vorlon

10 Questions to Ask About API Activity

Introduction

When you're diving into the depths of API activity logs, it feels a bit like detective work, doesn't it? Have no fear --we've put together a comprehensive guide to interpreting API logs, including a sample log that we'll dissect together.

Analysis of an API Log

The sample log details an API request to create a new order, made by an external partner at 03:45 AM UTC from Berlin, Germany. It shows a successful **POST** to the **/api/v1/orders** endpoint, with details of the order including items count and total price. The request, completed in 482 milliseconds with a **201 Created** status, involved **create_order** permissions. The log also notes the frequency of requests and includes a session ID.

sample API log


```
{
  "timestamp": "2024-02-02T03:45:00.123Z",
  "user_id": "8562",
  "user_role": "external_partner",
  "api_version": "v1",
  "endpoint": "/api/v1/orders",
  "http_method": "POST",
  "ip_address": "198.51.100.25",
  "location": "Berlin, Germany",
  "user_agent": "Mozilla/5.0 (Linux; Android 10; K)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0 Mobile
Safari/537.36",
  "response_status": 201,
  "response_time_ms": 482,
  "data_accessed": {
    "order_id": "4521",
    "action": "create_order",
    "details": {
      "items_count": 3,
      "total_price": 150.75,
      "currency": "EUR"
    }
  },
  "modification": {
    "type": "creation",
    "affected_fields": ["order_id", "items_count", "total_price",
"currency"]
  },
  "frequency": "3 requests in 5 seconds",
  "permissions_used": ["create_order"],
  "session_id": "Xf53JK2l3kj53"
}
```

1. Who accessed the API?

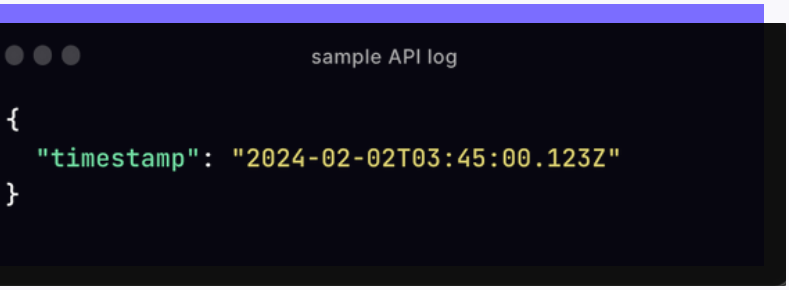
Start by identifying the user or service account that accessed the API. Legitimate requests usually come from recognized users or services within your organization. An unknown or unexpected accessor could be a red flag indicating unauthorized access.

The API was accessed by a user with the ***ID 8562***, who has the role of ***external_partner***. Verification is needed to ensure this user has the appropriate permissions for the actions they are performing.

2. What time did the access occur?

The timing of access can be quite telling. Access during off-hours, especially in the dead of night, might indicate a sneaky attempt at avoiding detection, particularly if it's out of character for the accessing entity.

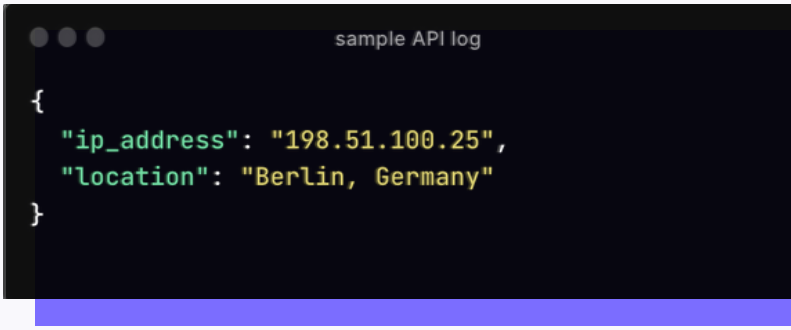
The access occurred at ***03:45 AM UTC***, which might be unusual depending on the user's typical activity patterns and the timezone they operate in.



3. From where was the API accessed?

Geolocation can be a dead giveaway. An API call originating from a location where your company doesn't operate or from a country known for harboring cybercriminals deserves a second look.

The access originated from Berlin, Germany, with the specific IP address ***198.51.100.25***. This detail helps assess whether the access location is expected for the user role.



4. What data was accessed?

Understanding the type of data accessed can help gauge the potential impact. Access to sensitive information like personal data or proprietary business information is more concerning than access to public or less sensitive data.

The POST request to the ***/api/v1/orders*** endpoint indicates an order creation with details about items and total price, which is sensitive business operation data.

sample API log

```
{
  "api_version": "v1",
  "endpoint": "/api/v1/orders"
}
```

sample API log

```
{
  "frequency": "3 requests in 5 seconds"
}
```

5. How frequently was the API accessed?

An unusually high frequency of access within a short period might indicate a brute force attack or data scraping attempt, especially if it deviates significantly from the norm.

The log shows a moderate frequency: ***3 requests in 5 seconds***, which might be normal for order creation but should be contextualized within typical user behavior.

6. Were any modifications made?

Check if the API call resulted in any data being modified or deleted. Unauthorized changes to data or configurations are clear indicators of malicious activity.

A new order was created (***type: creation***), affecting several fields related to the order. This is a significant modification that should be cross-referenced with the user's permissions.

sample API log

```
{
  "modification": {
    "type": "creation",
    "affected_fields": ["order_id", "items_count", "total_price",
"currency"]
  }
}
```

7. What was the response status?

Investigate the response status of the API calls. A high volume of failed attempts (e.g., 401 Unauthorized or 403 Forbidden responses) might suggest someone is probing for vulnerabilities.

The response status was ***201 (Created)***, indicating that the request successfully resulted in a new resource being created, which aligns with the POST action performed.

sample API log


```
{
  "response_status": 201
}
```


sample API log


```
{
  "timestamp": "2024-02-02T03:45:00.123Z",
  "user_id": "8562",
  "user_role": "external_partner",
  "api_version": "v1",
  "endpoint": "/api/v1/orders",
  "http_method": "POST",
  "ip_address": "198.51.100.25",
  "location": "Berlin, Germany",
  "user_agent": "Mozilla/5.0 (Linux; Android 10; K)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0 Mobile
Safari/537.36",
  "response_status": 201,
  "response_time_ms": 482,
  "data_accessed": {
    "order_id": "4521",
    "action": "create_order",
    "details": {
      "items_count": 3,
      "total_price": 150.75,
      "currency": "EUR"
    }
  },
  "modification": {
    "type": "creation",
    "affected_fields": ["order_id", "items_count", "total_price",
"currency"]
  },
  "frequency": "3 requests in 5 seconds",
  "permissions_used": ["create_order"],
  "session_id": "Xf53JK2l3kj53"
}
```


8. Is there a pattern or anomaly?

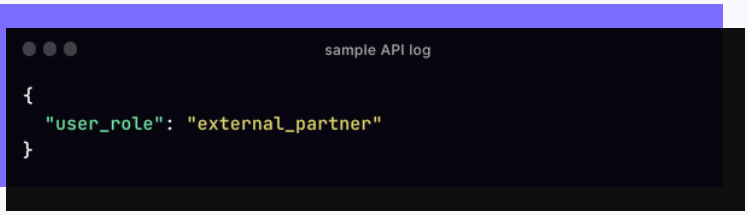
Look for patterns or anomalies in the API access logs. Repeated attempts to access the same data or systematic exploration of different endpoints could be a sign of reconnaissance by an attacker.

Without historical data for comparison, it's hard to say definitively. However, the log doesn't immediately suggest an anomaly, assuming the user role aligns with the action taken.

9. What permissions are used?

Determine the permissions invoked during the API call. Usage of elevated permissions or permissions not typically used by the accessing account should raise eyebrows.

The permission ***create_order*** was used, which should be reviewed against the external_partner role to ensure it aligns with their access rights.



10. Does the activity align with known behaviors?

Finally, compare the activity against the baseline of known good behaviors for your APIs. Deviations from established patterns of normal behavior are often the earliest signs of compromise.

If ***external_partner*** roles are typically allowed to create orders, this activity may align with expected behaviors. However, it's crucial to continually validate that such actions remain within the scope of the user's legitimate business needs.

Conclusion

By asking these questions, you're not just reviewing logs; you're conducting a full-fledged investigation into the heart of your API's security. The key to a fortified API security posture is continuous monitoring and regular rotation of access and service account credentials. This approach helps you not only identify potential threats but also understand normal behaviors and traffic patterns, making it easier to spot anomalies in the future.