DOMAIN API

TABLE OF CONTENTS

DOMAIN API documentation

AUTHENTICATION AND COMMANDS

The API interface uses a HTTP Basic authentication where username is your admin username and password is your API password. The authentication will only success if the connection come from permitted IP. The API password and permitted IP addresses adjustable on the user interface Edit Account Details page.

The required or optional fields are accepted via GET or POST method.

All requests are submitted using the following HTTP interface:

https:// <BASE_URL><COMMAND>?parameter1=value1&parameter2=value2&parameter3=value3...

For example: https://tryoutapi.webspacecontrol.com/account/modify?address1=Expo%20Tér%1

ACCOUNT ACTIONS

GET ACCOUNT DETAILS

Command request: /account/get

Response:

```
{
    "account": {
        "id": "1",
        "firstname": "John",
        "lastname": "Doe",
        "companyname": "",
        "email": "john@example.com",
        "address1": "Expo ter 5-7.",
        "address2": "",
        "city": "Budapest",
        "state": "",
        "postcode": "1101",
        "country": "HU",
        "phonenumber": "3614323232",
        "credit": "0.00"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET PRICES

Command request: /account/prices

Response:

```json
{
    "prices": {
        "domain": [
            {
                "tld": ".hu",
                "period": "1, 2",
                "register": "0.00 Ft \/2 év",
                "transfer": "0.00 Ft \/1 év",
                "renew": "0.00 Ft \/1 év"
            },
            {
                "tld": ".com",
                "period": 1,
                "register": "0.00 Ft",
                "transfer": "0.00 Ft",
                "renew": "0.00 Ft"
            }
        ],
        "product": [
            {
                "name": "Product type",
                "products": [
                    {
                        "pid": "12",
                        "name": "Product name 1",
                        "description": "",
                        "pricing": {
                            "annually": "1.00 Ft \/év",
                            "biennially": "2.00 Ft \/2 év",
                            "triennially": "3.00 Ft \/3 év"
                        }
                    }
                ]
            }
        ]
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

MODIFY ACCOUNT

Command request: /account/modify

Allowed fields for modify: address1, address2, city, state, postcode, country, phonenumber

Parameter	Obligation	Definition	Type
address1	optional	Billing street, street number	TEXT
address2	optional	Billing address	TEXT
city	optional	Billing address city name	TEXT
state	optional	Billing address state	TEXT
postcode	optional	Billing postcode	TEXT
country	optional	Billing address country 2 letters code	TEXT
phonenumber	optional	Phone number following format: 3614323232	NUMBER

Please note: the optional fields mean if you don't send new details the stored details do not changed.
Response:

```
{
    "account": {
        "clientid": "1"
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

CONTACT ACTIONS

LIST CONTACTS

Command request: /contacts/list

Response:

```json
{
    "contacts": [
        {
            "id": "1",
            "firstname": "John",
            "lastname": "Doe",
            "companyname": "John Doe Ltd.",
            "email": "john@example.com",
            "address1": "Expo ter 5-7.",
            "address2": "",
            "city": "Budapest",
            "state": "",
            "postcode": "1101",
            "country": "HU",
            "phonenumber": "3614323232",
            "tax_id": "12345678-1-01",
            "identity": "", // Deprecated field, use tax_id instead
            "vatnumber": "12345678-1-01", // Deprecated field, use tax_id
            "euvatnumber": "" // Deprecated field, use tax_id instead
        },
        {
            "id": "2",
            "firstname": "Jane",
            "lastname": "Doe",
            "companyname": "",
            "email": "jane@example.com",
            "address1": "Expo ter 5-7.",
            "address2": "",
            "city": "Budapest",
            "state": "",
            "postcode": "1101",
            "country": "HU",
            "phonenumber": "3614323232",
            "tax_id": "12345678-1-01",
            "identity": "", // Deprecated field, use tax_id instead
            "vatnumber": "12345678-1-01", // Deprecated field, use tax_id
            "euvatnumber": "" // Deprecated field, use tax_id instead
        },
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET CONTACT DETAILS

Command request: /contacts/get/:contactid

Response:

```json
{
    "contact": {
        "id": "1",
        "firstname": "John",
        "lastname": "Doe",
        "companyname": "",
        "email": "john@example.com",
        "address1": "Expo tér 5-7.",
        "address2": "",
        "city": "Budapest",
        "state": "Pest",
        "postcode": "1101",
        "country": "HU",
        "phonenumber": "3614323232",
        "tax_id": "",
        "identity": "", // Deprecated field, use tax_id instead
        "vatnumber": "", // Deprecated field, use tax_id instead
        "euvatnumber": "" // Deprecated field, use tax_id instead
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

ADD CONTACT

Command request: /contacts/add

Parameter	Obligation	Definition	Type
firstname	optional	First name	TEXT
lastname	optional	Last name	TEXT
companyname	optional	If company, Company name	TEXT
email	optional	Email address	TEXT
address1	optional	Address	TEXT
address2	optional	Address	TEXT
city	optional	City name	TEXT
state	optional	State name	TEXT
postcode	optional	Postcode	TEXT
country	optional	Country with 2 letters	TEXT
phonenumber	optional	Phone number with following format: 3614323232	NUMBER
tax_id	optional	VAT number OR Personal identity	TEXT
identity	deprecated	If not company, Personal identity	TEXT
vatnumber	deprecated	If Hungarian company, VAT number	TEXT
euvatnumber	deprecated	If European country, but nut Hungarian, EU VAT number	TEXT

Response:

```
{
    "contact": {
        "contactid": 1
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

MODIFY CONTACT

Command request: /contacts/modify/:contactid (:contactid is Contact ID)

Parameter	Obligation	Definition	Type
firstname	optional	First name	TEXT
lastname	optional	Last name	TEXT
companyname	optional	If company, Company name	TEXT
email	optional	Email address	TEXT
address1	optional	Address	TEXT
address2	optional	Address	TEXT
city	optional	City name	TEXT
state	optional	State name	TEXT
postcode	optional	Postcode	TEXT
country	optional	Country with 2 letters	TEXT
phonenumber	optional	Phone number with following format: 3614323232	NUMBER
tax_id	optional	VAT number OR Personal identity	TEXT
identity	deprecated	If not company, Personal identity	TEXT
vatnumber	deprecated	If Hungarian company, VAT number	TEXT
euvatnumber	deprecated	If European country, but nut Hungarian, EU VAT number	TEXT

Please note: the optional fields mean if you don't send new details the stored details do not changed.

Response:

```
{
    "contact": {
        "contactid": "1"
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

DELETE CONTACT

Command request: /contacts/delete/:contactid (:contactid is Contact ID)

Response:

```
{
    "contact": {
        "contactid": "1"
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

MESSAGE ACTIONS

LIST MESSAGES

Command request: /messages/list

Response:

```
{
    "messages": [
        {
            "id": "1",
            "subject": "Subject 1",
            "date": "2015-01-01 12:00:00",
            "to": "john@example.com"
        },
        {
            "id": "2",
            "subject": "Subject 2",
            "date": "2015-01-01 12:00:00",
            "to": "john@example.com"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET MESSAGE

Command request: /messages/get/:messageid (:messageid is Message ID)

Response:

```json
{
    "message": {
        "subject": "Subject 1",
        "message": "Email text",
        "date": "2015-01-01 12:00:00",
        "to": "john@example.com"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

INVOICE ACTIONS

LIST INVOICES

Command request: /invoices/list

Response:

```
{
    "invoices": [
        {
            "id": "1",
            "firstname": "John",
            "lastname": "Doe",
            "companyname": "",
            "invoicenum": "2015-1",
            "date": "2015-01-01",
            "duedate": "2015-01-01",
            "datepaid": "2015-01-01 12:00:00",
            "subtotal": "0.00",
            "credit": "0.00",
            "tax": "0.00",
            "total": "0.00",
            "taxrate": "0.00",
            "status": "Paid",
            "notes": "",
            "currencycode": "HUF"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET INVOICE

Command request: /invoices/get/:invoiceid|:invoicenum (:invoiceid is Invoice ID OR :invoicenum is Invoice Number)

Response:

```json
{
    "invoice": {
        "invoicenum": "2015-1",
        "date": "2015-01-01",
        "duedate": "2015-01-01",
        "datepaid": "2015-01-01 12:00:00",
        "subtotal": "0.00",
        "credit": "0.00",
        "tax": "0.00",
        "total": "0.00",
        "balance": "0.00",
        "taxrate": "27.00",
        "status": "Paid",
        "notes": "",
        "items": [
            {
                "description": "Domain renewal - example.com - 1 Year",
                "amount": "2999.00"
            }
        ]
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

HOSTING ACTIONS

LIST HOSTING SERVICES

Command request: /hostings/list

Response:

```json
{
    "hostings": [
        {
            "id": "1",
            "regdate": "2015-01-01",
            "name": "Product name 1",
            "groupname": "Tárhely",
            "domain": "example.com",
            "status": "Active",
            "expiration": "2016-01-01"
        },
        {
            "id": "2",
            "regdate": "2015-01-01",
            "name": "Product name 2",
            "groupname": "Tárhely",
            "domain": "example.tld",
            "status": "Active",
            "expiration": "2016-01-01"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET HOSTING SERVICE

Command request: /hostings/get/:id|:domain (:id is Hosting ID or :domain is Hosting main domain name)

Response:

```
{
    "hosting": {
        "regdate": "2015-01-01",
        "name": "Mikro",
        "groupname": "Tárhely",
        "domain": "example.com",
        "serverip": "127.0.0.1",
        "serverhostname": "cl0x.webspacecontrol.com",
        "recurringamount": "0.00",
        "billingcycle": "Biennially",
        "status": "Active",
        "username": "example"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

ORDER NEW HOSTING SERVICE

Command request: /hostings/add

Parameter	Obligation	Definition	Type
pid	required	Product ID (available in Get Prices command)	NUMBER
domain	required	Web hosting main domain name	TEXT
billingcycle	required	Billing method (available in Get Prices command)	TEXT
addons	optional	A comma separated list of addons ID (available in Get Prices command)	TEXT

Response:

```
{
    "hosting": {
        "orderid": 1,
        "productids": "1",
        "invoiceid": 1
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

UPGRADE HOSTING SERVICE

Command request: /hostings/upgrade/:id|:domain (:id is Hosting ID or :domain is Hosting main domain name)

Parameter	Obligation	Definition	Type
newproductid	*required*	*Product ID (available in Get Prices command)*	*NUMBER*
newproductbillingcycle	*required*	*Billing method (available in Get Prices command)*	*TEXT*

Response:

```
{
    "hosting": {
        "oldproductid": "1",
        "oldproductname": "Old product name",
        "newproductid": "2",
        "newproductname": "New prduct name",
        "daysuntilrenewal": 366,
        "totaldays": 366,
        "newproductbillingcycle": "annually",
        "price": "0.00 Ft",
        "id": "1",
        "orderid": 1,
        "order_number": "123456789",
        "invoiceid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

HOSTING CANCELLATION REQUEST

Command request: /hostings/cancel/:id|:domain (:id is Hosting ID or :domain is Hosting main domain name)

Parameter	Obligation	Definition	Type
type	required	Cancellation type: "Immediate" OR "End of Billing Period"	TEXT

Response:

```
{
    "hosting": {
        "serviceid": "1",
        "userid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

DOMAIN ACTIONS

LIST DOMAINS

Command request: /domains/list

Response:

```json
{
    "domains": [
        {
            "id": "1",
            "regtype": "Register",
            "domainname": "example.com",
            "status": "Active",
            "expiration": "2016-01-01"
        },
        {
            "id": "2",
            "regtype": "Register",
            "domainname": "example.tld",
            "status": "Expired",
            "expiration": "2015-01-01"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET DOMAIN DATA

Command request: /domains/get/:id|:name (:id is Domain ID or :name is Domain name)

Response:

```json
{
    "domain": {
        "regtype": "Register",
        "domainname": "example.com",
        "regperiod": "1",
        "regdate": "2015-01-01",
        "expirydate": "2015-01-01",
        "status": "Active",
        "ns": [
            "ns1.example.com",
            "ns2.example.com"
        ],
        "lockstatus": "N/A",
        "contacts": {
            "owner": {
                "firstname": "John",
                "lastname": "Doe",
                "companyname": "",
                "address1": "Expo ter",
                "address2": "5-7.",
                "city": "Budapest",
                "state": "",
                "postcode": "1101",
                "country": "HU",
                "phonenumber": "+36-14323232",
                "faxnumber": "",
                "email": "",
                "vatnumber": "AA999999",
                "type": "person"
            },
            "admin": {
                "firstname": "John",
                "lastname": "Doe",
                "companyname": "",
                "address1": "Expo ter",
                "address2": "5-7.",
                "city": "Budapest",
                "state": "",
                "postcode": "1101",
                "country": "HU",
                "phonenumber": "+36-14323232",
                "faxnumber": "",
                "email": "",
                "vatnumber": "AA999999",
                "type": "person"
            },
            "tech": {
                "firstname": "John",
                "lastname": "Doe",
                "companyname": "",
                "address1": "Expo ter",
```

DOMAIN API documentation

```json
                    "address2": "5-7.",
                    "city": "Budapest",
                    "state": "",
                    "postcode": "1101",
                    "country": "HU",
                    "phonenumber": "+36-14323232",
                    "faxnumber": "",
                    "email": "",
                    "vatnumber": "AA999999",
                    "type": "person"
                }
            }
        },
        "result": {
            "code": 200,
            "message": "OK"
        }
    }
```

22

ORDER DOMAIN REGISTRATION

Command request: /domains/register

Parameter	Obligation	Definition	Type
domain	required	Domain name	TEXT
years	required	Registration years	NUMBER
ns1	required	Domain name server 1	TEXT
ns2	optional	Domain name server 2	TEXT
ns3	optional	Domain name server 3	TEXT
ns4	optional	Domain name server 4	TEXT
ns5	optional	Domain name server 5	TEXT
domainfields	conditional	If TLD need (JSON encoded array with required and/or optional parameters)	TEXT
owner	required	Domain owner (Registrant) contact ID	NUMBER
admin	required	Domain admin contact ID	NUMBER
tech	required	Domain tech contact ID	NUMBER
type	optional	Domain type (doc=document, 1f=one factor or 2f=two factor based) – Only .HU namespace	TEXT

Response:

```
{
    "domain": {
        "orderid": 1,
        "domainids": "1",
        "invoiceid": 1
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

ORDER DOMAIN TRANSFER

Command request: /domains/transfer

Parameter	Obligation	Definition	Type
domain	required	Domain name	TEXT
years	required	Registration years	NUMBER
eppcode	conditional	If TLD required EPP code	TEXT
ns1	required	Domain name server 1	TEXT
ns2	optional	Domain name server 2	TEXT
ns3	optional	Domain name server 3	TEXT
ns4	optional	Domain name server 4	TEXT
ns5	optional	Domain name server 5	TEXT
domainfields	conditional	If TLD need (JSON encoded array with required and/or optional parameters)	TEXT
owner	required	Domain owner (Registrant) contact ID	NUMBER
admin	required	Domain admin contact ID	NUMBER
tech	required	Domain tech contact ID	NUMBER
trade	optional	Transfer with owner change (Only .HU namespace)	BOOLEAN

Response:

```
{
    "domain": {
        "orderid": 1,
        "domainids": "1",
        "invoiceid": 1
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

UPDATE THE NAMESERVERS OF A DOMAIN

Command request: /domains/updatens/:id|:name (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
ns1	required	Domain name server 1	TEXT
ns2	required	Domain name server 2	TEXT
ns3	optional	Domain name server 3	TEXT
ns4	optional	Domain name server 4	TEXT
ns5	optional	Domain name server 5	TEXT

Response:

```
{
    "domain": {
        "domainid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

SET DOMAIN AUTORENEWAL TYPE

Command request: /domains/autorenew/:id|:name (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
autorenewal	required	"yes" OR "no"	TEXT

Response:

```
{
    "domain": {
        "domainid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

ORDER DOMAIN RENEWAL

Command request: /domains/renew/:id|:name (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
years	required	Renewal years	NUMBER

Response:

```json
{
    "domain": {
        "orderid": 1,
        "invoiceid": 1
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

ORDER MULTIPLE DOMAIN RENEWAL

Command request: /domains/bulkrenew

Parameters name is domain name and value is years.

Response:

```json
{
    "domain": {
        "orderid": 1,
        "invoiceid": 1
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

UPDATE THE TRANSFER LOCK STATUS OF A DOMAIN

Command request: /domains/lock/:id|:name (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
lock	required	"yes" OR "no"	TEXT

Response:

```
{
    "domain": {
        "domainid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

UPDATE THE CONTACT INFORMATION ON A DOMAIN

Command request: /domains/contact/:id|:name/:type (:id is Domain ID or :name is Domain name AND :type is Contact type)

Response:

```json
{
    "domain": {
        "domainid": "1"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET .HU DOMAIN DECLARATION

Command request: /domains/declaration/:id|:name/get (:id is Domain ID or :name is Domain name)

Response:

```json
{
    "domain": {
        "declare_id": 1,
        "declare_text": "Declare text",
        "declare_captcha": "Declare captcha",
        "declare_hash": "Declare hash",
        "datetime": "2015-01-01 12:00:00",
        "active": 0
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

LIST .HU DOMAIN DECLARATION

Command request: /domains/declaration/list

Response:

```
{
    "domains": [
        {
            "id": 1,
            "domain": "example.hu"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

SEND .HU DOMAIN DECLARATION

Command request: /domains/declaration/:id|:name/add (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
captcha	required	Captcha text	TEXT
ip	optional	IP address	TEXT

Response:

```json
{
    "domain": {
        "message": "PGP aláírás helyes"
    },
    "result": {
        "code": 201,
        "message": "Created"
    }
}
```

DOMAIN API documentation

OBTAIN THE EPP CODE OF A DOMAIN

Command request: /domains/epp/:id|:name (:id is Domain ID or :name is Domain name)

Response:

```
{
    "domain": {
        "eppcode": "EPPCODE"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET .HU DOMAIN CONTRACT

Command request: /domains/contract/:id|:name/get (:id is Domain ID or :name is Domain name)

Response:

```
{
    "domain": "Base64 encoded pdf file",
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

UPLOAD .HU DOMAIN CONTRACT

Command request: /domains/contract/:id|:name/add (:id is Domain ID or :name is Domain name)

Need to upload a signed file for jpg, jpeg, gif, png, pdf or zip format and extension.

Response:

```
{
    "domain": "Domain document uploaded",
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

UPLOAD .HU DOMAIN DOCUMENT

Command request: /domains/document/:id|:name/add (:id is Domain ID or :name is Domain name)

For example: extract from the trade register, specimen of signature, domain owner change etc. Files with jpg, jpeg, gif, png, pdf or zip format and extension.

Parameter	Obligation	Definition	Type
name	required	Document name	TEXT

Response:

```
{
    "domain": "Domain document uploaded",
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

LIST TANSFERRED AWAY DOMAINS

Command request: /domains/transfer/away/:tld

Command request example: /domains/transfer/away/.hu

Response:

```
{
    "away": [
        "example.hu"
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN LIST WAITING FOR INTERVENTION

Command request: /domains/todo/list

Response:

```json
{
    "domains": [
        {
            "id": 1,
            "domain": "example.hu",
            "statusCode": 23,
            "statusMsg": "Preliminary (queuing, waiting)",
            "statusUzenet": "Preliminary (queuing, waiting)"
        },
        {
            "id": 2,
            "domain": "example2.hu",
            "statusCode": 23,
            "statusMsg": "Preliminary (queuing, waiting)",
            "statusUzenet": "Preliminary (queuing, waiting)",
            "declaration": true
        },
        {
            "id": 3,
            "domain": "example.com",
            "registrantValidation": "Pending"
        },
        {
            "id": 4,
            "domain": "example2.com",
            "accountValidation": "Failed"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET DOMAIN DNSSEC DATA

Command request: /domains/dnssec/:id|:name/get (:id is Domain ID or :name is Domain name)

Command request example (DS Data Interface, digest): /domains/dnssec/example.hu/get

Response:

```json
{
    "dnssec": [
        {
            "keytag": "7949",
            "algorithm": "8",
            "algorithmName": "8 - RSA\/SHA-256",
            "digesttype": "2",
            "digesttypeName": "2 - SHA-256",
            "digest": "abcde..xyz"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

Command request example (Key Data Interface, public key): /domains/dnssec/example.eu/get

Response:

```json
{
    "dnssec": [
        {
            "flags": "257",
            "flagsName": "257 - KSK",
            "protocol": "3",
            "algorithm": "13",
            "algorithmName": " 13 - ECDSA_P256_SHA256",
            "publicKey": "abcde..xyz"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

UPDATE DOMAIN DNSSEC DATA

Command request: /domains/dnssec/:id|:name/update (:id is Domain ID or :name is Domain name)

Operation type disable or enable only available: If the domain TLD is .hu or the domain name servers are the same as the registrar's default or webhosting name servers. Enable or disable doesn't require additional parameter(s).

In case of enabling, also tries to create the DS records in the name server and send the records to the registry.

In case of disabling, after deleting all the records in the registry, it also tries to delete it in the name servers.

The # indicates an associated index.

Parameter	Obligation	Definition	Type
operation	required	Action type (disable, enable or update)	TEXT

Only DS Data Interface with update action:

Parameter	Obligation	Definition	Type
operation	required	**update**	TEXT
add_#_keytag	conditional	Key Tag, Required if add record(s)	NUMBER
add_#_algorithm	conditional	Algorithm, Required if add record(s)	NUMBER
add_#_digesttype	conditional	Digest Type, Required if add record(s)	NUMBER
add_#_digest	conditional	Digest, Required if add record(s)	TEXT
rem_#_keytag	conditional	Key Tag, Required if remove record(s)	NUMBER
rem_#_algorithm	conditional	Algorithm, Required if remove record(s)	NUMBER
rem_#_digesttype	conditional	Digest Type, Required if remove record(s)	NUMBER
rem_#_digest	conditional	Digest, Required if remove record(s)	TEXT

Only Key Data Interface with update action:

Parameter	Obligation	Definition	Type
operation	required	**update**	TEXT
add_#_flags	conditional	Flags, Required if add record(s)	NUMBER
add_#_protocol	conditional	Protocol, Required if add record(s)	NUMBER
add_#_algorithm	conditional	Algorithm, Required if add record(s)	NUMBER
add_#_publicKey	conditional	Public key, Required if add record(s)	TEXT
rem_#_flags	conditional	Flags, Required if remove record(s)	NUMBER
rem_#_protocol	conditional	Protocol, Required if remove record(s)	NUMBER
rem_#_algorithm	conditional	Algorithm, Required if remove record(s)	NUMBER
rem_#_publicKey	conditional	Public key, Required if remove record(s)	TEXT

DOMAIN API documentation

Response example (DS Data Interface, digest):

```json
{
    "dnssec": [
        {
            "keytag": "7949",
            "algorithm": "8",
            "algorithmName": "8 - RSA\/SHA-256",
            "digesttype": "2",
            "digesttypeName": "2 - SHA-256",
            "digest": "abcde..xyz"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

Response example (Key Data Interface, public key):

```json
{
    "dnssec": [
        {
            "flags": "257",
            "flagsName": "257 - KSK",
            "protocol": "3",
            "algorithm": "13",
            "algorithmName": "13 - ECDSA_P256_SHA256",
            "publicKey": "abcde..xyz"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET ZONE RECORDS

Command request: /domains/zone/:id|:name/get (:id is Domain ID or :name is Domain name)

Available types: SOA, NS, A, AAAA, CNAME, MX, TXT, SRV, Park, Redirect

General parameters:

- *name: The record name*
- *type: The DNS record type*
- *ttl: The record's Time To Live (TTL)*

SOA record parameters:

- *mname: The domain's authoritative nameserver*
- *rname: The Responsible Person's (RP's) email address*
- *serial: The zone file's revision number*
- *refresh: The amount of time to wait before the secondary DNS server queries the primary DNS server's SOA records for changes*
- *retry: The amount of time to wait before the secondary server retries a failed zone transfer*
- *expire: The amount of time to wait before the secondary server attempts to complete a zone transfer*
- *minimum: The negative result TTL (for example, how long a resolver should consider a negative result for a subdomain to be valid before retrying)*

NS record parameter:

- *nsdname: The domain's authoritative nameserver*

A record parameter:

- *address: The zone record's IPv4 address*

AAAA record parameter:

- *address: The zone record's IPv6 address*

CNAME record parameter:

- *cname: The canonical name (CNAME) alias*

MX record parameters:

- *preference: The record's preference*
- *exchange: The server's location's canonical name (CNAME)*

TXT record parameter:

- *txtdata: The record's text data*

SRV record parameters:

- *priority: The target host's preference*

43

DOMAIN API documentation

- *weight: A relative weight. The system uses this value to rank entries with the same priority value*
- *port: The target host's port*
- *target: The service's target host*

Response:

```json
{
    "domain": {
        "1": {
            "name": "example.com.",
            "type": "SOA",
            "ttl": "3600",
            "mname": "ns1.example.com.",
            "rname": "hostmaster.example.com.",
            "serial": "2015010101",
            "refresh": "86400",
            "retry": "1800",
            "expire": "604800",
            "minimum": "3600"
        },
        "2": {
            "name": "example.com.",
            "type": "NS",
            "ttl": "3600",
            "nsdname": "ns1.example.com."
        },
        "3": {
            "name": "example.com.",
            "type": "NS",
            "ttl": "3600",
            "nsdname": "ns2.example.com."
        },
        "4": {
            "name": "example.com.",
            "type": "MX",
            "ttl": "3600",
            "preference": "5",
            "exchange": "mx.example.com."
        },
        "5": {
            "name": "example.com.",
            "type": "Park",
            "ttl": "3600"
        },
        "6": {
            "name": "*.example.com.",
            "type": "Park",
            "ttl": "3600"
        }
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

UPDATE ZONE RECORDS

Command request: /domains/zone/:id|:name/modify (:id is Domain ID or :name is Domain name)

Parameter	Obligation	Definition	Type
modify	required	Deleted, unmodified and modified zone records (without new records) JSON encoded array	TEXT
new	optional	Only new records in JSON ecoded array	TEXT

Please note: If you do not send an existing record the stored record will be deleted. You have to send the whole required zone with all of the required records.

Response:

```
{
    "domain": {
        "138": {
            "name": "example.com.",
            "type": "SOA",
            "ttl": "3600",
            "mname": "ns1.examlpe.com.",
            "rname": "hostmaster.example.com",
            "serial": "2015010101",
            "refresh": "3600",
            "retry": "600",
            "expire": "604800",
            "minimum": "3600"
        },
        "1202": {
            "name": "one.example.com.",
            "type": "CNAME",
            "ttl": "3600",
            "cname": "example.com."
        },
        "1203": {
            "name": "example.com.",
            "type": "A",
            "ttl": "3600",
            "address": "127.0.0.1"
        }
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

CREATE NEW ZONE

Command request: /domains/zone/:id|:name/create (:id is Domain ID or :name is Domain name)

New zone only created if not exist yet. The zone will be created with our default records what you can modify after, with the update zone records command.

Response:

```json
{
    "domain": {
        "1": {
            "name": "example.com.",
            "type": "SOA",
            "ttl": "3600",
            "mname": "ns1.example.com.",
            "rname": "hostmaster.example.com.",
            "serial": "2015010101",
            "refresh": "86400",
            "retry": "1800",
            "expire": "604800",
            "minimum": "3600"
        },
        "2": {
            "name": "example.com.",
            "type": "NS",
            "ttl": "3600",
            "nsdname": "ns1.example.com."
        },
        "3": {
            "name": "example.com.",
            "type": "NS",
            "ttl": "3600",
            "nsdname": "ns2.example.com."
        },
        "4": {
            "name": "example.com.",
            "type": "MX",
            "ttl": "3600",
            "preference": "5",
            "exchange": "mx.example.com."
        },
        "5": {
            "name": "example.com.",
            "type": "Park",
            "ttl": "3600"
        },

    },
    "result": {
        "code": 200,
        "message": "OK"
    }


}
```

DOMAIN API documentation

GET ZONE DS RECORDS (DNSSEC)

Command request: /domains/zone/:id|:name/getds (:id is Domain ID or :name is Domain name)

Response:

```json
{
    "ds": {
        "algorithm": 13,
        "digestType": "SHA256",
        "flags ": 257,
        "publicKey": "ABCDE . . .==",
        "keyTag": 12345,
        "dsRecord": "exmple.hu. IN DS 12345 13 2 ABCDEFG",
        "digest": "ABCDEFG"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

REMOVE ZONE DS RECORDS (DNSSEC)

Command request: /domains/zone/:id|:name/removeds (:id is Domain ID or :name is Domain name)

Response:

```json
{
    "ds": [],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

CREATE ZONE DS RECORDS (DNSSEC)

Command request: /domains/zone/:id|:name/createds (:id is Domain ID or :name is Domain name)

Response:

```
{
    "ds": {
        "algorithm": 13,
        "digestType": "SHA256",
        "flags ": 257,
        "publicKey": "ABCDE . . .==",
        "keyTag": 12345,
        "dsRecord": "exmple.hu. IN DS 12345 13 2 ABCDEFG",
        "digest": "ABCDEFG"
    },
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

E-INVOICE ACTIONS

LIST E-INVOICES

Command request: /einvoices/list

Response:

```
{
    "einvoices": [
        {
            "id": "1",
            "invoiceNumber": "E-PRE-2017-1",
            "gross": "111.00",
            "net": "222.00",
            "electronic": 1,
            "paid": 1,
            "paiddate": "2017-01-01",
            "createdate": "2017-01-01",
            "completiondate": "2017-01-09",
            "paidedgross": "222.00"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

GET E-INVOICE

Command request: /einvoices/get/:invoiceid (:invoiceid is Invoice ID)

Response:

```json
{
    "einvoices": [
        {
            "id": "1",
            "invoiceNumber": "E-PRE-2017-1",
            "gross": "111.00",
            "net": "222.00",
            "electronic": 1,
            "paid": 1,
            "paiddate": "2017-01-01",
            "createdate": "2017-01-01",
            "completiondate": "2017-01-09",
            "paidedgross": "222.00"
            "file": "Base64 encoded pdf file"
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

POLL ACTIONS

REQUEST POLL

Command request: /poll/req

Response:

```json
{
    "poll": [
        {
            "id": "aaaaaaaa-bbbb-cccc-dddd-eeeeeeeeeeee",
            "count": 2,
            "type": 1,
            "event": "REGISTRATION",
            "datetime": "2001-01-01 01:01:01",
            "domain": "domain.tld",
            "msg": "Domain: domain.tld\r\nAction: Domain
registration\r\nMessage: Domain Registered Successfully\r\nExpiry date:
2020-02-30",
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

Available types and events:

Type	Event
0	NO_MESSAGES
1	REGISTRATION
2	RENEWAL
3	SUSPENSION
4	DELETE
5	TRANSFER_PENDING
6	TRANSFER_SUCCESS
6	NS_CHANGE
7	CONTACT_SAVE
9	CONTACT_CHANGE
10	TAKE_AWAY
11	HOST_ADD
12	HOST_MODIFY
13	HOST_DELETE
14	RELEASE
15	LOCK
16	DNSSEC
17	REACTIVATE
18	REMARK

DOMAIN API documentation

ACKNOWLEDGE POLL

Command request: /poll/ack/:id (:id is Poll message ID)

Response:

```
{
    " poll": [
        {
            "msg": "Command completed successfully",
            "count": 1,
        }
    ],
    "result": {
        "code": 200,
        "message": "OK"
    }
}
```

DOMAIN API documentation

RESPONSE CODES AND MESSAGES

Code	Message	Definition
200	OK	Command completed successfully
201	Created	Command completed and related resources created or modified successful
400	Bad Request	Command request failed
401	Unauthorized	Authentication failed
404	Not Found	Command or related resource not exist or missing privileges to access them
501	Not Implemented	Command not implemented yet

DOMAIN API documentation

ERROR NUMBERS AND MESSAGES

Number	Message
10000	Internal error: Client Not Found
10001	Internal error: The Client ID was not sent or not recognized
10002	A client already exists with the specified email address
10003	You can not change the specified field
10010	The specified contact does not exist or does not belong to the logged in user
10011	You can not use the specified field
10012	A contact already exists with the specified email address
10020	The specified message does not exist or does not belong to the logged in user
10030	The specified invoice does not exist or does not belong to the logged in user
10040	The specified e-invoice does not exist or does not belong to the logged in user
10100	The specified service does not exist or does not belong to the logged in user
10101	Required field (pid) missing or is not valid
10102	Field pid is not avaliable
10103	Required field (domain) missing or is not valid
10104	Required field (billingcycle) missing or is not valid
10105	Internal error: Unable to add order when client status is Closed
10106	Internal error: Invalid Payment Method. Valid options include
10107	Internal error: Addon ID invalid
10108	Internal error: Service ID not owned by Client ID provided
10109	Internal error: Domain status is set to 'Pending|Cancelled|Pending Transfer|Transferred|Fraud' and cannot be renewed
10110	Internal error: Domain not owned by Client ID provided
10111	Internal error: No items added to cart so order cannot proceed
10112	Addons field is not valid
10113	Required field (newproductid) missing or is not valid
10114	Field newproductid is not available
10115	Required field (newproductbillingcycle) missing or is not valid
10116	Internal error: Service ID Not Found
10117	Unable to accept upgrade order. Previous upgrade invoice for service is still unpaid.
10118	Internal error: Invalid Payment Method. Valid options include xxx
10119	Internal error: Invalid Upgrade Type
10120	Required field (type) missing or is not valid
10121	Existing Cancellation Request Exists
10200	Internal error: Domain Not Found
10201	Internal error: Not found proper method for getting domain nameservers
10202	Internal error: Registry error via getting domain nameservers
10203	Internal error: Unrecognized error via getting domain nameservers
10204	Internal error: No available worker for this job.
10205	Internal error: Missing status in the gearman result fot this job.
10206	Internal error: Registry error via getting domain epp/auth code
10207	Internal error: Registry error via getting domain contact details
10208	Internal error: Registry error via updating domain nameservers.
10209	NS1 and NS2 required.

DOMAIN API documentation

10210	Internal error: Unrecognized error via updating domain nameservers.
10211	Internal error: Unrecognized error occurred when defining a job.
10212	Required field autorenewal missing or is not valid.
10213	Internal error via updating domain renewal.
10214	Required field years missing or is not valid
10215	Unable to add order when client status is Closed
10216	Internal error: Invalid Payment Method. Valid options include
10217	Addon ID invalid
10218	Service ID not owned by Client ID provided
10219	Domain status is set to 'Pending|Cancelled|Pending Transfer|Transferred|Fraud' and cannot be renewed
10220	Domain not owned by Client ID provided
10221	No items added to cart so order cannot proceed
10222	Some domains are not yours or wrong renewal years
10223	Not found renewable domains
10224	Internal error via updating domain lock status.
10225	Required field lockstatus missing or is not valid
10226	Registrar/transfer lock not supported.
10227	Registrar Lock Disabled by previous registrant changes.
10228	Unrecognized error via updating domain lock status
10229	Internal error via getting domain epp/transfer code
10230	Unrecognized error via getting domain information
10231	Required field type missing
10232	The contact type field is not valid
10233	Not found contact modification
10234	Unrecognized error via getting domain contact information
10235	Internal error via updating domain contact
10236	Internal error: Registry error via updating domain contact
10237	Declaration only acceptable .hu domain
10238	Getting domain declaration failed
10239	IP address is not valid
10240	Required field captcha missing or is not valid
10241	Zone is not exist
10242	Getting domain zone failed
10243	Creating domain zone failed
10244	Modifying domain zone failed
10245	Getting domain contract failed
10246	Required uploaded document missing
10247	Maximum file size 4194304 bytes
10248	Only jpg, gif, jpeg, png, pdf, zip file upload allowed
10249	Required field name missing
10250	Uploading domain document failed
10251	Getting domain info failed
10252	TLD not allowed
10253	TLD not exist
10254	Required field domain missing or is not valid

DOMAIN API documentation

10255	Years not valid or not acceptable
10256	Registered domain is not available for registration
10257	Only registered domain available for transfer
10258	Domain already exist's
10259	EPP code required
10260	Domain name not valid or not acceptable
10261	Domain name required
10262	Required field ns1 missing or is not valid
10263	Required field owner missing or is not valid
10264	Required field admin missing or is not valid
10265	Required field tech missing or is not valid
10266	You did not enter owner/registrant phone number
10267	Owner/registrant phone number is not valid, valid format +36123456789
10268	You did not enter admin/on site contact phone number
10269	Admin/on site contact phone number is not valid, valid format +36123456789
10270	You did not enter tech contact phone number
10271	Tech contact phone number is not valid, valid format +36123456789
10272	The V.A.T. number provided for owner/registrant in the tax number field does not comply with the format (format: 12345678-1-01)!
10273	The V.A.T. number provided for admin/on site contact in the tax number field does not comply with the format (format: 12345678-1-01)!
10274	The V.A.T. number provided for tech contact in the tax number field does not comply with the format (format: 12345678-1-01)!
10275	The EU V.A.T. number provided for owner/registrant in the tax number field does not comply with the format (format: HU12345678)!
10276	The EU V.A.T. number provided for admin/on site contact in the tax number field does not comply with the format (format: HU12345678)!
10277	The EU V.A.T. number provided for tech contact in the tax number field does not comply with the format (format: HU12345678)!
10278	You did not enter owner/registrant last name
10279	You did not enter admin/on site contact last name
10280	You did not enter tech contact last name
10281	You did not enter owner/registrant first name
10282	You did not enter admin/on site contact first name
10283	You did not enter tech contact first name
10284	You did not enter owner/registrant e-mail address
10285	You did not enter admin/on site contact e-mail address
10286	You did not enter tech contact e-mail address
10287	The e-mail address you entered for owner/registrant was not valid
10288	The e-mail address you entered for admin/on site contact was not valid
10289	The e-mail address you entered for tech contact was not valid
10290	You did not enter owner/registrant address
10291	You did not enter admin/on site contact address
10292	You did not enter tech contact address
10293	You did not enter owner/registrant city
10294	You did not enter admin/on site contact city
10295	You did not enter tech contact city
10296	You did not enter owner/registrant postcode

DOMAIN API documentation

10297	*You did not enter admin/on site contact postcode*
10298	*You did not enter tech contact postcode*
10299	*You did not enter owner/registrant country*
10300	*You did not enter admin/on site contact country*
10301	*You did not enter tech contact country*
10302	*The domain owner may only be a person or organsation registered in the EU, EGT, EFTA or Hungary neighbour country*
10303	*You did not enter identity number*
10304	*You did not enter EU vat number*
10305	*You did not enter your vat number*
10306	*Administrator contact person must have a hungarian address*
10307	*The domain owner may only be a person or organsation registered in the EU*
10308	*Unknown owner/registrant country*
10309	*Unknown admin/on site contact country*
10310	*Unknown tech contact country*
10311	*Unknown owner/registrant address*
10312	*Unknown admin/on site contact address*
10313	*Unknown tech contact address*
10314	*Multiple owner/registrant addresses*
10315	*Multiple admin/on site contact addresses*
10316	*Multiple tech contact addresses*
10317	*Missing owner/registrant street*
10318	*Missing admin/on site contact street*
10319	*Missing tech contact street*
10320	*Different owner/registrant state*
10321	*Different admin/on site contact state*
10322	*Different tech contact state*
10323	*Unknown owner/registrant city*
10324	*Unknown admin/on site contact city*
10325	*Unknown tech contact city*
10326	*Different owner/registrant city*
10327	*Different admin/on site contact city*
10328	*Different tech contact city*
10329	*Unknown owner/registrant postcode*
10330	*Unknown admin/on site contact postcode*
10331	*Unknown tech contact postcode*
10332	*Different owner/registrant postcode*
10333	*Different admin/on site contact postcode*
10334	*Different tech contact postcode*
10335	*Years not valid or not acceptable! Minimum 2 years need!*
10336	*Transfer not allowed until domain state is: Bejegyzett feltételes használat*
10337	*Unrecognized error via submitting domain declaration*
10338	*Unrecognized error via domain hold*
10339	*Unrecognized error via domain unhold*
10340	*Unrecognized error via domain delete*
10341	*Unrecognized error via domain renew*

DOMAIN API documentation

10342	*Unrecognized error via domain transfer*
10343	*Unrecognized error via add contact*
10344	*Unrecognized error via modify contact*
10345	*Unrecognized error via check contact*
10346	*Unrecognized error via domain update*
10347	*Unrecognized error via update domain contacts*
10348	*Unrecognized error via register domain*
10349	*Unrecognized error via validating declaration*
10350	*Unrecognized error via getting declaration*
10351	*Unrecognized error via getting suggestion info*
10352	*Unrecognized error via getting domain availability*
10353	*Unrecognized error via adding domain description*
10354	*Unrecognized error via getting registrar information*
10355	*Unrecognized error via acknowledging poll message*
10356	*Unrecognized error via transfering domain*
10357	*Registrant contact approval is pending*
10358	*Missing authcode*
10359	*Missing contact*
10360	*Domain status missing*
10361	*Failed renew from delete*
10362	*Failed backorder registration*
10363	*Documents for this domain are not accepted*
10364	*NS change is pending*
10365	*Register namserver is pending*
10366	*Renew is pending*
10367	*Registration is pending*
10368	*Transfer is pending*
10369	*Domain delete is pending*
10370	*Getting domain zone DS records failed*
10371	*Removing domain zone DS records failed*
10372	*Creating domain zone DS records failed*
10373	*Required field operation missing or is not valid*
10374	*Domain status is not Active*
10375	*Domain TLD or assigned registry/regisrar not supported*
10376	*Update domain DNSSEC failed*
10377	*Not found DNNSEC modification*
10378	*Required field (id) missing or is not valid*
10379	*In case of .hu domains, if the administrative contact and the domain owner are different, the admin contact must be an organization*
10380	*In case of .hu domains, if the technical contact and the domain owner are different, the admin contact must be an organization*
10400	*Uninplemented error*
10401	*Authentication failed*
10402	*Internal error: Referenced API call not found*
10403	*Internal error: Referenced API call hangs out*
10404	*Entry point not found*

DOMAIN API documentation

10501	Command for this entry point not implemented yet
10600	Internal error: GEARMAN_SUCCESS Success
10601	Internal error: GEARMAN_IO_WAIT Blocking IO was found.
10602	Internal error: GEARMAN_SHUTDOWN This is a server only related error, and will not be found in any client or worker return.
10603	Internal error: GEARMAN_SHUTDOWN_GRACEFUL This is a server only related error, and will not be found in any client or worker return.
10604	Internal error: GEARMAN_ERRNO System error occurred.
10605	Internal error: GEARMAN_EVENT This is a server only related error, and will not be found in any client or worker return.
10606	Internal error: GEARMAN_TOO_MANY_ARGS
10607	Internal error: GEARMAN_NO_ACTIVE_FDS No active connections were available.
10608	Internal error: GEARMAN_INVALID_MAGIC
10609	Internal error: GEARMAN_INVALID_COMMAND Wrong command sent in response to ECHO request
10610	Internal error: GEARMAN_INVALID_PACKET Packet not complete
10611	Internal error: GEARMAN_UNEXPECTED_PACKET
10612	Internal error: GEARMAN_GETADDRINFO Name resolution failed for a host.
10613	Internal error: GEARMAN_NO_SERVERS No servers have been provided for the client/worker.
10614	Internal error: GEARMAN_LOST_CONNECTION Connection was lost to the given server.
10615	Internal error: GEARMAN_MEMORY_ALLOCATION_FAILURE Memory allocation failed.
10616	Internal error: GEARMAN_JOB_EXISTS Gearman client job status has been called and the Job is currently known by a server, but is not being run by a worker.
10617	Internal error: GEARMAN_JOB_QUEUE_FULL This is a server only error code.
10618	Internal error: GEARMAN_SERVER_ERROR An error occurred on the server.
10619	Internal error: GEARMAN_WORK_ERROR A task has had an error and will be retried.
10620	Internal error: GEARMAN_WORK_DATA Worker has sent a chunked piece of data to the client
10621	Internal error: GEARMAN_WORK_WARNING Worker has issued a warning to the client
10622	Internal error: GEARMAN_WORK_STATUS Status has been updated by the worker
10623	Internal error: GEARMAN_WORK_EXCEPTION Worker has sent an exception the client
10624	Internal error: GEARMAN_WORK_FAIL A task has failed, and the worker has exited with an error
10625	Internal error: GEARMAN_NOT_CONNECTED Client/Worker is not currently connected to the server.
10626	Internal error: GEARMAN_COULD_NOT_CONNECT Server name was valid, but a connection could not be made.
10627	Internal error: GEARMAN_SEND_IN_PROGRESS This is a server only related error, and will not be found in any client or worker return.
10628	Internal error: GEARMAN_RECV_IN_PROGRESS This is a server only related error, and will not be found in any client or worker return.
10629	Internal error: GEARMAN_NOT_FLUSHING Gearman task send workload failed, it was not in the correct state.
10630	Internal error: GEARMAN_DATA_TOO_LARGE Gearman task send workload failed, the data was too large to be sent.
10631	Internal error: GEARMAN_INVALID_FUNCTION_NAME A worker was sent a request for a job that it did not have a valid function for.
10632	Internal error: GEARMAN_INVALID_WORKER_FUNCTION No callback was provided by the worker for a given function.

DOMAIN API documentation

10634	Internal error: GEARMAN_NO_REGISTERED_FUNCTIONS The worker has not registered any functions.
10635	Internal error: GEARMAN_NO_JOBS No jobs were found for the worker. This error code is only returned if the worker has just made a request for jobs.
10636	Internal error: GEARMAN_ECHO_DATA_CORRUPTION Echo was unsuccessful because the data was returned corrupted.
10637	Internal error: GEARMAN_NEED_WORKLOAD_FN A client was asked for work, but no callback was specified.
10638	Internal error: GEARMAN_PAUSE Used only in custom application for client can be used to check for this value.
10639	Internal error: GEARMAN_UNKNOWN_STATE The gearman return was never set.
10640	Internal error: GEARMAN_PTHREAD This is a server only related error, and will not be found in any client or worker return.
10641	Internal error: GEARMAN_PIPE_EOF Server only error.
10642	Internal error: GEARMAN_QUEUE_ERROR This is a server only related error, and will not be found in any client or worker return.
10643	Internal error: GEARMAN_FLUSH_DATA Internal state, should never be seen by either client or worker.
10644	Internal error: GEARMAN_SEND_BUFFER_TOO_SMALL Send buffer was too small.
10645	Internal error: GEARMAN_IGNORE_PACKET Internal only
10646	Internal error: GEARMAN_UNKNOWN_OPTION Default state of task return value.
10647	Internal error: GEARMAN_TIMEOUT A timeout occurred when making a request to the server.
10652	Internal error: GEARMAN_MAX_RETURN
10701	Internal error: The URL you passed to libcurl used a protocol that this libcurl does not support. The support might be a compile-time option that you didn't use, it can be a misspelled protocol string or just a protocol libcurl has no code for.
10702	Internal error: Very early initialization code failed. This is likely to be an internal error or problem, or a resource problem where something fundamental couldn't get done at init time.
10703	Internal error: The URL was not properly formatted.
10704	Internal error: A requested feature, protocol or option was not found built-in in this libcurl due to a build-time decision. This means that a feature or option was not enabled or explicitly disabled when libcurl was built and in order to get it to function you have to get a rebuilt libcurl.
10705	Internal error: Couldn't resolve proxy. The given proxy host could not be resolved.
10706	Internal error: Couldn't resolve host. The given remote host was not resolved.
10707	Internal error: Failed to connect() to host or proxy.
10708	Internal error: The server sent data libcurl couldn't parse. This error code is used for more than just FTP and is aliased as CURLE_WEIRD_SERVER_REPLY since 7.51.0.
10709	Internal error: We were denied access to the resource given in the URL. For FTP, this occurs while trying to change to the remote directory.
10710	Internal error: While waiting for the server to connect back when an active FTP session is used, an error code was sent over the control connection or similar.
10711	Internal error: After having sent the FTP password to the server, libcurl expects a proper reply. This error code indicates that an unexpected code was returned.
10712	Internal error: During an active FTP session while waiting for the server to connect, the CURLOPT_ACCEPTTIMEOUT_MS (or the internal default) timeout expired.
10713	Internal error: libcurl failed to get a sensible result back from the server as a response to either a PASV or a EPSV command. The server is flawed.
10714	Internal error: FTP servers return a 227-line as a response to a PASV command. If libcurl fails to parse that line, this return code is passed back.
10715	Internal error: An internal failure to lookup the host used for the new connection.
10716	Internal error: A problem was detected in the HTTP2 framing layer. This is somewhat generic and can be one out of several problems, see the error buffer for details.

10717	Internal error: Received an error when trying to set the transfer mode to binary or ASCII.
10718	Internal error: A file transfer was shorter or larger than expected. This happens when the server first reports an expected transfer size, and then delivers data that doesn't match the previously given size.
10719	Internal error: This was either a weird reply to a 'RETR' command or a zero byte transfer complete.
10721	Internal error: When sending custom 'QUOTE' commands to the remote server, one of the commands returned an error code that was 400 or higher (for FTP) or otherwise indicated unsuccessful completion of the command.
10722	Internal error: This is returned if CURLOPT_FAILONERROR is set TRUE and the HTTP server returns an error code that is >= 400.
10723	Internal error: An error occurred when writing received data to a local file, or an error was returned to libcurl from a write callback.
10725	Internal error: Failed starting the upload. For FTP, the server typically denied the STOR command. The error buffer usually contains the server's explanation for this.
10726	Internal error: There was a problem reading a local file or an error returned by the read callback.
10727	Internal error: A memory allocation request failed. This is serious badness and things are severely screwed up if this ever occurs.
10728	Internal error: Operation timeout. The specified time-out period was reached according to the conditions.
10730	Internal error: The FTP PORT command returned error. This mostly happens when you haven't specified a good enough address for libcurl to use. See CURLOPT_FTPPORT.
10731	Internal error: The FTP REST command returned error. This should never happen if the server is sane.
10733	Internal error: The server does not support or accept range requests.
10734	Internal error: This is an odd error that mainly occurs due to internal confusion.
10735	Internal error: A problem occurred somewhere in the SSL/TLS handshake. You really want the error buffer and read the message there as it pinpoints the problem slightly more. Could be certificates (file formats, paths, permissions), passwords, and others.
10736	Internal error: The download could not be resumed because the specified offset was out of the file boundary.
10737	Internal error: A file given with FILE:// couldn't be opened. Most likely because the file path doesn't identify an existing file. Did you check file permissions?
10738	Internal error: LDAP cannot bind. LDAP bind operation failed.
10739	Internal error: LDAP search failed.
10741	Internal error: Function not found. A required zlib function was not found.
10742	Internal error: Aborted by callback. A callback returned 'abort' to libcurl.
10743	Internal error: Internal error. A function was called with a bad parameter.
10745	Internal error: Interface error. A specified outgoing interface could not be used. Set which interface to use for outgoing connections' source IP address with CURLOPT_INTERFACE.
10747	Internal error: Too many redirects. When following redirects, libcurl hit the maximum amount. Set your limit with CURLOPT_MAXREDIRS.
10748	Internal error: An option passed to libcurl is not recognized/known. Refer to the appropriate documentation. This is most likely a problem in the program that uses libcurl. The error buffer might contain more specific information about which exact option it concerns.
10749	Internal error: A telnet option string was Illegally formatted.
10752	Internal error: Nothing was returned from the server, and under the circumstances, getting nothing is considered an error.
10753	Internal error: The specified crypto engine wasn't found.
10754	Internal error: Failed setting the selected SSL crypto engine as default!

DOMAIN API documentation

10755	Internal error: Failed sending network data.
10756	Internal error: Failure with receiving network data.
10758	Internal error: problem with the local client certificate.
10759	Internal error: Couldn't use specified cipher.
10760	Internal error: The remote server's SSL certificate or SSH md5 fingerprint was deemed not OK. This error code has been unified with CURLE_SSL_CACERT since 7.62.0. Its previous value was 51.
10761	Internal error: Unrecognized transfer encoding.
10762	Internal error: Invalid LDAP URL.
10763	Internal error: Maximum file size exceeded.
10764	Internal error: Requested FTP SSL level failed.
10765	Internal error: When doing a send operation curl had to rewind the data to retransmit, but the rewinding operation failed.
10766	Internal error: Initiating the SSL Engine failed.
10767	Internal error: The remote server denied curl to login (Added in 7.13.1)
10768	Internal error: File not found on TFTP server.
10769	Internal error: Permission problem on TFTP server.
10770	Internal error: Out of disk space on the server.
10771	Internal error: Illegal TFTP operation.
10772	Internal error: Unknown TFTP transfer ID.
10773	Internal error: File already exists and will not be overwritten.
10774	Internal error: This error should never be returned by a properly functioning TFTP server.
10775	Internal error: Character conversion failed.
10776	Internal error: Caller must register conversion callbacks.
10777	Internal error: Problem with reading the SSL CA cert (path? access rights?)
10778	Internal error: The resource referenced in the URL does not exist.
10779	Internal error: An unspecified error occurred during the SSH session.
10780	Internal error: Failed to shut down the SSL connection.
10781	Internal error: Socket is not ready for send/recv wait till it's ready and try again. This return code is only returned from curl_easy_recv and curl_easy_send (Added in 7.18.2)
10782	Internal error: Failed to load CRL file (Added in 7.19.0)
10783	Internal error: Issuer check failed (Added in 7.19.0)
10784	Internal error: The FTP server does not understand the PRET command at all or does not support the given argument. Be careful when using CURLOPT_CUSTOMREQUEST, a custom LIST command will be sent with PRET CMD before PASV as well. (Added in 7.20.0)
10785	Internal error: Mismatch of RTSP CSeq numbers.
10786	Internal error: Mismatch of RTSP Session Identifiers.
10787	Internal error: Unable to parse FTP file list (during FTP wildcard downloading).
10788	Internal error: Chunk callback reported error.
10789	Internal error: (For internal use only, will never be returned by libcurl) No connection available, the session will be queued. (added in 7.30.0)
10790	Internal error: Failed to match the pinned key specified with CURLOPT_PINNEDPUBLICKEY.
10791	Internal error: Status returned failure when asked with CURLOPT_SSL_VERIFYSTATUS.
10792	Internal error: Stream error in the HTTP/2 framing layer.
10793	Internal error: An API function was called from inside a callback.

10800	*Uninplemented .hu regcheck error*
10801	*At least one NS record points to a CNAME. NS records must point to A records*
10802	*The domain nameservers are on the same network. Nameservers must be either in a different C class IP block, or on a different network*
10803	*The SOA records differ on the nameservers. The two nameservers must contain the same SOA records*
10804	*At least one nameserver doesn't serve the domain*
10805	*The domains MX record does not point to an A record. MX records can only point to A records*
10806	*No A record can be found for at least one nameserver*
10807	*The SOA record contains a nameserver, that is not listed in an NS record*
10808	*The domain does not contain any MX or A records. To validate the postmaster@domain address at least one must exist*
10809	*The domain name does not have a secondary nameserver*
10810	*At least one nameserver has no A record*
10811	*The domain has no postmaster@domainname e-mail address*
10812	*At least one nameserver is not authoritative for the domain*
10813	*At least one IPv6 nameserver is not authoritative for the domain*
10814	*Could not get the SOA record for the domain*
10815	*The SOA record has a syntax error*
10816	*The e-mail address specified in the SOA record does not exist*
10817	*Unable to send mail to the e-mail address provided in the SOA record because the domain does not exist or has invalid configuration*
10818	*The verification procedure failed due to a timeout*
22000	*Registry error: Unknown command*
22001	*Registry error: Command syntax error*
22002	*Registry error: Command use error*
22003	*Registry error: Required parameter missing*
22004	*Registry error: Parameter value range error*
22005	*Registry error: Parameter value syntax error*
22100	*Registry error: Unimplemented protocol version*
22101	*Registry error: Unimplemented command*
22102	*Registry error: Unimplemented option*
22103	*Registry error: Unimplemented extension*
22104	*Registry error: Billing failure*
22105	*Registry error: Object is not eligible for renewal*
22106	*Registry error: Object is not eligible for transfer*
22200	*Registry error: Authentication error*
22201	*Registry error: Authorization error*
22202	*Registry error: Invalid authorization information*
22300	*Registry error: Object pending transfer*
22301	*Registry error: Object not pending transfer*
22302	*Registry error: Object exists*
22303	*Registry error: Object does not exist*
22304	*Registry error: Object status prohibits operation*
22305	*Registry error: Object association prohibits operation*
22306	*Registry error: Parameter value policy error*
22307	*Registry error: Unimplemented object service*

DOMAIN API documentation

22308	*Registry error: Data management policy violation*
22400	*Registry error: Command failed*
22500	*Registry error: Command failed; server closing connection*
22501	*Registry error: Authentication error; server closing connection*
22502	*Registry error: Session limit exceeded; server closing connection*
30000	*XML decoding error*
30002	*You have no right to perform locking operations*
30007	*ATTRIBUTES definition is missing*
30100	*The domain name is too long*
30101	*The domain name is missing*
30102	*The domain name is incorrect*
30103	*The domain name is incorrect*
30104	*The domain name is incorrect*
30105	*The domain name is incorrect*
30106	*The domain name is incorrect*
30107	*The domain name is incorrect*
30108	*The domain name is incorrect*
30109	*The starting name server name is missing*
30110	*The starting name server name is incorrect*
30111	*The starting name server name is incorrect*
30112	*The starting name server name is incorrect*
30113	*The starting name server name is incorrect*
30114	*The starting name server name is too short*
30115	*The starting name server name is incorrect*
30116	*The domain is already registered*
30117	*Domain registration already in progress*
30118	*Only 1 registration attempt is allowed every 5 seconds*
31001	*The domain object is missing*
31011	*Owner's English name is missing*
31012	*Owner's country code is missing*
31013	*Owner's zipcode is missing*
31014	*Owner's city is missing*
31015	*Owner's street is missing*
31016	*Owner's house number missing*
31017	*Owner's phone number is missing*
31018	*Owner's e-mail address is invalid*
31019	*The owner can not be identified as an organization or as a person*
31020	*The owner can not be identified by the NIC ID*
31021	*Owner's NIC ID is missing*
31022	*The owner can not be identified by ALIAS*
31023	*The owner's ALIAS id was not specified*
31030	*Owner's country code must be an EU Member State*
31040	*Admin contact NIC ID does not exist*
31041	*Admin contact NIC ID is missing*
31042	*The given admin contact alias does not exist or is unclear*

DOMAIN API documentation

31043	The admin-alias id is missing
31044	Only one Administrator can be specified
31045	The details of the Admin person are missing
31050	Tech contact NIC ID does not exist
31051	Tech contact NIC ID is missing
31052	The given tech contact alias does not exist or is unclear
31053	The tech contact alias id is missing
31054	Only one technical contact can be specified
31055	The details of the Tech contact are missing
31060	Zone contact NIC ID does not exist
31061	Admin contact NIC ID is missing
31062	The given zone contact alias does not exist or is unclear
31063	The zone contact alias id is missing
31064	Only one zone contact can be specified
31065	The details of the Zone contact are missing
31100	Admin contact surname is missing
31101	Admin contact forename is missing
31102	Admin contact country code is missing
31103	Admin contact zip code is missing
31104	Admin contact city is missing
31105	Admin contact street is missing
31106	Admin contact house number is missing
31107	Admin contact phone number is missing
31108	Admin contact e-mail address is missing
31109	Admin contact e-mail address is invalid
31130	Admin contact country code is invalid
31200	Zone contact surname is missing
31201	Zone contact forename is missing
31202	Zone contact country code is missing
31203	Zone contact zip code is missing
31204	Zone contact city is missing
31205	Zone contact street is missing
31206	Zone contact house number is missing
31207	Zone contact phone number is missing
31208	Zone contact e-mail address is missing
31209	Zone contact e-mail address is invalid
31230	Zone contact country code is invalid
31300	Tech contact surname is missing
31301	Tech contact forename is missing
31302	Tech contact country code is missing
31303	Tech contact zip code is missing
31304	Tech contact city is missing
31305	Tech contact street is missing
31306	Tech contact house number is missing
31307	Tech contact phone number is missing

DOMAIN API documentation

31308	Tech contact e-mail address is missing
31309	Tech contact e-mail address is invalid
31330	Tech contact country code is invalid
32000	Domain does not exist
32001	Permission denied
32002	Permission denied
32003	Permission denied
32004	Permission denied
32100	E-mail address is invalid
32101	The name is too short
32102	Domain has already received a complaint, only a complaint can be filed
32103	Compliant can be filled only if the domain status is conditional registered
32109	An error has occurred during the compliant creation
32110	The RQDOMAIN can be 0 or 1
32111	The complaint does not indicate whether the complainant requires a domain name or not
32112	The complainant's e-mail address is incorrect
32113	The complainant's e-mail address is missing
32114	The complainant's name is too short
32115	The complainant's name is missing
32116	Complaint can be only submitted to a domain that is in a state of complaint status
32117	Domain does not exist
32118	The domain ID must be a number
32119	Domain ID is missing
32120	The definition of the complaint is missing
32121	The domain ID for the deletion request is missing
32122	The domain ID must be a number
32123	Domain does not exist
32124	The domain name is handled by another registrar
32125	Deletion can be requested only in waiting for compliant or compliant sent status
32126	System error has occurred
32127	The complaint already contains justification
32128	The complaint ID must be number
32129	Domain and Complaint ID can not be submitted a same time
32130	The domain name is handled by another registrar
32131	The complaint must contain a justification or a file
32132	The domain and complaint ID are misssing
32133	The complaint does not exist
32134	The compliant already contain file
32140	The complaint did not contain a valid justification or file
32141	The complaint belongs to another registrar
32142	The deletion request definition is missing
32143	The RQPASSWORD definition is missing
32160	The RQPADDRESS definition is missing
32170	The RQBILLING definition is missing
32172	The BILLEMAIL format is incorrect, It can be a valid email address.

DOMAIN API documentation

32173	The EMAILINVOICE format is incorrect, It can be a valid email address.
32174	System error has occurred
32180	The RQPADDRESS definition is missing
32181	The RQEMAIL definition is missing
32182	The DELEGATION format is incorrect, It can be a valid email address.
32183	The DNSERROR format is incorrect, It can be a valid email address.
32184	The SUBMISSION format is incorrect, It can be a valid email address.
32199	System error, domain not found
32200	The date format is incorrect. Expected format: yyyy-MM-dd HH: mm: ss
32201	The DATETIME definition is missing
32202	No registration date in this state
32203	The domain name is handled by another registrar
32204	Domain does not exist
32205	The domain ID must be a number
32206	The DOMAIN_ID definition is missing
32207	The DATEMOD definition is missing
32208	The DATEMOD definition is missing
32209	The STATUSMOD definition is missing
32210	The domain is not in evaluated status
32211	The domain name is handled by another registrar
32212	Domain does not exist
32213	The domain ID must be a number
32220	The domain is not in change requested status
32231	The Zone contact belongs to another registrar
32232	The Tech contact belongs to another registrar
32233	The Admin contact belongs to another registrar
32234	The Owner belongs to another registrar
32700	The password must be at least 8 characters long and must contain at least one number, small and capital letter
33000	Object ID is missing
33001	Attribute is missing
33002	Object ID is missing
33003	The attributes does not all refer to the same object
33004	The modified value is missing
33005	Permission denied
33006	The Admin contact does not exist
33007	The owner does not exist
33008	You cannot transfer a domain that is managed by you
33009	Permission denied
33010	Permission denied
33011	The specified registrar does not exist
33012	The Tech contact does not exist
33013	The zone contact does not exist
33014	Permission denied
33015	The expiration date format is incorrect. yyyy-MM-dd
33016	The specified attribute cannot be changed in the domain object at this time

DOMAIN API documentation

33017	*The value of domain_replace_lockauth is only N or Y*
33018	*The specified attribute cannot be changed in the domain object*
33019	*Permission denied*
33020	*Permission denied*
33021	*Permission denied*
33022	*The requested change cannot be made due to DNS or authorization issues*
33023	*Admin contact error*
33024	*Zone contact error*
33025	*Tech contact error*
33026	*Domain data error*
33027	*Domain state error*
33028	*Domain already registered*
33029	*Domain registration in progress*
33030	*Owner error*
33031	*Domain not found*
33109	*The initial nameserver name is missing*
33110	*The initial nameserver name is incorrect*
33111	*The initial nameserver name is incorrect*
33112	*The initial nameserver name is incorrect*
33113	*The initial nameserver name is incorrect*
33114	*The initial name server name is too short*
33115	*The initial nameserver name is incorrect*
33200	*The specified attribute cannot be changed in the domain object at this time. It can only be configured for domains with locked and pending changes.*
33201	*To enable the lock, you need at least 1 authorizer*
33202	*The specified values do not change the status of domain_lock*
33203	*Domain_lock can only be N or Y*
33204	*The specified attribute cannot be changed in the domain object at this time*
33205	*There is no right to lock the domain or modify a locked domain*
33300	*The domain_ds_flag can only be 0 or 1*
33301	*The specified attribute cannot be changed in the domain object at this time*
33400	*The domain_desired_ds is invalid*
33401	*The specified attribute cannot be changed in the domain object at this time*
33500	*The attribute to modify is missing*
33501	*The identifier of the object to modify is missing*
33502	*The specified attributes do not all refer to the same object*
33503	*The modified value is missing*
33504	*The email address is incorrect*
33505	*The specified attribute cannot be changed in the person object at this time*
33506	*Permission denied*
33507	*The new data is incorrect*
33508	*The person not found*
33530	*The country code is incorrect*
33600	*The attribute to modify is missing*
33601	*The identifier of the object to modify is missing*
33602	*The specified attributes do not all refer to the same object*

DOMAIN API documentation

33603	The modified value is missing
33604	The email address is incorrect
33605	The specified attribute cannot be changed in the organization object at this time
33606	Permission denied
33607	The new data is incorrect
33608	The organization could not be found
33630	The country code you entered is not an EU country
33700	The attribute to modify is missing
33701	The identifier of the object to modify is missing
33702	The specified attributes do not all refer to the same object
33703	The modified value is missing
33704	The email address is incorrect
33705	The owner organization does not exist
33706	The specified attribute cannot be changed in the role object at this time
33707	Permission denied
33708	The new data is incorrect
33709	The role not found
33730	The country code is incorrect
33800	The object type cannot be interpreted
34000	The object type does not exist or cannot be searched
34001	The domain specified in org_domain_name does not exist
34002	The domain specified in person_domain_name does not exist
34003	The domain specified in role_domain_name does not exist
34004	The role_domain_name, person_domain_name, and org_domain_name can only be used with = opertator
34005	In the case of a search, the values of ATTRNAME and VALUE are mandatory
34006	Operator can only be lt,gt,=,like
34007	Searches can only contain OBJ items
34008	Authin can only be searched for a specific domain
34009	Lockauth is only searchable for a specific domain
34010	You can only search for one object type at a time
34011	The search defines a non-searchable object type
34020	In case of a complaint, only compl_id and compl_domain_hun_id can be searched
34022	The domain marked by compl_domain_hun_id does not exist
34023	The domain designated by compl_domain_hun_id is not in any of the complainant states
34024	The value of compl_id can only be a number
34025	The value of compl_domain_hun_id can only be a number
34088	Complaints can only be searched for by entering the domain ID (compl_domain_hun_id)
35000	Person last name is missing
35001	Person first name is missing
35002	Person country code is missing
35003	Person postcode is missing
35004	Person city is missing
35005	Person street is missing
35006	Person house number is missing

DOMAIN API documentation

35007	Person phone number is missing
35008	Person email format is incorrect
35009	Person email address is missing
35010	Cannot create person
35011	Cannot create person
35012	Person definition is missing
35030	Person country code is incorrect
35100	Owner english name is missing
35101	Owner country code is missing
35102	Owner postcode is missing
35103	Owner city is missing
35104	Owner street is missing
35105	Owner house number is missing
35106	Owner phone number is missing
35107	Owner email format is incorrec
35108	Cannot create owner
35109	Cannot create owner
35110	The owner cannot be identified as an organization or person
35111	Owner definition is missing
35130	The owner country code is not an EU Member State
35200	The domain referenced by the object identifier could not be found
35201	The domain referenced by the object ID does not have permission to flag
35202	Object ID missing
35203	The subject is missing
35205	The BASE64 encoded data is incorrect
35206	The BASE64 encoded data is incorrect
35207	The type of uploaded file can only be zip, rar, gif, jpg, png or pdf
35208	The type of uploaded file is missing
35209	The description cannot be created
35210	The description definition is missing
35300	The email address format is incorrect
35301	The email address is missing
35302	The phone number format is incorrect
35303	The phone number is missing
35304	The authorizer cannot be created
35305	Up to 5 authorizer can be applied
35306	You can only add an approver for a domain under modification, only in incomplete condition and prior request for change of authorizer
35307	You can only add an authorizer to a domain that is either a delegated or a modification request incomplete statuses
35308	You can only add an authorizer to your own managed domain
35309	Domain not found
35310	The Domain ID format is invalid
35400	For a locked domain, the last authorizer cannot be deleted
35401	The authorizer can not be deleted
35402	The authorizer does not belong to the designated domain

DOMAIN API documentation

35403	Authorizer not found or has been deleted
35404	In the case of a domain under modification, only an incomplete condition and a prior request for the replacement of the authorizer can be deleted
35405	Only a delegated or modification request incomplete status of domain status my be possible to delete authorizer
35406	Only a self-managed domain can be delete the authorizer
35407	Domain not found
35408	The lockauth id is missing or the format is incorrect
35409	The domain id is missing or the format is incorrect
35500	Failed approval
35501	The domain is not waiting for approval
35502	Only a self-managed domain can be approved
35503	The domain you want to approve does not exist
35504	The domain id is missing or the format is incorrect
35600	Role name is missing
35601	Role owner is missing
35602	Role support persons names are missing
35603	Role country code is missing
35604	Role postcode is missing
35605	Role city is missing
35606	Role street is missing
35607	Role house number is missing
35608	Role phone number is missing
35609	Role e-mail address is incorrect
35610	Role e-mail address is missing
35611	The role cannot be created
35612	The role cannot be created
35613	Role definition missing
35623	Attachment is damaged
35624	The description does not contain an attachment
35625	The description is related to a domain that is not in the management of the registrar organization
35626	The description does not belong to a domain
35627	Description not found
35628	The description identifier can only be a number
35629	The description id is missing
35630	Role country code is incorrect
35631	The document was uploaded by another registrar
35643	Attachment is damaged
35644	The complaint does not contain an attachment
35645	The appendix to the complaint can only be downloaded by specifying the specific domain or the designated domain does not exist
35646	The domain of the complaint is not in one of the complaint states
35647	The complaint can not be found
35648	The complaint ID can only be a number
35649	The complaint id is missing
35650	There is no complaint definition

DOMAIN API documentation

35651	*You do not have permission to view the complaint attachment*
36000	*You cannot request an electronic declaration for the domain in the request for confirmation because there is no domain or the domain is not in a delegated state, or the domain owner does not match the owner specified in the declaration or the domain is not in the registrar organization*
36001	*An accepted electronic declaration already exists for the domain in the request for confirmation*
36002	*The domain OBJ parameter is incorrect*
36003	*The OBJ parameter cannot be empty*
36004	*At least one domain OBJ parameter is required*
36005	*The OWNER parameter is incorrect*
36100	*You cannot request an electronic declaration for the domain in the request for confirmation because there is no domain or the domain is not in a delegated state, or the domain owner does not match the owner specified in the declaration or the domain is not in the registrar organization*
36101	*An accepted electronic declaration already exists for the domain in the request for confirmation*
36102	*The domain OBJ parameter is incorrect*
36103	*The OBJ parameter cannot be empty*
36104	*At least one domain OBJ parameter is required*
36105	*The OWNER parameter is incorrect*
36200	*The declaration ID must be a number*
36201	*The declaration does not exist or can no longer be fulfilled*
36202	*The declaration has already been verified*
36203	*The declaration does not exist*
36204	*The type of declaration is incorrect*
36205	*The IP address is invalid*
36206	*The timestamp cannot be in the future*
36207	*The timestamp cannot be earlier than the date of creation*
36208	*The timestamp is invalid, not in yyyy-MM-dd HH:mm:ss format*
36209	*HASH value of the text of the declaration is not correct*
36210	*The captcha is wrong*
36300	*The verification declaration ID must be a number*
36301	*The confirmation declaration does not exist or can no longer be fulfilled*
36302	*The confirmation declaration has already been verified*
36303	*There is no confirmation declaration*
36304	*The type of declaration is incorrect*
36305	*The IP address is invalid*
36306	*The timestamp cannot be in the future*
36307	*The timestamp cannot be earlier than the date of creation*
36308	*The timestamp is invalid, not in yyyy-MM-dd HH:mm:ss format*
36309	*HASH is not valid for the text of the confirmation declaration*
36310	*The captcha is wrong*
39892	*The domain name is not free or locked*
39893	*Owner/registrant error*
39894	*Domain state error*
39895	*Domain data error*
39896	*Tech contact error*

DOMAIN API documentation

39897	*Zone contact error*
39898	*Admin contact error*
39899	*There was an unspecified error*
39986	*XML format error*
39987	*The database could not accept the request because of overload*
39988	*Incorrect login or IP address not assigned to registrar organization*
39989	*The command received is incorrect and cannot be processed*
39990	*PGP signature wrong*
39991	*The PGP key is not assigned to the organization*
39992	*The command cannot be interpreted*
40200	*Command failed*
40220	*This domain name has a wrong syntax*
40225	*Unsupported IDN characters detected*
40240	*Internationalized Domain Names (IDN) not available*
40270	*This TLD is not in claims period*
40280	*This domain is premium, please contact support*
40300	*This domain is not eligible to claim*
40301	*An error occured while querying the claim key of this domain*
40302	*XML Notice for this domain is currently unavailable*
40303	*Bad XML notice id*
40304	*Accepted date can't be in future*
40305	*Accepted date too early*
40306	*Accepted date too late*
40307	*Domain name invalid for this claim key*
40308	*An error occured while loading the config*
40309	*Claim key not specified*
40310	*Notice id not specified*
40311	*Accepted date not specified*
40312	*Accepted ip not specified*
40313	*Bad claim key*
40314	*Too many domains specified*
40400	*Internal Error*
40666	*Drop catching methods are strictly forbidden*
41000	*Command completed successfully*
41001	*Command completed successfully, action pending*
42101	*Internal SQL Error*
42201	*XML Missing*
42202	*XML Syntax Error*
42203	*Internal XML Error*
42301	*Internal OBJECT Error*
42306	*Parameter value policy error*
43101	*There is already a pending action for this domain*
44101	*There are no domains corresponding to your search criteria in your account*
44102	*Insufficient funds to perform this operation*
44103	*Payment method not specified*

DOMAIN API documentation

44105	This action can only by made with accounts having the invoice payments enabled
45000	You may not perform this action
45001	The Licensee for .EU domains has to be in the European Union
45002	This domain cannot be registered in 'No Sunrise'
45003	This domain has already been registered with this sunrise
45004	You have already registered this domain for this type
45010	Only generic TLDs may be locked and/or unlocked
45011	The specified domain will expire soon, you may not update it
45012	The specified domain expires too soon or is already expired, you may not renew it
45013	A domain can be renewed at the earliest 60 days before the expiration date
45014	Update for this tld is not allowed
45015	The registry probibite update for domain bought in sunrise period
45016	This domain cannot not be transfered
45017	The specified domain expires too soon, you may not change its renewal mode
45025	There is already a registration process for this domain
45026	The specified domain has already been scheduled to be transfered
45027	Transferring this domain requires an authorisation code
45029	Currently we do not offer transfers for domains under the specified top level domain
45030	The specified contact is not in your account or does not exist
45111	The specified domain is not in your account or does not exist
45112	The specified domain does not exist or has not been ordered by you
45211	The specified domain is already locked
45212	The specified domain has already been scheduled to be locked
45221	The specified domain is not locked
45222	The specified domain has already been scheduled to be unlocked
45301	The provided data is incomplete and/or incorrect
45401	The specified subuser already exists
45403	The specified subuser does not exists
45404	The specified subuser is not attached to this API account
45405	The specified subuser may not be deleted; it is the main user of your API account
45411	The password and/or the login of the subuser is not correct
45421	There are no subUsers for your account
45501	Invalid contact profile ID specified
45521	Invalid nameserver profile ID specified
45531	Invalid zone profile ID specified
45541	This zone profile could not be created
45542	This zone profile could not be updated
45543	This zone profile could not be deleted
45544	This zone profile name is not available
45545	This record could not be found for the requested zone profile
45546	This record could not be updated for the requested zone profile
45551	This contact profile may not be deleted
45561	This contact profile could not be created
45571	This user account could not be created
45601	No poll message for the specified id found

DOMAIN API documentation

45651	There are currently no poll messages in your account
46101	Invalid domain name specified
46102	Invalid domain roid specified
46103	Invalid ordered domain name specified
46104	This domain seems to be free. Try to register instead of transferring it
46105	This domain is not available
46106	This domain has already been queued for a similar action
46107	Invalid top level domain specified
46108	This domain already exists in our database
46109	This domain is already expired and will be deleted shortly
46111	Invalid subscription period specified
46121	Subscription period too short
46122	Subscription period too long
46131	Invalid format for contactID specified
46132	Invalid format for recordID specified
46133	Invalid renewal mode specified
46141	Invalid priority for this nameserver specified
46142	Invalid domain for this nameserver specified
46143	Invalid ip for this nameserver specified
46144	Not enough nameservers specified
46145	Too many nameservers specified
46146	For domains under this TLD the nameservers may not be updated
46147	This nameserver profile could not be created
46161	Invalid IPv4 format specified
46162	This IP address does not exist in your account
46163	The specified IP address already exists
46164	Maximum of IP addresses reached
46171	You do not need to perform a trade to modify the organisation contact of this domain
46172	You have to perform a trade to modify any field of the organisation contact of this domain
46173	You have to perform a trade to modify the first/last name or the company of the organisation contact of this domain
46174	To perform a trade you have to specify the organisation contact
46175	Currently we do not offer the possbility to perform trades for domains under the specified top level domain
46502	There is no folder with the specified folderid
46506	An error occured while creating the the folder
46508	An error occured while removing the the folder
46510	An error occured while deleting the the folder
46602	The specified mailbox has not been found
46603	Maximum number of mailboxes reached
46604	The specified mailbox already exists
46605	You have to specify a password for the mailbox
46606	The specified alias has not been found
46607	This mailbox has not been flagged for deletion, it cannot be deleted
46608	The specified alias already exists

DOMAIN API documentation

46609	*The local-part of the address is not valid*
46610	*Maximum number of aliases reached for this mailbox*
46611	*Maximum number of emails reached for this mailing list*
46612	*The specified email has already been black/whitelisted*
46613	*The specified email has already been whitelisted*
46614	*The specified email has already been blacklisted*
46615	*The specified email has not been black or whitelisted*
46616	*The specified email has not been whitelisted*
46617	*The specified email has not been blacklisted*
46618	*The specified password doesnt match with your current password*
46619	*The email is already defined as an alias or a mailing list.*
46620	*Invalid policy specified*
46621	*The specified email already exist in the mailing list*
46622	*Invalid Eurodns UserID*
46624	*Flag not set for user deletion*
46625	*Your mailing list name can not be the same as one of your email name*
46626	*Unable to send e-Mail from support to the new e-Mail account.*
46630	*The specified quota is too large*
46634	*User name too long. The user name exceeds 48 characters.*
46640	*The alias can only be created for the same domain*
46651	*The alias is not a valid email address*
46652	*The specified email address is not valid*
46660	*Invalid username. Only letters [a-z] [0-9] [.-_] are allowed.*
46665	*Invalid password. Only letters [a-z] [0-9] [.-_#&] are allowed.*
46670	*Invalid e-mail address for black/white list.*
46675	*Username not specified.*
46676	*The owner of one of the mail in the mailing list does not exist*
46677	*One of the email in the mailing list does not exist*
46678	*The mailing list does not exist*
46685	*Question is not defined.*
46686	*The question is empty.*
46687	*The answer is empty.*
46688	*The maximum number of answer tries has been reached*
46689	*The data is invalid.*
47001	*Unable to load user information*
47002	*Unable to load user discounts*
47101	*Command sucessful, there was however an error debiting your account, you will be invoiced*
47201	*Unable to create a new order*
47301	*Failed to create a domain from your XML code*
47302	*Failed to load a domain using your XML code*
47311	*Failed to check domain availability*
47401	*Could not load specified object*
48001	*Unknown Internal Error*
48002	*Unknown command received*
49997	*Username and/or password missing*

DOMAIN API documentation

49998	*The IP address is not in the list of allowed IP addresses*
49999	*Wrong username and/or wrong password specified*
50420	*Command failed due to server error. Server closing connection*
50421	*Command failed due to server error. Client should try again*
50423	*Command failed due to server error. Client should try again (Could not get session)*
50425	*Service temporarily locked; usage exceeded*
50500	*Invalid command name*
50501	*Invalid command option*
50502	*Invalid entity value*
50503	*Invalid attribute name*
50504	*Missing required attribute*
50505	*Invalid attribute value syntax*
50506	*Invalid option value*
50507	*Invalid command format*
50508	*Missing required entity*
50509	*Missing command option*
50510	*Command not supported for this class*
50520	*Server closing connection. Client should try opening new connection*
50521	*Too many sessions open. Server closing connection*
50530	*Authentication failed*
50531	*Authorization failed*
50532	*Domain names linked with name server*
50533	*Domain name has active name servers*
50534	*Object has not been flagged for transfer*
50535	*Restricted IP address*
50536	*Object already flagged for transfer*
50540	*Attribute value is not unique*
50541	*Invalid attribute value*
50542	*Invalid old value for an attribute*
50543	*Final or implicit attribute cannot be updated*
50544	*Entity on hold*
50545	*Object not found*
50546	*Credit limit exceeded*
50547	*Invalid command sequence*
50548	*Object is not up for renewal*
50549	*Command failed*
50550	*Parent object not registered*
50551	*Parent object status does not allow for operation*
50552	*Object status does not allow for operation*
50553	*Operation not allowed. Object pending transfer*
50554	*Object already registered*
50555	*Object already renewed*
50556	*Maximum registration period exceeded*

DOMAIN API documentation