

Интеграционная платформа CONTACT NG Technology



Часть 5

Сервер приложений Contact NG Server

ОГЛАВЛЕНИЕ

1. ВВЕДЕНИЕ	1
2. АЛГОРИТМ ПЕРЕДАЧИ ЗАПРОСА ОПЕРАТОРУ СИСТЕМЫ И ПОЛУЧЕНИЯ ОТВЕТА ОТ ОПЕРАТОРА СИСТЕМЫ	2
3. ПРИМЕР ИСПОЛЬЗОВАНИЯ СЕРВЕРА ПРИЛОЖЕНИЙ	3
3.1. ФУНКЦИЯ <i>PROCESSREQUEST</i>	3
3.2. ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ ФУНКЦИИ <i>PROCESSREQUEST</i>	4
3.2.1. Пример 1. Вычисление комиссии	4
3.2.2. Пример 2. Неуспешное получение информации о переводе	4

1. Введение

В данном документе дано описание использования Сервера приложений Contact NG Server в качестве шлюза, обеспечивающего функции транспортного уровня:

- Прием запросов бизнесуровня от интеграционного приложения.
- Формирование запросов транспортного уровня, включая формирование ЭЦП
- Передачу запроса в операционный центр.
- Получение ответа из операционного центра, включая проверку ЭЦП.
- Передачу ответа бизнесуровня интеграционному приложению.
- Ведение журнала (логов).

Сервер приложений Contact NG Server входит в состав распространяемого Системой CONTACT программного обеспечения Contact NG и служит для отправки xml-запросов в операционный центр. Сервер приложений представляет собой HTTP-сервер и реализован в виде Windows-службы. Скачать дистрибутив можно по адресу: <https://contact.ruslavbank.com/progcont.nsf/ContactNGDistrRU>.

2. Алгоритм передачи запроса Оператору системы и получения ответа от Оператора системы

1. Клиентская часть ПО Contact NG формирует xml-запрос бизнес-уровня и передает его на обработку серверу приложений Contact NG Server.
2. Сервер приложений Contact NG Server подписывает полученный запрос (создает ЭЦП) и формирует xml транспортного уровня для передачи в операционный центр.
3. Сервер приложений Contact NG Server передает xml транспортного уровня на обработку в операционный центр.
4. операционный центр выполняет обработку запроса и формирует xml-ответ транспортного уровня.
5. Сервер приложений Contact NG Server, получив ответ, преобразует его в xml бизнес-уровня, проверяет ЭЦП и передает xml бизнес-уровня на обработку в клиентскую часть.

В качестве клиентской части может использоваться не только клиентская часть ПО Contact NG, но и любое другое приложение. Для отправки запроса необходимо использовать метод PUT http-сервера Contact NG Server. Для управления параметрами ЭЦП в тексте запроса необходимо использовать следующие атрибуты и значения:

- SignOut = "Yes" – запрос необходимо подписать (необходимо передавать данное значение во всех запросах)
- ExpectSigned = "Yes" – необходимо проверить подпись ответа (необходимо передавать для всех запросов, кроме запросов на получение справочников)
- ExpectSigned = "No" – проверять подпись ответа не нужно (необходимо передавать для запросов на получение справочников)

3. Пример использования сервера приложений

Ниже приведен пример модуля (Delphi 7), который содержит текст функции отправки xml-запроса в операционный центр, используя сервер приложений Contact NG Server.

3.1. Функция ProcessRequest

Функция ProcessRequest имеет три входных параметра:

- AHost – имя или IP-адрес компьютера, на котором установлен сервер приложений
- APort – порт, на котором работает сервер приложений
- Request – xml-запрос

ProcessRequest возвращает xml-ответ бизнес-уровня.

```
unit Transport;
interface
uses SysUtils, Classes, IdHTTP, Variants, msxml;
function ProcessRequest(AHost, APort: String; Request: IXMLDOMElement): IXMLDOMElement;
implementation
function ProcessRequest(AHost, APort: String; Request: IXMLDOMElement): IXMLDOMElement;
var
  Ms: TMemoryStream;
  Resp: TMemoryStream;
  s: string;
  List: TStringList;
  doc: IXMLDOMDocument;
  IdClient: TIdHTTP;
begin
  Result := nil;
  Ms := TMemoryStream.Create;
  List := TStringList.Create;
  IdClient := TIdHTTP.Create(nil);
  try
    Resp := TMemoryStream.Create;
    List.Text := Request.xml;
    List.SaveToStream(Ms);
    if Ms.Size < 1 then
      raise Exception.Create('Request is empty');

    Ms.Position := 0;
    Resp.Clear;
    s := 'http://' + AHost + ':' + APort;

    IdClient.DoRequest(hmPut, s, Ms, Resp);

    if Resp.Size > 0 then
      try
        SetLength(s, Resp.Size);
        Resp.Position := 0;
        Resp.Read(s[1], Resp.Size);
        doc := CoDOMDocument.Create;
        doc.loadXML(s);
        Result := doc.documentElement;
      finally
        s := '';
      end;
  end;
```

```
if Result <> nil then
begin
  if VarToStr(Result.getAttribute('RE')) <> '0' then
    raise Exception.Create(VarToStr(Result.getAttribute('ERR_TEXT')));
  end;
finally
  FreeAndNil(Resp);
  FreeAndNil(Ms);
  List.Free;
  IdClient.Free;
end;
end;

end.
```

3.2. Примеры использования функции ProcessRequest.

3.2.1. Пример 1. Вычисление комиссии

```
var
  Request, Response: IXMLDOMElement;
begin
  Request := CoDOMDocument.Create.createElement('REQUEST');
  Request.setAttribute('OBJECT_CLASS', 'TMoneyOrderObject');
  Request.setAttribute('ACTION', 'CalcFee');
  Request.setAttribute('POINT_CODE', 'XXXX'); // XXXX – код участника в системе Contact
  Request.setAttribute('LANG', 'RU');
  Request.setAttribute('TRANSACTION_AMOUNT', '100.00');
  Request.setAttribute('TRANSACTION_CUR', 'RUR');
  Request.setAttribute('ENTER_CUR', 'RUR');
  Request.setAttribute('TO_BANK_CODE', 'MRSB');
  Request.setAttribute('SignOut', 'Yes');
  Request.setAttribute('ExpectSigned', 'Yes');
  Response := ProcessRequest('localhost', '8080', Request);
  ShowMessage(Response.xml);
end;
```

3.2.2. Пример 2. Неуспешное получение информации о переводе

```
var
  Request, Response: IXMLDOMElement;
begin
  Request := CoDOMDocument.Create.createElement('REQUEST');
  Request.setAttribute('OBJECT_CLASS', 'TMoneyOrderObject');
  Request.setAttribute('ACTION', 'Get');
  Request.setAttribute('DOC_ID', '-1');
  Request.setAttribute('IN_OUT', 'O');
  Request.setAttribute('POINT_CODE', 'XXXX'); // XXXX – код участника в системе Contact
  Request.setAttribute('LANG', 'RU');
  Request.setAttribute('SignOut', 'Yes');
  Request.setAttribute('ExpectSigned', 'Yes');
  Response := ProcessRequest('localhost', '8080', Request);
end;
```

Если все сделано правильно, то на экране появится сообщение с ошибкой: “Перевод не найден”.