**API Банка v0.01**

**Изменения**

**v0.01**

* Скорректирован раздел «Формат запроса и ответа»;
* Обновлены возможные коды ответа сервера;
* Добавлен раздел «Пример отправки зашифрованного запроса и расшифровка ответа сервера»;
* Сообщения об ошибках приведены в единый формат;
* Документация полностью на русском;
* Теперь заголовок «Content-type» не нужен если нет тела запроса;
* Время жизни токена обновляется, после отправки запроса;
* СМС код создается при помощи OTP;

**Содержание**

[**Формат запроса и ответа** 4](#_Toc35617531)

[**Описание ошибок выполнения** 6](#_Toc35617532)

[**Пример отправки зашифрованного запроса и расшифровка ответа сервера** 8](#_Toc35617533)

[**Метод Получения сертификата** 11](#_Toc35617534)

[**Метод Регистрации** 12](#_Toc35617535)

[**Метод Подтверждения телефона** 13](#_Toc35617536)

[**Метод Повторной отправки смс на телефон** 14](#_Toc35617537)

[**Метод Создание кода доступа** 15](#_Toc35617538)

[**Метод Аутентификации** 16](#_Toc35617539)

[**Метод Получения списка карт пользователя** 17](#_Toc35617540)

[**Метод Получения информации по карте** 18](#_Toc35617541)

[**Метод Получения истории по всем картам пользователя** 19](#_Toc35617542)

[**Метод Получения истории по карте** 20](#_Toc35617543)

# **Формат запроса и ответа**

**Формат запроса к серверу.**

Заголовки запроса должны содержать (кроме вызова метода Получения сертификата):

|  |  |
| --- | --- |
| Имя ключа | Описание |
| X-Request-Signature | JSON Web Signature (JWS) подписанный сертификатом отправителя запроса c помощью алгоритма RS256 |
| Content-type | Должен содержать “application/jwt” |

JWS заголовки состоят из:

|  |  |  |
| --- | --- | --- |
| Имя ключа | Описание | Пример |
| alg | Алгоритм, используемый для подписи | “RS256” |
| typ | Тип токена | “JWT” |
| x5c | Сертификат X.509 отправителя запроса | “c2c60b99044f12f1d6a…” |

JWS полезная нагрузка состоит из:

|  |  |  |
| --- | --- | --- |
| Имя ключа | Описание | Пример |
| method | HTTP метод запрос | “POST” |
| body | Хэш тела запроса, полученный с помощью алгоритма SHA256 | “c2c60b99044f12f1d6a…” |

Тело запроса содержит JSON Web Encryption (JWE), зашифрованный при помощи сертификата сервера, в полезной нагрузке которого содержатся параметры запроса.

**Формат ответа сервера.**

Заголовки ответа содержат:

|  |  |
| --- | --- |
| Имя ключа | Описание |
| X-Response-Signature | JWS подписанный сертификатом сервера c помощью алгоритма RS256 |
| Content-type | Содержит “application/jwt” (кроме вызова метода Получения сертификата) |

JWS заголовки состоят из:

|  |  |  |
| --- | --- | --- |
| Имя ключа | Описание | Пример |
| alg | Алгоритм, используемый для подписи | “RS256” |
| typ | Тип токена | “JWT” |

JWS полезная нагрузка состоит из:

|  |  |  |
| --- | --- | --- |
| Имя ключа | Описание | Пример |
| method | HTTP метод запрос | “POST” |
| body | Хэш тела запроса, полученный с помощью алгоритма SHA256 | “c2c60b99044f12f1d6a…” |

JWE заголовки состоят из:

|  |  |  |
| --- | --- | --- |
| Имя ключа | Описание | Пример |
| alg | Алгоритм, используемый для подписи | “RSA1\_5” |
| enc | Алгоритм, используемый для шифрования полезной нагрузки | “A128CBC-HS256” |
| typ | Тип токена | “JWT” |

Тело ответа содержит JWE, зашифрованный сертификатом отправителя, в полезной нагрузке которого размещены результаты метода.

# **Описание ошибок выполнения**

**HTTP Code: 400 – Bad request.**

Причины:

* Некорректные параметры запроса;

Примеры ответа сервера:

1. **JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| messages | Array (String) | " PAN length should be 19.",  "Phone length should be 11." |

**HTTP Code: 401 – Unauthorized.**

Причины:

* Отсутствует необходимый токен;
* Некорректный токен;

Пример ответа сервера: отсутствует;

**HTTP Code: 460 – JWE Error.**

Причины:

* Не удалось декодировать тело JWE;
* Запрос не зашифрован;

Пример ответа сервера: отсутствует;

**HTTP Code: 461 – JWS Error.**

Причины:

* Отсутствует Header “X-Request-Signature”;
* Некорректная подпись запроса;

Пример ответа сервера: отсутствует;

**HTTP Code: 462 – Missing Content Type.**

Причины:

* Отсутствует Header “Content-type”;

Пример ответа сервера: отсутствует;

**HTTP Code: 500 – Server error.**

Причины:

* Неполадки в работе сервера;

Пример ответа сервера: отсутствует;

# **Пример отправки зашифрованного запроса и расшифровка ответа сервера**

1. Создаем JWE подписанный сертификатом сервера.

Полезная нагрузка:

{

"phone": "79916792924",

"PAN": "5100000000000412"

}

Результат:

eyJhbGciOiJSU0ExXzUiLCJlbmMiOiJBMTI4Q0JDLUhTMjU2Iiwia2lkIjoiQVByVG1VeHlXUVZVOFBuY1VGWXh4WWp6MlVvIiwidHlwIjoiSldUIn0.wRRfBB1xUC0gWLP2yQJt4iPXo-KFayPQ8b\_vdk0xHDcpc2PVjQyVez9ga4nb\_eeFEi2udZWMQ90cLTJR5vSseVJiR8o9ZB6tBPAsxHA7owENsUq9jvnAz7KQKZ\_7xSj321lndDnoIzaGT5Fqc3AyCfXIaRLEebGgGPyRL0oyXyEydkkhHSsga3-NqTRz-Z0MeqzeA8AbPPgnLen8Ot7tToEs5E\_SjnRaHQhLvt70gG8AhBV4KmG61sOvW-958AOZhnNVTZOwEKrYW3UHXxQG9URr1NJfMVF7W8zLlxAjeSzLL2kXKTnpEBK-mWDORSmOqkXI5mHgkFyWMfyY0RN4tg.qOij3KjA7qYTndk2l7-vYQ.p\_R1-HsVcxXTWjAw5uVgJhuEAsODP5BLbXbyvoHybUNTNRMc0B--nhmicgLQeaTTK7Pw\_TR9qUPeqNgy-USxT6I3qiefv-8pnIZ8CiZUZ6jRMmbNHKjYM07ujYRp0vSQHkbr8O99J6N0u2OQ9-OMRw.8veFCPq75HHDwtPclYDv0Q

1. Получаем хэш из JWE с помощью алгоритма SHA256.

Результат: b304d15a0d5dbf22dbf137e41efc49e33c7098b7949c58df47926a6a09ce337a

1. Создаем JWS подписанный сертификатом отправителя.

JWS заголовки состоят из:

|  |  |
| --- | --- |
| Имя ключа | Значение |
| alg | “RS256” |
| typ | “JWT” |
| x5c | “c2c60b99044f12f1d6a…” |

Полезная нагрузка:

{

"method":"POST",

"body":" b304d15a0d5dbf22dbf137e41efc49e33c7098b7949c58df47926a6a09ce337a"

}

Результат:

.eyJtZXRob2QiOiJQT1NUIiwiYm9keSI6ImIzMDRkMTVhMGQ1ZGJmMjJkYmYxMzdlNDFlZmM0OWUzM2M3MDk4Yjc5NDljNThkZjQ3OTI2YTZhMDljZTMzN2EiLCJpc3MiOiJQaG9uZSIsImF1ZCI6IlBob25lIn0.hcpkCddyNIVPoPWeHm08cK3Wl07dzVVcHIbvj2ZvtbYL-lEdRoUJ52BQIztBUUsBbFx5Z6HWu179RwsMaYmUNNHROmwnZrrows5VAdUkbBrXuwcevCZFJ8fDDhtnjtdQKNi8y3hlfcbbUkrY-3uIX59GFJCiHnFwfGDU0Y5WfHaW862LiD-XjBTnZnYIcGkyJv9NmsJk3R\_WxjAkyPpd4qAxfxstuQz5rPmRQ0fvxI0ZmigKXrXCg9Kb2jVktc\_aPHsgG1G54dtKUxCLYXdHxZ0wO7mcGH\_zE71zCv\_1V91V4mOjsDjkfkNwMsgNCoGL-brFbfYDRUZ6BqTmr1k-RA

1. Отправляем запрос серверу.

Заголовки запроса:

|  |  |
| --- | --- |
| Имя ключа | Значение |
| X-Request-Signature | (JWS полученный в пункте 3) |
| Content-type | ”application/jwt” |

Тело запроса состоит из JWE полученного на пункте 1

Результат запроса:

eyJhbGciOiJSU0ExXzUiLCJlbmMiOiJBMTI4Q0JDLUhTMjU2Iiwia2lkIjoiQTUxMUE4NDU2QUE2NzAyNDY1RTZDMDIxOUZEQzNFNkY4MDg4MjI5NSIsInR5cCI6IkpXVCJ9.c60AFwn4xEHhTc1h9Qktd02kGnxT7Fmxy6u1XB\_CA8\_y-5eETxCkH3rAGsnkxP2EG3xU0s7U7SgBYTpLh3azSqGK\_AxKqQ4103LzeNaiWWg-rqa2rh1HdAbhheHRHP5FgBjOVihIxRPm0y7478Y1deSYi7pLbFrt60DWp7qTfSHw\_FRvh0pfEz2N85R2fOxJ0\_cAjdJwpGSag3rA8dvFIELI4oJiMVaCePG2tMH\_9y2lcAdYSbmw2pI90Wvp3txsoCJ1TjMymb0DgCdtKLEWFEbNOVkSvFbtnGasOFSo3Q40e1RH6a2zonX7mUVXo8dDGjEXHcY25hF4pkKlWwUIRA.nUv1q1bo\_MQpLmaDWRp9Bg.bKUzVkDZ-AurwDiB0StTsUGpOeDZliP5sqGhFdzghUYcjBGBtQZBXciqxnovHjnVdQtkBlETnr0XsRGe-Tqu-3JUOXHDtY\_v2qB2vtxP6DZoGWJTJWJL1SYPx\_TPCNgpT9WFS8FuTzqGgD0jWpQL-YCQCLEYWsglSP4IijbAF0XCPF6IlV5wVivaYHrHr8FdNqXj3g72747cMDPfvIybd-qIvq\_ygvloCpq83TBMnGjNZHVZwiH7-33Nn3CGsxv04MhXKBamxCb6\_It1DTacSqUO87AJmuHvUGfNK0wrGGHknCC9o9egcpd1RG\_KwCygfUmiZzt-qOhRzUxkevA\_5X9RYWUSqc55y11n0axdqgcGIhmcL7JtXr7Liu3\_kOZcmaqY5qLLAY5HP50pBLkm9uoLvywep9ToemnyJj6OILiQTO31a1iDXAlURv6EqaLowI9YGp3NC\_2DmMCbRuDtmvlvs61W6UlbbPYY-6VEUtGAAWwiHTWGzwzL9UzE8h8e7brfU2utrtVDZ-VTKpwYgXzydEXPkbXXM9FcS-uGfvgHl-02G2AUUkh0LHvN2ip5VXr1fTaEkF7EDVYecJlIfUkoan3VtQFCnW-M\_pNXIcBHQm\_FgBQ1paxCXSbf0tPp2valaawBfOE9\_8BK8i3RC8VaNxZumyvQXGJYmxzn3GM7Hx5zdbyv0HpkOjoyterRnBm5DpV5l44dd7A9wduAknbYAsyQT-hR\_\_VcAa05aj7B94xuXkmtUBRsVMESQ8xsInmsYMtv8N7WOHun8CnXNCz2-P7ZNUrpC6ftKL7wjf1ewDJug4wBBFGHY10sErdRFJqhWTDrS1e\_6ChLVvcBkOTwl1sP2f2UE2bx1GXXhfDC1UcZNCa3bh-x1Vsww0pHgZRcqq2YVXzZeJtBCQ-qMrT8\_MyLbZbIWUtmOnpR9fJnyD8UW55cD8gu-5eTGeU33SrJMYFVA25V3T3s4dvhmuXHVgT93CvRQTA7mP\_aIKEdISxKRpFnCyAcGrP3DYXP58Vi0lsBuhQcDSla-T6a8MzFDKNej-LEFZBh5JIygiwJm4PPzP4Q1lhb0-kx0fcH\_9v9tukzjL1z34J36-MOTuclpFBnKb9E-NPnqtl4afl0sanXgLJAjJtXe2NIh71IyOU7QZ\_9UYVtmKdyh4OhX5pn0bome5R1-\_IUvxwBnL3MSGCj\_Gu3b6YI1co-8mS3.CKwjJLSGgCpJJTrmmogVQQ

1. Расшифровываем полученный ответ.

{

"alg":"RSA1\_5",

"enc":"A128CBCHS256"

"typ":"JWT"

}.

{

"message": "An SMS with a verification code has been sent to your number.",

"sessionToken":"eyJhbGciOiJSUzI1NiIsImtpZCI6IjAwRkFEMzk5NEM3MjU5MDU1NEYwRjlEQzUwNTYzMUM1ODhGM0Q5NEEiLCJ0eXAiOiJKV1QifQ.eyJpZFVzZXIiOiIyMyIsInR5cGUiOiJTZXNzaW9uIiwiaXNzIjoiQmFua0FwaSIsImF1ZCI6IkJhbmtBcGkifQ.JHD5ey4pwY2dz2uxkZcyEA4lR1Xkod0nur4G-HLBsLSZhawrD5QDF7riPpFLfgmNVvND7qFjDZveB1jvnskUaQNfOmAnGkcO1xma3-VcIfit0mY5hhtD9RDp242rtVY2F14\_KA78ls\_5uwZBGuqdUp2\_UNQx4d6a83BmtuC7Ejb-0hC-CBp9MPr\_vBKaw4rn0Dgm4ttDY\_dSoQBIs\_mqR1dRqmrkncM-LxDgIyAJ1uiYv2CU9PX\_tMqK\_6a\_qZJxXz6assPYWPXgR8thvxrfK8GB0CYFhMud\_JMUDXeZts3yABpzIgCL0HsDV8NJU4lo1fnocv\_QEN1O5zz1USorfA"

}

# **Метод Получения сертификата**

HTTP method: GET

Пример вызова: /jwt

**JSON Payload запроса:** отсутствует

**Header ответа**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-Type | application/x-x509-user-cert |
| X-Response-Signature | “eyJ0eXAiOiJKV1Qi…” |

**Ответ:**

-----BEGIN CERTIFICATE-----

MIIDEDCCAfigAwIBAgIQNhMZekHjOYRN8i9L…

-----END CERTIFICATE-----

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 500 – Server error

# **Метод Регистрации**

HTTP method: POST

Пример вызова: /user/register

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |

**JSON Payload запроса:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| pan | String | “5555555555554444” |
| phone | String | “79220000001” |

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| message | String | "An SMS with a verification code has been sent to your number." |
| sessionToken | String | “eyJ0eXAiOiJKV1Qi…” |

Успешное выполнение:

HTTP Code: 201 – Created

Ошибка выполнения:

HTTP Code: 400 – Bad request

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Подтверждения телефона**

HTTP method: POST

Пример вызова: /verify/phone

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | Bearer eyJ0eXAiOiJKV1Qi… |

\*Authorization содержит токен Сессии

**JSON Payload запроса:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| code | String | “123456” |

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| authToken | String | “eyJ0eXAiOiJKV1Qi…” |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 400 – Bad request

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Повторной отправки смс на телефон**

HTTP method: POST

Пример вызова: /send/sms

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Сессии

**JSON Payload запроса:** отсутствует

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| message | String | “SMS with a verification code has been sent to your number.” |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Создание кода доступа**

HTTP method: POST

Пример вызова: /user/passcode

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Авторизации

**JSON Payload запроса:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| passcode | String | “1234” |

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| message | String | “Passcode created successfully.” |

Успешное выполнение:

HTTP Code: 201 – OK

Ошибка выполнения:

HTTP Code: 400 – Bad request

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Аутентификации**

HTTP method: POST

Пример вызова: / auth

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Сессии

**JSON Payload запроса:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| passcode | String | “1234” |

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| authToken | String | “eyJ0eXAiOiJKV1Qi…” |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 400 – Bad request

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Получения списка карт пользователя**

HTTP method: GET

Пример вызова: /card

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Авторизации

**JSON Payload запроса:** отсутствует

**JSON ответ (Массив):**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| id | Int | 9 |
| numberAccount | String | “11112222333344445555” |
| pan | String | “5555555555555599” |
| cvv | String | “123” |
| name | String | “TestCard8” |
| balance | Double | 100 |
| activationDate | String | "2019-11-10T00:00:00" |
| expireDate | String | "03/24" |
| userId | Int | 28 |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Получения информации по карте**

HTTP method: GET

Пример вызова: /card/{id}

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Авторизации

**JSON Payload запроса:** отсутствует

**JSON ответ:**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| id | Int | 9 |
| numberAccount | String | “11112222333344445555” |
| pan | String | “5555555555555599” |
| cvv | String | “123” |
| name | String | “TestCard8” |
| balance | Double | 100 |
| activationDate | String | "2019-11-10T00:00:00" |
| expireDate | String | "03/24" |
| userId | Int | 28 |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 401 – Unauthorized

HTTP Code: 403 – Forbidden

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Получения истории по всем картам пользователя**

HTTP method: GET

Пример вызова: /history

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Авторизации

**JSON Payload запроса:** отсутствует

**JSON ответ (Массив):**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| cardName | String | “TestCard3” |
| operationName | String | "OperationName6" |
| operationStatus | String | "Accept" |
| Amount | Int | 133 |
| createdAt | String | “2020-02-24T14:00:00” |

Успешное выполнение:

HTTP Code: 200 – OK

Ошибка выполнения:

HTTP Code: 401 – Unauthorized

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error

# **Метод Получения истории по карте**

HTTP method: GET

Пример вызова: /history/{id}

**Header запросa**

|  |  |
| --- | --- |
| Имя ключа | Пример |
| Content-type | application/jwt |
| X-Request-Signature | eyJ0eXAiOiJKV1Qi… |
| Authorization | “Bearer eyJ0eXAiOiJKV1Qi…” |

\*Authorization содержит токен Авторизации

**JSON Payload запроса:** отсутствует

**JSON ответ (Массив):**

|  |  |  |
| --- | --- | --- |
| Имя ключа | Тип | Пример |
| cardName | String | “TestCard3” |
| operationName | String | "OperationName6" |
| operationStatus | String | "Accept" |
| Amount | Int | 133 |
| createdAt | String | “2020-02-24T14:00:00” |

Успешное выполнение:

HTTP Code: 200 – OK

HTTP Code: 204 – No Content

Ошибка выполнения:

HTTP Code: 401 – Unauthorized

HTTP Code: 403 – Forbidden

HTTP Code: 460 – JWE Error

HTTP Code: 461 – JWS Error

HTTP Code: 462 – Missing Content Type

HTTP Code: 500 – Server error