Analysis Report for: eb.txt

Decoded using latin-1...

Overall Functionality

The VBA code interacts with a Microsoft Excel spreadsheet named "B-Daten". It appears to be a reporting tool that extracts data from this spreadsheet and populates a "Tagesbericht" (daily report) sheet. The code processes data related to plant operation status, various measurements (e.g., Zulaufmenge - inflow volume, Fäkalien - fecal matter, energy consumption, CSB/BSB - chemical/biological oxygen demand), and personnel information. The "B-Daten" sheet seems to contain raw data organized across numerous columns (at least 138), and the code iterates through specific columns to find the latest valid data entry. The report then displays this information, including conditional formatting based on thresholds and the date.

Function Summaries

* **`Bericht()`**: This is the main subroutine. It initializes various string arrays (`AStatus`, `MStatus`, `ZGStatus`, `BStatus`, `UStatus`) containing status descriptions. Then, it calls three other subroutines (`SetFigures1`, `SetFigures2`, `SetFigures3`, `SetFigures4`) to populate different sections of the "Tagesbericht" sheet and finally selects cell A1. It has no parameters and no return value.

* **`SetFigures1()`**: This subroutine extracts and displays data related to plant operational status for four different plants. It finds the last numeric value in specific columns of the "B-Daten" sheet (columns 105-110), uses this to extract corresponding dates and values, and updates the "Tagesbericht" sheet with the extracted data and conditional formatting for the status based on numerical value. It has no parameters and no return value.

* **`SetFigures2()`**: This subroutine extracts and displays data related to various measurements from the "B-Daten" sheet (inflow volumes, fecal matter, energy, CSB/BSB/NH4-N/Pges for inflow and outflow). It finds the last numeric entry in the relevant columns and calculates sums and ratios. It also applies conditional formatting to highlight values outside specified ranges or those older than a certain number of days. It has no parameters and no return value.

* **`SetFigures3()`**: This subroutine extracts and displays personnel data from the "B-Daten" sheet (employee status, shift assignment, and additional notes). It uses data from columns 117-124 to populate status fields. It has no parameters and no return value.

* **`SetFigures4()`**: This subroutine extracts and displays data related to external work from the "B-Daten" sheet (description of work, duration, and completion status in columns 127-138) for up to five entries. It also applies conditional formatting based on the status. It has no parameters and no return value.

Control Flow

The control flow is largely sequential, with the `Bericht` subroutine orchestrating the execution of the four `SetFigures` subroutines. Each `SetFigures` subroutine follows a pattern:

1. **Finding the last data entry:** It iterates through rows of the "B-Daten" sheet (using nested loops for two ranges of rows: 2-373 and 374-745) to locate the last row containing a numeric value in a specified column. This is a crucial step, suggesting the spreadsheet receives daily updates, and the macro aims to extract the latest information.

2. **Extracting and displaying data:** Once the last row is found, the code extracts date and value data, writes formulas into cells on the "Tagesbericht" sheet to display values from the "B-Daten" sheet, or sets values directly using `.Value`.

3. **Conditional formatting:** It uses `Select Case` statements to apply conditional formatting (font color and cell background color) based on numeric values extracted from "B-Daten". This highlights important statuses or values that are outside of acceptable ranges.

The loops in each function use `Application.WorksheetFunction.IsNumber` to check if a cell contains a numerical value, ensuring that the macro gracefully handles missing or non-numeric data.

Data Structures

The primary data structures are:

* **String arrays:** `AStatus`, `MStatus`, `ZGStatus`, `BStatus`, `UStatus` are arrays of strings used to store status descriptions for plant operation, employee status, and shift assignments. They act as lookup tables to map numerical codes to human-readable descriptions.

* **Spreadsheet:** The Excel spreadsheet "B-Daten" serves as a crucial data source, storing the raw data that the VBA code processes and displays in the "Tagesbericht" sheet. The structure of "B-Daten" is implicitly defined by the column indices used in the VBA code. The columns appears to hold time-series data, with each row likely representing a day's measurements and status updates.

Malware Family Suggestion

While this code doesn't directly exhibit malicious behavior like network communication or file system manipulation, its functionality raises concerns about its potential use in a malicious context. It's designed to extract specific data from an Excel file. A malicious actor could modify this code to:

* **Data Exfiltration:** Instead of writing to a "Tagesbericht" sheet, the modified code could send the extracted data (plant status, measurements, personnel details) to a remote server. This would represent a data exfiltration attack.

* **Data Manipulation:** The code could be altered to silently modify values within the "B-Daten" sheet, potentially altering operational parameters or records.

* **Macro Trojan:** The macro itself, although seemingly benign in this instance, could be a vehicle for delivering a more sophisticated malware payload upon execution.

Therefore, while not definitively malware itself, the VBA code could be categorized as a potential **information stealer** component or a **macro-based dropper** depending on how a malicious actor would leverage it. The use of the code without proper authentication or authorization also opens the possibility for unauthorized access and modification of sensitive operational information. The analysis using Oletools is insufficient to detect all types of malware. Static analysis might not reveal all the potential risks. A dynamic analysis with sandboxing is strongly recommended to completely asses the risk.