessTokenLifetime": 43200,
  "identityTokenLifetime": 43200,
  "authentication": "WindowsCustom",
  "allowRememberLogin": false
 },
```

```
  "ldap": {
    "directories": [
      {
        "server": "demo.local",
        "port": 389,
        "secureSocketLayer": false,
        "verifyServerCertificate": false,
        "authType": "Basic",
        "userName": "DEMO\\servicecm",
        "password": "Q1w2e3r4",
        "domainDnsName": "demo.local",
        "domainNetbiosName": "DEMO"
      }
    ]
  },
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedHosts": "*"
}
```