

Opis protokołu RPC

Grzegorz Maj
nr indeksu: 236095

1 Streszczenie

Niniejszy dokument opisuje specyfikację protokołu RQP (Remote Queues Protocol). W jego skład wchodzi:

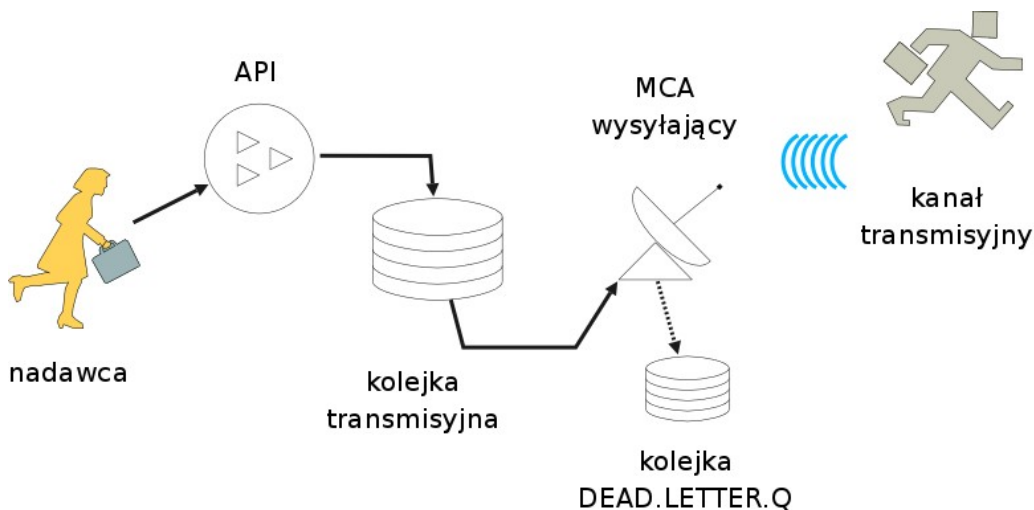
- opis celów protokołu;
- opis założeń protokołu;
- opis formatu komunikatów;
- opis wymienianych komunikatów;
- opis stanów.

2 Cele

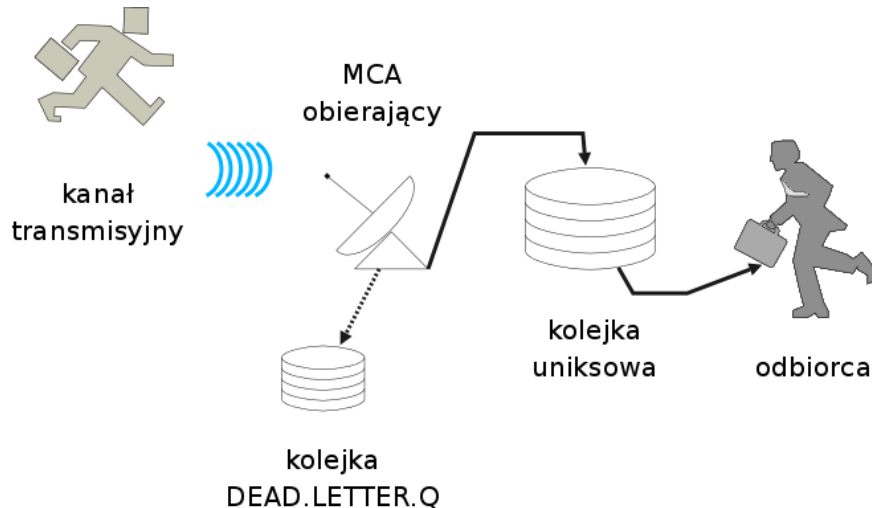
2.1 Protokół RQP służy do przesyłania informacji między kolejkami na zdalnych maszynach. Rozszerza on możliwość korzystania ze zwykłych uniksowych kolejek komunikatów w ten sposób, iż dane zapisywane są przez sieć do kolejek na systemach zdalnych.

2.2 Schemat działania

2.2.1 Strona nadawcza



2.2.2 Strona odbiorcza



3 Definicje

- agent kanałowy – (MCA) proces implementujący połączenie (patrz: nadawca, odbiorca);
- kanał transmisyjny – połączenie, implementowane w postaci dwóch agentów kanałowych;
- kolejka transmisyjna – bufor po stronie nadawcy, implementowany przez agenta wysyłającego;
- kolejka zdalna - kolejka uniksowa IPC pracująca na maszynie odbiorcy;
- komunikat - porcja danych przesyłana od nadawcy do odbiorcy;
- komunikat niepewny – komunikat, który nie wymaga potwierdzenia (patrz: komunikat pewny);
- komunikat pewny - komunikat z danymi, który nie powinien zostać usuwany z kolejki transmisyjnej do chwili, gdy nie mamy pewności, że trafił bezpiecznie do kolejki docelowej lub został odrzucony;
- nadawca - użytkownik instancji protokołu, który chce przesyłać informacje;
- odbiorca - użytkownik wstawiający informacje do docelowej kolejki uniksowej;
- połączenie – połączenie między stroną nadającą (wysyłającą dane) a stroną odbierającą;
- protokół - protokół RQP (Remote Queues Protocol) opisywany w niniejszym dokumencie;
- zestawianie - proces budowania połączenia;

4 Założenia

- Protokół RQP działa w warstwie aplikacji. Do transportu wykorzystywany jest protokół TCP/IP, w ramach którego port o numerze 6543 jest używany do nasłuchiwania po stronie MCA odbierających.

Połączenie fizyczne istnieje pomiędzy każdym MCA nadającym a odbierającym, o ile odbywa się pomiędzy nimi wymiana komunikatów. Jest nawiązywane podczas pierwszego odpytywania konkretnego MCA odbierającego o posiadanie kolejek przez MCA nadającego. Jest utrzymywane tak długo, aż któraś ze stron zakończy swoje działanie, bądź wszystkie kolejki po stronie odbierającej, związane z MCA nadającym, zostaną usunięte.

- Celem wykrycia ewentualnego zakończenia pracy przez MCA odbierające, a przez to braku odpowiedzi na komunikaty, wprowadza się stałą TIMEOUT określającą, jak długo czekamy na odpowiedź po wysłaniu komunikatu.

- Korzystanie z protokołu odbywa się poprzez interfejs programistyczny (API). Jest on zaimplementowany w postaci biblioteki dzielonej. Udostępnia następujące funkcje:

```
int openConnection(key_t key);
int sendMessage(int qdesc, key_t rkey, const void *buf, size_t count,
                 int msgtype, const void *options)
```

Funkcja `openConnection` przekształca klucz na deskryptor. Otwarcie połączenia uruchamia agenta wysyłającego związanego z tym kluczem. Użycie funkcji `openConnection` z tą samą wartością klucza w innym procesie (nadawcy) pozwala na skorzystanie z istniejącego agenta. Użycie funkcji `openConnection` z inną wartością klucza w przypadku, gdy istnieje już jeden agent wysyłający kończy się błędem i zwraca wartość `-1`.

Funkcja `sendMessage` przyjmuje jako parametr deskryptor połączenia, klucz kolejki docelowej, wiadomość do wysłania, jej rozmiar, typ („pewna” czy „niepewna” – patrz: definicje) oraz ewentualnie dodatkowe opcje.

5 Format komunikatów

5.1 Liczby używane w komunikatach są przesyłane w porządku sieciowym. Uzupełnienia formatu zapisu liczb są niedopuszczalne.

5.2 Typy komunikatów:

```
SZUKANIE_KOLEJKI_MSG {
    uint8          typ_komunikatu;
    uint64         klucz_kolejki;
}
```

```
ODPOWIEDZ_MSG {
    uint8          typ_komunikatu;
    uint64         odpowiedz;
}
```

```
WIADOMOSC_MSG {
    uint8          typ_komunikatu;
    uint64         dlugosc_komunikatu;
    uint8          typ_wiadomosci;
    uint64         klucz_kolejki;
    uint64         klucz_wiadomosci;
    octet[dlugosc] wiadomosc;
} gdzie dlugosc = dlugosc_komunikatu - 26.
```

5.3 Opisy długości:

dlugosc_komunikatu - pole zawierające długość całego pakietu, długość ta obejmuje pola: typ_komunikatu, dlugosc_komunikatu, typ_wiadomosci, klucz_kolejki, klucz_wiadomosci, wiadomosc.

6 Opis wymienianych komunikatów

SZUKANIE_KOLEJKI_MSG jest wysyłany przez MCA nadającego w celu wyszukania kolejki o podanym kluczu.

Pola:

- typ_komunikatu - ustawione na wartość SZUKANIE_KOLEJKI_TYP;
- klucz_kolejki - zawiera klucz poszukiwanej kolejki.

ODPOWIEDZ_MAM_KOLEJKE_MSG jest typu ODPOWIEDZ_MSG. Jest ona wysyłana przez MCA odbierającego w celu poinformowania żądającego o posiadaniu kolejki o danym kluczu.

Pola:

- typ_komunikatu - ustawione na wartość ODPOWIEDZ_MAM_KOLEJKE_TYP;
- odpowiedz - zawiera klucz kolejki, o której posiadaniu informujemy.

WIADOMOSC_MSG jest wysyłana przez MCA nadającego. Zawiera on właściwą wiadomość, która ma zostać wstawiona do kolejki zdalnej.

Pola:

- typ_komunikatu – ustawione na wartość WIADOMOSC_TYP;
- dlugosc_komunikatu – patrz pkt. 5.3;
- typ_wiadomosci – ustawione na wartość PEWNA_TYP, lub NIEPEWNA_TYP, w zależności od typu wysyłanej wiadomości (patrz: definicje);
- klucz_kolejki – klucz kolejki zdalnej, do której ma trafić komunikat;
- klucz_wiadomosci – identyfikator wiadomości unikalny w obrębie danego MCA nadającego;
- wiadomosc – treść wiadomości, która ma zostać wstawiona do kolejki zdalnej.

ODPOWIEDZ_DOSTALEM_PEWNA_MSG jest typu ODPOWIEDZ_MSG. Jest ona wysyłana przez MCA odbierającego w celu poinformowania żądającego o otrzymaniu i umieszczeniu w kolejce docelowej wiadomości „pewnej”.

Pola:

- typ_komunikatu – ustawione na wartość ODPOWIEDZ_DOSTALEM_PEWNA_TYP;
- odpowiedz – unikalny identyfikator wiadomości, ten sam, który był zapisany w polu klucz_wiadomosci podczas jej przesyłania.

ODPOWIEDZ_NIE_MAM_KOLEJKI_MSG jest typu ODPOWIEDZ_MSG. Jest ona wysyłana przez MCA odbierającego w celu poinformowania powiązanego MCA nadającego, że kolejka docelowa przestała istnieć, bądź zostały utracone prawa do zapisu do niej.

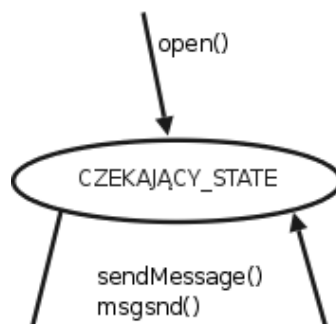
Pola:

- typ_komunikatu – ustawione na wartość ODPOWIEDZ_NIE_MAM_KOLEJKI_TYP;
- odpowiedz – numer kolejki, o której braku informujemy.

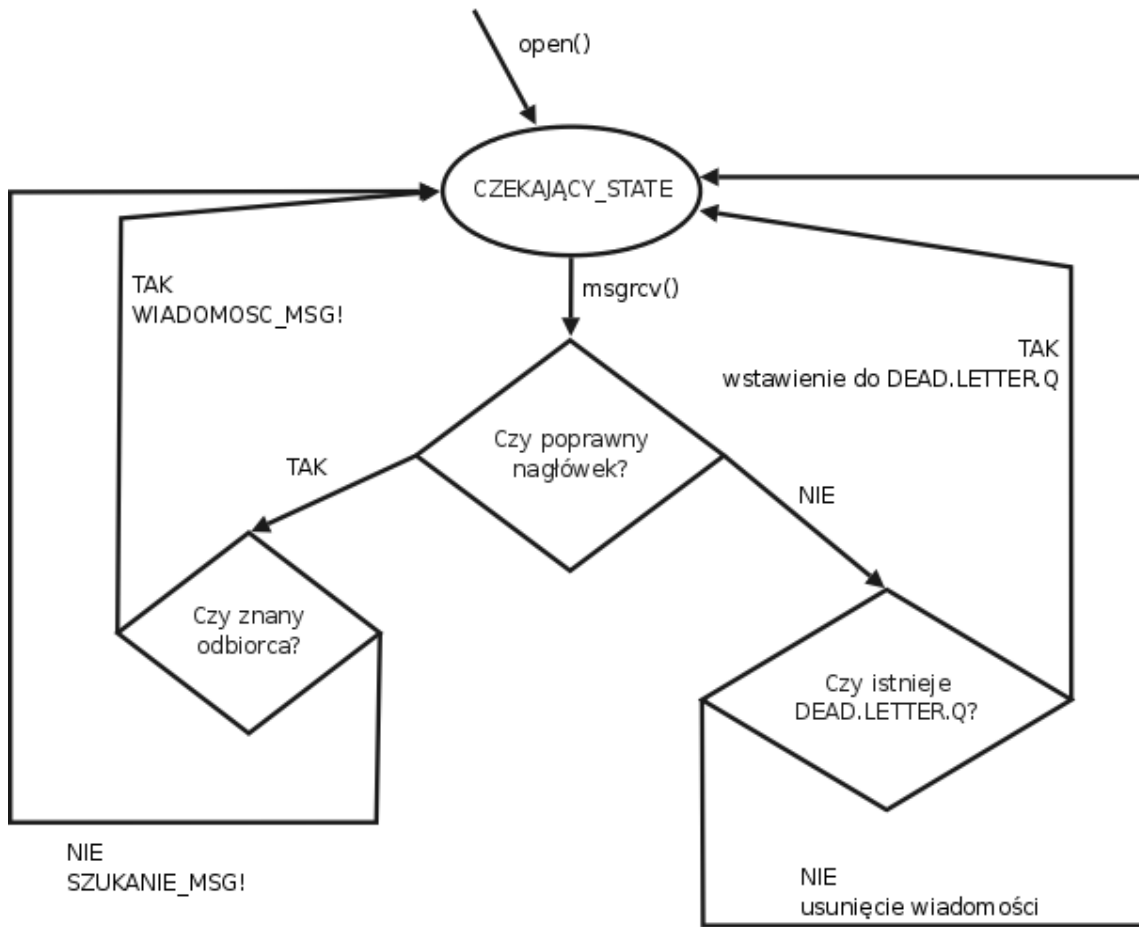
7 Opis stanów

7.1 Strona nadająca:

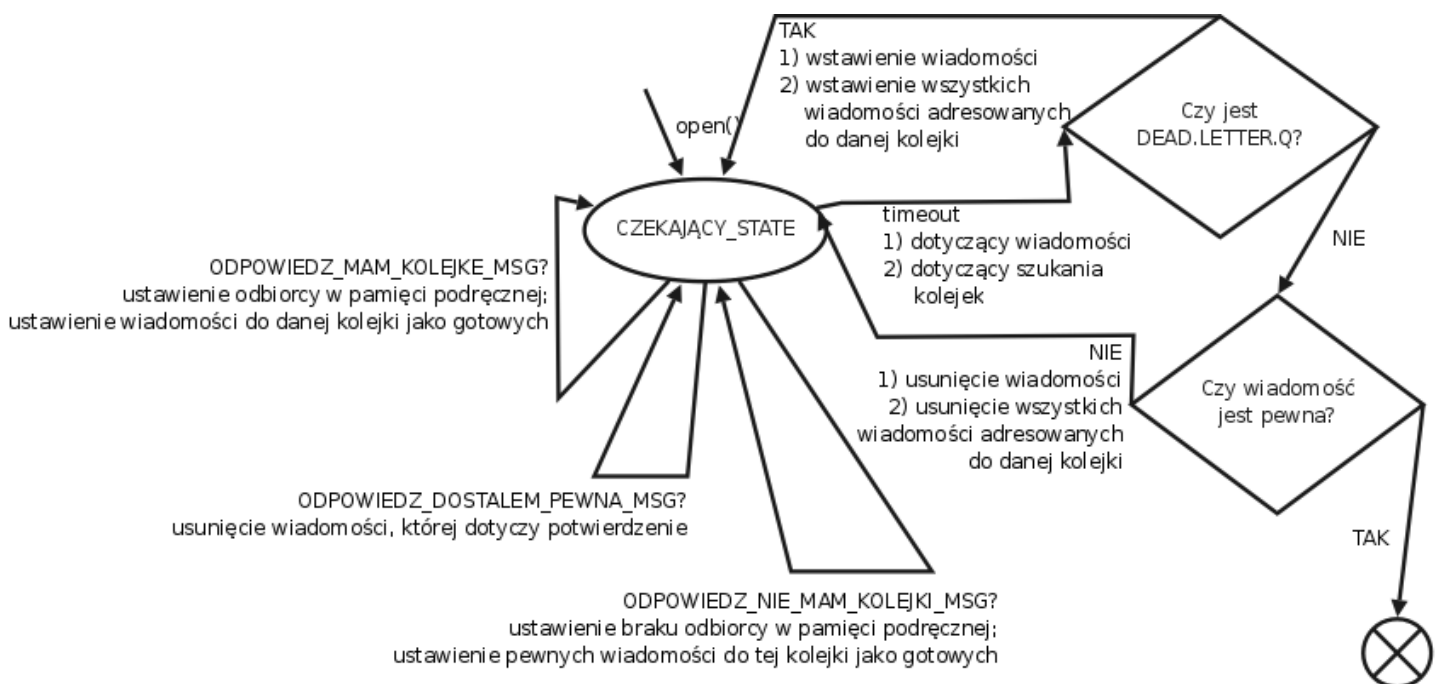
7.1.1 API



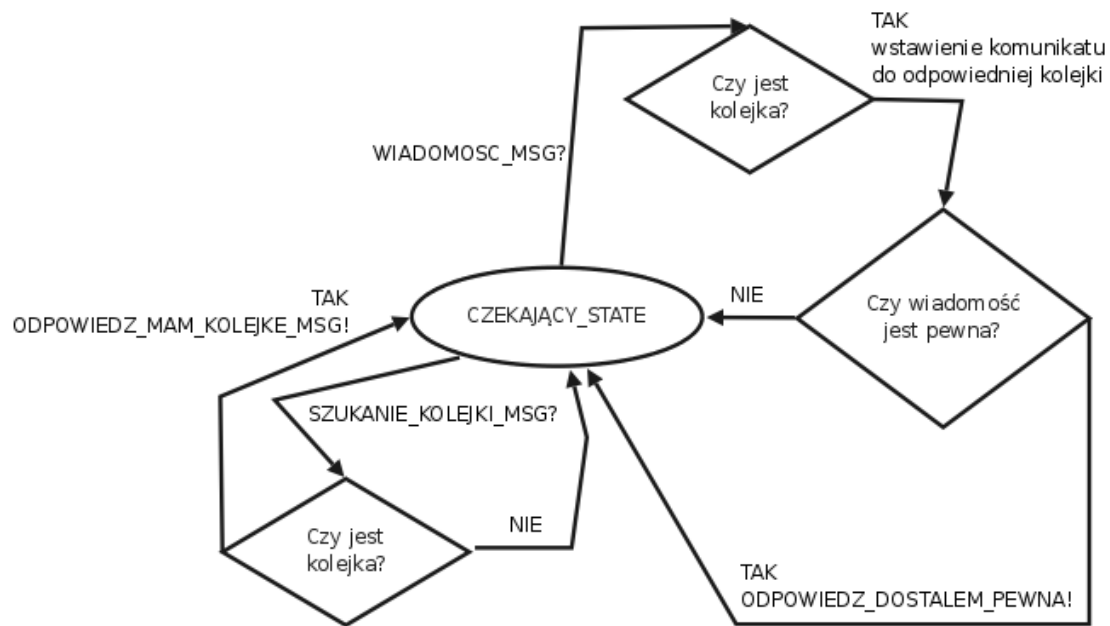
7.1.2 MCA wysyłający, wątek nadający



7.1.3 MCA wysyłający, wątek odbierający potwierdzenia



7.2 Strona odbierająca:



8 Podsumowanie używanych numerów

- NIEPEWNA_TYP = 0;
- ODPOWIEDZ_DOSTALEM_PEWNA_TYP = 3;
- ODPOWIEDZ_MAM_KOLEJKE_TYP = 2;
- ODPOWIEDZ_NIE_MAM_KOLEJKI_TYP = 4;
- PEWNA_TYP = 1;
- SZUKANIE_KOLEJKI_TYP = 1;
- WIADOMOSC_TYP = 5;
- TIMEOUT = 30sec.