The *Coffee Machine Incident*

GREGOR WEGBERG, MAX SCHRIMPF, KARL WÜST – LOVE TO WRITE LONG DOCUMENTS

You likely noticed the coffee machine being on vacation for quite some time during the semester break. Between the end of June and the end of July, students were not able to dispense their free coffee capsules. This article provides a brief overview how the coffee machine works (when it does) and summarizes the event itself. The post-mortem with details in regard to the incident itself can be found after this article and is highly recommended.

Let's start with the "coffee machine" itself. It consists of two parts: the two boilers converting precious capsules to coffee (lower part) and the coffee capsule dispenser (upper part). The coffee boilers are off-the-shelf products with a service contract in case of any trouble. So they keep working without any noteworthy problems. However, the capsule dispenser contains quite a lot of our own engineering and therefore is not covered by any contracts in case it stops working and requires a fix. To be more precise, it was the capsule dispensing system which broke and not the capsule dispenser. The latter would also be covered by the service contract, itself. In our case this system includes the off-the-shelf Nespresso capsule dispenser, an RFID reader, an MDB[1] to Serial converter (we call it "MDB board") and a computer running our "coffee service" software. The computer running the coffee service interacts with all other devices ➔

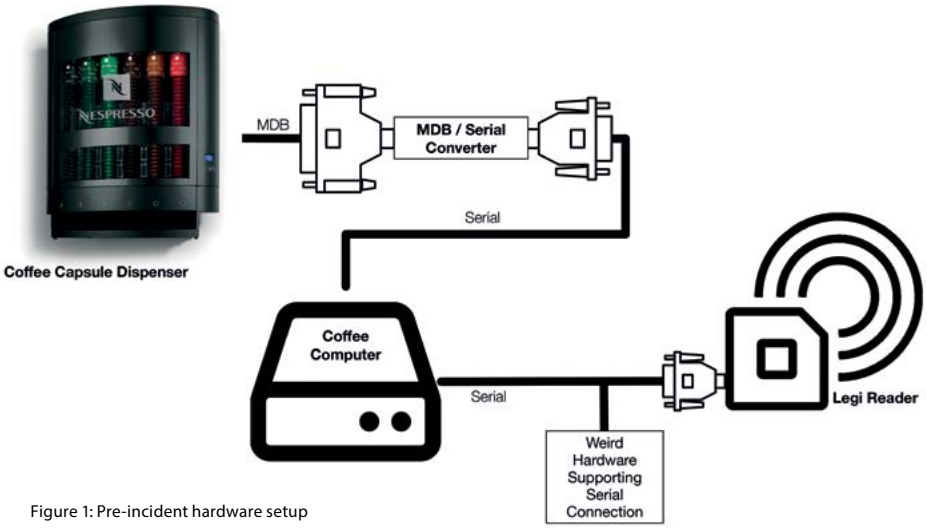

Figure 1: Pre-incident hardware setup

Adnovum

and orchestrates the whole process. A rough overview of the pre-incident hardware setup is shown in Figure 1.

At an abstract perspective (not containing the low level Hex string communication or the needed state machines) the process of dispensing a capsule is quite easy: The coffee service running on the computer constantly polls the RFID reader for new data. In case a student puts her Legi against the machine, the RFID reader answers with state data and the student's RFID number. This RFID number is sent over a web-service to all three student associations which provide free coffee to their students (VIS, AMIV, VMP) and requests all of them to check if they know this student. In case one of the student associations knows about the number, it answers with a success message including information indicating if this student is allowed to dispense a capsule right now. In this case, the coffee service instructs the capsule dispenser over the MDB board to dispense one capsule. The dispenser now releases a capsule after the student selected her favorite type of coffee. Afterwards, the dispenser informs the coffee service which type was chosen. This information, along with additional data (e.g. the student association which answered the request) is logged into a database. This data is used to feed the coffee statistics page [2] and is used by VIS to invoice the other student associations for the coffee dispensed by their members. At this point, the whole procedure starts anew.

The incident started likely due to the RFID reader breaking. We assume that either one of the soldering joints or one of the wires broke. It might also be the custom made board (Figure 1 "Weird Hardware Supporting Serial Connection") which gave up after being in use for quite a while. Or maybe the whole system just had some serious headache since it seems to be a widely spread theory that the electromagnetic fields of the RFID reader work better if you just hit the machine hard enough (Which is obviously bullshit. Please don't hit the coffee dispenser!). We would like to point out that these are just some working theories as we don't have a smoking gun pointing to the culprit. The post-mortem report following this article provides much more detail and lists all the observed events. We highly recommend to read it if you want to get a sense of how much work was involved in debugging, finding all problems and getting the system back up and running. We also tried to write the report in a way that is commonly used after IT incidents in the public or private sector so that aspect might also be of interest to you.

Please don't hit the coffee dispenser!

To be able to debug each component and the whole system, we decided to stop using Serial ports (e.g. RS232) and use Serial-to-USB converters resp. adapters. Only this way we were able to use our own hardware which nowadays does not commonly include Serial ports. This change was kept and the coffee computer is now connected through USB ports to peripherals (see Figure 2). We had terrific support from Bastli. They not just rewired our RFID reader to a Serial-to-USB converter, but also provided us with access to their source code used for the beer vending machine. With their help, we were able to get the system running again. However, the current state is just a short-term duct tape

solution. As stated in the post-mortem, there are still lots of problems waiting to be resolved, or even just be handled at all.

In the long run, a complete rewrite of the coffee service as well as replacing the current coffee computer is necessary. We aim to have a common codebase with Bastli and use the same software to run both the coffee dispenser and the beer vending machine. In addition, we aim to use the same, or very similar, hardware. This should lead to more stability, better documentation, a lot more people involved in the project and the ability to resolve future issues and many more positive effects. All in all, only such a long-term solution will make it possible to keep this central serVIS running and therefore freeing you of your tiredness.

At this point, we would like to thank everyone involved in resolving this incident. It was great working with you and we're very thankful for the time you invested to bring coffee to the students instead of studying!

Footnotes & References

[1] "MDB" stands for "Multi-Drop Bus" and is part of the "Multi-Drop Bus / Internal Communication Protocol" standardized by the National Automatic Merchandising Association (NAMA) [2]. This bus/protocol is used in many vending machines all over the world to connect, for example, a payment device to a vending machine. https://namanow.org/images/pdfs/technology/mdb_version_4-2.pdf

[2] https://www.vis.ethz.ch/de/services/lounge/statistics

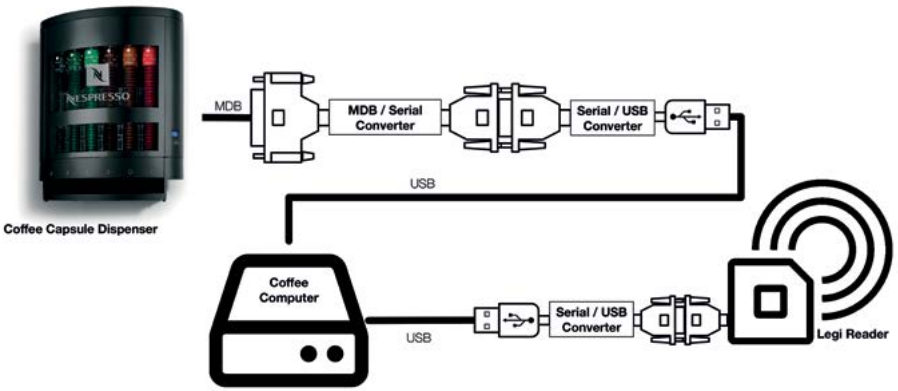

Figure 2: Post-incident hardware setup