

ЗАТВЕРЖДЕНО

«__» _____ 2013 р.

Інструкція по роботі з ЕРР

РОЗРОБЛЕНО:

«__» _____ 2013 р.

Зміст

1. Вступ	9
1.1. Процес обміну пакетами	10
1.2. Обмеження при роботі з EPP	11
1.3. Параметри доменних імен	11
1.4. Параметри з'єднання с реєстром	13
2. Загальний перелік команд EPP	13
2.1. Команди EPP. Загальні відомості	14
2.1.1. Команди управління сесією	14
2.1.2. Команди для отримання інформації про об'єкти	14
2.1.3. Команди зміни об'єктів	15
2.2. Формат команд EPP	15
2.2.1. Блок команди	15
2.2.2. Елементи команди	16
2.2.3. Загальна структура відповідей EPP-сервера	16
2.2.4. Елементи відповіді EPP-сервера	16
2.2.5. Коди відповідей EPP сервера (Response Codes)	17
3. Команди EPP	18
3.1. LOGIN	18
3.1.1. Елементи команди <login>	18
3.1.2. Приклад команди <login>	19
3.1.3. Приклад команди <login> із зміною пароля для наступної сесії	19
3.1.4. Приклад відповіді сервера на команду <login> при успішному відкритті сесії	20
3.1.5. Приклад відповіді сервера на команду <login> при невірно вказаному паролі	20
3.1.6. Приклад відповіді сервера на команду <login> при перевищенні числа невдалих спроб з'єднання	21

3.1.7. Коди відповідей при виконанні команди<login>.....	21
3.2. HELLO, GREETING	22
3.2.1. Елементи відповіді <greeting>	22
3.2.2. Приклад запиту <hello>	22
3.2.3. Приклад привітання сервера <greeting>	22
3.3. LOGOUT.....	24
3.3.1. Приклад команди <logout>	24
3.3.2. Приклад відповіді сервера на команду <logout>	24
3.3.3. Коди відповідей при виконанні команди <logout>.....	24
3.4. POLL	25
3.4.1. Приклад команди <poll op="req">	25
3.4.2. Приклад відповіді на команду <poll> з інформацією, специфічною для об'єкта26	
3.4.3. Приклад підтвердження за допомогою команди <poll op="ack">	27
3.4.4. Приклад відповіді на команду <poll> з підтвердженням читання повідомлення.....	27
3.4.5. Приклад відповіді на команду <poll op="req">, який показує, що черга повідомлень порожня	28
3.5. DOMAIN:CHECK.....	28
3.5.1. Елементи команди <domain:check>	28
3.5.2. Приклад команди <domain:check>	28
3.5.3. Приклад відповіді сервера на команду <domain:check>	29
3.5.4. Піделементи команди <domain:cd>.....	29
3.5.5. Приклад відповіді команди <domain:check>	29
3.5.6. Коди відповіді при виконанні команди<domain:check>	30
3.6. DOMAIN:INFO.....	31
3.6.1. Елементи команди<domain:info>	31
3.6.2. Приклад команди <domain:info>	31
3.6.3. Елементи відповіді на команду <domain:info>	31

3.6.4. Приклад відповіді сервера на команду <domain:info> для авторизованого користувача.....	32
3.6.5. Приклад відповіді сервера на неавторизований запит <domain:info>. 34	
3.6.6. Коди відповіді при виконанні команди <domain:info>	34
3.7. DOMAIN:CREATE	35
3.7.1. Елементи команди <domain:create>	35
3.7.2. Приклад команди <domain:create>	36
3.7.3. Приклади команди <domain:create> якщо вказується номер знака для товарів та послуг (для періоду пріоритетної реєстрації):.....	36
3.7.4. Коди відповіді при виконанні команди <domain:create>	37
3.8. DOMAIN:UPDATE	39
3.8.1. Елементи команди <domain:update>	39
3.8.2. Приклад команди <domain:update> для зміни контактів і name-серверів	40
3.8.3. Приклад команди <domain:update> для зміни clientTransferProhibited і обнуління authInfo	41
3.8.4. Приклад команди <domain:update> для відміни статусу clientTransferProhibited та завдання нового authInfo	41
3.8.5. Завдання елементу <domain:ns>	42
3.8.6. Приклад відповіді сервера на команду <domain:update>.....	42
3.8.7. Коди відповідей при виконанні команди <domain:update>	43
3.9. DOMAIN:RENEW	44
3.9.1. Елементи команди <domain:renew>	44
3.9.2. Приклад команди <domain:renew>	44
3.9.3. Елементи відповіді на команду <domain:renew>	45
3.9.4. Приклад відповіді сервера на команду <domain:renew>.....	45
3.9.5. Коди відповіді при виконанні команди <domain:renew>.....	46
3.10. DOMAIN:DELETE	47

3.10.1. Елементи команди<domain:delete>	47
3.10.2. Перелік перевірок під час обробки команди <domain:delete>	47
3.10.3. Приклад команди <domain:delete>	47
3.10.4. Приклад відповіді сервера на команду <domain:delete>	48
3.10.5. Коди відповіді при виконанні команди <domain:delete>	48
3.11. DOMAIN RESTORE	49
3.11.1. Елементи команди<rgp:update>	49
3.11.2. Приклад команди <rgp:update>	49
3.11.3. Приклад відповіді серверу на команду <rgp:update>	50
3.11.4. Коди відповідей при виконанні команди<rgp:update>	50
3.12. DOMAIN:TRANSFER	51
3.12.1. Атрибути команди <domain:transfer>	51
3.13. DOMAIN:TRANSFER REQUEST	52
3.13.1. Елементи команди<domain:transfer op="request">	52
3.13.2. Елементи відповіді на команду <domain:transfer op="request">	52
3.13.3. Приклад команди <domain:transfer op="request">	53
3.13.4. Приклад відповіді сервера на команду <domain:transfer op="request"> 54	
3.13.5. Коди відповіді при виконанні команди <domain:transfer op="request"> 54	
3.13.6. Дії після успішного виконання запиту на трансфер домену	56
3.14. DOMAIN:TRANSFER CANCEL	57
3.14.1. Елемент команди <domain:transfer op="cancel">	57
3.14.2. Приклад команди <domain:transfer op="cancel">	57
3.14.3. Елементи відповіді на команду <domain:transfer op="cancel">	57
3.14.4. Приклад відповіді сервера на команду <domain:transfer op="cancel"> 58	
3.14.5. Коди відповідей при виконанні команди<domain:transfer op="cancel"> 58	

3.15. DOMAIN:TRANSFER APPROVE	59
3.15.1. Елементи команди<domain:transfer op="approve">	60
3.15.2. Приклад команди <domain:transfer op="approve">	60
3.15.3. Елементи відповіді на команду<domain:transfer op="approve">	60
3.15.4. Приклад відповіді сервера на команду <domain:transfer op="approve"> 61	
3.15.5. Коди відповідей при виконанні команди<domain:transfer op="approve">	61
3.15.6. Дії після успішного виконання команди <domain:transfer op="approve">	62
3.16. DOMAIN:TRANSFER REJECT	63
3.16.1. Елементи команди<domain:transfer op="reject">	63
3.16.2. Приклад команди <domain:transfer op="reject">	63
3.16.3. Елементи відповіді на команду <domain:transfer op="reject">	63
3.16.4. Приклад відповіді на команду <domain:transfer op="reject">	64
3.16.5. Коди відповідей при виконанні команди<domain:transfer op="reject"> 64	
3.16.6. Дії після успішного виконання команди <domain:transfer op="reject"> 65	
3.17. DOMAIN:TRANSFER QUERY	66
3.17.1. Елементи команди <domain:transfer op="query">	66
3.17.2. Приклад команди <domain:transfer op="query">	66
3.17.3. Елементи відповіді на команду<domain:transfer op="query">	66
3.17.4. Приклад відповіді на команду <domain:transfer op="query">	67
3.17.5. Коди відповідей при виконанні команди<domain:transfer op="query"> 68	
3.18. HOST:CHECK	69
3.18.1. Елементи команди<host:check>	69
3.18.2. Приклад команди <host:check>	69

3.18.3. Елементи відповіді на команду<host:check>.....	70
3.18.4. Приклад відповіді серверу на команду <host:check>	70
3.18.5. Коди відповідей при виконанні команди<host:check>.....	71
3.19. HOST:INFO	72
3.19.1. Елементи команди<host:info>.....	72
3.19.2. Приклад команди <host:info>	72
3.19.3. Елементи відповіді на команду <host:info>	72
3.19.4. Приклад відповіді на команду <host:info>	73
3.19.5. Коди відповідей при виконанні команди<host:info>.....	74
3.20. HOST:CREATE.....	74
3.20.1. Елементи команди<host:create>.....	74
3.20.2. Приклад команди <host:create>	75
3.20.3. Елементи відповіді на команду <host:create>	75
3.20.4. Приклад відповіді на команду <host:create>	75
3.20.5. Коди відповідей при виконанні команди <host:create>	76
3.21. HOST:UPDATE	77
3.21.1. Елементи команди<host:update>	77
3.21.2. Приклад відповіді на команду <host:update>	78
3.21.3. Коди відповідей при виконанні команди<host:update>	78
3.22. HOST:DELETE	79
3.22.1. Елементи команди<host:delete>.....	80
3.22.2. Приклад команди <host:delete>	80
3.22.3. Приклад відповіді на команду <host:delete>	80
3.22.4. Коди відповідей при виконанні команди <host:delete>	81
3.23. CONTACT:CHECK.....	81
3.23.1. Елементи команди <contact:check>.....	82
3.23.2. Приклад команди <contact:check>.....	82
3.23.3. Елементи відповіді на команду <contact:check>.....	82

3.23.4. Приклад відповіді на команду <contact:check>	82
3.23.5. Коди відповідей при виконанні команди <contact:check>	83
3.24. CONTACT:INFO	84
3.24.1. Елементи команди<contact:info>	84
3.24.2. Приклад команди <contact:info>	84
3.24.3. Елементи відповіді на команду <contact:info>	84
3.24.4. Приклад відповіді на команду <contact:info>	86
3.24.5. Коди відповідей при виконанні команди <contact:info>	87
3.25. CONTACT:CREATE	88
3.25.1. Елементи команди<contact:create>	88
3.25.2. Приклад команди <contact:create>	89
3.25.3. Елементи відповіді на команду <contact:create>	90
3.25.4. Приклад відповіді на команду <contact:create>	91
3.25.5. Коди відповідей при виконанні команди <contact:create>	91
3.26. CONTACT:UPDATE	92
3.26.1. Елементи команди<contact:update>	92
3.26.2. Приклад команди <contact:update>	93
3.26.3. Приклад відповіді на команду <contact:update>	94
3.26.4. Коди відповідей при виконанні команди <contact:update>	95
3.27. CONTACT:DELETE	95
3.27.1. Елементи команди<contact:delete>	95
3.27.2. Приклад команди <contact:delete>	95
3.27.3. Приклад відповіді на команду <contact:delete>	96
3.27.4. Коди відповідей при виконанні команди <contact:delete>	96

1. Вступ

EPP (ExtensibleProvisioningProtocol) - це протокол взаємодії Реєстраторів доменів і реєстру доменної зони (<http://tools.ietf.org/html/rfc5730>).

Даний документ описує взаємодію програмного комплексу клієнта та EPP-сервера за допомогою ExtensibleMarkupLanguage (XML) 1.0, через EPP-протокол.

У даному документі не міститься повний і вичерпний опис системи команд EPP, що викладено в документах RFC 5730 - 5734, RFC 3915. У даному документі надана реалізація EPP ТОВ «Технічний центр Інтернет» (далі – ТЦІ) в системі реєстрації керування доменними іменами в домені .УКР. Реалізація протоколу EPP ТЦІ відрізняється від реалізації протоколу, що описаний в RFC. Відмінності обумовлені Правилами реєстрації і користування доменними іменами в домені .УКР (надалі – Правила).

У протоколі EPP ТЦІ реалізовані відповіді діагностики виконання запитів трьома мовами: українською, російською і англійською.

Після отримання доступу до EPP-серверу можна приступити до початку інтеграції. Надається 2 (два) доступи до серверів - тестовий і робочий (після завершення процедури акредитації Реєстратора). На тестовому сервері можна виконати всі доступні EPP-команди, що описані в даному документі, пройти набір EPP-тестів (є частиною процедури акредитації Реєстратора), і переконатися в тому, що ваша система готова до роботи з платними операціями з робочим EPP-сервером.

Будьте уважні, XML чутливий до регістру, всі приклади команд описані в даному документі повинні бути дотримані для коректної роботи систем.

EPP-сервер дозволяє здійснювати реєстрацію, подовження, трансфер, а також інші операції з доменними іменами.

Протокол EPP описує обмін даними між клієнтським і серверним програмним забезпеченням у вигляді пакетів даних. Кожен пакет даних складається з 4 байт службової інформації та тіла пакету. 4 байта службової інформації містять довжину всього пакету (довжина тіла пакета + 4). Те, що пакет даних містить довжину пакета, дозволяє клієнту і серверу визначити довжину даних, на основі перших чотирьох байт.

Наприклад, якщо EPP-клієнт відправляє на сервер команду hello:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>  
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
```

<hello/>

</err>

Необхідно обчислити довжину тіла, тобто XML-даних, наведених вище, в даному випадку це 119 байт. Загальна довжина пакету становитиме 4 байти (заголовок довжини) + 119 байт (тіло) = 123 байта.

Ця довжина записується в перших чотирьох байтах в наступному вигляді:

```
byte[] bytes = new byte[totalLength];  
bytes[0] = (byte)(0xff & (totalLength >> 24));  
bytes[1] = (byte)(0xff & (totalLength >> 16));  
bytes[2] = (byte)(0xff & (totalLength >> 8));  
bytes[3] = (byte)(0xff & totalLength);
```

Зверніть увагу! В даному випадку робота ведеться з байтами, а не із строковими даними (деякі бібліотеки, зокрема `err-rtk`, обчислюють кількість символів, а не байт; у випадку `utf-8` один символ може складатися з декількох байтів). Після формування 4-байтного заголовка довжини до нього додається сам запит (XML) і весь пакет відправляється до серверу. Якщо пакет даних не вказано, сервер не зможе визначити довжину даних, що надходять і не зможе правильно обробити запит.

Примітка. У більшості існуючих бібліотек для роботи з EPP, описаний вище механізм вже реалізований, і програмісту немає необхідності формувати довжину пакета - зазвичай, це здійснюється автоматично самою бібліотекою. У разі ж реалізації протоколу самостійно (або в разі якщо бібліотека не підтримує обчислення довжини), необхідно піклуватись про правильне формування пакетів.

1.1. Процес обміну пакетами

Процес обміну пакетами виглядає наступним чином:

1. Клієнт відкриває SSL з'єднання з сервером;
2. Сервер відправляє, а клієнт вичитує пакет вітання сервера;
3. Після цього кілька разів повторюється операція згідно з цільовим призначенням SSL-сесії з обміну пакетами - клієнт посилає пакет з командою на сервер, сервер його приймає, обробляє і надсилає пакет з відповіддю;
4. Клієнт закриває з'єднання з сервером.

Зверніть увагу! Після відкриття з'єднання з сервером, спочатку необхідно прийняти пакет вітання сервера, і тільки після цього можна починати

відправляти команди. По завершенню обміном командами з сервером, потрібно виконати команду <logout> і тільки потім розірвати з'єднання.

1.2. Обмеження при роботі з EPP

Параметр	Значення
Мінімальна кількість контактів одного типу для одного домену	1
Максимальна кількість контактів одного типу для одного домену	1
Мінімальна кількість name-серверів для одного домену	0
Максимальна кількість name-серверів для одного домену	16
Мінімальна кількість IP-адрес (v4 або v6) для одного name-сервера	1
Максимальна кількість IP-адрес (v4 та/або v6) для одного name-сервера	16
Максимальний час збереження повідомлень в черзі poll (міс.)	1
Максимальна кількість об'єктів в операції check	10
Приналежність IP-адрес до публічного простору (RFC 1918, 3330, 3927, 4193, 3879)	так
Мінімальна довжина authInfo домену/контакту	8 символів
Максимальна довжина authInfo домену/контакту	до 80 символів
Обов'язкові символи authInfo	a-zA-Z0-9 , також повинен бути щонайменше один зі спеціальних символів: +,- /,:;<=>?@[]^'_{ }~!"#\$%&()*

1.3. Параметри доменних імен

Параметр	Значення
Мінімальна кількість символів у домені (без урахування 4-х символів .УКР)	1
Максимальна кількість символів у домені в кодуванні Unicode (з урахуванням 4-х символів .УКР)	63
Доменне ім'я починається і закінчується буквою кириличного алфавіту (у т.ч. апостроф) або цифрою. Символами у середині доменного імені є букви кириличного алфавіту, цифри або дефіси.	Згідно IDN таблиці для ccTLD .УКР (https://namestore.u-registry.com/Ukrainian_IDN_Table_v3.txt). Забороняється використання будь-яких символів, якщо вони відсутні у IDN таблиці.
Мінімальна кількість років, на яку може бути зареєстровано домен	1
Максимальна кількість років, на яку може бути зареєстровано або подовжено термін реєстрації	У будь-який момент часу, термін дії періоду реєстрації домену не може перевищувати 10 років
Термін дії секретного коду домену (authInfo)	Згідно терміну реєстрації домену
Термін подовження періоду реєстрації домену при трансфері від іншого Реєстратора.	1 рік, при цьому сумарний термін дії реєстрації домену не може перевищувати 10 років. Якщо при трансфері майбутній термін реєстрації домену може перевищити максимально допустимий - трансфер відбувається без подовження терміну реєстрації домену.
Тривалість періоду автоматичного подовження домену (AutoRenewGracePeriod)	30 діб
Тривалість періоду очікування відновлення (redemptionPeriod)	25 діб

Тривалість періоду очікування видалення домену (Pending Delete)	5 діб
---	-------

1.4. Параметри з'єднання с реєстром

Параметр	Значення
Адреса робочого EPP сервера	надається після проходження акредитації
Адреса тестового EPP сервера	надається під час проходження акредитації
Термін дії SSL сертифікату Реєстратора	12 місяців
Кількість одночасних SSL-сесій від Реєстратора	15
Максимальна кількість IP адрес з яких допускаються підключення від Реєстратора	5
Максимально допустима кількість невдалих спроб логіну	до 3 в рамках однієї сесії
Період не активності, коли з'єднання з Реєстратором буде розірвано (idletimeout)	10 хвилин
Максимальний період непереривного з'єднання Реєстратора з реєстром (globaltimeout)	24 години
Ліміт запитів від одного Реєстратора для всіх сесій	10000 за 10 хвилин
Ліміт tcp-підключень з одного IP	не більше 50 за 1 секунду, або 1200 за 1 хвилину, або 3000 за 5 хвилин. У разі перевищення вказаних лімітів - блокування підключення на 15 хвилини.

2. Загальний перелік команд EPP

Команди обробляються на сервері EPP в порядку, в якому вони надійшли від клієнта. Команди обробляється негайно, про що клієнтові повідомляються у

відповіді сервера. У разі, якщо обробка команди вимагає підтвердження з боку клієнта, іншого клієнта або оператора Реєстру, клієнтові буде повідомлено, що команда прийнята, але дію відкладено (actionpending). (див. код відповіді 1001). У разі відкладеної дії передбачена можливість переглядати інформацію про хід виконання команди.

2.1. Команди EPP. Загальні відомості

Команди EPP поділяються на 3 категорії:

1. команди управління сесією - для підключення / відключення до EPP-серверу;
2. команди для отримання інформації про об'єкти - операції читання;
3. команди зміни об'єктів.

2.1.1. Команди управління сесією

Для управління сесією використовуються дві команди: <login> і <logout>.

Команда EPP <login> використовується для відкриття сесії з EPP сервером.

Команда EPP <logout> завершує сеанс доступу до сервера EPP. Сесія може бути також завершена після закінчення деякого часу бездіяльності або після закінчення максимального часу сесії. Дані параметри регламентуються цією Інструкцією.

2.1.2. Команди для отримання інформації про об'єкти

Для отримання інформації про те, чи зареєстрований об'єкт (домен, хост або контакт) в Реєстрі, використовується команда перевірки <check>

<domain:check>

<contact:check>

<host:check>

Для отримання інформації про вже зареєстровані об'єкти (домени, хости або контакти) використовується команда <info>

<domain:info>

<host:info>

<contact:info>

Для надсилання службових повідомлень, які сервер EPP направляє клієнтові, використовується команда <poll>.

Команда має атрибути:

1. "req" - читання останнього непрочитаного повідомлення з черги повідомлень,
2. "ack" - підтвердження прочитання повідомлення.

Для отримання інформації про поточний статус операції трансферу об'єкта використовується інформаційна команда `<transfer op="query">`. Команда доступна для клієнта, що ініціював трансфер, і поточного Реєстратора об'єкта.

2.1.3. Команди зміни об'єктів

Для створення нових об'єктів (доменів, хостів або контактів) використовується команда `<create>`

`<domain:create>`

`<host:create>`

`<contact:create>`

Для зміни даних в об'єктах (доменах, хостах або контактах) використовується команда `<update>`

`<domain:update>`

`<host:update>`

`<contact:update>`

Для продовження терміну реєстрації домену використовується команда `<renew>`

`<domain:renew>`

Для трансферу домену від іншого Реєстратора використовується команда `<transfer>`

`<domain:transfer>`

Для видалення об'єкту (домену, хосту або контакту) використовується команда `<delete>`

`<domain:delete>`

`<host:delete>`

`<contact:delete>`

2.2. Формат команд EPP

Всі EPP-команди розміщуються між мітками `<?xml>`, `<epp>` і `</epp>`. Сама команда описується блоком між мітками `<command>` та `</command>`.

2.2.1. Блок команди

Блок команди виглядає наступним чином (на прикладі команди `domain:info`):

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
```

```
<command>
```

```
<info>
```

```

<domain:info xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--e1aybc.xn--j1amh</domain:name>
</domain:info>
</info>
<clTRID>USER-1276595352</clTRID>
</command>
</epp>

```

2.2.2. Елементи команди

Команда містить наступні елементи:

Необов'язковий елемент <clTRID>, що містить унікальний ідентифікатор запиту, який може використовувати Реєстратор при роботі з сервером. Цей ідентифікатор буде присутній у відповіді сервера на дану команду. Використовується для визначення відповідності запитів і відповідей сервера EPP.

На кожну отриману від клієнта команду сервер направляє відповідь, в якій зазначено, чи була успішно виконана команда або при її аналізі і виконанні виникли помилки.

У подальшому, при описі елементів EPP-команд у таблицях (другий стовпець) застосовується позначення щодо властивості обов'язковості або опціональності (необов'язковості) використання елементів:

М - mandatory (обов'язковий елемент);

О - optional (необов'язковий елемент (опціональний));

2.2.3. Загальна структура відповідей EPP-сервера

Загальна структура відповідей сервера EPP:

Всі EPP-елементи розміщуються між мітками <?xml>, <epp> і </epp>. Сама відповідь розміщується у блоці між мітками <response> та </response>.

Блок відповіді виглядає наступним чином:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
<response>
- Відповідь сервера -
</response>
</epp>

```

2.2.4. Елементи відповіді EPP-сервера

Відповідь EPP сервера містить елементи:

Елемент		Опис
<result>	M	Елемент, який містить інформацію про результат виконання команди. Містить параметр “code”, в якому зазначено код відповіді сервера.
<msg>	M	Елемент, який містить код мови повідомлень в команді (у атрибуті “lang”).
<value>	O	Елементи <value> містять інформацію, яка викликала помилки сервера (значення помилкового параметра).
<extValue>	O	За наявності помилки, елемент містить опис причини помилки (елемент <reason>)
<msgQ>	O	Елемент може містити номер службового повідомлення.
<resData>	O	Елемент, що містить інформацію, специфічну для кожної команди.
<trID>	M	Унікальний ідентифікатор обробки команди. Він містить елементи <clTRID> (реєстраторський ідентифікатор транзакцій) і <svTRID> (серверний ідентифікатор транзакцій).

2.2.5. Коды відповідей EPP сервера (Response Codes)

Для опису результату виконання кожної EPP команди використовується код, що складається з 4 цифр. Кожна цифра має певне значення.

Перша цифра визначає результат виконання команди:

- **1yzz** Позитивна відповідь. - Команда виконана успішно.
- **2yzz** Негативна відповідь. - Команда не прийнята, дія не виконана.

Друга цифра відповіді визначає категорію:

- **x0zz** Синтаксис протоколу
- **x1zz** Логіка
- **x2zz** Безпека
- **x3zz** Управління даними
- **x4zz** Внутрішні проблеми сервера

- **x5zz** Управління з'єднанням з сервером

Третя і четверта цифри визначають порядковий номер відповіді у даній категорії.

Детальніше опис всіх кодів (Response Codes):

<http://tools.ietf.org/html/rfc5730#page-39>

Кожна відповідь сервера, окрім коду, містить текстовий опис. Мова, на якій видається опис, визначається атрибутом "lang". Можливі наступні варіанти значення атрибуту "lang": "ua", "ru", "en".

3. Команди EPP

3.1. LOGIN

Команда EPP <login> використовується для відкриття сесії з EPP сервером у відповідь на видане сервером вітання <greeting> після встановлення з'єднання.

Команда EPP <login> повинна бути відправлена перед будь-якою іншою командою. Число невдалих спроб, після яких сервер закриває з'єднання, обмежена і визначається параметрами сервера згідно обмежень, введених у цій Інструкції.

3.1.1. Елементи команди <login>

Елемент		Опис
<clID>	M	Містить ідентифікатор клієнта (login). Це рядок довжиною від 3 до 16 символів, що складається з літер латинського алфавіту, цифр та знака "-".
<pw>	M	Містить пароль клієнта - рядок довжиною від 6 до 16 символів. Інформація в цьому полі є регістрозалежною.
<newPW>	O	Необов'язковий, додатковий елемент, який містить НОВИЙ пароль клієнта. Це поле є регістрозалежним. Після успішного відкриття сесії пароль клієнта буде змінений і для відкриття наступної сесії буде потрібно вказати новий пароль.
<options>	M	Містить такі піделементи: <version> - версія протоколу. <lang> - мову для повідомлень сервера. Значення параметрів <version> і <lang> повинні співпадати зі значеннями, запропонованими сервером у вітанні

		<greeting>
<svcs>	M	Містить елементи <objURI>, що вказують на URI схем просторів імен (namespace), що описують об'єкти, з якими буде здійснюватися робота в ході сесії. Значення параметрів <objURI> повинні співпадати зі значеннями, запропонованими сервером у вітанні <greeting>.

3.1.2. Приклад команди <login>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<login>
<clID>user</clID>
<pw>SecretPassword</pw>
<options>
<version>1.0</version>
<lang>en</lang>
</options>
<svcs>
<objURI>urn:ietf:params:xml:ns:contact-1.0</objURI>
<objURI>urn:ietf:params:xml:ns:domain-1.0</objURI>
<objURI>urn:ietf:params:xml:ns:host-1.0</objURI></svcs>
</login>
<clTRID>USER-1275641748</clTRID>
</command>
</epp>
```

Ця команда <login> встановлює сесію з унікальним ідентифікатором запиту USER-1275641748 з інтерфейсом спілкування англійською мовою для клієнта user, якій має пароль SecretPassword, з метою доступу до об'єктів та отримання інформації про об'єкти типів Domain, Contact, Host, що розташовані за відповідними ідентифікаторами URI.

3.1.3. Приклад команди <login> із зміною пароля для наступної сесії

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
```

```

<login>
<clID>user</clID>
<pw>SecretPassword</pw>
<newPW>NewSecretPassword</newPW>
<options>
<version>1.0</version>
<lang>en</lang>
</options>
<svcs>
<objURI>urn:ietf:params:xml:ns:contact-1.0</objURI>
<objURI>urn:ietf:params:xml:ns:domain-1.0</objURI>
<objURI>urn:ietf:params:xml:ns:host-1.0</objURI></svcs>
</login>
<clTRID>USER-1275641748</clTRID>
</command>
</epp>

```

3.1.4. Приклад відповіді сервера на команду <login> при успішному відкритті сесії

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-1275641748</clTRID>
<svTRID>UKR-20100604115549</svTRID>
</trID>
</response>
</epp>

```

3.1.5. Приклад відповіді сервера на команду <login> при невірному вказаному паролі

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
<response>

```

```

<result code="2200">
<msg lang="en">Authentication failure.</msg>
</result>
<trID>
<clTRID>USER-1275641748</clTRID>
<svTRID>UKR-20100604115549</svTRID>
</trID>
</response>
</epp>

```

3.1.6. Приклад відповіді сервера на команду <login> при перевищенні числа невдалих спроб з'єднання

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
<response>
<result code="2501">
<msg lang="en">Authentication error; server closing connection.</msg>
</result>
<trID>
<clTRID>USER-1275641748</clTRID>
<svTRID>UKR-201003434343349</svTRID>
</trID>
</response>
</epp>

```

3.1.7. Коди відповідей при виконанні команди<login>

Код відповіді	Опис
1000	"Command completed successfully" "Команда виконана успішно" "Команда выполнена успешно"
2200	"Authentication error" "Помилка аутентифікації" "Ошибка аутентификации"
2501	"Authentication error; server closing connection" "Помилка аутентифікації; сервер закриває з'єднання" "Ошибка аутентификации; сервер закрывает соединение"

2502	"Session limit exceeded; server closing connection" "Перевищено кількість одночасних сесій; сервер закриває з'єднання" "Превышено количество одновременных сессий; сервер закрывает соединение"
------	---

3.2. HELLO, GREETING

Після встановлення з'єднання з EPP-сервером або після отримання запиту <hello> сервер надає відповідь <greeting> ("вітання") у вигляді XML-документа.

Команда <hello> та відповідь <greeting> мають інформаційно-довідковий характер.

3.2.1. Елементи відповіді <greeting>

Документ містить наступні елементи:

Елемент		Опис
<svID>	M	Ім'я сервера.
<svDate>	M	Поточні локальні дата і час на сервері.
<svcMenu>	M	Елемент описує сервіси, які підтримує сервіс: <version> - версії протоколу, які підтримує сервіс; <lang> - список мов, які підтримує сервіс.
<objURI>	M	елемент містить посилання на опис структури, які використовуються сервером об'єктів (namespace URIs). Надалі клієнт повинен вказувати в командах ці позначення при описі об'єктів в полях xmlns.
<dcp>	M	Datacollectionpolicy - елемент містить опис політики конфіденційності сервера при передачі даних. Детальний опис елементу є в RFC 5730.

EPP клієнт може запитати у сервера відповідь <greeting> в будь-який час між командами <login> і <logout>, відправивши запит <hello>.

3.2.2. Приклад запиту <hello>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
<hello/>
</epp>
```

3.2.3. Приклад привітання сервера <greeting>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
  <greeting>
    <svID>EPP Server</svID>
    <svDate>2013-07-06T14:25:01.532Z</svDate>
    <svcMenu>
      <version>1.0</version>
      <lang>en</lang>
      <lang>ru</lang>
      <lang>ua</lang>
    <objURI>urn:ietf:params:xml:ns:domain-1.0</objURI>
    <objURI>urn:ietf:params:xml:ns:host-1.0</objURI>
    <objURI>urn:ietf:params:xml:ns:contact-1.0</objURI>
    <svcExtension>
      <extURI>urn:ukr:params:xml:ns:rrpproxy-1.0</extURI>
      <extURI>urn:ukr:params:xml:ns:oxrs-1.0</extURI>
      <extURI>urn:ietf:params:xml:ns:rgp-1.0</extURI>
      <extURI>urn:ukr:params:xml:ns:idn-1.0</extURI>
      <extURI>urn:ukr:params:xml:ns:bulk-1.0</extURI>
    </svcExtension>
  </svcMenu>
  <dcp>
    <access>
      <personal/>
    </access>
    <statement>
      <purpose>
        <admin/>
        <other/>
      </purpose>
      <recipient>
        <unrelated/>
        <ours>
          <recDesc> U-Registry</recDesc>
        </ours>
        <public/>
      </recipient>
    </statement>
  </dcp>
</epp>
```

```
</recipient>
<retention>
<legal/>
</retention>
</statement>
<expiry>
<relative>P10Y0M0DT0H0M0.0S</relative>
</expiry>
</dcp>
</greeting>
</epp>
```

3.3. LOGOUT

Команда EPP <logout> використовується для закриття сесії з EPP сервером.

3.3.1. Приклад команди <logout>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<logout/>
<clTRID>USER-1275641748</clTRID>
</command>
</epp>
```

3.3.2. Приклад відповіді сервера на команду <logout>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1500">
<msg>Command completed successfully; ending session</msg>
</result>
<trID>
<clTRID>USER-1275641748</clTRID>
<svTRID>UKR-201006041155433</svTRID>
</trID>
</response>
</epp>
```

3.3.3. Коди відповідей при виконанні команди <logout>

Код відповіді	Опис
1500	"Command completed successfully; ending session" "Команда виконана успішно, завершення сесії" "Команда выполнена успешно, завершение сессии"

3.4. POLL

Команда EPP <poll> призначена для читання службових повідомлень, які сервер EPP направляє клієнтові.

Якщо черга повідомлень не порожня, то успішна відповідь на команду <poll> поверне перше (найстаріше) повідомлення з черги повідомлень. У відповіді сервера міститься унікальний ідентифікатор повідомлення (необхідний для підтвердження отримання повідомлення) і лічильник, що показує загальну кількість повідомлень у черзі. Загальна кількість повідомлень для одного клієнта в черзі необмежена. Але повідомлення, старші ніж 1 місяць, будуть автоматично видалені.

Отримавши повідомлення, клієнт повинен відповісти на нього явним підтвердженням прийому. Отримавши підтвердження, сервер видалить повідомлення з черги, зменшить лічильник у черзі і зробить доступним для читання наступне повідомлення в черзі (якщо воно існує).

Системні повідомлення створюються для клієнтів, що потрапили під дію, модифікації над об'єктом. Наприклад, інформація про те, що надійшла команда <transfer> додається у чергу Реєстратору, який уповноважений прийняти або відхилити запит на трансфер.

Модифікація об'єктів, які зберігаються в Реєстрі, можуть відбуватися:

1. з ініціативи поточного Реєстратора об'єкта,
2. як результат виконання команд іншого клієнта,
3. в наслідок дій самого сервера.

Повідомлення в черзі зберігаються обмежений час, який визначається параметрами Реєстру. Якщо клієнти занадто довго не видаляють прочитані повідомлення (підтверджують їх прийом), то самі старі повідомлення будуть автоматично видалятися за правилом "перший надійшов - перший видалений".

Команда <poll> має атрибут "op", що визначає наступне:

1. "req" - читання першого непрочитаного повідомлення з черги повідомлень;
2. "ack" - підтвердження прочитання повідомлення.

3.4.1. Приклад команди <poll op="req">

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<poll op="req"/>
<clTRID>USER2-1276609056</clTRID>
</command>
</epp>

```

3.4.2. Приклад відповіді на команду <poll> з інформацією, специфічною для об'єкта

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
  <response>
    <result code="1301">
      <msg lang="en">Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="5" id="12345">
      <qDate>2010-06-15T16:37:35+03:00</qDate>
      <msg>Transfer requested.</msg>
    </msgQ>
    <resData>
      <domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>xn--elaybc.xn--j1amh</domain:name>
        <domain:trStatus>pending</domain:trStatus>
        <domain:reID>test</domain:reID>
        <domain:reDate>2010-06-15T16:37:35+03:00</domain:reDate>
        <domain:acID>registrar</domain:acID>
        <domain:acDate>2010-06-22T16:37:35+03:00</domain:acDate>
        <domain:exDate>2012-06-10T22:58:28+03:00</domain:exDate>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>USER2-1276609056</clTRID>
      <svTRID>UKR-20100615163736-665862-00002</svTRID>
    </trID>
  </response>

```

</epp>

В атрибуті "id" елемента msgQ вказується номер поточного повідомлення та кількість повідомлень, що залишилися в черзі (у цьому прикладі id поточного повідомлення 12345, всього повідомлень у черзі 5).

Клієнт повинен підтвердити прийом кожної відповіді, щоб виключити повідомлення з черги і зробити подальші повідомлення доступними для читання.

Команда <poll> з атрибутом "ack" повинна мати додатковий атрибут "msgID", що вказує номер повідомлення в черзі.

```
3.4.3. Приклад підтвердження за допомогою команди <poll op="ack">
<? xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
<command>
<poll op="ack" msgID="12345" />
<clTRID> USER2-1276609057 </clTRID>
</command>
</epp>
```

По результатах виконання цієї команди повідомлення з поточним id=12345 буде видалено з черги. У черзі залишиться 4 повідомлення.

3.4.4. Приклад відповіді на команду <poll> з підтвердженням читання повідомлення

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<msgQ count="4" id="12345" />
<trID>
<clTRID>USER2-1276609057</clTRID>
<svTRID>UKR-20100615163737</svTRID>
</trID>
</response>
</epp>
```

В атрибуті "id" елементу msgQ міститься номер видаленого повідомлення та кількість повідомлень, що залишилися в черзі.

3.4.5. Приклад відповіді на команду <poll op="req">, який показує, що черга повідомлень порожня

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1300">
<msg lang="en">Command completed successfully; no messages</msg>
</result>
<trID>
<clTRID>USER2-1276609059</clTRID>
<svTRID>UKR-201026151637393</svTRID>
</trID>
</response>
</epp>
```

3.5. DOMAIN:CHECK

Команда EPP <domain:check> призначена для перевірки існування домену в реєстрі або перевірки можливості реєстрації нового домену з заданим ім'ям.

3.5.1. Елементи команди <domain:check>

Команда <domain:check> містить один або кілька (але не більше 10) елементів <domain:name>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я

<domain:name> - повністю визначене доменне ім'я, яке може бути вказано або в кодуванні utf-8, або в кодуванні Punycode.

3.5.2. Приклад команди <domain:check>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<check>
<domain:check xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--e1aybc.xn--j1amh</domain:name>
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:check>
</check>
```

```
<clTRID>USER-1275850648</clTRID>
</command>
</epp>
```

3.5.3. Приклад відповіді сервера на команду <domain:check>

При успішній обробці команди <domain:check> у відповіді сервера елемент <resData> містить під-елемент <domain:chkData>, в якому присутні елементи <domain:cd> - по одному на кожен <domain:name> в запиті.

3.5.4. Під-елементи команди <domain:cd>

Під-елемент		Опис
<domain:name avail="0/1">	М	Задане доменне ім'я і атрибут "avail", який може приймати значення "0" або "1". "0" означає, що такий домен не може бути створений, "1" означає, що домен синтаксично коректний і вільний для реєстрації.
<domain:reason>	О	Необов'язковий елемент. Задається при avail = "0" і описує причину, по якій домен не може бути зареєстрований. Причини можуть бути наступні: • Невірне ім'я домену, Об'єкт вже існує (Object exists); • Даний публічний домен не обслуговується (Object not supported); • Домен зарезервовано, географічна назва (Object reserved - placename); • Домен зарезервовано, для органів влади (Object reserved - authority); • Домен зарезервовано (Object reserved); • Домен внесено до стоп-листу (Object prohibited).

3.5.5. Приклад відповіді команди <domain:check>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
```

```

<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:chkData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:cd>
<domain:name avail="0">xn--e1aybc.xn--j1amh</domain:name>
<domain:reason>Object exists</domain:reason>
</domain:cd>
<domain:cd>
<domain:name avail="1">xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:cd>
</domain:chkData>
</resData>
<trID>
<clTRID>USER-1275850648</clTRID>
<svTRID>UKR-20100606215846</svTRID>
</trID>
</response>
</epp>

```

3.5.6. Коди відповіді при виконанні команди<domain:check>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2002	"Command use error" "Ошибка в использовании команды" "Помилка у використанні команди" (при спробі вказати або 0 доменів, або більш ніж 10 доменів, або одне ім'я використовується більш одного разу)

3.6. DOMAIN:INFO

Команда EPP <domain:info> призначена для отримання інформації про об'єкт Domain.

3.6.1. Елементи команди<domain:info>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:authInfo>	О	Додатковий необов'язковий елемент, що містить пароль об'єкта. Задається для отримання повного доступу до об'єкта клієнтом, який не є поточним Реєстратором об'єкта.

3.6.2. Приклад команди <domain:info>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<info>
<domain:info xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--el aybc.xn--j1 amh</domain:name>
</domain:info>
</info>
<clTRID>USER-1276595352</clTRID>
</command>
</epp>
```

3.6.3. Елементи відповіді на команду <domain:info>

Відповідь сервера містить секцію <resData><infData>, в яку входять наступні елементи

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:roid>	М	Унікальний внутрішній ідентифікатор об'єкта в реєстрі.
<domain:status>	М	Статус об'єкта (по одному елементу <domain:status> на кожен статус).
<domain:registrant>	М*	Ідентифікатор контакту - реєстранта

		домену.
<domain:contact>	M*	Кілька елементів, що вказують на адміністративні, технічні контакти домену.
<domain:ns>	M*	Список елементів <domain:hostObj>, що визначають сервера, на яких делеговано домен.
<domain:host>	O	Необов'язкові елементи, що визначають підлеглі сервера, тобто сервера, зареєстровані у вказаному домені (тільки в авторизованому запиті).
<domain:clID>	M	Ідентифікатор Реєстратора - поточного Реєстратора об'єкта
<domain:crID>	O	Ідентифікатор Реєстратора, що створив об'єкт
<domain:crDate>	M*	Дата і час створення об'єкта
<domain:upID>	O	Ідентифікатор Реєстратора, який вніс останню зміну (тільки в авторизованому запиті).
<domain:upDate>	O	Дата і час останньої зміни об'єкта (тільки в авторизованому запиті).
<domain:trDate>	O	Дата та час останнього трансферу об'єкта (тільки в авторизованому запиті).
<domain:exDate>	M*	Дата та час закінчення реєстрації домену.
<domain:authInfo><pw>	O	authInfo - пароль до об'єкта (тільки в авторизованому запиті).

* - обов'язкові у авторизованому режимі

3.6.4. Приклад відповіді сервера на команду <domain:info> для авторизованого користувача

Таку відповідь отримує поточний Реєстратор об'єкта Domain або клієнт, який вказав коректний пароль (authInfo) об'єкта Domain.


```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:infData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>example1.com</domain:name>
<domain:roid>UKR-D00001111111</domain:roid>
<domain:status s="ok"/>
<domain:registrant>ex123</domain:registrant>
<domain:contact type="admin">contact-id1</domain:contact>
<domain:contact type="tech">contact-id2</domain:contact>
<domain:ns>
<domain:hostObj>xn--1-ytbm.xn--e1aybc.xn--j1amh</domain:hostObj>
<domain:hostObj>ns2.example.com</domain:hostObj>
</Domain:ns>
<domain:host>xn--1-ytbm.xn--e1aybc.xn--j1amh</domain:host>
<domain:clID>registrar</domain:clID>
<domain:crID>registrar</domain:crID>
<domain:crDate>2011-07-10T12:13:12+03:00</domain:crDate>
<domain:upID>registrar</domain:upID>
<domain:upDate>2011-08-11T13:20:26+03:00</domain:upDate>
<domain:exDate>2012-07-10T12:13:12+03:00</domain:exDate>
<domain:authInfo>
<domain:pw>5r$dfewa76Y2ef32o</domain:pw>
</domain:authInfo>
</domain:infData>
</resData>
<trID>
<clTRID>USER-1776595352</clTRID>
<svTRID>UKR-20100615124912-000397-00002</svTRID>
</trID>
</response>
</epp>

```

3.6.5. Приклад відповіді сервера на неавторизований запит <domain:info>.

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg lang="en">Command completed successfully</msg>
    </result>
    <resData>
      <domain:infData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>третій-домен-второго-регистратора.укр</domain:name>
        <domain:roid>D00076630-ukr</domain:roid>
        <domain:status s="clientDeleteProhibited"/>
        <domain:clID>111</domain:clID>
      </domain:infData>
    </resData>
    <trID>
      <clTRID>USER-1276595352</clTRID>
      <svTRID>UKR-1382369335756-1764495986181856-356</svTRID>
    </trID>
  </response>
</epp>
```

3.6.6. Коды відповіді при виконанні команди <domain:info>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2202	"Invalid authorization information" "Недійсні дані аутентифікації" "Недействительные данные аутентификации"
2303	"Object does not exist"

	"Об'єкт не існує" "Объект не существует"
--	---

3.7. DOMAIN:CREATE

Команда EPP <domain:create> призначена для створення об'єкта Domain.

3.7.1. Елементи команди <domain:create>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:period>	О	Необов'язковий елемент, що визначає термін, на який реєструється домен (в роках). Якщо елемент не вказаний, реєстр встановлює термін, прийнятий як мінімальний для даного публічного домену.
<domain:ns>	О	Необов'язковий елемент, що містить список серверів, на яких буде делеговано домен. Сервера можуть бути зазначені або в якості посилань на вже створені об'єкти "Host" (<domain:hostObj>)
<domain:registrant>	М	Обов'язковий елемент, що визначає реєстранта домену. Це має бути посилання на об'єкт Contact, вже наявний (попередньо зареєстрований) в реєстрі.
<domain:contact>	М	Один або декілька елементів, що вказують на контакти домену. Це повинні бути посилання на об'єкти Contact, вже наявні (попередньо зареєстровані) в реєстрі. Необхідно вказати 4 контакти: Адміністративний контакт (Administrative contact), Технічний контакт (Technical contact), Фінансовий контакт (Billing contact), Реєстрант контакт (Registrant contact).
<domain:authInfo>	М	Обов'язковий елемент, що містить пароль

		домену.
--	--	---------

3.7.2. Приклад команди <domain:create>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<create>
<domain:create xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:period unit="y">1</domain:period>
<domain:ns>
<domain:hostObj>ns1.example.com</domain:hostObj>
<domain:hostObj>ns2.example.com</domain:hostObj>
</domain:ns>
<domain:registrant>contact-id1</domain:registrant>
<domain:contact type="admin">contact-id2</domain:contact>
<domain:contact type="tech">contact-id2</domain:contact>
<domain:contact type="billing">contact-id4</domain:contact>
<domain:authInfo>
<domain:pw>2fo$02xxBAR</domain:pw>
</domain:authInfo>
</domain:create>
</create>
<c|TRID>USER-98679171392</c|TRID>
</command>
</epp>
```

3.7.3. Приклади команди <domain:create> якщо вказується номер знака для товарів та послуг (для періоду пріоритетної реєстрації):

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<create>
<domain:create xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:period unit="y">1</domain:period>
<domain:ns>
```

```

<domain:hostObj>ns1.example.com</domain:hostObj>
<domain:hostObj>ns2.example.com</domain:hostObj>
</domain:ns>
<domain:registrant>contact-id1</domain:registrant>
<domain:contact type="admin">contact-id2</domain:contact>
<domain:contact type="tech">contact-id2</domain:contact>
<domain:contact type="billing">contact-id4</domain:contact>
    <domain:license>112233</domain:license>
    <domain:authInfo>
        <domain:pw>2fo$02xxBAR</domain:pw>
    </domain:authInfo>
</domain:create>
</create>
<extension>
    <sunrise:sunrise xmlns:sunrise="uregistry:epp:sunrise">
        <sunrise:part>5.7.6</sunrise:part>
        <sunrise:reference>xn--e1aybc.xn--j1amh-ukr-0099990088</sunrise:reference>
    </sunrise:sunrise>
</extension>
<clTRID>USER-55798768392</clTRID>
</command>
</epp>

```

3.7.4. Коди відповіді при виконанні команди <domain:create>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
1001	"Command completed successfully; action pending" "Команда выполнена успешно, действие отложено" "Команду виконано успішно, дію відкладено"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"

2003	"Required parameter missing" "Відсутній обов'язковий параметр" "Отсутствует обязательный параметр"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2302	"Object exists" "Об'єкт вже існує" "Объект уже существует"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует" <i>Повідомлення про помилку видається при спробі використовувати в якості контакту або хосту неіснуючий об'єкт.</i>
2306	"Parameter value policy error" "Значення параметра не відповідає правилам" "Значение параметра не соответствует правилам" <i>(додатково в блоці розширеної відповіді будуть надані детальні причини, що вікликали відмову в обробці:</i> <code><extValue></code> <code><value xmlns:obj="urn:ietf:params:xml:ns:obj"></code> <i>Тут буде зазначено фрагмент xml, що містить помилку</i> <code></value></code> <code><reason>детальна інформація про причину<</code> <code>reason></code>

	</extValue>)
2307	"Unimplemented object service" "Даний об'єкт не обслуговується" "Данный объект не обслуживается"

3.8. DOMAIN:UPDATE

Команда EPP <domain:update> призначена для зміни об'єкта Domain. Тільки поточний Реєстратор може оперувати з командою EPP <domain:update>. У команді <domain:update> повинен бути присутнім хоча б один з елементів <domain:add>, <domain:rem> або <domain:chg>, незважаючи на те, що кожен з них є опціональний. Якщо об'єкт перебуває в статусі clientUpdateProhibited, то всі зміни, окрім відміни статусу clientUpdateProhibited, будуть не доступні. В межах однієї команди не допускається одночасне використання одних й тих самих даних в розділах <domain:add> і <domain:rem>, наприклад, у об'єкта Domain не можна одночасно додати и видати однаковий name-сервер.

3.8.1. Елементи команди<domain:update>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:authInfo>	М	Елемент, який встановлює authInfo (пароль до об'єкту). Може містити елемент <domain:pw>, в якому вказується новий пароль, або елемент <domain:null/>, що означає обнуління пароля (автоматична генерація та зміна пароля реєстром).
<domain:add>	О	Необов'язковий елемент, що містить значення атрибутів, які необхідно додати до об'єкта.
<domain:rem>	О	Необов'язковий елемент, що містить значення атрибутів, які необхідно видалити з об'єкта.
<domain:chg>	О	Необов'язковий елемент, що містить значення атрибутів, які необхідно змінити.
<domain:ns>	О	Перелік name-серверів, що додаються або видаляються.

<domain:contact>	O	Перелік ідентифікаторів контактів, що додаються або видаляються.
<domain:status>	O	Перелік статусів, що додаються або видаляються з об'єкта Domain. Статуси які може встановити Реєстратор: • clientUpdateProhibited • clientDeleteProhibited • clientTransferProhibited • clientRenewProhibited • clientHold
<domain:hostObj>	O	Адреса name-сервера
<domain:hostAttr>	O	Елемент атрибутів хосту
<domain:hostName>	O	Ідентифікатор хосту (Ім'я хосту)
<domain:hostAddr ip="v4">	O	IPv4 адреса хосту
<domain:hostAddr ip="v6">	O	IPv6 адреса хосту

3.8.2. Приклад команди <domain:update> для зміни контактів і name-серверів

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<update>
<domain:update xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:add>
<domain:ns>
<domain:hostObj>ns2.domain.com</domain:hostObj>
</domain:ns>
<domain:contact type="tech">new-tech-contact</domain:contact>
</domain:add>
<domain:rem>
<domain:ns>

```



```

<domain:hostObj>ns1.domain.com</domain:hostObj>
</domain:ns>
<domain:contact type="tech">old-tech-contact</domain:contact>
</domain:rem>
<domain:chg>
<domain:registrant>current-registrant-contact</domain:registrant>
</domain:chg>
</domain:update>
</update>
<clTRID>USER-344324324342</clTRID>
</command>
</epp>

```

3.8.3. Приклад команди <domain:update> для зміни clientTransferProhibited і обнуління authInfo

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<update>
<domain:update xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:add>
<domain:status s="clientTransferProhibited"/>
</domain:add>
<domain:chg>
<domain:authInfo>
<domain:null/>
</domain:authInfo>
</domain:chg>
</domain:update>
</update>
<clTRID>USER-4546554654654</clTRID>
</command>
</epp>

```

3.8.4. Приклад команди <domain:update> для відміни статусу clientTransferProhibited та завдання нового authInfo

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<update>
<domain:update xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:rem>
<domain:status s="clientTransferProhibited"/>
</domain:rem>
<domain:chg>
<domain:authInfo>
<domain:pw>3434g#dSdr4</domain:pw>
</domain:authInfo>
</domain:chg>
</domain:update>
</update>
<clTRID>USER-5454545453</clTRID>
</command>
</epp>

```

3.8.5. Завдання елементу <domain:ns>

Для завдання елемента <domain:ns> можна використовувати декілька способів:

1. як посилання на вже наявні в реєстрі об'єкти "Host":

```

<domain:ns>
<domain:hostObj>xn--1-ytbm.xn--e1aybc.xn--j1amh</domain:hostObj>
<domain:hostObj>xn--2-ytbm.xn--e1aybc.xn--j1amh</domain:hostObj>
</domain:ns>

```

3.8.6. Приклад відповіді сервера на команду <domain:update>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>

```

<clTRID>USER-54654654654654</clTRID>

<svTRID>UKR-344545453453</svTRID>

</trID>

</response>

</epp>

3.8.7. Коди відповідей при виконанні команди<domain:update>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2002	"Command use error" "Ошибка в использовании команды" "Помилка у використанні команди"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (дане повідомлення надається, якщо виконується дія над доменом, який обслуговується іншим Реєстратором)
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует" Повідомлення про помилку видається при спробі

	<i>використання у команді неіснуючих об'єктів.</i>
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции" (дане повідомлення надається якщо виконується дія над доменом, що має статус <i>clientUpdateProhibited</i> , <i>serverUpdateProhibited</i> або статуси <i>pendingDelete</i> , <i>pendingTransfer</i> , <i>redemptionPeriod</i> , <i>changeProhibited</i>)
2309	"Application rejected" "Заявка відхилена" "Заявка отклонена"

3.9. DOMAIN:RENEW

Команда EPP <domain:renew> призначена для продовження терміну реєстрації домену. Команда доступна тільки для Реєстратора відповідного домену. Якщо домен має статус *clientRenewProhibited* або *serverRenewProhibited*, команда <domain:renew> буде відхилена.

3.9.1. Елементи команди <domain:renew>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я.
<domain:curExpDate>	М	Поточний термін закінчення реєстрації домену. Цей елемент призначений гарантувати, що при випадковому повторі команди не відбудеться несанкціонованого продовження терміну реєстрації.
<domain:period unit="y">	М	Кількість років, на які подовжується термін реєстрації. Період продовження реєстрації при додаванні до поточного періоду реєстрації домену не може перевищувати максимальне значення періоду реєстрації (визначається Правилами).

3.9.2. Приклад команди <domain:renew>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```

<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<renew>
<domain:renew xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:curExpDate>2012-07-10</domain:curExpDate>
<domain:period unit="y">2</domain:period>
</domain:renew>
</renew>
<clTRID>USER-127677777</clTRID>
</command>
</epp>

```

3.9.3. Елементи відповіді на команду <domain:renew>

Відповідь сервера містить секцію <resData><renData>, що містить такі елементи:

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я.
<domain:exDate>	М	Нова дата та час (локальне) закінчення реєстрації домену

3.9.4. Приклад відповіді сервера на команду <domain:renew>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:renData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:exDate>2014-07-10T15:13:12+03:00</domain:exDate>
</domain:renData>
</resData>
<trID>
<clTRID>DRID-1276599424</clTRID>

```

```

<svTRID>UKR-65567135704</svTRID>
</trID>
</response>
</epp>

```

3.9.5. Коди відповіді при виконанні команди <domain:renew>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2105	"Object is not eligible for renewal" "Об'єкт не підлягає продовженню" "Объект не подлежит продлению"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"
2306	"Parameter value policy error"

	"Значення параметра не відповідає правилам" "Значение параметра не соответствует правилам"
2309	"Application rejected" "Заявка відхилена" "Заявка отклонена"

3.10. DOMAIN:DELETE

Команда EPP <domain:delete> призначена для видалення об'єкта Domain.

3.10.1. Елементи команди<domain:delete>

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я

Команда видалення хоста доступна тільки для поточного Реєстратора домену. Якщо об'єкт Domain має статус clientDeleteProhibited або serverDeleteProhibited, команда буде відхилена.

Якщо об'єкт Domain має хоча б один зі статусів: clientDeleteProhibited, serverDeleteProhibited або pendingTransfer – він не може бути видалений.

3.10.2. Перелік перевірок під час обробки команди <domain:delete>

Під час обробки команди < domain:delete> проводяться наступні перевірки:

1. Чи відповідає доменне ім'я вимогам RFC 1034,
2. Чи існує інформація про домен в Реєстрі,
3. Чи надійшла заявка від належного Реєстратора,
4. Чи дозволяє статус об'єкту операцію видалення.

3.10.3. Приклад команди <domain:delete>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<delete>
<domain:delete xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:delete>
</delete>
<clTRID>USER-5678241099</clTRID>
</command>
</epp>
```

3.10.4. Приклад відповіді сервера на команду <domain:delete>

При успішній обробці команди <domain:delete> у відповіді сервера секція <resData> відсутня.

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1001">
<msg lang="en">Command completed successfully; action pending</msg>
</result>
<trID>
<clTRID>USER-137777777</clTRID>
<svTRID>UKR-25656561139-986804-02211</svTRID>
</trID>
</response>
</epp>
```

3.10.5. Коди відповіді при виконанні команди <domain:delete>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
1001	"Command completed successfully; action pending" "Команду виконано успішно, дію відкладено" "Команда выполнена успешно, действие отложено"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"

2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"
2305	"Object association prohibits operation" "Зв'язки об'єкта не дозволяють здійснення операції" "Объектные связи не допускают совершение операции"

3.11. DOMAIN RESTORE

Для відновлення об'єкта Domain, що знаходиться в стадії життєвого циклу "Період очікування відновлення" (Redemption Period) призначена команда EPP `<domain:update>` з розширенням `rgp` (Restore Grace Period).

Об'єкт Domain потрапляє в "Період очікування відновлення" (Redemption Period) після успішної обробки реєстром команди `<domain:delete>`. "Періоду очікування відновлення" (Redemption Period) відповідає статус об'єкта Domain `redemptionPeriod`. Коли домен знаходиться в цьому статусі, сервер для даного об'єкта приймає команду відновлення домену - `<rgp:update>`.

Команда відновлення домену доступна тільки поточного Реєстратора домену. Для успішної обробки команди, домен повинен мати статус `redemptionPeriod`. При успішному отриманні команди, домен відновлюється негайно, хости домену не відновлюються.

3.11.1. Елементи команди `<rgp:update>`

Елемент		Опис
<code><domain:name></code>	М	Повністю визначене доменне ім'я
<code><rgp:restore op="request"/></code>	М	Розширення команди <code><rgp:restore></code> , що вказує на запит відновлення домену.

3.11.2. Приклад команди `<rgp:update>`

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<update>
<domain:update xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
```

```

<domain:name>xn--e1aybc.xn--j1amh</domain:name>
</domain:update>
</update>
<extension>
<rgp:update
  xmlns:rgp="urn:ietf:params:xml:ns:rgp-1.0">
  <rgp:restore op="request"/>
</rgp:update>
</extension>
<clTRID>USER-1232323</clTRID>
</command>
</epp>

```

3.11.3. Приклад відповіді серверу на команду <rgp:update>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-1343545453</clTRID>
<svTRID>UKR-3454543543543</svTRID>
</trID>
</response>
</epp>

```

3.11.4. Коди відповідей при виконанні команди<rgp:update>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"

2002	"Command use error" "Ошибка в использовании команды" "Помилка у використанні команди"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (дане повідомлення надається, якщо виконується дія над доменом, який обслуговується іншим Реєстратором)
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"
2306	"Parameter value policy error" "Значення параметра не відповідає правилам" "Значение параметра не соответствует правилам" (дане повідомлення надається, якщо виконується команда <rgp:update> з одночасною зміною name-серверів, контактів та ін.)
2309	"Application rejected" "Заявка відхилена" "Заявка отклонена"

3.12. DOMAIN:TRANSFER

Команда EPP <domain:transfer> призначена для передачі домену між реєстраторами. Команда domain:transfer з різними атрибутами “op” доступна як для приймаючого так і віддаючого реєстраторів в залежності від ролі Реєстратора під час трансферу.

3.12.1. Атрибути команди <domain:transfer>

Можливі атрибути “op”:

1. "request" - запит на трансфер (від Реєстратора, який бажає прийняти на обслуговування домен);
2. "cancel" - відмова від запиту на трансфер (від Реєстратора, який подав запит; "request");
3. "approve" - згода поточного Реєстратора на трансфер;
4. "reject" - відмова поточного Реєстратора від трансферу;
5. "query" - перегляд поточного стану операції трансфер (може використовувати кожна зі сторін).

3.13. DOMAIN:TRANSFER REQUEST

Команда EPP <domain:transfer op="request"> домену призначена для відправки запиту приймаючим Реєстратором до поточного Реєстратору з метою здійснення трансферу домену.

Для того щоб подати команду <domain:transfer op="request">, приймаючий Реєстратор має вказати пароль об'єкта domain (authInfo). Якщо домен (об'єкт Domain) має хоча б один зі статусів: clientTransferProhibited, serverTransferProhibited, clientUpdateProhibited, serverUpdateProhibited, RedemptionPeriod, PendingDelete, pendingTransfer, changeProhibited команда буде відхилена.

3.13.1. Елементи команди<domain:transfer op="request">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я.
<domain:period unit="y">	М	Кількість років, на які продовжується термін реєстрації. Період продовження реєстрації при додаванні до поточного періоду реєстрації домену не може перевищувати максимальне значення періоду реєстрації. Якщо значення, вказане в елементі <domain:period unit="y">, перевищує максимальний термін реєстрації домену, команду <transfer op="request"> буде відхилено.
<domain:pw>		Пароль об'єкта domain (authInfo).

3.13.2. Елементи відповіді на команду <domain:transfer op="request">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я.
<domain:trStatus>	М	Поточний статус операції трансферу - "pending".
<domain:reID>	М	Ідентифікатор Реєстратора, який надіслав запит на трансфер (ідентифікатор приймаючого Реєстратора).
<domain:reDate>	М	Локальні дата і час, коли був надісланий запит на трансфер.
<domain:acID>	М	Ідентифікатор поточного Реєстратора домену.
<domain:acDate>	М	Локальні дата і час, до яких очікується відповідь на запит від поточного Реєстратора - дата до автоматичного підтвердження трансферу.
<domain:exDate>	М	Локальні дата і час закінчення реєстрації домену, які будуть встановлені в разі успішного проведення операції трансферу.

3.13.3. Приклад команди <domain:transfer op="request">

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<transfer op="request">
<domain:transfer xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:period unit="y">1</domain:period>
<domain:authInfo>
<domain:pw>FO1Obar22</domain:pw>
</domain:authInfo>
</domain:transfer>
</transfer>
<clTRID>USER-343434324232</clTRID>
</command>
</epp>

```

3.13.4. Приклад відповіді сервера на команду <domain:transfer op="request">

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1001">
<msg lang="en">Command completed successfully; action pending</msg>
</result>
<resData>
<domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:trStatus>pending</domain:trStatus>
<domain:reID>registrar-2</domain:reID>
<domain:reDate>2013-06-05T12:17:31+03:00</domain:reDate>
<domain:acID>registrar-1</domain:acID>
<domain:acDate>2013-06-10T12:17:31+03:00</domain:acDate>
<domain:exDate>2014-07-11T21:58:18+03:00</domain:exDate>
</domain:trnData>
</resData>
<trID>
<clTRID>USER-2323231113</clTRID>
<svTRID>UKR-23232323232</svTRID>
</trID>
</response>
</epp>
```

3.13.5. Коди відповіді при виконанні команди <domain:transfer op="request">

При виконанні команди <domain:transfer op="request"> можливі наступні відповіді серверу:

Код відповіді	Опис
1001	"Command completed successfully; action pending" "Команду виконано успішно, дію відкладено" "Команда выполнена успешно, действие отложено"

2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2106	"Object is not eligible for transfer" "Об'єкт не підлягає трансферу" "Объект не подлежит трансферу"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2202	"Invalid authorization information" "Недійсні дані аутентифікації" "Недействительные данные аутентификации"
2300	"Object pending transfer" "Об'єкт в очікуванні трансферу" "Объект в ожидании трансфера"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"
2307	"Unimplemented object service" "Об'єкт не обслуговується" "Об'єкт не обслуживается" (дане повідомлення надається, якщо Реєстратору не доступні операції в доменній зоні)

2309	"Application rejected" “Заявка відхилена” “Заявка отклонена” <i>(дане повідомлення надається, якщо у Реєстратора недостатньо коштів на поточному балансі або заявка не відповідає умовам здійснення трансферу)</i>
2310	"Application rejected; Maximum registration period exceed." “Заявка відхилена; Перевищено максимальний термін реєстрації доменів” “Заявка отклонена; Превышен максимальный период регистрации”

3.13.6. Дії після успішного виконання запиту на трансфер домену

Після отримання успішного запиту на трансфер, об’єкт Domain отримує статус pendingTransfer. Поточному Реєстратору домену надсилається повідомлення про отримання реєстром заявки на трансфер домену через систему повідомлень <poll> EPP. Повідомлення також дублюється по електронній пошті обом Реєстраторам.

Впродовж автоматичного опрацювання трансферу (5 діб) поточний Реєстратор має можливість достроково підтвердити або відхилити трансфер домену. При отриманні дострокової відповіді заявка на передачу негайно обробляється: при позитивній відповіді - заявка буде задоволена, при негативному - у виконанні трансферу буде відмовлено.

Якщо після закінчення часу автоматичного опрацювання дострокова відповідь не була отримана, трансфер автоматично підтверджується реєстром у наступній послідовності:

1. Заявці на трансфер присвоюється статус "serverApproved" (трансфер підтверджений сервером);
2. Домен і його хости передаються під керування приймаючого Реєстратора;
3. Ідентифікатори контактів залишаються незмінними;
4. Реєстратор який прийняв домен може самостійно задати нові ідентифікатори контактів;
5. Статус pendingTransfer у домену та підлеглих хостів знімається;
6. Домен продовжується на період що вказаний при подачі команди DOMAIN:TRANSFER REQUEST (якщо новий період вичерпання реєстрації не перевищує максимально-допустимий);

7. AuthInfo (пароль) домену автоматично змінюється;

3.14. DOMAIN:TRANSFER CANCEL

Команда EPP <domain:transfer op="cancel"> призначена для відмови від запиту на трансфер об'єкта Domain.

Команда <domain:transfer op="cancel"> приймається якщо об'єкт Domain перебуває в статусі pendingTransfer тільки від Реєстратора, що ініціював трансфер.

3.14.1. Елемент команди <domain:transfer op="cancel">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я

3.14.2. Приклад команди <domain:transfer op="cancel">

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<transfer op="cancel">
<domain:transfer xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:transfer>
</transfer>
<clTRID>USER-454545308</clTRID>
</command>
</epp>
```

3.14.3. Елементи відповіді на команду <domain:transfer op="cancel">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:trStatus>	М	Поточний статус операції трансферу - "clientCancelled".
<domain:reID>	М	Ідентифікатор Реєстратора, який надіслав запит на трансфер (ідентифікатор приймаючого Реєстратора).
<domain:reDate>	М	Локальні дата і час, коли був надісланий запит на

>		трансфер.
<domain:acID>	M	Ідентифікатор поточного Реєстратора домену.
<domain:acDate> >	M	Локальні дата і час відмови від трансферу.
<domain:exDate> >	M	Локальні дата і час закінчення терміну реєстрації домену.

3.14.4. Приклад відповіді сервера на команду <domain:transfer op="cancel">

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:trStatus>clientCancelled</domain:trStatus>
<domain:reID>registrar-2</domain:reID>
<domain:reDate>2010-07-15T16:37:35+03:00</domain:reDate>
<domain:acID>registrar-1</domain:acID>
<domain:acDate>2010-07-15T18:21:48+03:00</domain:acDate>
<domain:exDate>2011-07-10T22:58:28+03:00</domain:exDate>
</domain:trnData>
</resData>
<trID>
<clTRID>USER-2343243432</clTRID>
<svTRID>UKR-234343334343</svTRID>
</trID>
</response>
</epp>

```

3.14.5. Коди відповідей при виконанні команди<domain:transfer op="cancel">

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" <i>(дане повідомлення надається, якщо вказано не вірно один з параметрів)</i>
2106	"Object is not eligible for transfer" "Об'єкт не підлягає трансферу" "Объект не подлежит трансферу"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" <i>(дане повідомлення надається, якщо Реєстратор намагається відмінити, трансфер домену який до нього не трансфериться)</i>
2301	"Object not pending transfer" "Відсутній запит на трансфер об'єкту" "Отсутствие запроса на трансфер объекта" <i>(дане повідомлення надається якщо Реєстратор намагається відмінити трансфер домену трансфер якого взагалі не відбувається)</i>
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"

3.15. DOMAIN:TRANSFER APPROVE

Команда EPP <domain:transfer op="approve"> призначена для підтвердження запиту на трансфер об'єкта Domain.

Команда <domain:transfer op="approve"> приймається якщо об'єкт Domain перебуває в статусі pendingTransfer тільки від поточного Реєстратора домену.

3.15.1. Елементи команди<domain:transfer op="approve">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я

3.15.2. Приклад команди <domain:transfer op="approve">

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<transfer op="approve">
<domain:transfer xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:transfer>
</transfer>
<clTRID>USER-1272324338</clTRID>
</command>
</epp>
```

3.15.3. Елементи відповіді на команду<domain:transfer op="approve">

Відповідь на команду <domain:transfer op="approve"> містить наступні елементи:

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:trStatus>	М	Поточний статус операції трансферу - "clientApproved".
<domain:reID>	М	Ідентифікатор Реєстратора, який надіслав запит на трансфер (ідентифікатор приймаючого Реєстратора).
<domain:reDate>	М	Локальні дата і час, коли був надісланий запит на трансфер.
<domain:acID>	М	Ідентифікатор поточного Реєстратора домену.

<domain:acDate>	М	Локальні дата і час підтвердження трансферу.
<domain:exDate>	М	Локальні дата і час нового терміну закінчення реєстрації домену.

3.15.4. Приклад відповіді сервера на команду <domain:transfer op="approve">

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:trStatus>clientApproved</domain:trStatus>
<domain:reID>registrar-2</domain:reID>
<domain:reDate>2010-07-10T16:37:35+03:00</domain:reDate>
<domain:acID>registrar-1</domain:acID>
<domain:acDate>2010-07-10T18:21:48+03:00</domain:acDate>
<domain:exDate>2012-07-05T22:58:28+03:00</domain:exDate>
</domain:trnData>
</resData>
<trID>
<clTRID>USER-1276615308</clTRID>
<svTRID>UA-20100615182149-170357-00002</svTRID>
</trID>
</response>
</epp>
```

3.15.5. Коди відповідей при виконанні команди <domain:transfer op="approve">

При виконанні команди <domain:transfer op="approve"> можливі наступні коди відповіді серверу:

Код	Опис
-----	------

ВІДПОВІДІ	
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (дане повідомлення надається якщо Реєстратор намагається відмінити трансфер домену, який до нього не трансфериться)
2301	"Object not pending transfer" "Відсутній запит на трансфер об'єкту" "Отсутствие запроса на трансфер объекта" (дане повідомлення надається, якщо Реєстратор намагається відмінити трансфер домену трансфер, якого взагалі не відбувається)
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"

3.15.6. Дії після успішного виконання команди <domain:transfer op="approve">

Після успішного виконання команди <domain:transfer op="approve">реєстр виконує наступні дії:

1. Заявка на передачу задовольняється, їй присвоюється статус "clientApproved" (трансфер підтверджений поточним Реєстратором);
2. Домен і його підлеглі хости переводяться під керування нового Реєстратора;

3. Статус pendingTransfer у домену та підлеглих хостів знімається;
4. Домен подовжується на період, що вказаний при подачі команди DOMAIN:TRANSFER REQUEST (якщо новий період вичерпання реєстрації не перевищує максимально-допустимий);
5. AuthInfo (пароль) домену автоматично змінюється.

3.16. DOMAIN:TRANSFER REJECT

Команда EPP <domain:transfer op="reject"> призначена для відхилення запиту на трансфер об'єкта Domain.

Команда <domain:transfer op="reject"> приймається, якщо об'єкт Domain перебуває в статусі pendingTransfer тільки від поточного Реєстратора домену.

3.16.1. Елементи команди <domain:transfer op="reject">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я

3.16.2. Приклад команди <domain:transfer op="reject">

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<transfer op="reject">
<domain:transfer xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
</domain:transfer>
</transfer>
<clTRID>USER-3434342432</clTRID>
</command>
</epp>
```

3.16.3. Елементи відповіді на команду <domain:transfer op="reject">

Відповідь на команду <domain:transfer op="reject"> містить наступні елементи:

Елемент		Опис
<domain:trStatus>	М	Поточний статус операції трансферу - "clientrejectd".
<domain:reID>	М	Ідентифікатор Реєстратора, який надіслав запит на трансфер (ідентифікатор приймаючого Реєстратора).

<domain:reDate>	M	Локальні дата і час, коли був надісланий запит на трансфер.
<domain:acID>	M	Ідентифікатор поточного Реєстратора домену.
<domain:acDate>	M	Локальні дата і час відмови від трансферу.
<domain:exDate>	M	Локальні дата і час терміну закінчення реєстрації домену.

3.16.4. Приклад відповіді на команду <domain:transfer op="reject">

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:trStatus>clientrejectd</domain:trStatus>
<domain:reID>registrar-2</domain:reID>
<domain:reDate>2010-07-10T16:37:35+03:00</domain:reDate>
<domain:acID>registrar-1</domain:acID>
<domain:acDate>2010-07-10T18:21:48+03:00</domain:acDate>
<domain:exDate>2011-07-05T22:58:28+03:00</domain:exDate>
</domain:trnData>
</resData>
<trID>
<clTRID>USER-1276615308</clTRID>
<svTRID>UKR-2334343432</svTRID>
</trID>
</response>
</epp>

```

3.16.5. Коди відповідей при виконанні команди <domain:transfer op="reject">

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (дане повідомлення надається, якщо Реєстратор намагається відмінити трансфер домену, який до нього не трансфериться)
2301	"Object not pending transfer" "Відсутній запит на трансфер об'єкту" "Отсутствие запроса на трансфер объекта" (дане повідомлення надається, якщо Реєстратор намагається відмінити трансфер домену трансфер, якого взагалі не відбувається)
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"

3.16.6. Дії після успішного виконання команди <domain:transfer op="reject">

Після успішного виконання команди <domain:transfer op="reject">реєстр виконує наступні дії:

1. Заявка на transfer відхиляється, з присвоєнням статусу "clientRejected" (трансфер відхилений поточним Реєстратором);
2. Статус pendingTransfer у домену та підлеглих хостів знімається;

3. AuthInfo (пароль) домену автоматично змінюється.

3.17. DOMAIN:TRANSFER QUERY

Команда EPP <domain:transfer op="query"> призначена для отримання інформації про стан операції трансферу об'єкта Domain. Дана команда доступна для обох Реєстраторів на протязі проведення трансферу.

Для того щоб подати команду <domain:transfer op="query">, приймаючий Реєстратор має вказати authInfo домену. Поточний Реєстратор може не вказувати пароль authInfo при виконанні команди <domain:transfer op="query"> протягом всього періоду проведення трансферу.

3.17.1. Елементи команди <domain:transfer op="query">

Елемент		Опис
<domain:name>	М	Повністю визначене доменне ім'я
<domain:authInfo>	М	Елемент, який вказує authInfo (пароль до об'єкта Domain)

3.17.2. Приклад команди <domain:transfer op="query">

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<transfer op="query">
<domain:transfer xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:authInfo>
<domain:pw>FO22Owbar22</domain:pw>
</domain:authInfo>
</domain:transfer>
</transfer>
<clTRID>USER-12434343455</clTRID>
</command>
</epp>
```

3.17.3. Елементи відповіді на команду <domain:transfer op="query">

Елемент		Опис
<domain:trStatus>	М	Поточний статус операції трансферу.

		Він може бути: • "pending" - операція не закінчена, очікується підтвердження; • "clientApproved" - трансфер підтверджений поточним реєстратором домену, операція завершена; • "clientRejected" - поточний Реєстратор домену відмовив у трансфері, операція завершена. • "clientCancelled" - запит на трансфер відмінено приймаючим Реєстратором, операція завершена. • "serverApproved" - трансфер підтверджений сервером. • "serverCancelled" - трансфер відмінено сервером.
<domain:reID>	M	Ідентифікатор Реєстратора, який надіслав запит на трансфер (ідентифікатор приймаючого Реєстратора).
<domain:reDate>	M	Локальні дата і час, коли був надісланий запит на трансфер.
<domain:acID>	M	Ідентифікатор Реєстратора, який повинен обробити (при статусі pending) заявку на трансфер або вже обробив (при інших значення статусу).
<domain:acDate>	M	Локальні дата і час, до якого очікується відповідь від поточного Реєстратора (при статусі pending) або дата і час обробки відповіді або автоматичної дії сервера.
<domain:exDate>	M	Локальні дата і час закінчення терміну реєстрації домену (при статусі pending - це майбутня дата закінчення терміну реєстрації домену).

3.17.4. Приклад відповіді на команду <domain:transfer op="query">
<?xml version="1.0" encoding="UTF-8"?>

```

<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully;</msg>
</result>
<resData>
<domain:trnData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
<domain:name>xn--2-itb0bcc.xn--j1amh</domain:name>
<domain:trStatus>pending</domain:trStatus>
<domain:reID>registrar-1</domain:reID>
<domain:reDate>2010-07-15T16:37:35+03:00</domain:reDate>
<domain:acID>registrar-2</domain:acID>
<domain:acDate>2010-07-20T16:37:35+03:00</domain:acDate>
<domain:exDate>2012-07-10T22:58:28+03:00</domain:exDate>
</domain:trnData>
</resData>
<trID>
<clTRID>USER-1276609155</clTRID>
<svTRID>UKR-23234343434</svTRID>
</trID>
</response>
</epp>

```

3.17.5. Коди відповідей при виконанні команди<domain:transfer op="query">

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра"

	"Синтаксическая ошибка параметра" (дане повідомлення надається, якщо вказано не вірно один з параметрів)
2106	"Object is not eligible for transfer" "Об'єкт не підлягає трансферу" "Объект не подлежит трансферу"
2202	"Invalid authorization information" "Недійсні дані аутентифікації" "Недействительные данные аутентификации"
2301	"Object not pending transfer" "Відсутній запит на трансфер об'єкту" "Отсутствие запроса на трансфер объекта"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"

3.18. HOST:CHECK

Команда EPP <host:check> призначена для перевірки існування хоста або для перевірки можливості реєстрації нового хоста з заданим ім'ям. В межах виконання однієї команди <host:check> допустима перевірка до **10 хостів** одночасно.

3.18.1. Елементи команди<host:check>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хосту

3.18.2. Приклад команди <host:check>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<check>
<host:check xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:name>xn--2-ytbm.xn--e1aybc.xn--j1amh</host:name>
</host:check>
</check>
```

<c|TRID>USER-3232323232</c|TRID>

</command>

</epp>

3.18.3. Елементи відповіді на команду<host:check>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста.
<host:name avail="0/1">	М	Задане ім'я хоста з атрибутом "avail", який може приймати значення "0" або "1". "0" означає, що такий хост не може бути створений, "1" означає, що ім'я хоста синтаксично коректно і вільно.
<host:reason>	О	Необов'язковий елемент. Задається за avail = "0", описує причину, по якій хост не може бути створений. Якщо, ім'я хоста не відповідає [RFC1034] - avail = "0" та <host:reason> "Incorrect hostname" Якщо, інформації про вузол немає в реєстрі - avail = "0" та <host:reason> "Object exists". При наданні даного повідомлення визначається тип хоста (внутрішній або зовнішній). Якщо хост внутрішній, у реєстрі перевіряється чи зареєстрований батьківський домен для даного хоста. Якщо такий домен не зареєстрований - avail = "0" та <host:reason> "Parent domain does not exist" Якщо команда надійшла від Реєстратора, який не є Реєстратором батьківського домену хоста - avail = "0" та <host:reason> "You have no permissions to add this nameserver".

3.18.4. Приклад відповіді серверу на команду <host:check>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<host:chkData xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:cd>
<host:name avail="1">xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
</host:cd>
<host:cd>
<host:name avail="0">xn--2-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:reason lang="en">Object exists</host:reason>
</host:cd>
</host:chkData>
</resData>
<trID>
<clTRID>USER-3434534534534</clTRID>
<svTRID>UKR-34234324324234</svTRID>
</trID>
</response>
</epp>

```

3.18.5. Коди відповідей при виконанні команди<host:check>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2002	"Command use error"

	"Ошибка в использовании команды" "Помилка у використанні команди" <i>(дане повідомлення надається, якщо перевищена максимальна кількість хостів, що перевіряються, в межах однієї команди)</i>
--	---

3.19. HOST:INFO

Команда EPP <host:info> призначена для отримання інформації про об'єкт Host. Виконати команду може як поточний так і сторонній Реєстратор для об'єкта Host, що базується на домені, який обслуговує реєстр.

3.19.1. Елементи команди<host:info>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста

3.19.2. Приклад команди <host:info>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<info>
<host:info xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbn.xn--e1aybc.xn--j1amh</host:name>
</host:info>
</info>
<clTRID>USER-3434343232</clTRID>
</command>
</epp>
```

3.19.3. Елементи відповіді на команду <host:info>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста.
<host:roid>	М	Унікальний внутрішній ідентифікатор об'єкта в реєстрі.
<host:status>	М	Статус об'єкта хост.
<host:addr>	М	IP-адреси об'єкта хост. Один хост може мати до 10

		IPv4-адрес та IPv6-адрес.
<host:clID>	M	Ідентифікатор поточного Реєстратора домену.
<host:crID>	M	Ідентифікатор Реєстратора, який створив об'єкт хост.
<host:crDate>	M	Локальні час і дата створення об'єкту хост.
<host:upID>	M	Ідентифікатор Реєстратора, який останній вніс зміни до об'єкту хост.
<host:upDate>	M	Локальні дата і час внесення останніх змін до об'єкту хост.
<host:trDate>	M*	Дата і час останнього трансферу батьківського домену.

* - якщо такий трансфер був здійснений

3.19.4. Приклад відповіді на команду <host:info>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<host:infData xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:roid>HOST-123232312</host:roid>
<host:status s="ok"/>
<host:addr ip="v4">91.247.23.12</host:addr>
<host:addr ip="v6">FE80::0202:B3FF:FE1E:8329</host:addr>
<host:clID>registrar-1</host:clID>
<host:crID>registrar-112</host:crID>
<host:crDate>2011-07-22T12:27:31+03:00</host:crDate>
</host:infData>
</resData>

```

```

<trID>
<clTRID>USER-321321312312312</clTRID>
<svTRID>UKR-2323232132323213</svTRID>
</trID>
</response>
</epp>

```

3.19.5. Коди відповідей при виконанні команди<host:info>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2003	"Required parameter missing" "Відсутній обов'язковий параметр" "Отсутствует обязательный параметр"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует" (Повідомлення надається якщо запитуваний хост не існує)

3.20. HOST:CREATE

Команда EPP <host:create> призначена для створення об'єкта Host. Поточний Реєстратор домену може створити хост з вказівкою IP-адреси для такого домену. При цьому, домен обов'язково повинен обслуговуватися Реєстром. Якщо створюється хост в домені, що не обслуговується Реєстром, IP-адресу вказувати не потрібно.

3.20.1. Елементи команди<host:create>

Елемент	Опис
---------	------

<host:name>	М	Повністю визначене ім'я хоста
<host:addr>	М	Елемент містить IP-адреси, які необхідно присвоїти даному хосту.

3.20.2. Приклад команди <host:create>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<create>
<host:create xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:addr ip='v4'>195.64.12.23</host:addr>
<host:addr ip='v6'>FE80::0202:B3FF:FE1E:8329</host:addr>
</host:create>
</create>
<clTRID>USER-3435435353</clTRID>
</command>
</epp>
```

3.20.3. Елементи відповіді на команду <host:create>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста.
<host:crDate>	М	Локальні час і дата створення об'єкту хост.

3.20.4. Приклад відповіді на команду <host:create>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<host:creData xmlns:host="http://dotukr.com/epp/host-1.1">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:crDate>2011-05-11T11:31:51+03:00</host:crDate>
</host:creData>
```

```

</resData>
<trID>
<clTRID>USER-4343243243243</clTRID>
<svTRID>UKR-4324343545</svTRID>
</trID>
</response>
</epp>

```

3.20.5. Коди відповідей при виконанні команди <host:create>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2003	"Required parameter missing" "Відсутній обов'язковий параметр" "Отсутствует обязательный параметр"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2302	"Object exists" "Об'єкт вже існує" "Объект уже существует"
2303	"Object does not exist" "Об'єкт не існує"

	"Объект не существует" (Повідомлення надається, якщо реєструється хост в домені, якого не існує)
2306	"Parameter value policy error" "Значення параметра не відповідає правилам" "Значение параметра не соответствует правилам"

3.21. HOST:UPDATE

Команда EPP <host:update> призначена для зміни інформації про об'єкт Host. Змінювати інформацію про хост може лише поточний Реєстратор батьківського домену.

3.21.1. Елементи команди <host:update>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста
<host:add>	О	Елемент, що містить значення атрибутів, які необхідно додати до об'єкта.
<host:rem>	О	Елемент, що містить значення атрибутів, які необхідно видалити з об'єкта.
<host:chg>	О	Елемент, що містить значення атрибутів, які необхідно змінити у об'єкті
<host:addr>	О	Елемент містить IP-адреси, які необхідно присвоїти або видалити з даного хосту, в залежності від розміщення в тілі команди. Вказуються тільки для хостів доменних імен, що обслуговуються реєстром.
<host:status>	О	Елемент містить статуси (clientUpdateProhibited, clientDeleteProhibited), які необхідно присвоїти або видалити з даного хосту, в залежності від розміщення в тілі команди.

Під час виконання команди <host:update>, необхідно задати щонайменше один з елементів <host:add> або <host:rem>.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

```
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
```

```
<command>
```

```

<update>
<host:update xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
<host:add>
<host:status s="clientDeleteProhibited"></host:status>
<host:status s="clientUpdateProhibited"></host:status>
</host:add>
</host:update>
</update>
<clTRID>USER-545454546</clTRID>
</command>
</epp>

```

3.21.2. Приклад відповіді на команду <host:update>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-45545435435345</clTRID>
<svTRID>UKR-454543543534534</svTRID>
</trID>
</response>
</epp>

```

3.21.3. Коди відповідей при виконанні команди <host:update>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"

2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (Повідомлення надається, якщо редагується хост, що належить домену іншого Реєстратора)
2302	"Object exists" "Об'єкт вже існує" "Объект уже существует"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует" (Повідомлення надається, якщо редагується хост в домені, якого не існує)
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции" (Повідомлення надається, якщо реєстром отримано команду редагування хоста зі статусом <i>clientUpdateProhibited</i> або <i>serverUpdateProhibited</i>)

3.22. HOST:DELETE

Команда EPP <host:delete> призначена для видалення об'єкта Host.

Команда <host:delete> доступна для поточного Реєстратора батьківського домену хоста. Видалення хоста буде відхилено у разі, якщо він має статуси: *clientDeleteProhibited* або *serverDeleteProhibited*, або в реєстрі існують домени, які використовують даний хост у якості name-серверів. У разі якщо, хост використовується існуючим доменом, реєстр для такого хоста автоматично присвоює статус "linked" (пов'язаний). Отримати інформацію чи має хост статус "linked" можна за допомогою команди <host:info>

Якщо протягом періоду автоматичного подовження від поточного Реєстратора надійшла команда на видалення домену (<domain:delete>), і домен, який видаляється має хости, що використовуються іншими доменами реєстру - такий домен буде видалено разом з хостами. При цьому Реєстратори, в доменах яких даний хост використовувався як Name-сервера, будуть повідомлені про видалення даного хоста.

3.22.1. Елементи команди<host:delete>

Елемент		Опис
<host:name>	М	Повністю визначене ім'я хоста

3.22.2. Приклад команди <host:delete>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<delete>
<host:delete xmlns:host="urn:ietf:params:xml:ns:host-1.0">
<host:name>xn--1-ytbm.xn--e1aybc.xn--j1amh</host:name>
</host:delete>
</delete>
<clTRID>USER-445454544</clTRID>
</command>
</epp>
```

3.22.3. Приклад відповіді на команду <host:delete>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-345345454543</clTRID>
<svTRID>UKR-3434354354353</svTRID>
</trID>
</response>
</epp>
```


3.22.4. Коди відповідей при виконанні команди <host:delete>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации" (Повідомлення надається, якщо видаляється хост, що належить домену іншого Реєстратора)
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует" (Повідомлення надається, якщо видаляється хост в домені якого не існує)
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции" (Повідомлення надається, якщо реєстром отримано команду на видалення хоста зі статусом <i>clientDeleteProhibited</i> або <i>serverDeleteProhibited</i>)
2305	"Object association prohibits operation" "Зв'язки об'єкта не дозволяють здійснення операції" "Объектные связи не допускают совершение операции" (Повідомлення надається, якщо реєстром отримано команду на видалення хоста зі статусом <i>linked</i> для домену в життєвому періоді "Період дії реєстрації")

3.23. CONTACT:CHECK

Команда EPP <contact:check> призначена для перевірки наявності в реєстрі контакту із заданим ідентифікатором або перевірки можливості реєстрації нового контакту із заданим ідентифікатором. Команда <contact:check> може містити від одного до 10 елементів - ідентифікаторів контакту.

3.23.1. Елементи команди <contact:check>

Елемент		Опис
<contact:id>	М	Ідентифікатор контакту

3.23.2. Приклад команди <contact:check>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<check>
<contact:check xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-id-1</contact:id>
<contact:id>contact-id-2</contact:id>
</contact:check>
</check>
<clTRID>USER-323232323</clTRID>
</command>
</epp>
```

3.23.3. Елементи відповіді на команду <contact:check>

Елемент		Опис
<contact:id avail="0/1">	М	Ідентифікатор контакту з атрибутом "avail", який може приймати значення "0" або "1". "0" означає, що такий ідентифікатор не може бути створений, "1" означає, що ідентифікатор синтаксично коректний і не зареєстрований.
<contact:reason>	О	Надається за avail = "0", описує причину, по якій ідентифікатор не може бути створений.

3.23.4. Приклад відповіді на команду <contact:check>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
```

```

<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<contact:chkData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:cd>
<contact:id avail="0">contact-id-1</contact:id>
<contact:reason>Object exists</contact:reason>
</contact:cd>
<contact:cd>
<contact:id avail="1">contact-id-2</contact:id>
</contact:cd>
</contact:chkData>
</resData>
<trID>
<clTRID>USER-43545454354</clTRID>
<svTRID>UKR-2323232323232</svTRID>
</trID>
</response>
</epp>

```

3.23.5. Коди відповідей при виконанні команди <contact:check>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2002	"Command use error" "Ошибка в использовании команды" "Помилка у використанні команди" (дане повідомлення надається, якщо перевищена максимальна кількість контактів, що перевіряються, в межах однієї команди)

3.24. CONTACT:INFO

Команда EPP <contact:info> призначена для отримання інформації про об'єкт Contact. Отримати дані може будь-який акредитований реєстратор не залежно від того, чи являється він реєстратором об'єкт Contact про, який отримується інформація.

3.24.1. Елементи команди<contact:info>

Елемент		Опис
<contact:id>	M	Ідентифікатор контакту
<contact:authInfo>	O	Необов'язковий елемент, що містить пароль об'єкта. Здається для отримання повного доступу до об'єкта не поточним Реєстратором домену.

3.24.2. Приклад команди <contact:info>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<info>
<contact:info xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-id-1</contact:id>
</contact:info>
</info>
<clTRID>USER-34343434344</clTRID>
</command>
</epp>
```

3.24.3. Елементи відповіді на команду <contact:info>

Елемент		Опис
<contact:id>	M	Унікальний ідентифікатор об'єкта Contact.
<contact:roid>	M	Унікальний внутрішній ідентифікатор об'єкта в реєстрі.
<contact:status>	M	Статус об'єкта (по одному елементу <contact:status> на кожен статус).
<contact:postalInfo>	M	До двох елементів з адресною інформацією. Елемент type = "int" містить текст тільки

		латиницею; type = "loc" - текст на кирилиці в кодуванні UTF-8.
<contact:name>	M	Ім'я або посада людини.
<contact:org>	M	Назва організації
<contact:addr>	M	Адреса, в яку входять під-елементи: <contact:street>, <contact:city>, <contact:sp>, <contact:pc>, <contact:cc>
<contact:street>	M	Адреса, до трьох рядків
<contact:city>	M	Назва міста
<contact:sp>	O	Область, район
<contact:pc>	M	Поштовий індекс
<contact:cc>	M	Двосимвольний код країни
<contact:voice>	M	Номер телефону
<contact:fax>	O	Номер факсу
<contact:email>	M	Адреса Email
<contact:authInfo><pw>	M*	authInfo (пароль до об'єкта).
<contact:clID>	M	Ідентифікатор поточного Реєстратора домену
<contact:crID>	M	Ідентифікатор Реєстратора який створив об'єкт Contact
<contact:crDate>	M	Дата і час створення об'єкта контакт.
<contact:upID>	M	Ідентифікатор Реєстратора, який вніс останні зміни до об'єкту Contact
<contact:upDate>	M	Дата і час внесення останньої зміни до об'єкту Contact
<contact:trDate>	O	Дата і час останнього трансферу об'єкту Contact

* - у авторизованому режимі

3.24.4. Приклад відповіді на команду <contact:info>

```
<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<contact:infData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-1</contact:id>
<contact:roid>UKR-34344343</contact:roid>
<contact:status s="ok"/>
<contact:postalInfo type="int">
<contact:name>Dmitro Andreev</contact:name>
<contact:org>My Corporation</contact:org>
<contact:addr>
<contact:street>Street</contact:street>
<contact:city>Lviv</contact:city>
<contact:pc>02000</contact:pc>
<contact:cc>UA</contact:cc>
</contact:addr>
</contact:postalInfo>
<contact:postalInfo type="loc">
<contact:name>Дмитро Андрєєв</contact:name>
<contact:org>Моя Корпорація</contact:org>
<contact:addr>
<contact:street>Вулиця 1</contact:street>
<contact:city>Львів</contact:city>
<contact:pc>02000</contact:pc>
<contact:cc>UA</contact:cc>
</contact:addr>
</contact:postalInfo>
<contact:voice>+380.320000000</contact:voice>
```

```

<contact:email>email@email.tld</contact:email>
<contact:clID>registrar-1</contact:clID>
<contact:crID>registrar-1</contact:crID>
<contact:crDate>2010-02-03T22:59:31+03:00</contact:crDate>
<contact:authInfo>
<contact:pw>wewqew@!2e</contact:pw>
</contact:authInfo>
<contact:disclose flag="0">
<contact:name type="loc"/>
<contact:org type="loc"/>
<contact:addr type="int"/>
<contact:addr type="loc"/>
<contact:voice/>
<contact:fax/>
<contact:email/>
</contact:disclose>
</contact:infData>
</resData>
<trID>
<clTRID>USER-34324324234</clTRID>
<svTRID>UKR-343432432322</svTRI*D>
</trID>
</response>
</epp>

```

3.24.5. Коди відповідей при виконанні команди <contact:info>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error"

	"Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2202	"Invalid authorization information" "Недійсні дані аутентифікації" "Недействительные данные аутентификации"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"

3.25. CONTACT:CREATE

Команда EPP <contact:create> призначена для створення об'єкта Contact.

Існує чотири типи контактів:

1. Реєстрант (registrant),
2. Адміністратор (admin),
3. Контакт з технічних питань (tech),
4. Контакт з фінансових питань (billing) домену.

3.25.1. Елементи команди<contact:create>

Елемент		Опис
<contact:id>	М	Ідентифікатор контакту.
<contact:postalInfo>	М	До двох елементів з адресною інформацією. Елемент type = "int" містить текст тільки латиницею; type = "loc" - текст на кирилиці в кодуванні UTF-8 (в межах одного контакту може бути задано щонайменше один type, або два різних type).
<contact:name>	М	Ім'я або посада людини.
<contact:org>	О	Назва організації
<contact:addr>	М	Адреса, в яку входять під-елементи: <contact:street>, <contact:city>, <contact:sp>, <contact:pc>, <contact:cc>
<contact:street>	М	Адреса, до трьох рядків

<contact:city>	M	Назва міста
<contact:sp>	O	Область, район
<contact:pc>	M	Поштовий індекс
<contact:cc>	M	ISO- код країни (Alpha-2)
<contact:voice>	M	Номер телефону
<contact:fax>	O	Номер факсу
<contact:email>	M	Адреса Email
<contact:authInfo><pw>	M	authInfo (пароль до об'єкта).
<contact:disclose>	O	Набір прапорів для не відображення (flag = "0") або розкриття (flag = "1") персональної інформації в WHOIS. Має містити щонайменше один з елементів, режим відображення якого необхідно змінити: • <contact:name type="int"/> • <contact:name type="loc"/> • <contact:org type="int"/> • <contact:org type="loc"/> • <contact:addr type="int"/> • <contact:addr type="loc"/> • <contact:voice/> • <contact:fax/> • <contact:email/>

3.25.2. Приклад команди <contact:create>

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<create>
<contact:create xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-id-1</contact:id>
<contact:postalInfo type="int">
<contact:name>Dmitro Andreev</contact:name>
<contact:org>My Corporation</contact:org>

```

```

<contact:addr>
<contact:street>Street, 2</contact:street>
<contact:city>Lviv</contact:city>
<contact:pc>02000</contact:pc>
<contact:cc>UA</contact:cc>
</contact:addr>
</contact:postalInfo>
<contact:postalInfo type="loc">
<contact:name>Дмитро Андрєєв</contact:name>
<contact:org>Моя Корпорація</contact:org>
<contact:addr>
<contact:street>Вулиця, 2</contact:street>
<contact:city>Львів</contact:city>
<contact:pc>02000</contact:pc>
<contact:cc>UA</contact:cc>
</contact:addr>
</contact:postalInfo>
<contact:voice>+380.320000000</contact:voice>
<contact:email>email@email.tld</contact:email>
<contact:authInfo>
<contact:pw>fddf3rewe</contact:pw>
</contact:authInfo>
<contact:disclose flag="0">
<contact:name type="loc"/>
<contact:org type="loc"/>
<contact:addr type="loc"/>
<contact:voice/>
<contact:email/>
</contact:disclose>
</contact:create>
</create>
<clTRID>USER-45454534534</clTRID>
</command>
</epp>

```

3.25.3. Елементи відповіді на команду <contact:create>

Елемент		Опис
<contact:id>	M	Унікальний ідентифікатор об'єкта Contact.
<contact:crDate>	M	Локальні дата і час створення об'єкта Contact.

3.25.4. Приклад відповіді на команду <contact:create>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<resData>
<contact:creData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>nnn</contact:id>
<contact:crDate>2011-07-02T11:11:11+03:00</contact:crDate>
</contact:creData>
</resData>
<trID>
<clTRID>USER-34324234234</clTRID>
<svTRID>UKR-5656546546</svTRID>
</trID>
</response>
</epp>

```

3.25.5. Коди відповідей при виконанні команди <contact:create>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"

2003	"Required parameter missing" "Відсутній обов'язковий параметр" "Отсутствует обязательный параметр"
2004	"Parameter value range error" "Помилка в значенні параметра" "Ошибка в значении параметра"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2302	"Object exists" "Об'єкт вже існує" "Объект уже существует"
2306	"Parameter value policy error" "Значення параметра не відповідає правилам" "Значение параметра не соответствует правилам"

3.26. CONTACT:UPDATE

Команда EPP <contact:update> призначена для зміни інформації про об'єкт Contact. Команда доступна тільки для поточного Реєстратора контакту.

3.26.1. Елементи команди<contact:update>

Елемент		Опис
<contact:id>	М	Ідентифікатор контакту.
<contact:add>	О	Елемент, що містить значення атрибутів, які необхідно додати до об'єкта Contact. Може містити один або декілька статусів: • clientUpdateProhibited • clientDeleteProhibited • clientTransferProhibited
<contact:rem>	О	Елемент, що містить значення атрибутів, які потрібно видалити з об'єкта Contact. Може містити один або декілька статусів: • clientUpdateProhibited • clientDeleteProhibited • clientTransferProhibited

<contact:chg>	О	Елемент, що містить значення атрибутів, які потрібно змінити. Може містити наступні елементи: • <contact:postalInfo> - елементи з адресною інформацією. Елемент type = "int" містить текст тільки латиницею; type = "loc" - текст на кирилиці в кодуванні UTF-8; • <contact:name> - ім'я або посада людини. • <contact:org> - назва організації. • <contact:addr> - адреса, яка складається з 3-х рядків; ○ <contact:street> - назва вулиці, дім ○ <contact:city> - місто ○ <contact:sp> - район, область ○ <contact:pc> - поштовий індекс ○ <contact:cc> - ISO- код країни (Alpha-2) • <contact:voice> - номер телефону • <contact:fax> - номер факсу • <contact:email> - адреса email • <contact:authInfo><pw> - authInfo (пароль до об'єкта). • <contact:disclose> - набір прапорів для приховування (flag = "0") або розкриття (flag = "1") персональної інформації:
---------------	---	---

При виконанні команди <contact:update> має бути присутній щонайменше один з елементів: <contact:add>, <contact:rem> або <contact:chg>.

3.26.2. Приклад команди <contact:update>

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<command>
<update>
<contact:update xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-1</contact:id>
<contact:add>
<contact:status s="clientDeleteProhibited"></contact:status>
</contact:add>
<contact:chg>
<contact:postalInfo type="loc">

```

```

<contact:name>Ім'я Прізвище</contact:name>
<contact:org>ТОВ Компанія</contact:org>
<contact:addr>
<contact:street>Вулиця, 1</contact:street>
<contact:city>Львів</contact:city>
<contact:cc>UA</contact:cc>
</contact:addr>
</contact:postalInfo>
<contact:voice>+380.320000000</contact:voice>
<contact:authInfo>
<contact:pw>EdsER23</contact:pw>
</contact:authInfo>
<contact:disclose flag="0">
<contact:name type="loc"/>
<contact:voice/>
</contact:disclose>
</contact:chg>
</contact:update>
</update>
<clTRID>USER-4545454578</clTRID>
</command>
</epp>

```

3.26.3. Приклад відповіді на команду <contact:update>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-2343242342</clTRID>
<svTRID>UKR-3434243423</svTRID>
</trID>
</response>
</epp>

```

3.26.4. Коди відповідей при виконанні команди <contact:update>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"
2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"

3.27. CONTACT:DELETE

Команда EPP <contact:delete> призначена для видалення інформації про об'єкт Contact. Команда доступна тільки для поточного Реєстратора контакту. Контакт не може бути видалений, якщо він використовується об'єктом Domain або має будь-який зі статусів: clientDeleteProhibited, serverDeleteProhibited.

3.27.1. Елементи команди<contact:delete>

Елемент		Опис
<contact:id>	М	Ідентифікатор контакту.

3.27.2. Приклад команди <contact:delete>

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
```

```

<command>
<delete>
<contact:delete xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
<contact:id>contact-id1</contact:id>
</contact:delete>
</delete>
<clTRID>USER-34343434343</clTRID>
</command>
</epp>

```

3.27.3. Приклад відповіді на команду <contact:delete>

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns='urn:ietf:params:xml:ns:epp-1.0'>
<response>
<result code="1000">
<msg lang="en">Command completed successfully</msg>
</result>
<trID>
<clTRID>USER-14545444</clTRID>
<svTRID>UKR-435454545435</svTRID>
</trID>
</response>
</epp>

```

3.27.4. Коди відповідей при виконанні команди <contact:delete>

Код відповіді	Опис
1000	"Command completed successfully" "Команду виконано успішно" "Команда выполнена успешно"
2001	"Command syntax error" "Синтаксична помилка команди" "Синтаксическая ошибка команды"
2005	"Parameter value syntax error" "Синтаксична помилка параметра" "Синтаксическая ошибка параметра"

2201	"Authorization error" "Помилка авторизації" "Ошибка авторизации"
2303	"Object does not exist" "Об'єкт не існує" "Объект не существует"
2304	"Object status prohibits operation" "Статус об'єкта не дозволяє здійснення операції" "Статус объекта не позволяет совершение операции"
2305	"Object association prohibits operation" "Зв'язки об'єкта не дозволяють здійснення операції" "Объектные связи не допускают совершение операции"