



[오델로 프로토콜 명세서]

Best of Best 7 기

보안제품개발트랙

임대동

2019-01-15

목차

1. 개요	3
1.1 적용 범위	3
1.2 Protocol 기본 규약	3
2. PROTOCOL HEADER	4
2.1 HEADER 패킷 필드 구조	4
2.2 VERSION	4
2.3 ROLE	4
2.4 MODE	5
2.5 COMMAND	5
2.6 ROOM_NUM	7
2.7 TIMER	7
2.8 BODY	8
3. REQUEST (Client → Server) BODY	9
3.1 REQUEST BODY 패킷 필드 구조	9
3.2 TURN_NUM	9
3.3 COLOR	9
3.4 POSITION_X	10
3.5 POSITION_Y	10
3.6 DELAY	10
3.7 BLACK_SCORE	10
3.8 WHITE_SCORE	11
4. RESPONSE (Server → Client) BODY	12
4.1 RESPONSE BODY 패킷 필드 구조	12
4.2 BOARD_LINE_N	12

1. 개요

본 문서는 “OTHELLO” 혹은 “REVERSI”라 불리는 네트워크 게임(이하 “오델로”라 한다.) 애플리케이션을 제어하는데 필요한 프로토콜을 정의한다.

1.1 적용 범위

- 오델로 Server 프로그램
- 오델로 Client 프로그램

1.2 Protocol 기본 규약

- 본 프로토콜은 아래와 같은 오델로 기본 규칙만을 지원한다.
 - 두 명이 가로 세로 8칸의 오델로 판 위에서 한쪽은 검은색, 다른 한쪽은 흰색인 돌을 번갈아 놓으며 진행된다.
 - 처음에 판 가운데에 사각형으로 엇갈리게 배치된 돌([백돌, 흑돌], [흑돌, 백돌]) 4개를 놓고 시작한다.
 - 돌은 반드시 상대방 돌을 양쪽에서 포위하여 뒤집을 수 있는 곳에 놓아야 한다.
 - 돌을 뒤집을 곳이 없는 경우에는 차례가 자동적으로 상대방에게 넘어가게 된다.
 - 아래와 같은 조건에 의해 양쪽 모두 더 이상 돌을 놓을 수 없게 되면 게임이 끝나게 된다.
 1. 64개의 돌 모두가 판에 가득 찬 경우 (가장 일반적)
 2. 어느 한 쪽이 돌을 모두 뒤집은 경우
 3. 한 차례에 양 쪽 모두 서로 차례를 넘겨야 하는 경우
 - 게임이 끝났을 때 돌이 많이 있는 플레이어가 승자가 된다. 만일 돌의 개수가 같을 경우는 무승부가 된다.
- 본 프로토콜은 Server와 Client 프로그램간 TCP/IP 네트워크 인터페이스를 기반으로 데이터 송/수신되는 Binary-based 프로토콜이다.
- 본 프로토콜은 Client가 Server에게 전달하는 REQUEST와 REQUEST에 대한 응답으로 Server가 Client에게 전달하는 RESPONSE로 나뉜다.

2. PROTOCOL HEADER

프로토콜의 헤더는 8 byte 고정 길이로 VERSION(1 byte), ROLE(1 byte), MODE(1 byte), COMMAND(1 byte), ROOM_NUM(2 byte), TIMER(1 byte), BODY(1 byte)로 이루어져 있다. 프로토콜의 헤더는 REQUEST와 RESPONSE에서 동일하게 사용된다.

2.1 HEADER 패킷 필드 구조

VERSION (1 byte)	ROLE (1 byte)	MODE (1 byte)	COMMAND (1 byte)	ROOM_NUM (2 byte)	TIMER (1 byte)	BODY (1 byte)
---------------------	------------------	------------------	---------------------	----------------------	-------------------	------------------

2.2 VERSION

VERSION(1 byte)은 프로토콜의 버전을 나타내는 필드로 <Major Number>.<Minor Number>로 표현된다. 1~4번째 비트는 Major Number로 사용되고, 5~8번째 비트는 Minor Number로 사용된다. 기본값은 프로토콜 버전 1.0인 0x10(0b00010000)이다.

VERSION (1 byte)	설명
0x10	프로토콜 버전 1.0

2.3 ROLE

ROLE(1 byte)은 해당 헤더가 SERVER가 보낸 헤더인지, 혹은 CLIENT가 보낸 헤더인지 알려주는 필드이다. 해당 필드의 값이 1이면 CLIENT를 나타내고, 2이면 SERVER를 나타낸다.

1. CLIENT
 - 헤더를 보낸 주체가 CLIENT
2. SERVER
 - 헤더를 보낸 주체가 SERVER

ROLE (1 byte)	설명
0x01	CLIENT
0x02	SERVER

2.4 MODE

MODE(1 byte)는 모델로의 게임 모드를 나타내는 필드이다. 본 프로토콜은 4가지 게임 모드를 지원한다.

1. AI_AI
 - AI vs AI 게임을 지원하는 모드이다.
2. AI_HUMAN
 - AI vs HUMAN 게임을 지원하는 모드이다. [플레이 주체가 AI]
3. HUMAN_AI
 - HUMAN vs AI 게임을 지원하는 모드이다. [플레이 주체가 HUMAN]
4. HUMAN_HUMAN
 - HUMAN vs HUMAN 게임을 지원하는 모드이다.

MODE (1 byte)	설명
0x01	AI_AI
0x02	AI_HUMAN
0x03	HUMAN_AI
0x04	HUMAN_HUMAN

2.5 COMMAND

COMMAND(1 byte)는 Server와 Client의 명령을 나타내는 필드이다. COMMAND는 Client가 Server에게 보내는 REQUEST COMMAND와 REQUEST에 응답해 Server가 Client에게 보내는 RESPONSE COMMAND로 나뉜다.

2.5.1 REQUEST COMMAND

1. ENTER_ROOM
 - Client가 ROOM_NUM 필드 값의 방에 들어가는 것을 요청
2. REQUEST_PENDING
 - 게임 중 게임 일시 정지를 요청
3. REQUEST_SURRENDER
 - 게임 중 항복을 요청
4. PUT_STONE

- 게임 중 돌을 놓는 것을 요청

5. LEAVE_ROOM

- Client가 방을 나가는 것을 요청

COMMAND (1 byte)	설명	BODY 유무
0x01	ENTER_ROOM	X
0x02	REQUEST_PENDING	X
0x03	REQUEST_SURRENDER	X
0x04	PUT_STONE	O
0x05	LEAVE_ROOM	X

2.5.2 RESPONSE COMMAND

1. START

- 게임이 시작됨을 응답

2. WAITING_PLAYER

- 방에 플레이어가 부족해 플레이어를 기다린다는 정보를 응답

3. FULL_ROOM

- 방에 수용 인원이 가득차서 들어갈 수 없음을 응답

4. PLAYING

- 게임 중 게임 정보를 응답

5. PENDING

- 게임 일시정지 상태임을 응답

6. PLAYER_BLACK_WIN

- 검은색 플레이어가 이겼다는 정보를 응답

7. PLAYER_WHITE_WIN

- 흰색 플레이어가 이겼다는 정보를 응답

8. DRAW

- 동점으로 비겼다는 정보를 응답

9. ERROR

- 잘못된 요청으로 오류가 발생했음을 응답

10. LEAVE

- 방을 정상적으로 나갔다는 것을 응답

COMMAND (1 byte)	설명	BODY 유무
0x10	START	X
0x20	WAITING_PLAYER	X
0x30	FULL_ROOM	X
0x40	PLAYING	0
0x50	PENDING	X
0x60	PLAYER_BLACK_WIN	X
0x70	PLAYER_WHITE_WIN	X
0x80	DRAW	X
0x90	ERROR	X
0xA0	LEAVE	X

2.6 ROOM_NUM

ROOM_NUM(2 byte)은 방 번호를 나타내는 영역으로 0번 방은 대기실로 사용되고, 1번부터 65,535번 방은 게임 플레이를 위한 방으로 사용되어 진다. 0번 방을 제외한 각 방의 플레이어 수용 인원은 2인으로 제한된다.

ROOM_NUM (2 byte)	설명
0x0000 (0)	0번 방, 대기실
0x0001 ~ 0xFFFF (1 ~ 65,535)	1 ~ 65535번 방

2.7 TIMER

TIMER(1 byte)는 돌을 놓는데 걸리는 제한 시간 옵션 필드로 제한 시간내에 돌을 놓지 않으면 애플리케이션에서 랜덤으로 돌을 놓도록 설계한다.

1. NONE_TIMER

- 시간 제한 없음 (무제한)

2. TIMER_15SEC
 - 시간 제한 15초
3. TIMER_30SEC
 - 시간 제한 30초
4. TIMER_60SEC
 - 시간 제한 60초

TIMER (1 byte)	설명
0x00 (0)	NONE_TIMER
0x0F (15)	TIMER_15SEC
0x1E (30)	TIMER_30SEC
0x3C (60)	TIMER_60SEC

2.8 BODY

BODY(1 byte)는 프로토콜 BODY의 존재 유무를 나타내는 필드이다. 해당 필드의 값이 0이면 프로토콜 BODY가 존재하지 않고, 1이면 프로토콜 BODY가 존재한다.

1. NONE_BODY
 - BODY가 존재하지 않음
2. EXIST_BODY
 - BODY가 존재함

BODY (1 byte)	설명
0x00	NONE_BODY
0x01	EXIST_BODY

3. REQUEST (Client → Server) BODY

REQUEST BODY는 REQUEST 헤더의 COMMAND 값이 PUT_STONE(0x04)일 때 헤더의 BODY 필드의 값이 EXIST_BODY(0x01)로 세팅이 되어 BODY가 존재한다. REQUEST BODY는 8 byte 고정 길이로 TURN_NUM(1 byte), COLOR(1 byte), POSITION_X(1 byte), POSITION_Y(1 byte), DELAY(2 byte), BLACK_SCORE(1 byte), WHITE_SCORE(1 byte)로 이루어져 있다.

3.1 REQUEST BODY 패킷 필드 구조

VERSION (1 byte)	ROLE (1 byte)	MODE (1 byte)	COMMAND (1 byte)	ROOM_NUM (2 byte)	TIMER (1 byte)	BODY (1 byte)
TURN_NUM (1 byte)	COLOR (1 byte)	POSITION_X (1 byte)	POSITION_Y (1 byte)	DELAY (2 byte)	BLACK_SCORE (1 byte)	WHITE_SCORE (1 byte)

3.2 TURN_NUM

TURN_NUM(1 byte)은 게임 진행 중 몇 턴인지 알려주는 필드이다. 오델로 최대 턴은 60턴이다.

TURN_NUM (1 byte)	설명
0x01 ~ 0x3C (1~60)	턴 숫자, 게임 최대 턴: 60

3.3 COLOR

COLOR(1 byte)는 게임을 플레이 중인 플레이어가 어떤 색의 돌을 플레이 중인지 나타내는 필드이다. 필드의 값이 1이면 검은색 플레이어, 2이면 흰색 플레이어, 0xFF이면 관전자를 나타낸다.

COLOR (1 byte)	설명
0x01	BLACK_PLAYER
0x02	WHITE_PLAYER
0xFF	OBSERVER

3.4 POSITION_X

POSITION_X(1 byte)는 돌을 놓는 가로 X축 좌표를 나타내는 필드이다. 오델로는 8x8 판이므로 0부터 7까지의 값을 가질 수 있다.

POSITION_X (1 byte)	설명
0x00 ~ 0x07 (0~7)	가로 X 축 좌표 값

3.5 POSITION_Y

POSITION_Y(1 byte)는 돌을 놓는 세로 Y축 좌표를 나타내는 필드이다. 오델로는 8x8 판이므로 0부터 7까지의 값을 가질 수 있다.

POSITION_Y (1 byte)	설명
0x00 ~ 0x07 (0~7)	세로 Y 축 좌표 값

3.6 DELAY

DELAY(2 byte)는 일반적으로 AI vs AI 게임 모드(AI_AI)에서 사용되는 필드로 AI간의 너무 빠른 게임 진행에 의해 게임을 제대로 관전할 수 없을 때 돌이 놓이는 딜레이 시간을 설정할 수 있다. DELAY 값은 MS(millisecond) 기준으로 설정할 수 있다. DELAY 값이 0이면 딜레이가 없고, 1ms부터 65,535ms까지 설정할 수 있다.

DELAY (2 byte)	설명
0x0000	NONE_DELAY
0x0064	DELAY_100MS
0x03E8	DELAY_1000MS
0x0BB8	DELAY_3000MS

3.7 BLACK_SCORE

BLACK_SCORE(1 byte)는 검은색 플레이어의 점수(검은색 돌 개수)를 나타내는 필드이다. Client 는 점수를 계산해 서버에 보내고, 서버는 그것이 올바른 과정에서 나온 점수인지 검증한다.

BLACK_SCORE (1 byte)	설명
0x00 ~ 0x64 (0 ~ 64)	검은색 플레이어의 점수

3.8 WHITE_SCORE

WHITE_SCORE(1 byte)는 흰색 플레이어의 점수(흰색 돌 개수)를 나타내는 필드이다. Client 는 점수를 계산해 서버에 보내고, 서버는 그것이 올바른 과정에서 나온 점수인지 검증한다.

WHITE_SCORE (1 byte)	설명
0x00 ~ 0x64 (0 ~ 64)	흰색 플레이어의 점수

4. RESPONSE (Server → Client) BODY

RESPONSE BODY는 RESPONSE 헤더의 COMMAND 값이 PLAYING(0x40)일 때 헤더의 BODY 필드의 값이 EXIST_BODY(0x01)로 세팅이 되어 BODY가 존재한다. 해당 요청에 대한 응답이라는 뜻으로 Client가 보내온 8 byte의 REQUEST BODY를 헤더 뒤에 붙이고, 그 뒤에 BOARD_LINE_1부터 BOARD_LINE_8까지 8줄의 게임 판의 상태를 함께 보낸다.

4.1 RESPONSE BODY 패킷 필드 구조

VERSION (1 byte)	ROLE (1 byte)	MODE (1 byte)	COMMAND (1 byte)	ROOM_NUM (2 byte)	TIMER (1 byte)	BODY (1 byte)
TURN_NUM (1 byte)	COLOR (1 byte)	POSITON_X (1 byte)	POSITION_Y (1 byte)	DELAY (2 byte)	BLACK_SCORE (1 byte)	WHITE_SCORE (1 byte)
BOARD_LINE_1		BOARD_LINE_2		BOARD_LINE_3	BOARD_LINE_4	
BOARD_LINE_5		BOARD_LINE_6		BOARD_LINE_7	BOARD_LINE_8	

4.2 BOARD_LINE_N

BOARD_LINE_N(2 byte)은 모델로 8x8 판에서 각 가로줄에 해당하는 돌의 상태를 나타내는 필드이다. 하나의 칸은 2개의 bit로 표현되어 각 라인당 8개의 칸의 상태를 표현한다. 2개의 bit가 0이면 돌이 없는 상태, 1이면 검은색 돌이 놓인 상태, 2이면 흰색 돌이 놓인 상태, 3이면 돌을 놓을 수 있는 상태이다.

BOARD_LINE_N (2 byte)	설명
0b00 (0)	NONE_STONE
0b01 (1)	BLACK_STONE
0b02 (2)	WHITE_STONE

0b03 (3)	POSSIBLE_SPACE
----------	----------------