

Scoreboard API Module Specification

Overview

The Scoreboard API Module is responsible for managing and updating user scores in real-time. It provides secure endpoints to receive score updates and ensures that only authorized actions can increment user scores. The module also supports broadcasting live updates to all clients to display the top 10 user scores.

Objectives

- Receive and process score updates from user actions.
- Ensure only authorized updates are processed to prevent score manipulation.
- Maintain a real-time scoreboard reflecting the top 10 user scores.
- Provide a public API to fetch the current scoreboard.
- Broadcast updates to connected clients in real-time.

API Endpoints

1. Update User Score

Endpoint: POST /api/score/update

Description: Accepts a user action and updates the corresponding user's score.

Request:

```
{  
  
  "userId": "string",  
  
  "action": "string",  
  
  "signature": "string"  
}
```

- userId: Unique identifier for the user.
- action: Action performed by the user.
- signature: HMAC-based signature to verify authenticity.

Response:

```
{  
  
  "success": true,  
  
  "message": "Score updated successfully"
```

```
}
```

Security Considerations:

- Use HMAC with a shared secret to sign each request.
- Validate the signature before processing the update.
- Rate limit score update requests per user.

2. Get Top 10 Scores

Endpoint: GET /api/score/top

Description: Fetches the top 10 users with the highest scores.

Response:

```
{
  "scores": [
    {
      "userId": "string",
      "score": 12345
    }
  ]
}
```

3. WebSocket Live Updates

Endpoint: ws://<host>/api/score/live

Description: Provides a WebSocket endpoint to broadcast real-time scoreboard updates.

Message Structure:

```
{
  "type": "update",
  "scores": [
    {
      "userId": "string",
      "score": 12345
    }
  ]
}
```

```
}  
  
]  
  
}
```

Data Model

User Score

```
interface UserScore {  
  
    userId: string;  
  
    score: number;  
  
}
```

Execution Flow

1. User Action: User performs an action that triggers a score update.
2. API Request: Client sends a signed request to the /api/score/update endpoint.
3. Validation: The server validates the request signature and checks rate limits.
4. Score Update: If valid, the user's score is updated.
5. Broadcast: Updated top 10 scores are broadcast via WebSockets.
6. Client Update: Connected clients update the displayed scoreboard.

Security Measures

1. Request Validation: Ensure all score update requests include a valid HMAC signature.
2. Rate Limiting: Implement rate limiting to prevent abuse.
3. Input Sanitization: Sanitize and validate all inputs.
4. Logging: Log suspicious activity for further investigation.

Suggested Improvements

1. Implement a caching layer (e.g., Redis) for faster scoreboard retrieval.
2. Use JWT for user authentication alongside HMAC for action validation.
3. Introduce replay protection by tracking recent actions.
4. Optimize score calculations with batch processing.

Diagram



|

v

+-----+

| Signed API Request (HMAC) |

+-----+

|

v

+-----+

| Validate Signature & Rate-Limit |

+-----+

|

v

+-----+

| Update User Score |

+-----+

|

v

+-----+

| Update Top 10 & Broadcast |

+-----+

|

v

+-----+

| Real-Time WebSocket |

+-----+

This specification defines a robust and secure module to handle real-time scoreboard updates while safeguarding against unauthorized score manipulation.