OCTOPAI

Octopai API: UserEvents Documentation for User Audit Trails

Objective

This documentation provides a step-by-step guide to interact with the Octopai API. It specifically covers two endpoints: /api/UserAccount/Login and /api/UserAccount/UserEvents. The utilization of these endpoints can serve various business use cases including User Behavior Analysis, Security and Fraud Detection, Compliance and Auditing, System Monitoring and Performance, Customer Support, and Product Development.

Business Use Cases

An audit trail is a security-relevant chronological record that provides documentary evidence of the sequence of activities that have affected a specific operation, procedure, or event. In the context of this API, it records the sequence of user activities or events.
Audit trails are a crucial aspect of security and compliance for many organizations. They are used to detect security incidents, performance issues, and to aid in the recovery from incidents. Additionally, they support the investigation and forensic analysis of how an incident occurred.

Here is how the UserEvents functionality contributes to a User Audit Trail:

- User Authentication: The API logs events related to user authentication, such as successful and failed login attempts. This can help detect potential security risks, like repeated failed login attempts that might indicate a brute-force attack.
- User Activity: The API records various user activities like page loads. By tracking these events, administrators can establish a pattern of normal behavior per user, making it easier to identify anomalous actions that could signify a breach.
- Timestamps: Every event logged by the API includes a timestamp. This allows administrators to reconstruct the sequence of events leading up to a particular incident, which is vital in forensic investigations.
- Data Source: The IP addresses from which events originate are also recorded. This can be used to identify suspicious activity from unfamiliar sources.

By extracting and analyzing data from the UserEvents API, organizations can maintain a comprehensive audit trail that helps uphold security, facilitate incident response, and ensure regulatory compliance.

Pre-requirements

- Familiarity with HTTP methods, specifically POST
- Ability to use command-line tools like **‘curl’**
- Valid Octopai User credentials (email and password)

Instructions

Step 1: User Login

To authenticate a user, send a POST request to the following endpoint:

```
https://<your URL name>.octopai.com/api/UserAccount/Login
```

Here is how to construct and send this request using **‘curl’**:

```
curl --location 'https://<YOUR URL NAME>.octopai.com/api/UserAccount/Login' \
--header 'Content-Type: application/json' \
--data-raw '{
  "Username": "<YOUR USER EMAIL>",
  "Password": "<YOUR PASSWORD>"
}'
```

Replace ‘**<YOUR URL NAME>’**, ‘<**YOUR USER EMAIL**>’, and ‘<**YOUR PASSWORD**>’ with your actual values.

Step 2: Extract Access Token

After a successful login, the API returns a JSON response containing an accessToken, which is needed for subsequent authenticated requests. Here is an example of a successful response:

```
{
  "accessToken": "<YOUR ACCESS TOKEN>",
  "expiration": "2023-07-23T12:36:37.0355877Z",
  "refreshToken": {
    "token": "<your refresh token>",
    "expiration": "2023-08-22T10:36:37.035615Z"
  },
  "userName": "<YOUR USER NAME>",
  "displayName": "<YOUR DISPLAY NAME>",
  "userEmail": null,
  "isAdmin": true,
  "error": null,
  "groupMember": [],
  "authLevel": "ADMIN",
  "displayModules": "..."
}
```

Extract the ‘**accessToken**’ from this response for the next step.

Step 3: User Events Request

To retrieve user events, send a POST request to the following endpoint:

```
/api/UserAccount/UserEvents
```

This request requires the '**accessToken**' obtained from the previous step and the user must have an Admin role. Here is how to construct and send the request:

```
curl --location 'https://<YOUR URL NAME>.octopai.com/api/UserAccount/UserEvents' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer <ACCESS TOKEN>' \
--data '{
  "PageNumber":"1",
  "RowsInPage":"50"
}'
```

Replace '<**YOUR URL NAME**>' and '<**ACCESS TOKEN**>' with your actual values.

Step 4: Understanding the Response

The response from the '**/api/UserAccount/UserEvents**' endpoint is an array of objects, each representing a user event. Here is an example of a successful response:

```
[
  {
    "type": "LOGIN SUCCESS",
    "userName": null,
    "source": "<IP>",
    "time": "2023-07-23T10:39:33.84"
  },
  {
    "type": "LOGIN FAIL",
    "userName": null,
    "source": "<IP>",
    "time": "2023-07-18T13:11:17.6"
  },
  {
    "type": "PAGE LOAD",
    "userName": null,
    "source": "<IP>",
    "time": "2023-07-18T13:10:58.2"
  }
]
```

Each object contains the following properties:

- **'type'**: The type of the event (e.g., "LOGIN SUCCESS", "LOGIN FAIL").
- **'userName'**: The username associated with the event. This may be null or "UNKNOWN USER".
- **'source'**: The IP address from which the event originated.
- **'time'**: The timestamp of when the event occurred.

Security Considerations

Store your accessToken securely and refresh it as needed to prevent it from being compromised. Always use HTTPS to make your requests to ensure data security during transit.

For further queries, reach out to support at support@octopai.com.