Justin Min
01/09/2026

Deloitte.

Deloitte Cyber Program

Project Overview

In an increasingly complex digital landscape, Deloitte helps organizations navigate cyber threats, adapt to evolving risks, and protect their most critical assets so they can deliver sustained outcomes for their stakeholders. Deloitte's Cyber Risk platform has capabilities across industries, focusing on resilience, vigilance, and security.

During this program, you will get the opportunity to step into the shoes of a Deloitte team member working with the Cyber Intelligence Centre (CIC) and complete tasks that replicate the critical work this team does during a live security incident. You will learn how to support a client during a breach, interpret technical logs, and identify malicious actors to mitigate data loss.

Skills you will learn and practice: Forensic Analysis, Incident Response, Log Inspection, Threat Hunting, Critical Thinking, Client Communication, and Data Privacy Management.

Scenario

A major news publication has revealed sensitive private information about Daikibo Industrials, our client. A production problem has caused its assembly lines to stop, threatening the smooth operation of supply chains relying on Daikibo's products. The client suspects the security of their new status board may have been breached.

Task

In this task you will be joining our cyber security team. Your job is to:

1. Determine if the alleged breach could have happened from an attacker on the internet directly (i.e. no access to Daikibo's VPN).
2. Inspect a web_requests.log file (listing only data from a period when the alleged attack has to have happened):
    - Try to spot suspicious requests
        - Hint: In the Resources section, you can find a diagram example of how to read the logs file
        - Hint: Look for longer sequences of user requests
        - Hint: Notice the order of requests from Login → to requests for the dashboard page's resources (styles, scripts, images, etc.) → to API requests for the actual statuses of the machines
        - Hint: How would you recognise if an automated request to the API happens at an exact interval of time (assume no such functionality is available in the dashboard)?
    - If you've identified such requests make sure to write down the ID of the user (it's part of the requests)

Here is how the web_requests.log file is structured:

- There is a sequence of blocks of text divided by empty lines
- Each block represents the activity of a unique IP address (no 2 blocks have the same IP)
- The block starts with the IP address followed by a table of the requests made to Daikibo's telemetry dashboard (the dashboard lives in Daikibo's intranet) by the device with this IP address, sorted by time
- The IP addresses are from the internal Daikibo network and are static
- 1 block can represent 1 or multiple browsing sessions
- Sessions made on different dates require new logins
- There is no continuous polling/pushing of data between client and server - the users need to refresh the page to get the latest data
    - Hint: For an easier visual inspection, open up the file in a code editor like Sublime Text or Visual Studio Code, expand the window to the full width of your screen and decrease font size until no text breaks on a new line

When you believe you have completed the 2 tasks above, submit your work by taking a quick quiz to check your discoveries. Start the quiz by clicking 'Start your quiz' below. Good luck!

Here are some resources to help you:

1. web_activity.log:
   https://cdn.theforage.com/vinternships/companyassets/9PBTqmSxAf6zZTseP/web_activity.log
2. How to Read the Log (Diagram):
   https://cdn.theforage.com/vinternships/companyassets/9PBTqmSxAf6zZTseP/how-to-read-the-logs.pdf

```
  2021-06-25T12:15:48.000Z GET    "/index.css" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                                                                  200 (SUCCESS)
  2021-06-25T12:15:49.000Z GET    "/index.js" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                                                                   200 (SUCCESS)
  2021-06-25T12:15:50.000Z GET    "/api/factory/status?factory=*" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                                               200 (SUCCESS)
  2021-06-25T12:16:17.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                        200 (SUCCESS)
  2021-06-25T12:17:16.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                     200 (SUCCESS)
  2021-06-25T12:17:34.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}                     200 (SUCCESS)
  2021-06-25T12:17:42.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=HeavyDutyDrill" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}      200 (SUCCESS)
  2021-06-25T12:19:20.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=SpotWelder" {authorizedUserId: "thZTtwb2Avvzsnw4X9R2MD"}          200 (SUCCESS)

192.168.0.32:
  TIME                    METHOD REQUEST                                                                                                                 STATUS
  2021-06-25T12:35:01.000Z GET    "/"                                                                                                                    401 (UNAUTHORIZED)
  2021-06-25T12:35:01.000Z GET    "/login"                                                                                                               200 (SUCCESS)
  2021-06-25T12:35:01.000Z GET    "/login.css"                                                                                                           200 (SUCCESS)
  2021-06-25T12:35:01.000Z GET    "/login.js"                                                                                                            200 (SUCCESS)
  2021-06-25T12:35:29.000Z POST   "/login"                                                                                                               200 (SUCCESS)
  2021-06-25T12:35:30.000Z GET    "/" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                                       200 (SUCCESS)
  2021-06-25T12:35:31.000Z GET    "/index.css" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                              200 (SUCCESS)
  2021-06-25T12:35:32.000Z GET    "/index.js" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                               200 (SUCCESS)
  2021-06-25T12:35:32.000Z GET    "/api/factory/status?factory=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                           200 (SUCCESS)
  2021-06-25T12:35:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                    200 (SUCCESS)
  2021-06-25T12:36:08.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=Furnace" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}             200 (SUCCESS)
  2021-06-25T12:37:30.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                   200 (SUCCESS)
  2021-06-25T12:38:24.000Z GET    "/api/factory/machine/status?factory=berlin&machine=HeavyDutyDrill" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}    200 (SUCCESS)
  2021-06-26T12:52:39.000Z GET    "/"                                                                                                                    401 (UNAUTHORIZED)
  2021-06-26T12:52:39.000Z GET    "/login"                                                                                                               200 (SUCCESS)
  2021-06-26T12:52:39.000Z GET    "/login.css"                                                                                                           200 (SUCCESS)
  2021-06-26T12:52:39.000Z GET    "/login.js"                                                                                                            200 (SUCCESS)
  2021-06-26T12:53:12.000Z POST   "/login"                                                                                                               200 (SUCCESS)
  2021-06-26T12:53:12.000Z GET    "/" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                                       200 (SUCCESS)
  2021-06-26T12:53:12.000Z GET    "/index.css" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                              200 (SUCCESS)
  2021-06-26T12:53:12.000Z GET    "/index.js" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                               200 (SUCCESS)
  2021-06-26T12:53:12.000Z GET    "/api/factory/status?factory=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                           200 (SUCCESS)
  2021-06-26T12:53:56.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                 200 (SUCCESS)
  2021-06-26T12:54:43.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=LaserCutter" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}      200 (SUCCESS)
  2021-06-26T12:56:04.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                   200 (SUCCESS)
  2021-06-26T12:56:29.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                   200 (SUCCESS)
  2021-06-27T13:13:29.000Z GET    "/"                                                                                                                    401 (UNAUTHORIZED)
  2021-06-27T13:13:29.000Z GET    "/login"                                                                                                               200 (SUCCESS)
  2021-06-27T13:13:30.000Z GET    "/login.css"                                                                                                           200 (SUCCESS)
  2021-06-27T13:13:30.000Z GET    "/login.js"                                                                                                            200 (SUCCESS)
  2021-06-27T13:14:06.000Z POST   "/login"                                                                                                               200 (SUCCESS)
  2021-06-27T13:14:07.000Z GET    "/" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                                       200 (SUCCESS)
  2021-06-27T13:14:08.000Z GET    "/index.css" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                              200 (SUCCESS)
  2021-06-27T13:14:09.000Z GET    "/index.js" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                                               200 (SUCCESS)
  2021-06-27T13:14:09.000Z GET    "/api/factory/status?factory=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                                           200 (SUCCESS)
  2021-06-27T13:14:46.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                 200 (SUCCESS)
  2021-06-27T13:14:51.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}                    200 (SUCCESS)
  2021-06-27T13:15:42.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=MetalPress" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}         200 (SUCCESS)
  2021-06-27T13:16:46.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=AirWrench" {authorizedUserId: "mTDKsgxXBn64f2yFgPUDMg"}          200 (SUCCESS)
```

Figure 1.1 Snippet of the activity log

Action

1. Network Access Assessment

I started by reviewing the provided project documentation to determine the dashboard's hosting environment.

- Analysis: I verified that the Daikibo manufacturing dashboard is hosted on the company's internal intranet, not the public internet.
- Conclusion: This confirmed that an attacker could not access the dashboard directly from the web without a VPN connection, effectively narrowing the scope to an internal or VPN-connected threat.

2. Log File Inspection

I opened the provided web_activity.log file to identify the specific source of the suspicious traffic.

- Baseline Comparison: I analyzed standard user behavior, noting that a typical session involves a sequence of Login → Dashboard Resources (CSS/Images) → User Interaction.
- Identifying the Anomaly: I identified a specific User ID (mdB7yD2dp1BFZPontHBQ1Z) that initially appeared normal—logging in and browsing the dashboard—but then deviated significantly. Unlike a human user, subsequent requests retrieved data (checking statuses for all 4 factories) without loading any page resources (CSS/JS), indicating a "headless" script.
- Timing & Error Analysis: I reviewed the timestamps and observed two distinct mechanical behaviors:
    a. Precision: The requests occurred at exact one-hour intervals (e.g., 17:00:48, 18:00:48), showing an automated attempt.
    b. Blind Persistence: From 00:00 to 16:00 on 06/26, the user's session expired (401 Unauthorized). Despite 16 consecutive failures, the source continued to send requests exactly on the hour. These constant and consistent attempts definitively confirmed the activity was an automated script rather than a person.

Results

Upon completing the analysis of the web_activity.log, I successfully isolated the source of the anomalous traffic.

The Actor: The traffic was traced to User ID mdB7yD2dp1BFZPontHBQ1Z.

The Verdict: The behavior confirmed that this was not a human user, but an automated script.

Key Evidence:

1. "Headless" Browsing: The user repeatedly queried API endpoints for factory data without ever requesting the associated Dashboard resources (CSS, JavaScript, Images) required for a human interface.
2. Mechanical Precision: The requests followed non-human punctuality, executing exactly at the 48th second of every hour (e.g., 17:00:48, 18:00:48).
3. Blind Persistence: Log analysis from 00:00 to 16:00 showed the script continuing to send requests despite receiving 16 consecutive 401 Unauthorized errors. This lack of error handling definitively proved the absence of human logic.

Conclusion:

The incident was identified as an internal automation misconfiguration rather than an external malicious attack. The script was likely checking system health but was poorly configured, creating unnecessary noise and error logs.

```
File    Edit    View
2021-06-25T16:14:00.000Z GET    "/"                                                                                                    401 (UNAUTHORIZED)
2021-06-25T16:14:01.000Z GET    "/login"                                                                                               200 (SUCCESS)
2021-06-25T16:14:01.000Z GET    "/login.css"                                                                                           200 (SUCCESS)
2021-06-25T16:14:02.000Z GET    "/login.js"                                                                                            200 (SUCCESS)
2021-06-25T16:14:54.000Z POST   "/login"                                                                                               200 (SUCCESS)
2021-06-25T16:14:54.000Z GET    "/" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}                                                       200 (SUCCESS)
2021-06-25T16:14:55.000Z GET    "/index.css" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}                                              200 (SUCCESS)
2021-06-25T16:14:56.000Z GET    "/index.js" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}                                               200 (SUCCESS)
2021-06-25T16:14:57.000Z GET    "/api/factory/status?factory=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}                           200 (SUCCESS)
2021-06-25T16:15:18.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T16:16:09.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=Furnace" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T16:16:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T16:17:30.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=AirWrench" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T16:18:39.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=HeavyDutyDrill" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"} 200 (SUCCESS)
2021-06-25T17:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T17:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T17:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T17:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T18:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T18:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T18:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T18:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T19:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T19:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T19:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T19:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T20:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T20:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T20:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T20:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T21:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T21:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T21:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T21:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T22:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T22:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T22:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T22:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-25T23:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T23:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     200 (SUCCESS)
2021-06-25T23:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  200 (SUCCESS)
2021-06-25T23:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    200 (SUCCESS)
2021-06-26T00:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
2021-06-26T00:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
2021-06-26T00:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  401 (UNAUTHORIZED)
2021-06-26T00:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    401 (UNAUTHORIZED)
2021-06-26T01:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
2021-06-26T01:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
2021-06-26T01:00:48.000Z GET    "/api/factory/machine/status?factory=shenzhen&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}  401 (UNAUTHORIZED)
2021-06-26T01:00:48.000Z GET    "/api/factory/machine/status?factory=berlin&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}    401 (UNAUTHORIZED)
2021-06-26T02:00:48.000Z GET    "/api/factory/machine/status?factory=meiyo&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
2021-06-26T02:00:48.000Z GET    "/api/factory/machine/status?factory=seiko&machine=*" {authorizedUserId: "mdB7yD2dp1BFZPontHBQ1Z"}     401 (UNAUTHORIZED)
```

Figure 1.2 Log sample of the mdB7yD2dp1BFZPontHBQ1Z

Results

I had previous experience reading logs thanks to my Cybersecurity Cloud Risk Analyst capstone internship at Georgia State University, but this Deloitte program offered a different perspective. During the internship, I utilized tools that aggregated the data and was more focused on mapping the controls, whereas here I was tasked with analyzing raw text data. Thousands of lines of timestamps and codes is a bit jarring at first, but as I began filtering the data and looking for patterns, that I adapted quickly.

Spotting an anomaly and distinguishing it from the ocean of other logs was a great moment of satisfaction. Realizing the 401 errors were happening on the dot was a big indicator of non-human intervention, and noticing that the User ID mdB7yD2dp1BFZPontHBQ1Z skipped straight to the raw data (API data) without asking for the files needed to display the data visually (the dashboard resources) was a reassuring moment that my attention to detail and interpreting technical data has sharpened.

Skills learned from this experience include:

- Log forensics and analysis, parsing server logs to identify anomalies
- Distinguishing human interaction (resource-heavy, need for display resources) from machine automation (rhythmic, efficient, punctual)
- Web security and computer networking

Certificate of Completion

Deloitte.

Justin Min

Cyber Job Simulation

Certificate of Completion

January 1st, 2026

Over the period of January 2026, Justin Min has completed practical tasks in:

Cyber security

Tina McCreery
Chief Human Resources Officer, Deloitte

Enrolment Verification Code w8LE6sCtdz9hQrxaK | User Verification Code 6956beacf76d215bcfaa943f | Issued by **Forage**