

Передача дополнительной информации о пользователе (пользовательский контекст)

Оглавление

- # Модель данных пользовательского контекста UserDeviceContext
- # Настройки UIDM, управляющие функциональностью передачи атрибутов пользовательского контекста
 - # Настройка создания дополнительных атрибутов пользовательского контекста
 - # Настройки добавления атрибутов пользовательского контекста в клеймы токена
 - # Настройки добавления атрибутов пользовательского контекста в протоколируемое событие (аудит)
- # Примеры конфигурации
- # Примеры передачи параметров пользовательского контекста
 - # Передача параметров контекста в протоколе M2M
 - # Передача параметров контекста в протоколе Login API
 - # Передача параметров пользовательского контекста при подписании документов электронной подписью

Пользовательский контекст — сведения об аутентифицирующемся пользователе, дополнительные к его аутентификационным данным. Пользовательский контекст собирается UIDM в момент аутентификации пользователя, и может включать в себя:

- свойства устройства пользователя (номер телефона, MAC-адрес устройства);
- его местоположение (по IP-адресу и геопространственным данным устройства);
- имя приложения на устройстве пользователя и данные о таком приложении;
- внутренний, внешний IP-адреса;
- метки времени и т. д.

Часть этой информации может быть получена от клиента UIDM (например, от мобильного или Web-приложения), а часть определяется самим сервером UIDM.

Пользовательский контекст может быть использован:

- для принятия решения об аутентификации (например, разрешать аутентификацию только с разрешенного IP-адреса);
- для протоколирования в базе данных аудита;
- для передачи во внешние системы (например, в системы противодействия мошенничеству (антифрод)).

UIDM может быть настроен помещать данные из пользовательского контекста в клеймы токена доступа, выпускаемого при успешной аутентификации. Такие данные могут быть прочитаны из токена доступа защищаемым ресурсом, и на их основе для пользователя (предоставившего токен с такими данными контекста), может быть принято решение о дополнительном ограничении или предоставлении доступа к данным защищаемого ресурса.

Пользовательское приложение или сервис могут предоставлять обновлённые сведения о пользовательском контексте, которые будут заменять предыдущие. Однако, такие обновлённые сведения будут помещены в клеймы нового токена только при его выпуске.

Дополнительные параметры пользователя можно передавать в запросах на аутентификацию протоколов OAuth2 и M2M. Пример запроса приведен в разделе [Пример передачи параметров пользовательского контекста](#).

Текущая реализация протоколов OAuth2 и M2M позволяет клиенту переопределять атрибуты контекста при вызове каждого последующего запроса в сценарии аутентификации. При этом для атрибутов, которые были переданы в очередном запросе, значения в контексте будут заменены на полученные новые. Остальные атрибуты, полученные ранее, останутся без изменений.

Модель данных пользовательского контекста UserDeviceContext

Собранная дополнительная информация о контексте пользователя внутри UIDM собирается в модель данных UserDeviceContext.

Атрибуты контекста могут быть переданы в клеймы токена, выпускаемого при аутентификации, и сохраняются в поле `data` протоколируемого события создания токена.

Структура модели данных UserDeviceContext

```
{
  "deviceDeterminedLocationContext": {
    "coordinates": {
      "lat": {
        "valueDegrees": double
      },
      "lon": {
        "valueDegrees": double
      },
      "height": {
        "valueMeters": double
      }
    },
    "city": {
      "cityId": string,
      "nameNat": string,
      "nameInt": string
    },
    "region": {
      "regionId": string,
      "nameNat": string,
      "nameInt": string
    },
    "country": {
      "isoCode": string,
      "nameNat": string,
      "nameInt": string
    }
  },
  "geoIpDeterminedLocationContext": {
    "coordinates": {
      "lat": {
        "valueDegrees": double
      },
      "lon": {
        "valueDegrees": double
      },
      "height": {
        "valueMeters": double
      }
    },
    "city": {
```

```

        "cityId": string,
        "nameNat": string,
        "nameInt": string
    },
    "region": {
        "regionId": string,
        "nameNat": string,
        "nameInt": string
    },
    "country": {
        "isoCode": string,
        "nameNat": string,
        "nameInt": string
    },
    "serverDeterminedIpNetworkContext": {
        "remoteAddress": string
    },
    "deviceDeterminedNetworkContext": {
        "mac": {
            "macAddress": string
        },
        "innerIp": {
            "remoteAddress": string
        },
        "extIp": {
            "remoteAddress": string
        }
    },
    "userAgentContext": {
        "userAgentString": string,
        "deviceType": string,
        "deviceBrand": string,
        "deviceModel": string,
        "osFamily": string,
        "osNameVersion": string,
        "browserType": string,
        "browserFamily": string,
        "browserNameVersion": string
    },
    "mobileDeviceContext": {
        "deviceId": string,
        "deviceLocale": string,
        "deviceOS": string,
        "deviceOSVersion": string,
        "appVersion": string,
        "deviceRoot": boolean,
        "deviceName": string
    },
    "additionalContextAttributes": {
        "customParam1": string,
        "customParam2": string,
        "customParam3": string
    }
}

```

deviceDeterminedLocationContext

Передается клиентом UIDM.

Содержит:

- географические координаты — долготу и широту, выраженную в градусах. Отрицательные значения

соответствуют западному и южным полушариям соответственно;

- высоту над уровнем моря в метрах.

geoIpDeterminedLocationContext

Информация о местоположении, **определяемая сервером UIDM** по данным базы Geolp.

serverDeterminedIpNetworkContext

Сетевой контекст, **определяемый сервером UIDM** на основании HTTP-заголовков запроса. `ipAddress` может передаваться в стандартном формате IPv4 или IPv6.

deviceDeterminedNetworkContext

Передается клиентом UIDM.

Сетевой контекст, определяемый клиентом (устройством).

Передается с параметрами `mac`, `extIp`, `innerIp` запроса. IP-адреса могут передаваться в стандартных форматах IPv4 или IPv6.

userAgentContext

Определяется сервером UIDM.

Информация об устройстве пользователя, полученная сервером UIDM на основе анализа HTTP заголовка UserAgent.

mobileDeviceContext

Передается клиентом UIDM.

Контекст, передаваемый с клиента (устройства) с данными по мобильному устройству.

Может быть передан с параметром `device_info` запроса в виде JSON-строки:

```
{
  "deviceId": string,
  "deviceLocale": string,
  "deviceOS": string,
  "deviceOSVersion": string,
  "appVersion": string,
  "deviceRoot": boolean,
  "deviceName": string,
}
```

additionalContextAttributes

Передается клиентом UIDM.

Кастомные атрибуты контекста передаются с одноименными параметрами запроса и определяются конфигурацией UIDM.

Настройки UIDM, управляющие функциональностью передачи атрибутов пользовательского контекста

Настройка создания дополнительных атрибутов пользовательского контекста

```
com.rooxteam.sso.userContext.additionalAttributes.{custom_attr_name}.max_length={length}
```

Разрешить использовать дополнительный параметр контекста с именем `{custom_attr_name}` и задать его максимально возможную длину `{length}` (целое число от `1` до `2,147,483,647`).

В дополнительные атрибуты контекста будут добавлены только параметры, для которых добавлена данная конфигурация.

Значения переданных параметров будут обрезаться до максимально возможной длины строки, установленной данной конфигурацией.

Значение по умолчанию

Если ключ не задан, дополнительный параметр с именем `{custom_attr_name}` не передаётся.

Настройки добавления атрибутов пользовательского контекста в клеймы токена

```
com.rooxteam.sso.userContext.claim_properties
```

Задаёт имя клейма токена и соответствующий ему атрибут пользовательского контекста, который будет помещён в этот клейм.

Значения передаются в виде разделённого запятыми списка `{attr_name}={nested_property}`, где `{attr_name}` — наименование клейма в токене, `{nested_property}` — полное (с учётом пути атрибута в модели данных [UserDeviceContext](#)) имя атрибута контекста, которое надо передать в клейм.

Например, чтобы передать в клейм токена атрибут с именем `mac` и значением MAC-адреса из `deviceDeterminedNetworkContext`, нужно задать соответствие `mac=deviceDeterminedNetworkContext.mac.macAddress`.

Значение по умолчанию

Если параметр не задан, никакие атрибуты пользовательского контекста в токен не помещаются.

```
com.rooxteam.sso.userContext.claim_name
```

Устанавливает наименование клейма токена, в который будут попадать атрибуты [UserDeviceContext](#), заданные настройкой `com.rooxteam.sso.userContext.claim_properties`

Значение по умолчанию

`device_ctx`

Настройки добавления атрибутов пользовательского контекста в протоколируемое событие (аудит)

Любой атрибут контекста можно передать в аудит (поле `data`) или внешнюю систему (например, антифрод). Наименование параметра и атрибутов устанавливается конфигурацией, аналогичной добавлению клейма в токен.

```
com.rooxteam.sso.userContext.audit_name
```

Устанавливает наименование атрибута в поле `data` аудита, в который будут попадать атрибуты пользовательского контекста, заданные настройкой `com.rooxteam.sso.userContext.claim_properties`

Возможные значения

Строка, способная быть именем XML-тэга

Значение по умолчанию

`device_ctx`

```
com.rooxteam.sso.userContext.audit_properties
```

Устанавливает соответствие между параметрами атрибута для аудита и атрибутами контекста.

ВАЖНО

Данный параметр работает только в [протоколе M2M](#)! В протоколе Login-API в аудит попадают атрибуты контекста из клеймов токена. См. конфигурационный ключ

```
com.rooxteam.sso.userContext.claim_properties
```

Значения передаются в виде списка `{attr_name}={nested_property}`, где `{attr_name}` - наименование параметра атрибута аудита, `{nested_property}` — полное с учетом пути имя атрибута контекста, которое надо сохранить в аудит

Например, чтобы сохранить в аудит параметр с именем `mac` и значением MAC-адреса из

`deviceDeterminedNetworkContext`, нужно задать соответствие

```
mac=deviceDeterminedNetworkContext.mac.macAddress
```

Список разделяется запятыми.

Значение по умолчанию

Если параметр не задан, никакие атрибуты в аудит не передаются.

Примеры конфигурации

Задача

Необходимо передать в токен в клейм с именем `devctx` значения пользовательского контекста:

- MAC-адрес клиента (`deviceDeterminedNetworkContext.mac.macAddress` в [модели данных](#) пользовательского контекста)
- внутренний IP-адрес клиента (`deviceDeterminedNetworkContext.innerIp.remoteAddress` в модели данных пользовательского контекста)
- внешний IP-адрес клиента (`deviceDeterminedNetworkContext.externalIp.remoteAddress` в модели данных пользовательского контекста)
- дополнительный параметр пользовательского контекста (`additionalContextAttributes.customParam1` в модели данных пользовательского контекста)

Решение

1. Установим, что требуемые поля пользовательского контекста должны быть помещены в токен в клейм `devctx` :

```
com.rooxteam.sso.userContext.claim_name=devctx
```

2. Включим возможность использования дополнительного произвольного параметра `additionalAttributes.customParam1`, одновременно задав ему максимальную длину `10`.

```
com.rooxteam.sso.userContext.additionalAttributes.customParam1.max_length=10
```

3. Установим, что

- поле клейма `mac` будет содержать данные поля `deviceDeterminedNetworkContext.mac.macAddress` модели данных пользовательского контекста,
- поле клейма `innerIp` будет содержать данные поля `deviceDeterminedNetworkContext.innerIp.remoteAddress` модели данных пользовательского контекста,
- поле клейма `extIp` будет содержать данные поля `deviceDeterminedNetworkContext.externalIp.remoteAddress` модели данных пользовательского контекста,
- поле клейма `customParam1` будет содержать данные поля `deviceDeterminedNetworkContext.additionalContextAttributes.customParam1` модели данных пользовательского контекста:

```
com.rooxteam.sso.userContext.claim_properties=mac=deviceDeterminedNetworkContext.mac.macAddress,innerIp=deviceDeterminedNetworkContext.innerIp.remoteAddress,extIp=deviceDeterminedNetworkContext.extIp.remoteAddress,customParam1=additionalContextAttributes.customParam1
```

В результате такой настройки в клеймы токена переданные значения попадут в следующем виде:

```
{
  ...
  "devctx": {
    "mac": "01:23:45:67:89:ab",
    "innerIp": "192.168.0.42",
    "extIp": "179.253.12.11",
    "customParam1": "value1"
  }
  ...
}
```

Примеры передачи параметров пользовательского контекста

Передача параметров контекста в протоколе M2M

Пример отправки параметров пользовательского контекста в составе второго запроса в цепочке аутентификации (на шаге аутентификации пользователя по логину и паролю) по протоколу M2M

```
POST /sso/oauth2/access_token HTTP/1.1
Host: <sso_host>
Accept: application/json
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

client_id=<client_id>&
client_secret=<client_secret>&
scope=<scope>&
grant_type=urn:roox:params:oauth:grant-type:m2m&
realm=%2Fcustomer&
service=dispatcher&
execution=<execution>&
username=<username>&
```

```
password=<password>&  
mac=<mac>&  
innerIp=<innerIp>&  
extIp=<extIp>&  
device_info=<device_info>&  
custom_param=<custom_param>&  
_eventId=next
```

ЗАМЕТКА

В настоящем документе информация приведена для справки, подробнее см. спецификацию на [протокол M2M](#).

Базовые параметры запроса

`<sso_host>`

Базовый адрес сервера WebSSO (например, `sso.rooxteam.com`)

`<client_id>`

Идентификатор клиента (например `selfcare`)

`<client_secret>`

Пароль клиента

`<scope>`

Область действия (scope) OAuth2.0 в кодировке UTF-8 (**опционально, регистрозависимо**)

`<grant_type>`

Способ авторизации пользователя. Всегда используется значение `urn:roox:params:oauth:grant-type:m2m`

`<realm>`

Группа пользователей WebSSO. Всегда используется значение `%2Fcustomer` , которое является uri-encoded значением `/customer`

`<execution>`

Идентификатор предсессии аутентификации. Значение берется из предыдущего ответа сервера

`<service>`

Имя цепочки аутентификации, всегда `dispatcher`

`<username>`

Имя пользователя (номер телефона)

`<password>`

Пароль пользователя

`<_eventId>`

Идентификатор действия, всегда `next`

Параметры пользовательского контекста

<mac>

MAC-адрес сетевого адаптера (поле `deviceDeterminedNetworkContext.mac.macAddress`)

<innerIp>

Внутренний IP-пользователя пользователя (поле `deviceDeterminedNetworkContext.innerIp.remoteAddress`)

<extIp>

Внешний IP-адрес пользователя (поле `deviceDeterminedNetworkContext.extIp.remoteAddress`)

<device_info>

JSON-строка с информацией об устройстве пользователя (поле `mobileDeviceContext`)

<custom_param>

Дополнительный произвольный параметр контекста (поле `additionalContextAttributes.custom_param`)

Передача параметров контекста в протоколе Login API

Настройка имени объекта в событии аудита

```
com.rooxteam.sso.userContext.audit_name=user_audit_ctx
```

Настройки параметра `deviceId` , соответствующие дополнительному произвольному атрибуту

`additionalContextAttributes.deviceId`

```
com.rooxteam.sso.userContext.additionalAttributes.deviceId.max_length=500
com.rooxteam.sso.userContext.audit_properties=deviceId=additionalContextAttributes.deviceId
```

Пример передачи параметров пользовательского контекста на этапе ввода пользователем логина и пароля (Login API)

```
POST https://example.com/sso/auth/login-widget-router
Accept: application/json
Content-Type: application/x-www-form-urlencoded
Cookie: RX_SID=089DF75FE057960105525B5413B9DE46;
gotoOnFail=https://example.com/sso/oauth2/authorize?
response_type=code&realm=%2Fcustomer&client_id=smeportal&redirect_uri=https%3A%2F%2Fexample.com%2Foauth2-
consumer%2Fauthorize&state=client_id%3Dsmeportal%26goto%3Dhttps%253A%252F%252Fexample.com%
252F%252Fsecure%252Fflk.jsp%26gotoOnFail%3Dhttps%253A%252F%252Fexample.com%252F%252F%252Fsecure%252Fflk.jsp
deviceId=custom_param_value&
_eventId=next&
username=tester2&
```

```
password=password&[redacted]  
execution=0b6e52f9-be27-42de-84ae-917886f30a44...
```

Ответ сервера об успешной аутентификации пользователя

```
HTTP/1.1 200 OK
```

```
{  
  "step": "redirect",  
  "location": "/sso/auth/complete"  
}
```

При таких настройках при успешной аутентификации пользователя в событиях аудита будет сохранено событие типа `sso.auth.success`, а в поле `data` события будут сохранены следующие данные:

```
<?xml version="1.0"?>  
<data key="data" type="object">  
  <user_audit_ctx key="user_audit_ctx" type="object">  
    <deviceId key="deviceId" type="text"><![CDATA[custom_param_value]]></deviceId>  
  </user_audit_ctx>  
  <realm key="realm" type="text"><![CDATA[customer]]></realm>  
  <issuer key="issuer" type="object">  
    <id key="id" type="text"><![CDATA[sso_____0be41f1f-c1d0-44ca-8afe-4d875aea0870]]></id>  
    <type key="type" type="text"><![CDATA[PRINCIPAL]]></type>  
  </issuer>  
</data>
```

Параметры пользовательского контекста могут быть переданы и в других сценариях.

Передача параметров пользовательского контекста при подписании документов электронной подписью

Пример передачи параметров пользовательского контекста на этапе запроса одноразового пароля (OTP) для подписания документов электронной подписью

```
POST https://example.com/sso/oauth2/access_token  
Content-Type: application/x-www-form-urlencoded  
  
client_id=smeportal_m2m&  
client_secret=password&[redacted]  
access_token=eyJraWQiOiJwdWJsaWN0b2t1bjpqd2UiLCJj...&  
grant_type=urn%3Aroox%3Aparams%3Aoauth%3Agrant-type%3Am2m&  
realm=%2Fcustomer&[redacted]  
service=sign_document_batch&  
category=otp-sign&[redacted]  
deviceId=custom_param_value&  
operation=...
```

В случае успешного подписания документа в событиях аудита будет также сохранено событие типа `sso.auth.success` с данными, идентичными указанным [выше](#).

