



ТОО "Мегахост Казахстан»  
Республика Казахстан, г. Уральск  
Проспект Евразия, 59/3, офис 107

8 800 080 63 42

+7 727 356 27 97

Skype (только звонки) megahostkaz

Клиентская служба: [sl@megahost.kz](mailto:sl@megahost.kz)

Служба поддержки: [sp@megahost.kz](mailto:sp@megahost.kz)

## Описание API системы управления доменами

## Содержание

<b>1. Введение</b>	<b>3</b>
<b>2. Описание HTTP шлюза</b>	<b>3</b>
2.1. Реальный доступ	3
2.2. Тестовый доступ	3
<b>3. Отправка HTTP-запросов</b>	<b>4</b>
3.1. Команды	4
3.2. Формат входных данных	4
3.3. Формат результата выполнения команды	5
3.4. Общие поля запросов	5
3.5. Сообщения об ошибках	5
3.6. Способы аутентификации	8
3.6.1. Аутентификация по паролю	8
3.6.2. Аутентификация по ключу API	8
<b>4. Описание команд</b>	<b>8</b>
4.1. Создание профайла	8
4.2. Изменение профайла	12
4.3. Регистрация доменного имени	13
4.4. Продление доменного имени	15
4.5. Изменение контактов для доменного имени	15
4.6. Получение DNS-серверов для доменного имени	16
4.7. Изменение DNS-серверов для доменного имени	17
4.8. Получение списка областей для страны	18
4.9. Получение списка регистраторов для доменной зоны	18
4.10. Получение статуса регистрации доменного имени	19
4.11. Получение данных профайла	20
4.12. Получение суммы внутреннего баланса пользователя	20
4.13. Трансфер доменного имени	21
4.14. Проверка доступности доменного имени	22

# 1. Введение

Это справочное руководство описывает HTTP-шлюз к системе распределённой регистрации ТОО "Мегахост Казахстан" (далее API).

HTTP-шлюз — это метод взаимодействия с системой распределённой регистрации API, позволяющий осуществлять операции в реальном времени за один шаг.

Для осуществления одношаговых (одноэтапных) операций, вся информация должна быть представлена в одном единственном HTTP-запросе. В интерфейсе API нет понятия "состояния" и все запросы независимы друг от друга. HTTP-интерфейс поддерживает такие операции как регистрация доменного имени, продление доменного имени, смена DNS серверов доменного имени, смена регистрационных данных доменного имени и т.п. Доступные операции описаны ниже в этом документе.

## 2. Описание HTTP шлюза

Компания ТОО "Мегахост Казахстан" предоставляет не только реальный доступ к HTTP-шлюзу, но также и тестовый доступ для отладки взаимодействия с системой API.

### 2.1. Реальный доступ

Запросы к HTTP-шлюзу должны направляться на URL

<https://billing.megahost.kz/api.php>

или

<http://billing.megahost.kz/api.php>

Рекомендуется использовать безопасный SSL-протокол, если его поддерживает регистратор.

Среднее время ответа при нормальных нагрузках сервера должно быть не более 5 или 10 секунд.

### 2.2. Тестовый доступ

ТОО "Мегахост Казахстан" предоставляет тестовый доступ к своему шлюзу для тестирования системы регистрации. Отличия тестового доступа от реального таковы:

- Плата за операции не взимается
- Операции с доменами реально не производятся, домены не регистрируются
- Тестовая система не содержит информации о доменах, которая присутствует в реальном реестре.

Для использования тестовой системы, HTTP запросы должны направляться на тот же URL, что и для реальной системы. При этом используются следующие авторизационные данные:

- login: test
- pass: test

## 3. Отправка HTTP-запросов

### 3.1. Команды

HTTP-шлюз позволяет выполнять различные команды. В таблице ниже приведён список команд, которые могут быть осуществлены с использованием HTTP-шлюза. Для каждой операции требуется указание различных параметров (полей), которые описаны ниже в этом документе.

Команда (значение <code>command</code> )	Описание
<code>createProfile</code>	Создание профайла
<code>updateProfile</code>	Изменение профайла
<code>registerDomain</code>	Регистрация доменного имени
<code>renewDomain</code>	Продление доменного имени
<code>updateDomainContacts</code>	Изменение контактов для доменного имени
<code>updateDNS</code>	Изменение DNS-серверов для доменного имени
<code>getAvailOblast</code>	Получение списка областей для страны
<code>getAvailRegistrators</code>	Получение списка регистраторов для доменной зоны
<code>getStatus</code>	Получение статуса регистрации доменного имени
<code>getProfile</code>	Получение данных профайла

### 3.2. Формат входных данных

Команды передаются в виде стандартного запроса HTTP/1.0 методом POST или GET. Параметры команды передаются в виде HTTP параметров. При этом действуют следующие правила:

- Значения всех полей являются строками.
- Значения полей передаются в кодировке utf-8.
- Все обязательные поля должны присутствовать в запросе и должны содержать как минимум один символ.
- Названия параметров HTTP-запроса должны в точности соответствовать названиям полей с учётом регистра символов.

- Значения всех полей должны быть urlencoded.

### 3.3. Формат результата выполнения команды

Ответом интерфейса API является сериализованная строка, содержащая в себе массив параметров, полученная с помощью PHP-функции `serialize`.

Кодировка строки ответа `utf-8`.

Для преобразования сериализованной строки обратно в массив параметров, необходимо использовать PHP-функцию `unserialize`.

Так же есть возможность получать ответ в формате `JSON`.

### 3.4. Общие поля запросов

В таблице перечислены все обязательные поля, которые должны присутствовать в любом запросе.

Имя поля	Описание
<code>command</code>	Определяет команду, которая должна быть выполнена, например <code>registerDomain</code>
<code>login</code>	Логин пользователя в биллинговой системе
<code>pass</code>	Пароль пользователя в биллинговой системе.  Поля <code>pass</code> и <code>apikey</code> являются взаимоисключающими и не могут встречаться в одном запросе. См. раздел 3.6.
<code>apikey</code>	Ключ для доступа к интерфейсу API.
<code>language</code>	Язык подробного описания ошибок ( <code>russian</code> , <code>english</code> , <code>ukrainian</code> ). Поле не обязательное. По умолчанию: <code>russian</code>
<code>json</code>	Если для поля задано значение равное единице, то система будет выдавать ответ в <code>JSON</code> -формате. Поле не обязательное.

### 3.5. Сообщения об ошибках

Существует два типа ошибок - критические и не критические.

В случае критической ошибки считается, что команда не выполнена.

В случае не критической ошибки считается, что команда выполнена, либо будет выполнена позже.

В случае ошибки при выполнении команды, API возвращает параметры, перечисленные в таблице ниже.

Имя поля	Описание
status	Результат выполнения команды. В случае критической ошибки значение всегда равно ERROR. В случае не критической SUCCESS.
errorCode	Код ошибки
errorMsg	Подробное описание ошибки

В таблице ниже приведены возможные критические ошибки при работе с API.

Код	Описание
1	Ошибка подключения к БД
2	Ошибка сохранения данных в БД
3	Не указан логин пользователя
4	Пользователь не найден
5	Доступ к API отключен
6	Не указан пароль или ключ API
7	Указан неправильный пароль или ключ API
8	Неизвестная команда
9	Запрещено использовать пароль и ключ API в одном запросе
10	Профайл не найден
11	Ошибки в полях профайла: <перечень полей>
12	Не указано ни одно из полей профайла для изменения
13	Не указано доменное имя
14	Не указана доменная зона
15	Доменная зона указана неверно, либо она не активна
16	Длина доменного имени должна быть от до символов
17	Доменное имя может содержать только a-z0-9a-я и символ дефиса, при чем не должно начинаться с дефиса или заканчиваться на него
18	Доменное имя занято
19	Доменное имя уже заказано другим пользователем биллинговой системы
20	Превышен лимит на количество регистрируемых бесплатных доменных имен в день
21	Не указан срок регистрации доменного имени
22	Допустимый срок регистрации доменного имени - г.
23	Возможность использования DNS-серверов по умолчанию отключена для доменной зоны

Код	Описание
24	Не указан или указан неверно первичный DNS-сервер
25	Не указан или указан неверно вторичный DNS-сервер
26	Указан неверно третий DNS-сервер
27	Указан неверно четвертый DNS-сервер
28	API не поддерживает работу с собственными DNS-серверами для данной доменной зоны
29	Ошибка проверки DNS-серверов
31	Недостаточно средств на внутреннем балансе
32	Тип профайла не должен отличаться от типа профайла на который изначально регистрировался домен
33	Доменная зона для заказа не найдена
34	Не указан срок продления доменного имени
35	Допустимый срок продления доменного имени - г.
36	Операция не возможна, т.к. для заказа есть неоплаченные счета
37	Операция не возможна, т.к. заказ не обработан
38	Продление доменного имени не возможно ранее чем за дней до окончания срока регистрации
39	Тип профайла не должен отличаться от типа профайла на который изначально регистрировался домен
40	Операция не возможна, т.к. заказ просрочен
41	Операция не поддерживается для данной доменной зоны
42	Не указан ISO-код страны
43	Список областей для страны не задан
44	Для доменной зоны отсутствует возможность выбора регистратора
45	Не указан IP-адрес для собственного NS1
46	Указан IP-адрес для NS1, который не является собственным DNS-сервером домена
47	Не указан IP-адрес для собственного NS2
48	Указан IP-адрес для NS2, который не является собственным DNS-сервером домена
49	Не указан IP-адрес для собственного NS3
50	Указан IP-адрес для NS3, который не является собственным DNS-сервером домена
51	Не указан IP-адрес для собственного NS4
52	Указан IP-адрес для NS4, который не является собственным DNS-сервером домена
53	Ошибка получения DNS-серверов
54	Доступ запрещен. IP отсутствует в списке разрешенных

Код	Описание
55	Для данной доменной зоны трансфер не поддерживается
56	Доменное имя не зарегистрировано
57	API не поддерживает работу с собственными DNS-серверами при трансфере
58	Не указан AUTH-CODE доменного имени
59	ID профайла не должен отличаться от ID профайла на который изначально регистрировался домен
60	Ошибка проверки доступности доменного имени

В таблице ниже приведены возможные не критические ошибки при работе с API.

Код	Описание
30	Заявка принята, но по техническим причинам будет обработана в ручном режиме.

## 3.6. Способы аутентификации

В системе регистрации API поддерживается два способа аутентификации: по логину и паролю, а также аутентификация по ключу API.

### 3.6.1. Аутентификация по паролю

Аутентификация осуществляется с использованием полей запроса **login** и **pass**.

Пользователь с указанным логином и паролем должен существовать в биллинговой системе для успешного прохождения аутентификации. Так же для него должен быть включен доступ к API.

### 3.6.2. Аутентификация по ключу API

Аутентификация осуществляется с использованием полей запроса **login** и **apikey**.

Пользователь с указанным логином и ключом API должен существовать в биллинговой системе для успешного прохождения аутентификации. Так же для него должен быть включен доступ к API.

## 4. Описание команд

### 4.1. Создание профайла

Эта команда служит для создания профайла (контакта) в биллинговой системе. В качестве значения поля **command** для этой команды должно быть указано **createProfile**.



Профайлы используются при регистрации доменных имен, а так же при изменении контактов доменного имени.

Профайлы бывают трех типов - физическое лицо, частный/индивидуальный предприниматель и организация.

В таблице ниже перечислены общие поля, используемые при создании профайла, а так же поля, необходимые для конкретных типов профайла.

Типы полей:

R - обязательное поле

RR - обязательное поле, если профайл используется для регистрации доменов RU/SU/РФ

RPPUA - обязательное поле, если профайл используется для регистрации доменов PP.UA

R\_XX - обязательное поле, если для профайла указана страна XX (XX - ISO код страны)

O - необязательное поле

O\_XX - необязательное поле, если для профайла указана страна XX (XX - ISO код страны)

Общие поля для всех типов профайлов

Имя поля	Тип	Описание
org	R	Тип профайла: 1 - физическое лицо 2 - частный/индивидуальный предприниматель 3 - организация
surname	R	Для организаций - фамилия директора. Для физ.лиц/ЧП/ИП - фамилия физ.лица/ЧП/ИП. Пример: Иванов
name	R	Для организаций - имя директора. Для физ.лиц/ЧП/ИП - имя физ.лица/ЧП/ИП. Пример: Александр
otchestvo	R	Для организаций - отчество директора. Для физ.лиц/ЧП/ИП - отчество физ.лица/ЧП/ИП. Пример: Иванович
gender	O	Пол физического лица / директора. Может принимать значения male (мужской) или female (женский)
country	R	Двухбуквенный ISO-код страны. Пример: KZ
oblast	R_RU R_UA R_BY R_KZ	Область. Пример: Акмолинская обл. Список допустимых для конкретной страны областей можно получить, выполнив команду описанную в разделе 4.8.
post	R	Почтовый индекс. Пример: 189343

Имя поля	Тип	Описание
city	R	Название населенного пункта. Пример: Астана
street	R	Почтовый адрес. Пример: ул. Саратовская, д. 5, кв. 34; ул. Крымская, д. 172 (частный дом); ул. Сумская, д. 2, офис 4
komu	R O_FI	Адресат в дательном падеже. Пример: Иванову С.В.
phone	R	Телефон с международным кодом (включая символ +); международный код, код города и местный номер разделяются пробелами. Скобки и дефисы не допускаются. Пример: +7 495 8102233
mobile	RPPUA	Мобильный телефон с международным кодом (включая символ +); международный код, код оператора и местный номер разделяются пробелами. Скобки и дефисы не допускаются. Пример: +7 926 8102233
fax	O	Факс с международным кодом (включая символ +); международный код, код города и местный номер разделяются пробелами. Скобки и дефисы не допускаются. Пример: +7 495 8102233
email	R	E-mail адрес. Пример: ivan@krutov.ru
icq	O	Номер ICQ. Пример: 345224532
skype	O	Логин Skype.
ripn	O	RIPN-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-RIPN
ripe	O	RIPE-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-RIPE
uanic	O	UANIC-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-UANIC
eunic	O	EUNIC-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-EUNIC
dpnic	O	DPNIC-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-DPNIC
epnic	O	EPNIC-никхендел. Поле может быть обязательным для определенных доменных зон. Пример: RIP-EPNIC

## Дополнительные поля для физического лица

Имя поля	Тип	Описание
seriya	R	Серия и номер паспорта. Знак номера перед номером паспорта не ставится. Нужно указать серию паспорта и через пробел его номер. Примеры: Россия: 34 02 651241 Украина: HC 223344
by	R	Полное наименование органа, выдавшего паспорт. Пример: 48 о/м г.Москвы
date	R	Дата выдачи паспорта. Пример: 23.07.1999
birthday	R	Дата рождения. Пример: 20.07.1983
socstrahnumber	R_FI	Номер социального страхования.
oib	R_HR	Company or Personal ID (OIB).
idnum	R_BY	Идентификационный номер.

## Дополнительные поля для частного/индивидуального предпринимателя

Имя поля	Тип	Описание
seriya	R	Серия и номер паспорта. Знак номера перед номером паспорта не ставится. Нужно указать серию паспорта и через пробел его номер. Примеры: Россия: 34 02 651241 Украина: HC 223344
by	R	Полное наименование органа, выдавшего паспорт. Пример: 48 о/м г.Москвы
date	R	Дата выдачи паспорта. Пример: 23.07.1999
birthday	R	Дата рождения. Пример: 20.07.1983
edrpou	R_UA	ЄДРПОУ Код в Государственном реестре предприятий и организаций. Пример: 2354231243
inn	R_BY R_RU	Идентификационный номер налогоплательщика (ИНН). Пример: 789012345678
ogrn	R_HU	Регистрационный номер частного предпринимателя.
idnum	R_BY	Идентификационный номер.

## Дополнительные поля для организации

Имя поля	Тип	Описание
firma	R	Полное наименование организации на русском языке в соответствии с учредительными документами. Пример: Общество с ограниченной ответственностью "Рога и Копыта"
firmaeng	R	Полное наименование организации латинскими буквами. Пример: "ROGA I KOPYTA", LTD.
address_org	R	Юридический адрес организации в соответствии с учредительными документами. Пример: 101000, Москва, ул.Пупкина, 1, стр. 2
inn	R	Идентификационный номер налогоплательщика (ИНН). Пример: 789012345678
kpp	R_RU	КПП организации. Пример: 632946014
oknh	R_RU	Код в Общем классификаторе областей народного хозяйства. Пример: 72000
okpo	R_BY	Код в Общем классификаторе областей народного хозяйства. Пример: 72000
ogrn	R	ОГРН. Сведения о документе, удостоверяющем регистрацию в соответствии с правилами страны инкорпорации (серия и номер документа).
ogrn_by	R	Кем выдан ОГРН.
ogrn_date	R	Дата выдачи ОГРН.
bank	O	Банковские реквизиты: Банк
bank_bik	O	Банковские реквизиты: БИК (МФО)
bank_schet	O	Банковские реквизиты: Расчётный счет

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
profileid	ID профайла в системе API. Необходим для дальнейшего изменения профайла, а так же для регистрации доменов на данный профайл.

## 4.2. Изменение профайла

Эта команда служит для изменения профайла (контакта) в биллинговой системе.

В качестве значения поля **command** для этой команды должно быть указано **updateProfile**.

Внимание! Контакт доменного имени при изменении профайла в биллинговой системе остается неизменным. Для изменения непосредственно контактов доменного имени, необходимо использовать команду, описанную в разделе 4.5.

В таблице ниже перечислены поля, используемые при изменении профайла.

Имя поля	Описание
profileid	ID профайла, возвращаемый командой createProfile. Если не указать, то будет изменен профайл по умолчанию для пользователя.
<параметры>	В качестве параметров могут быть использованы любые поля, описанные в разделе 4.1., как по отдельности, так и вместе. Запрещено изменять значение поля org (тип профайла).

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
profileid	ID профайла в системе биллинга API.

### 4.3. Регистрация доменного имени

Эта команда служит для регистрации доменного имени. В качестве значения поля **command** для этой команды должно быть указано **registerDomain**.

В таблице ниже перечислены поля, используемые при регистрации доменного имени.

Имя поля	Описание
domain	Полное доменное имя, которое необходимо зарегистрировать, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.
registratorid	ID регистратора. Поле используется только в том случае, если для доменной зоны доступны несколько регистраторов. Если не задано, используется регистратор по умолчанию. Список доступных регистраторов для доменной зоны можно получить, выполнив команду, описанную в разделе 4.9.
period	Период, на который производится регистрация домена. Значение необходимо указывать в годах. Пример: 1. Доменные зоны KZ, COM.KZ, ORG.KZ, KAZ регистрируются максимум на 10 лет.
profileid	ID профайла, возвращаемый командой createProfile, данные из которого необходимо использовать для создания контактов владельца домена. Если не указано, то используется профайл по умолчанию для пользователя.
defaultns	1 - использовать DNS-сервера по умолчанию (данная возможность может быть отключена) 0 - указать собственные DNS-сервера.

Имя поля	Описание
ns1	Первичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns2	Вторичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns3	Третий DNS-сервер. Поле не обязательное.
ns4	Четвертый DNS-сервер. Поле не обязательное.
ns1ip	IP-адрес для первичного DNS-сервера. Поле является обязательным, только в том случае, если ns1 - собственный DNS-сервер регистрируемого домена.
ns2ip	IP-адрес для вторичного DNS-сервера. Поле является обязательным, только в том случае, если ns2 - собственный DNS-сервер регистрируемого домена.
ns3ip	IP-адрес для третьего DNS-сервера. Поле является обязательным, только в том случае, если ns3 - собственный DNS-сервер регистрируемого домена.
ns4ip	IP-адрес для четвертого DNS-сервера. Поле является обязательным, только в том случае, если ns4 - собственный DNS-сервер регистрируемого домена.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Доменное имя, которое было зарегистрировано.
period	Период на который было зарегистрировано доменное имя.
profileid	ID профайла в системе API, который использовался при регистрации доменного имени.
defaultns	Флаг использования DNS-серверов по умолчанию. Если были использованы, то значение 1.
ns1	Первичный DNS-сервер.
ns2	Вторичный DNS-сервер.
ns3	Третий DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns4	Четвертый DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns1ip	IP-адрес вторичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns2ip	IP-адрес вторичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns3ip	IP-адрес третьего DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns4ip	IP-адрес четвертого DNS-сервера. Поле будет возвращено только в случае, если оно было.
balance	Текущий баланс пользователя.

Имя поля	Описание
cost	Стоимость регистрации домена.
currency	Код валюты в которой возвращены стоимость и баланс. Идентична валюте пользователя в биллинговой системе.

#### 4.4. Продление доменного имени

Эта команда служит для продления доменного имени. В качестве значения поля **command** для этой команды должно быть указано **renewDomain**.

В таблице ниже перечислены поля, используемые при продлении доменного имени.

Имя поля	Описание
domain	Полное доменное имя, которое необходимо продлить, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.
period	Период, на который производится регистрация домена. Допустимые значения для данного поля по каждой доменной зоне можно увидеть здесь. Значение необходимо указывать в годах. Пример: 1.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Текущий баланс пользователя.
domain	Доменное имя, которое было зарегистрировано.
period	Период на который было зарегистрировано доменное имя.
balance	Текущий баланс пользователя.
cost	Стоимость регистрации домена.
currency	Код валюты в которой возвращены стоимость и баланс. Идентична валюте пользователя в биллинговой системе.

#### 4.5. Изменение контактов для доменного имени

Эта команда служит для изменения контактов для доменного имени.

В качестве значения поля **command** для этой команды должно быть указано **updateDomainContacts**.

В таблице ниже перечислены поля, используемые при изменении контактов для доменного имени.

Имя поля	Описание
domain	Полное доменное имя, которое необходимо продлить, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.
profileid	ID профайла, возвращаемый командой createProfile, данные из которого необходимо использовать для изменения контактов владельца домена. Если не указано, то используется профайл по умолчанию для пользователя.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Доменное имя, для которого было произведено изменения контактов.

## 4.6. Получение DNS-серверов для доменного имени

Эта команда служит для получения DNS-серверов для доменного имени. В качестве значения поля **command** для этой команды должно быть указано **getDNS**.

В таблице ниже перечислены поля, используемые при получении DNS-серверов.

Имя поля	Описание
domain	Полное доменное имя, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Доменное имя для которого получаем DNS-сервера.
ns1	Первичный DNS-сервер.
ns2	Вторичный DNS-сервер.
ns3	Третий DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns4	Четвертый DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns1ip	IP-адрес вторичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns2ip	IP-адрес вторичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.



Имя поля	Описание
ns3ip	IP-адрес третьего DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns4ip	IP-адрес четвертого DNS-сервера. Поле будет возвращено только в случае, если оно было.

## 4.7. Изменение DNS-серверов для доменного имени

Эта команда служит для изменения DNS-серверов для доменного имени. В качестве значения поля **command** для этой команды должно быть указано **updateDNS**.

В таблице ниже перечислены поля, используемые при изменении DNS-серверов.

Имя поля	Описание
domain	Полное доменное имя, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.
defaultns	1 - использовать DNS-сервера по умолчанию (данная возможность может быть отключена) 0 - указать собственные DNS-сервера.
ns1	Первичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns2	Вторичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns3	Третий DNS-сервер. Поле не обязательное.
ns4	Четвертый DNS-сервер. Поле не обязательное.
ns1ip	IP-адрес для первичного DNS-сервера. Поле является обязательным, только в том случае, если ns1 - собственный DNS-сервер регистрируемого домена.
ns2ip	IP-адрес для вторичного DNS-сервера. Поле является обязательным, только в том случае, если ns2 - собственный DNS-сервер регистрируемого домена.
ns3ip	IP-адрес для третьего DNS-сервера. Поле является обязательным, только в том случае, если ns3 - собственный DNS-сервер регистрируемого домена.
ns4ip	IP-адрес для четвертого DNS-сервера. Поле является обязательным, только в том случае, если ns4 - собственный DNS-сервер регистрируемого домена.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Доменное имя, которое было зарегистрировано.
defaultns	Флаг использования DNS-серверов по умолчанию. Если были использованы, то значение 1.

Имя поля	Описание
ns1	Первичный DNS-сервер.
ns2	Вторичный DNS-сервер.
ns3	Третий DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns4	Четвертый DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns1ip	IP-адрес первичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns2ip	IP-адрес вторичного DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns3ip	IP-адрес третьего DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.
ns4ip	IP-адрес четвертого DNS-сервера. Поле будет возвращено только в случае, если для него установлено значение.

## 4.8. Получение списка областей для страны

Эта команда служит для получения списка корректных областей для страны. В качестве значения поля **command** для этой команды должно быть указано **getAvailOblast**.

В таблице ниже перечислены поля, используемые при получении списка областей для страны.

Имя поля	Описание
country	Двухбуквенный ISO-код страны. Пример: UA

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
country	Двухбуквенный ISO-код страны. Пример: UA

## 4.9. Получение списка регистраторов для доменной зоны

Эта команда служит для получения списка доступных регистраторов для доменной зоны. В качестве значения поля **command** для этой команды должно быть указано **getAvailRegistrators**.

В таблице ниже перечислены поля, используемые при получении списка регистраторов для доменной зоны.

Имя поля	Описание
zone	Доменная зона без точки в начале. Пример: ru

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS. Успешное выполнение команды возможно только в том случае, если в системе API для доменной зоны включена возможность выбора регистратора.

registrators	<p>В данном поле возвращается массив полей. Каждое поле массива в свою очередь так же является массивом и содержит в себе следующие поля:</p> <p>registratorid - ID регистратора  registratorname - название регистратора  registratorisdefault - если 1, то данный регистратор является регистратором по умолчанию для доменной зоны  zoneregcost - стоимость регистрации  zonerenewcost - стоимость продления  zoneregperiod - доступный срок регистрации  zonerenewperiod - доступный срок продления  currency - код валюты, в которой указана стоимость</p> <p>Если для доменной зоны разрешен трансфер, то выводятся дополнительные поля:</p> <p>zonetransfercost - стоимость трансфера  zonetransferneedauthcode - если 1, то необходимо указать AUTH-CODE при оформлении заказа</p>
--------------	--

#### 4.10. Получение статуса регистрации доменного имени

Эта команда служит для получения статуса регистрации доменного имени. В качестве значения поля **command** для этой команды должно быть указано **getStatus**.

В таблице ниже перечислены поля, используемые при получении статуса.

Имя поля	Описание
domain	Полное доменное имя, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.

Имя поля	Описание
domain	Доменное имя для которого получаем статус.
domainStatus	Статус регистрации доменного имени. Возможные значения:  unregistered - домен не зарегистрирован (не активен) registered - домен зарегистрирован (активен) expired - домен просрочен (не активен) suspended - домен приостановлен (не активен) inprocess - домен находится в процессе регистрации
todate	Дата, до которой зарегистрировано доменное имя. Доступно только для статусов registered, suspended, expired Пример: 20.07.1983

#### 4.11. Получение данных профайла

Эта команда служит для получения данных профайла (контакта) в биллинговой системе. В качестве значения поля **command** для этой команды должно быть указано **getProfile**.

В таблице ниже перечислены поля, используемые при получении профайла.

Имя поля	Описание
profileid	ID профайла, возвращаемый командой createProfile. Если не указать, то будет получен профайл по умолчанию для пользователя.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
profileid	ID профайла в системе API.
fields	Массив полей профайла, описанных в разделе 4.1.

#### 4.12. Получение суммы внутреннего баланса пользователя

Эта команда служит для получения суммы внутреннего баланса пользователя биллинговой системы. В качестве значения поля **command** для этой команды должно быть указано **getBalance**.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
balance	Текущий баланс пользователя.

Имя поля	Описание
currency	Код валюты в которой возвращен баланс. Идентична валюте пользователя в биллинговой системе.

### 4.13. Трансфер доменного имени

Эта команда служит для запроса на трансфер доменного имени от другого регистратора. В качестве значения поля **command** для этой команды должно быть указано **transferDomain**.

В таблице ниже перечислены поля, используемые при регистрации доменного имени.

Имя поля	Описание
domain	Полное доменное имя, которое необходимо зарегистрировать, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.
registratorid	ID регистратора. Поле используется только в том случае, если для доменной зоны доступны несколько регистраторов. Если не задано, используется регистратор по умолчанию. Список доступных регистраторов для доменной зоны можно получить, выполнив команду, описанную в разделе 4.9.
period	Период, на который производится регистрация домена. Значение необходимо указывать в годах. Пример: 1.
authcode	AUTH-CODE доменного имени. Поле может быть как обязательным, так и нет, в зависимости от настроек доменной зоны.
profileid	ID профайла, возвращаемый командой createProfile, данные из которого необходимо использовать для создания контактов владельца домена. Если не указано, то используется профайл по умолчанию для пользователя.
defaultns	1 - использовать DNS-сервера по умолчанию (данная возможность может быть отключена) 0 - указать собственные DNS-сервера.
ns1	Первичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns2	Вторичный DNS-сервер. Поле является обязательным, если не указано использование DNS-серверов по умолчанию.
ns3	Третий DNS-сервер. Поле не обязательное.
ns4	Четвертый DNS-сервер. Поле не обязательное.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Доменное имя, которое было зарегистрировано.

Имя поля	Описание
period	Период на который было зарегистрировано доменное имя.
profileid	ID профайла в системе API, который использовался при регистрации доменного имени.
defaultns	Флаг использования DNS-серверов по умолчанию. Если были использованы, то значение 1.
ns1	Первичный DNS-сервер.
ns2	Вторичный DNS-сервер.
ns3	Третий DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
ns4	Четвертый DNS-сервер. Поле будет возвращено только в случае, если для него установлено значение.
balance	Текущий баланс пользователя.
cost	Стоимость регистрации домена.
currency	Код валюты в которой возвращены стоимость и баланс. Идентична валюте пользователя в биллинговой системе.

#### 4.14. Проверка доступности доменного имени

Эта команда служит для проверки доступности доменного имени для регистрации. В качестве значения поля **command** для этой команды должно быть указано **checkDomain**.

В таблице ниже перечислены поля, используемые при проверке доступности доменного имени.

Имя поля	Описание
domain	Полное доменное имя, например example.com. Допустимы алфавитно-цифровые символы и символ дефиса. Русские имена доменов указываются в кодировке utf-8.

В случае успешного выполнения команды, API вернет поля, перечисленные в таблице ниже.

Имя поля	Описание
status	Если команда выполнена успешно, значение всегда будет SUCCESS.
domain	Проверяемое доменное имя.
avail	1 - доменное имя доступно для регистрации 0 - доменное имя не доступно для регистрации