JOHANNES KEPLER
UNIVERSITY LINZ

Michael Roland,
Tobias Höller,
Daniel Hofer,
Daniel Pekarek,
Michael Preisach
Institute of
Networks and Security,
LIT Secure and Correct
Systems Lab

@ michael.roland@ins.jku.at

DOI 10.35011/ww2q-d522

June 2023
(revised in Sept. 2023)

An analysis of PoS/ cashIT! cash registers

Abstract This report summarizes our findings about vulnerabilities in cashIT!, a cash register system implementing the Austrian cash registers security regulation (RKSV). Besides lack of encryption, outdated software components and low-entropy passwords, these weaknesses include a bypass of origin checks (CVE-2023-3654), unauthenticated remote database exfiltration (CVE-2023-3655), and unauthenticated remote code with administrative privileges on the cash register host machines (CVE-2023-3656). Based on our analysis result, these vulnerabilities affect over 200 cash register installations in Austrian restaurants that are accessible over the Internet. In addition, daily cloud backups of more than 300 active cash register installations (and over 600 including historic backups of presumably inactive installations) are freely downloadable from cashIT! servers. These cloud backups contain detailed sales data, user account information (potentially with data about current and former employees), and may contain customer contact information, credentials for the online signature creation unit, and credentials to the backend system of the Austrian card payment processor Hobex.

We acknowledge partial support by the LIT Secure and Correct Systems Lab funded by the State of Upper Austria.

JOHANNES KEPLER
UNIVERSITY LINZ
Altenberger Straße 69
4040 Linz, Austria
jku.at

Contents

Executive summary

This report summarizes our findings about the cashIT! cash register system. We stumbled upon this software product in November 2022 during an ongoing research project to systematically analyze services operated within the campus network of Johannes Kepler University Linz.

We already reported a number of issues to the system vendor, PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH, in December 2022:

1. Remote management functionality available only over HTTP without transport layer security resulting in potential transmission of sensitive data (login passwords, sales data, etc.) in clear-text over the public Internet and in making remote management susceptible to man-in-the-middle attacks (see section 3);
2. Enumeration of remote management interfaces of cash register devices accessible on the Internet (see section 4);
3. Access default password "admin" in production systems (see section 5);
4. Read-out of clear-text passwords of all users by privileged users through remote management (see section 7);
5. The risk of usage of low-entropy/low-complexity passwords for remote management access (see section 6).

Following a responsible disclosure strategy, we waited for vendor response and aimed for publication of our findings after a reasonable time frame (of at least 90 days). Before actual publication, we wanted to re-assess the status of the mentioned potential security issues, to make sure that the vendor (and/or its customers) had adequately addressed the issues, and we would, consequently, not cause any harm by this publication. We obtained permission to run tests on their production systems by the owner of the cash register system located on our campus.

During the course of this re-evaluation, we discovered a couple of further security issues that eventually gained us full remote code execution resulting in the capability to analyze the underlying program code of the application. This was possible through a chain of exploits that made it possible to read out clear-text passwords of privileged users by arbitrary authenticated remote users (see section 8) and use of that password to trigger a remote code execution available to privileged users (see section 9).

Once we had access to the program code of the cash register application, we found numerous endpoints that could potentially be abused to bypass authentication and/or authorization checks and could be usable for full remote code execution and full read-out of all data stored in the cash register by unauthenticated remote users. This report only lists a few examples to highlight the potential security impact (see section 10).

We further found that a server operated by the system vendor potentially contains recent backups of all[1] active cash registers (even if the cash register indicates that no cloud backups are made). We found backup archives of a total of 660 cash registers. Figure 9 gives an overview of the amount of discovered backup archives and the distribution of their modification timestamps.

[1]We only verified that these archives contained actual backups (by downloading them) for instances for which we had permission to perform our tests. Nevertheless, we suspect that all instances make these backups.

These findings are confirmed to apply to version 03.A06rks 2023.02.37 (which was the latest version when we performed our evaluation) and earlier. Appendix A lists the distribution of different software versions on systems that we could discover during our evaluation.

As a result, malicious unauthenticated remote actors could

1. execute arbitrary code with administrative privileges on the cash register host machines eventually resulting in complete take-over of these devices,
2. download (and modify) all data on any cash register that is accessible over the Internet, and
3. download (and modify) backup data of any cash register even without direct remote access to the affected cash register (and likely beyond control of the owner of the cash register).

We consider this particularly problematic as the backups may contain potentially sensitive data, such as (but not limited to)

1. user account information of current (and former) employees,
2. detailed sales data,
3. customer contact information,
4. credentials for the online signature creation unit (according to the Austrian RKSV), and
5. credentials to the backend system of the Austrian card payment processor Hobex.

Beyond that we also found a public web dashboard operated by the system vendor that exposes further details about customers and their cash register installations (see section 13). Moreover, we identified a few other issues like outdated software and firmware versions (see section 11) and generic default passwords (see section 12) used in components bundled with the cash register system.

1. Background

During the course of a systematic analysis of services operated within the campus network of Johannes Kepler University Linz, we came across a system identified as "*PoS/ cash/IT Kassa, Version 03.A03rks (2022.03.15)*" at http://===REDACTED===:80 run by ===REDACTED===. This service raised our interest because

1. it operated over plain HTTP without transport layer security,
2. it did not offer an equivalent HTTPS service, and
3. its front page (see Figure 1) exposed user input fields that indicated a user authentication method (using either an on-screen keypad or a password input field).

Obviously, we could not resist to try the word "admin" as a login credential, which immediately gave us administrative access to the cash register system. We also found that "admin" was the credential for access to their public demo offered at https://www.cashit.at/testen/.

Further analysis revealed that the system is not only accessible from within the university network (including eduroam on our campuses), but also from the public Internet.

Since this was not following state-of-the-art best practices for web applications, we assumed a configuration error on the operator side. As part of responsible disclosure, we immediately informed them that they had accidentally made their internal cash register system available to the public Internet. However, during that disclosure, we were informed by the responsible contact of the operator that

1. the service was intentionally accessible over the Internet for remote management purposes,
2. this form of remote management access was supposedly the intended purpose of the product as marketed by the retailer (with a dynamic redirection service offered at http://<installation-name>.posdev.online/ to facilitate easy access even with dynamically changing IP addresses), and
3. that their administrative password is a presumably strong password and certainly not the word "admin" (the account was named "admin" though).

This led to a more thorough evaluation. The findings of this evaluation are documented in this report.

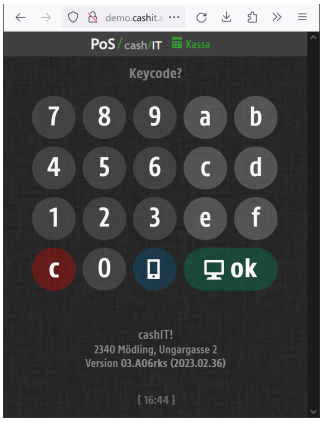


Figure 1: Login screen at demo instance

2. Timeline of findings and disclosure

- **2022-11-22**
  Initial discovery of suspicious service within our campus network (see section 1).
- **2022-11-23**
  Reported initial findings (see sections 1, 3, 5, and 6) to system operator (phone call with ===REDACTED=== at ===REDACTED===).
- **2022-11-23**
  Continued further investigation.
- **2022-12-07**
  Reported findings (see sections 1, 3, 4, 5, 6, and 7) to vendor (via e-mail to support@cashit.at and support@posdev.eu) and CERT.at (Ticket number ===REDACTED===).
- **2022-12-07**
  Received vendor response (via e-mail by ===REDACTED=== at PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH).
- **2022-12-08**
  Internally discussed and verified claimed vendor response.
- **2023-03-26**
  Continued analysis of dynamic redirection service (after we had given the vendor a sufficiently long period for applying fixes to production systems).
- **2023-06-13**
  Assessment of potential publication of findings and evaluation of results from the dynamic redirection service analysis.
- **2023-06-13**
  Discovery of a range of further potential issues (see sections 3.2, 4.2, 5.2, 6.2, 8, 9, 10, 11, 12, and 13).
- **2023-06-15**
  Approached ===REDACTED=== at ===REDACTED=== to obtain permission to verify assumptions on their production systems.
- **2023-06-19**
  Received permission to verify preliminary findings on their production systems from ===REDACTED=== at ===REDACTED=== (via phone call).
- **2023-06-20**
  Continued analysis and verification of findings.
- **2023-06-27**
  Reported findings (see sections 3.2, 4.2, 5.2, 6.2, 8, 9, 10, 11, 12, and 13) to vendor (via e-mail to ===REDACTED===@posdev.eu) and CERT.at (Ticket number ===REDACTED===).
- **2023-06-28**
  Received vendor response (via e-mail by ===REDACTED=== at PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH).
- **2023-07-03**
  Approached CERT.at regarding assignment of CVE numbers and assistance in further vendor communication (Ticket number ===REDACTED===).
- **2023-07-03**
  Received vendor response (via e-mail by ===REDACTED=== at PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH).

- **2023-07-04**
  Received response from CERT.at about initiation of CVE assignment process via an external CNA (via phone call with ===REDACTED=== at CERT.at).
- **2023-07-04**
  Approached CyberDanube regarding assignment of CVE numbers (contact to ===REDACTED=== established by CERT.at).
- **2023-07-13**
  CyberDanube Security Technologies GmbH confirmed reservation of CVE-2023-3654, CVE-2023-3655, and CVE-2023-3656.
- **2023-08-08**
  Submitted talk proposal to IKT-Sicherheitskonferenz 2023.
- **2023-08-17**
  Talk "Wie man alle OWASP Top 10 abkassiert!" accepted and added to official program of IKT-Sicherheitskonferenz 2023.
- **2023-08-21**
  Reported updated publication timeline to vendor (via e-mail to ===REDACTED=== at PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH) and CERT.at (Ticket number ===REDACTED===).
- **2023-10-03**
  Public disclosure of this report and presentation of findings at IKT-Sicherheitskonferenz 2023.

3. HTTP-only management web interface (no transport layer security)

While we initially believed that the cash register web interface HTTP service had been accidentally exposed to the public Internet due to misconfiguration, we quickly realized that this was intended for remote management purposes. The system vendor also offers a service that redirects http://<installation-name>.posdev.online/ and http://<installation-name>.cashit.info/ to the HTTP server of the cash register to facilitate remote management access even if the service is operated at a dynamically changing IP address. This service seems to always redirect to a URL of the form http://<IPv4-address>:80/ (i.e. plain HTTP on port 80) and is itself not responding through HTTPS (though port 443 seems to be open).

We believe that access over unsecure HTTP is problematic for several reasons:

1. Login credentials are transmitted in cleartext. A potential on-the-wire attacker could intercept and obtain such credentials. Since remote management would typically be performed by privileged users, this is a particular risk.
2. Potentially sensitive data (such as customer account data and sales data) can be accessed through remote management. A typical use-case discussed with one of the operators of the cash register is indeed that their tax accountant can directly collect sales data from their cash registers. Again, a potential on-the-wire attacker could intercept such data.
3. Since the communication is unauthenticated, an active man-in-the-middle (MITM) attack could potentially be performed, not only intercepting but also modifying data on the wire. In the case of sales data, such an attack could be used to falsify data reported to the tax accountant. Similarly, configuration changes performed by a remote administrator could be tampered with by such an attacker.
4. Since the redirector service at *.posdev.online (and *.cashit.info) does not use HTTPS, these capabilities extend to MITM attackers intercepting communication with this service as well (since an attacker would be able to manipulate the redirect to later intercept communication).

3.1 Vendor response

We reported this finding to the vendor on 2022-12-07 and received a response on the same day. In their response (see Appendix C), the vendor indicated that TLS is supported as an option and that they offer configuration of TLS (with or without browser-trusted server certificates) as a paid add-on service.

3.2 Re-evaluation

We strongly believe that using transport layer security is a state-of-the-art best practice and HTTPS has become the de facto standard for websites. While we acknowledge the additional effort required on the customer side, we believe that at least the redirector service should facilitate HTTPS to eliminate one additional attack path. We would also like to point out that the entry barrier to HTTPS has become very low due to free-of-charge certificate issuance services/certificate authorities such as Let's Encrypt[2].

[2]https://letsencrypt.org/

We further leveraged a device enumeration possible with the *.posdev.online redirector service (see section 4) to analyze currently accessible systems. We found that the redirector service usually points to a URL of the form http://<IPv4-address>:80/, regardless of the actual port that the cash register was made available on (i.e. some responded on a different port, some did not respond at all, even though the host was up; only for 5 cash registers, the redirector responded with a different port value). Not a single cash register responded with HTTPS on port 443 (though 44 of the enumerated hosts had other services, such as web management interfaces of Internet gateways, operating on port 443). Therefore, we assume it is not widely known that HTTPS could be enabled for the cash register remote management web service. However, we found that some of the enumerated hosts exposed ports typically used for VPN tunnels instead of the cash register remote management web service.

4. Dynamic IP resolution service permits enumeration of devices and customers

The system vendor offers a dynamic redirector service that redirects http://<installation-name>.posdev.online/ and http://<installation-name>.cashit.info/ to the HTTP server of the cash register to facilitate remote management access even if the service is operated at a dynamically changing IP address. <installation-name> is a customer defined name for the cash register. For instance, http://===REDACTED===.posdev.online/ displays an IFRAME that points to http://===REDACTED===:80.

We found that the service does not only accept the full installation name but also partial names if they match exactly one installation name. For instance, if only the two instances “mycashregister” and “someregister”, the URL http://cas.posdev.online/ would match the instance “mycashregister”, but the URL http://reg.posdev.online/ would match none of them.

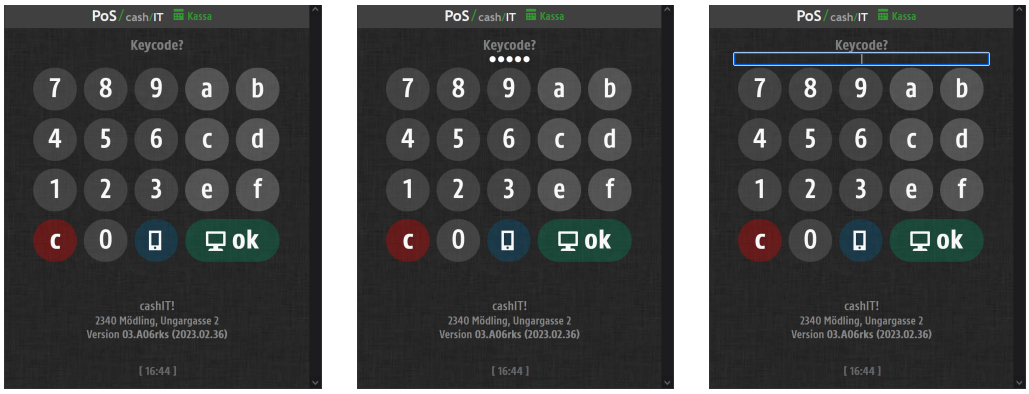
This permits to quickly enumerate a large number of installations through systematic search starting with single letters a..z for the installations and continuously adding more letters (aa..az, ba..bz, ...) until an IFRAME pointing to some installation address is returned by the service.

4.1 Vendor response

We reported this finding to the vendor on 2022-12-07 and received a response on the same day. In their response (see Appendix C), the vendor indicated that they immediately fixed this issue and no longer redirect for partial names. Customers would now need the complete installation names (customer-defined “keyword” according to the vendor response). They further explained, that customers can completely disable registration of their cash registers with this dynamic redirector service.

4.2 Re-evaluation

We could confirm that access through partial installation names (and consequently enumeration through systematic testing of installation names) is no longer possible since we received the vendor response.



(a) (b) (c)

Figure 2: Login screen (a) before and (b) after typing a keycode, and (c) with the password input field highlighted.

However, we discovered that besides customer-chosen installation names, the dynamic redirector service also redirects the cash register identifier. Cash register identifiers have the form "PosNNNNN", where NNNNN is a 5-digit number that seems to be sequentially given to newly installed cash registers. The first assigned number is "Pos10000", the last assigned number seems to be "Pos11031" (though continuously growing, indicating registration of new devices). The last possible number seems to be "Pos29999" (pointing to the IPv4 address where the redirector service is hosted itself). For 5 of the enumerated IDs, the redirector asnwered with a different port number, where one of the values was not a valid port number (valid: 81, 88, 6060, 8080; invalid: 2017.1).

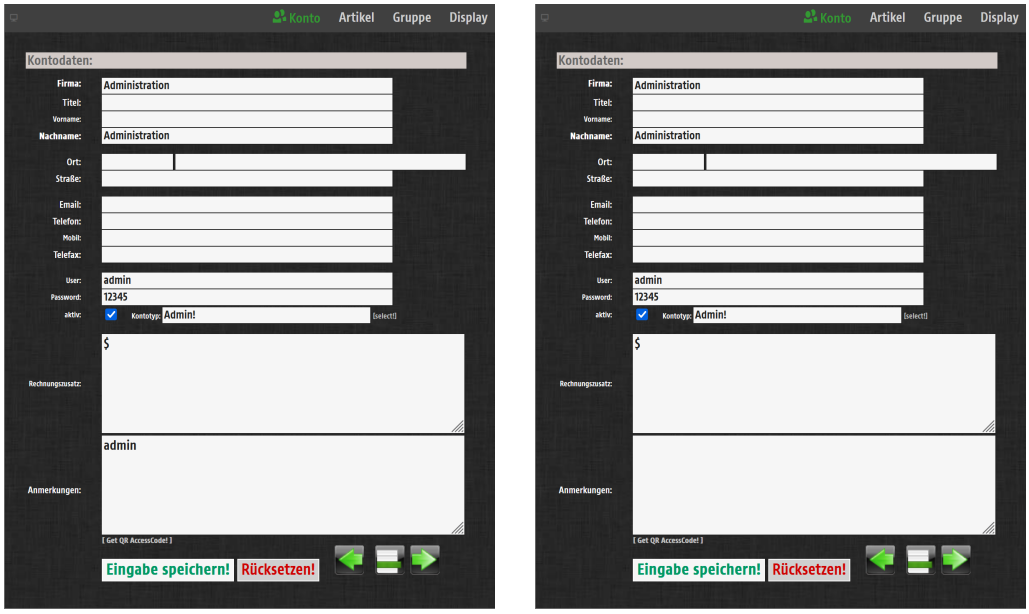
Given that 597 of the 1032 IDs in the allocated range returned an IP address (and that the IDs correlated with a list of active devices obtainable through another customer information disclosure, see section 13), we believe that assignment of ID-based redirects is not subject to customer choice but enabled for all cash registers that were not locked out ("GESPERRT!") by the system vendor.

We further found 223 of these devices actually accessible on the returned IP address (though sometimes on a different port).

5. User authentication with backdoor password

When accessing the web interface of the cash register, it presents itself with an on-screen keyboard captioned "Keycode?". The on-screen keyboard allows to input hexadecimal digits (0..9, a..f), which, when pressed, appear as characters in an editable (but slightly hidden) password input field (see Figure 2). After typing "admin" into the password input field of the cash register that we discovered at http://===REDACTED===:80/, we immediately gained access as administrative user. We subsequently successfully tested this keycode with a few other installations discovered through device enumeration (cf. section 4) and, further, found a demo instance of the cash register available at http://demo.cashit.at/ that used the same password as default.

We initially assumed that this is the password of a default administrative user that was not removed or disabled on several production systems by accident. As administrative users are able to read the credentials of all user accounts of an installation (see section 7), we were able to obtain a list of usernames



(a) (b)

Figure 3: User account details of user “admin” (a) before and (b) after the update.

and passwords used at http://===REDACTED===.posdev.online/. We discovered that none of the existing users in this installation had their password set to “admin”, but that a user with that username existed. Hence, we initially assumed that authentication worked with either username or password as the keycode. This assumption was further assured by the fact that authentication also worked for some of the other existing usernames. However, we could not identify a clear pattern of when a username would work as keycode and when it would not.

5.1 Vendor response

We reported the availability of access through the keycode “admin” to the vendor on 2022-12-07 and received a response on the same day. The vendor explained that it is up to the customer to change default passwords in their installations. Nevertheless, the vendor explained that they published a software update that would remove all default passwords from existing installations.

5.2 Re-evaluation

We fully agree that using strong passwords throughout their installation is up to the customers. However, we were still suspicious why some of the usernames worked as keycodes to access the cash register even if these values did not match any of the existing user passwords.

Further evaluation on the vendor’s demo installation at http://demo.cashit.at/ revealed that the update had removed the value of another field in the profile of the user “admin” (see Figure 3). The field “Anmerkungen” no longer contained the value “admin”. We validated our assumption by changing the value of that field to different values and testing if these values worked as keycodes. These

values indeed worked as alternate keycodes then. Thus, our initial assumption that keycode could be username or passwords was wrong. It had, in fact, been password or comment ("Anmerkungen" is German for comments, notes, or remarks).

6. Weak user authentication

The expected password complexity seems to be rather low. A typical keycode would consist of characters picked from the set of the 16 hexadecimal digits and be of the length of a typical PIN code (4-8 characters) to be typeable by waiters. Remote management currently seems to require a minimum password length of 5 characters.

While we acknowledge that remote management users could be configured with stronger passwords, we doubt that this would happen in practice.

6.1 Vendor response

We reported our observation to the vendor on 2022-12-07 and received a response on the same day. The vendor confirmed the minimum length of 5 characters for keycodes used through remote management. They explained that the system would be primarily used with waiter keys ("Kellnerschlüssel") that emit a 16 digit hexadecimal code.

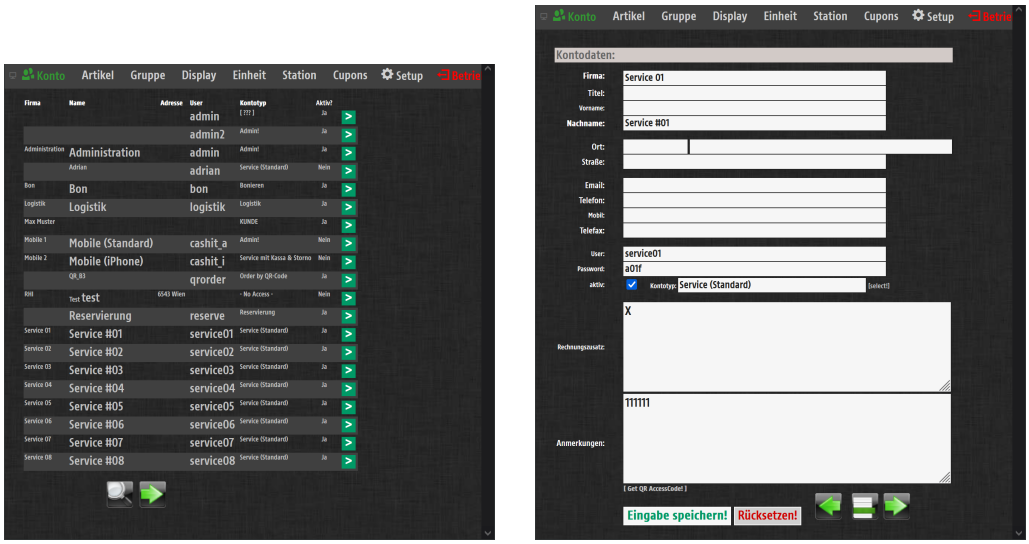
6.2 Re-evaluation

In the production system that we analyzed (thanks to permission by its owner), waiter keys are not used. Instead, they use short numeric PINs. Also, we wonder if waiter keys use a random or a sequential (and, thus, easily guessable) numbering scheme. However, the evaluated sample size does not permit any assumptions on the general use of waiter keys or the entropy of waiter key codes.

We initially intended to perform a brute-force attack to enumerate user accounts on that production system. However, before we managed to implement this, alternative approaches eventually lead to unauthenticated full data exfiltration and full access without the need to enumerate user keycodes through brute-force.

7. Read access to clear-text passwords by privileged users

Administrative users may list and edit all user accounts (see Figure 4). While this itself is not problematic, the form to edit individual users also displays the current clear-text password value of any user (note that this is also typically performed over an unencrypted HTTP transport). Additionally, it displays the comment field that can be used as alternative password (cf. section 5). Thus an administrative user could potentially impersonate any existing user.



(a) (b)

Figure 4: List of user accounts and user account details of user “service1” with revealed password field.

7.1 Vendor response

We reported this finding to the vendor on 2022-12-07 and received a response on the same day. The vendor confirmed this and explained that “this permission” can be restricted to the local network. Note that we are unsure if “this permission” referred to administrators’ capability of reading clear-text passwords or to the capability of editing user accounts in general.

8. Read access to clear-text passwords by unprivileged authenticated users

While remote management can be enabled/disabled for individual user groups, the production instance that we analyzed (with permission by its owner) also allows remote login with several unprivileged users. We believe that remote access would likely only be useful for administrative purposes and could potentially be restricted to only these user accounts. However, we are unsure if unprivileged users with remote access permission is intended by design or happened due to configuration mistakes.

Nevertheless, we evaluated capabilities of unprivileged authenticated users and found that any authenticated user has read and write access to the user database. This privilege escalation is possible due to API endpoints with broken session access control handling.

During our re-evaluation based on the vendor response, we took a closer look at the URL structure used within the application and noticed that the application URLs follow a specific pattern:

```
http://<HOST>/page/start.mv?[NOFRAMES]+<SID>+<N1>+<N2>+<N3>[+<MODULE>][+&<key>=<value>]
```

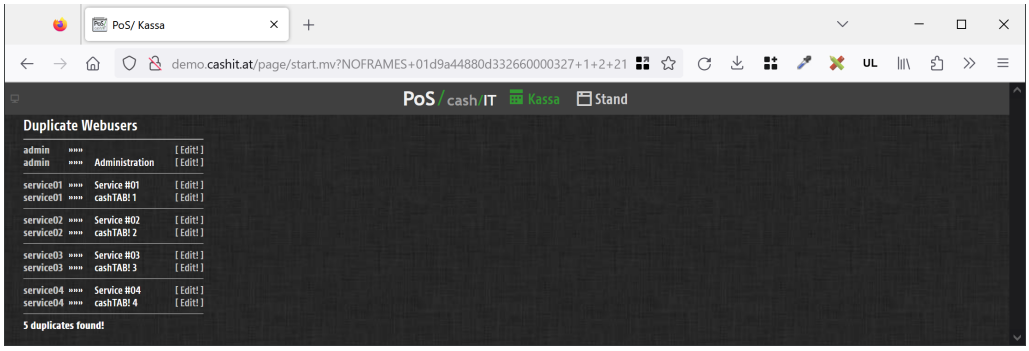


Figure 5: Page with ID 1/2/21 displays duplicate users with edit button to unprivileged user (reproduced on public demo instance to avoid disclosure of sensitive information from production instance).

SID clearly looked as if it was a session ID derived from date/time and IP address (i.e. it contains monotonically increasing hexadecimal numbers and some digits that only change for requests originating from different IP subnets). N1, N2, and N3 are numeric values and changed when displaying different pages. Particularly, N1 seems to be 1 for regular pages ("Betrieb") and 2 for maintenance pages ("Wartung"). N2 and N3 seem to identify the section and sub-section of the viewed page. All three numbers seem to be indices starting at one. MODULE seems to be a space-separated string further selecting a specific functionality of the sub-page. The query string is sometimes followed by additional key-value pairs added as query-string parameters.

Thus, by incrementing these three numbers, logged in users can easily enumerate all pages and sub-pages available (and accessible to them) within the application. This enabled us to quickly enumerate all pages accessible to an unprivileged user. By setting N3 to 21, we gained access to a page that lists duplicate users (see Figure 5). While we did not consider that potential information disclosure an issue, the "Edit!" links next to each duplicate user immediately raised our attention. Clicking that link opens a popup dialog (see Figure 6) that shows the full user record, again including the plain-text password (hidden behind a password input field) and "Memo1", which resembled the comment field that is also usable as password. This was possible while logged in as unprivileged user and even allowed access to the passwords of privileged users. Again, enumeration of the whole user database is possible by changing the numeric arguments in the URL.

The form also contained two more memo fields ("Memo2" and "Memo3"; which also seemed to work as keycodes when not empty) and a save button ("Datensatz speichern"). The save button permits the unprivileged user to even update any record.

This enables arbitrary users of the cash register to steal or change the passwords of any user (by using the memo fields potentially even unnoticed by the legitimate user). The form also permits to create new users. A malicious user could abuse this to raise their own permissions to (remote) administrator level.

8.1 Vendor response

We reported this finding to the vendor on 2023-06-27 (explaining that we would aim for responsible disclosure after 90 days with an option to extend if necessary) and received a response on the next day. The vendor confirmed that

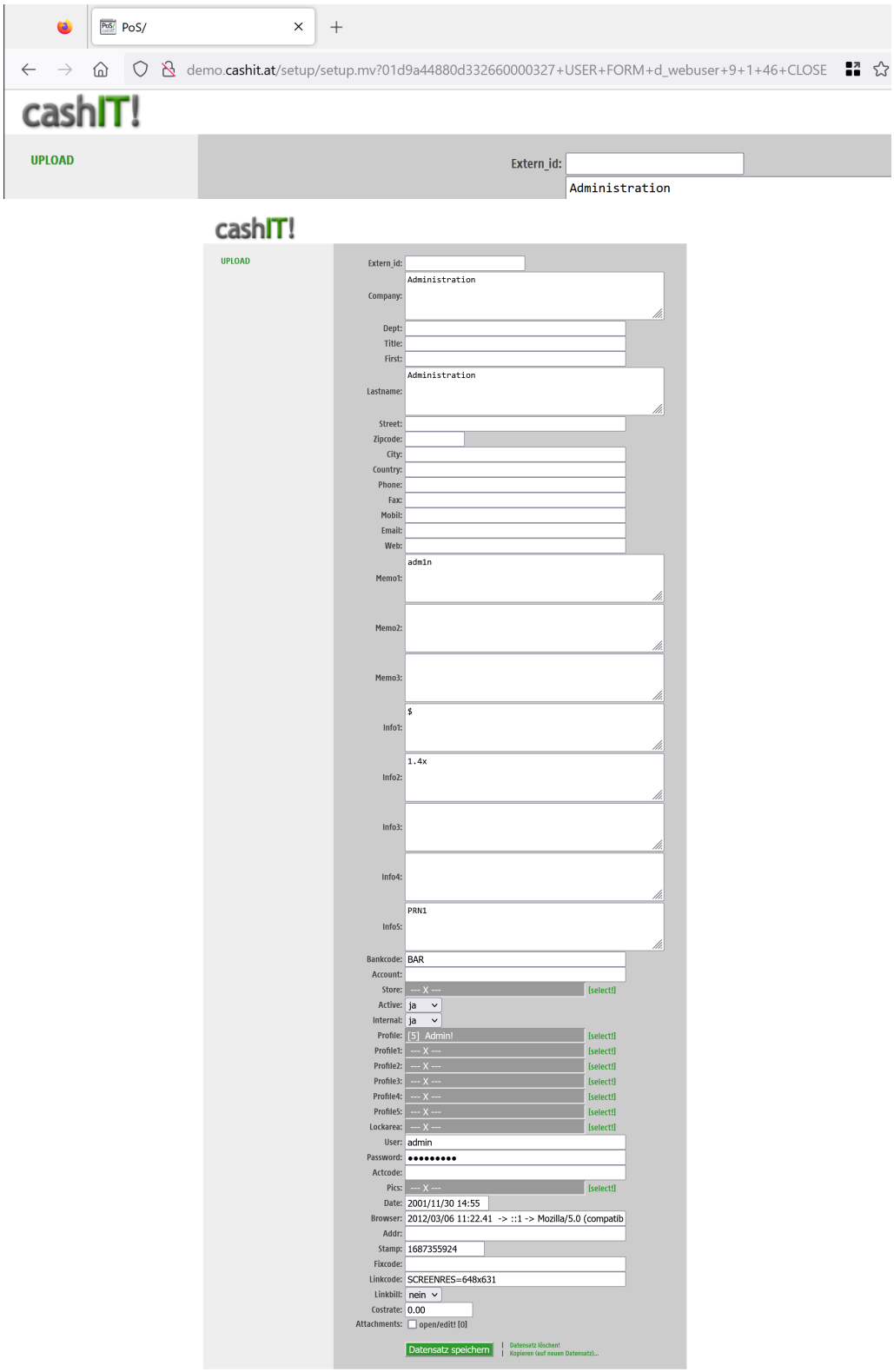


Figure 6: Setup page displays user details to unprivileged user (reproduced on public demo instance to avoid disclosure of sensitive information from production instance).

unprivileged users accidentally had access to obsolete maintenance routines and that these routines have been removed. In addition, the vendor explained that passwords can no longer be read through the user account details page (not even by privileged users, cf. section 7).

9. Remote code execution by privileged users through dedicated command injection endpoint

Browsing the administrative sections, we came across a link to restart the Windows print spooler from within the web interface:

```
http://<HOST>/page/start.mv?NOFRAMES+<SID>+2+26+44+PLUGIN+SETUP+DOSCMD+&dosline=spooler
```

We were triggered by the keywords "DOSCMD" and "dosline", which suggested that this endpoint may actually execute the passed command on the host system.

We verified this by changing the command "spooler" to the commands "whoami" and "hostname", which we expected to exist on a standard Windows system. After a longer load delay, this gave us the execution results shown in Figure 7 at the lower left end of the rendered page. Consequently, while some characters seem to be dropped from the resulting output, this endpoint, indeed, permits execution of arbitrary commands with the privileges of a user named "cmd". Further experiments revealed that this user has administrative privileges on the host machine, resulting in arbitrary remote code execution by privileged remote users. Combined with the issue in section 8, this results in arbitrary remote code execution for any authenticated remote user. A malicious user could abuse this to remotely take over the entire cash register host system.

9.1 Vendor response

We reported this finding to the vendor on 2023-06-27 (explaining that we would aim for responsible disclosure after 90 days with an option to extend if necessary) and received a response on the next day. The vendor confirmed that this endpoint exists and explained that additional filters have been put into place to restrict this functionality to "internal" admins.

(a) whoami (b) hostname

Figure 7: Command execution results displayed at the lower left end of the page (tested on production system after obtaining permission).

10. When everything started falling apart ...

From this point on, more and more things started to fall apart. Through the remote code execution exploit, we were able to exfiltrate the program code of the web application. We found the download location for updates on the vendor's update service—a simple unauthenticated HTTP URL.

```
http://upload.posdev.eu/_cashit_update.zip
```

This permitted us to compare with the latest version (03.A06rks 2023.02.37) of the application.

Through analysis of the application, we discovered dozens of endpoints that potentially allow bypassing authentication and/or authorization checking. This reports provides a few examples to illustrate the potential security impact. However, it is not a comprehensive list of all the potential issues we found.

10.1 Bypass of origin check permits remote management access for any user and unauthenticated station login

It seems that the authentication logic is split between local user authentication (when the web frontend is displayed in a web browser on the host system itself), local network authentication (when the web frontend is displayed on a device in a network with a prefix defined to be local), and remote network authentication (in all other cases). The authentication logic is tricked to believe that a request comes from the local machine if the hostname in the request URL is "localhost":

```
http://localhost/page/start.mv?...
```

This can be achieved by changing the "Host" header of the HTTP request to "localhost":

```
curl -H "Host: localhost" "http://<IP>/page/start.mv?..."
```

In this mode, the cash register accepts a station ID parameter, e.g. http://<IP>/page/start.mv?STATION1 to enable login without a keycode through a single button press. Also, user accounts without remote access permission can be used to login. Combining this with the previously discovered vulnerabilities, enables arbitrary remote code execution even with user accounts that do not have remote access capabilities (and potentially even without knowledge of any keycode).

Note that changing the host header with regular web browsers is not easily possible. Besides using curl (or similar tools), one could trick the browser to directly use http://localhost/ to access a remote cash register by, e.g.,

- adding an entry for localhost pointing to the remote IP address to `/etc/hosts` (though this may require additional effort to trick modern web browsers into accepting the address change for localhost) or
- setting up a NAT rule on the local machine that redirects traffic from localhost:80 to the external address.

CVE-2023-3654

Title	Bypass of origin check
Description	The system distinguishes between 3 classes of origin: localhost, local network and remote access. Remote attackers can become categorized as localhost by setting the HTTP Host header to "localhost". This enables login with accounts without remote access permissions and unauthenticated service station login.
Vendor	PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH
Product	cashIT! - serving solutions.
Homepage	https://www.cashit.at/
Vulnerable versions	$\leq$ 03.A06rks 2023.02.37
Type	CWE-346: Origin Validation Error
CVSS score	9.4
CVSS vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:L

10.2 Unauthenticated web proxy

We further found an endpoint that permits proxying of web requests (actually HTTP(S) GET requests) through the cash register host:

```
http://<HOST>/page/plugin/extcall/start.mv?NOFRAMES+0+0+0+0+0+<URL>
```

The endpoint does not require any authentication. This may permit access to systems normally only accessible from the cash register host or its surrounding network which are otherwise protected by a firewall, host-based authentication etc.

10.3 Unauthenticated full database dump through remote management interface

We found multiple endpoints that permit full database exfiltration without authentication. For instance,

```
http://<HOST>/page/plugin/syncdata/export.mv?<DATABASE>+GETFILE
```

where DATABASE is one of the system databases storing configuration values and operative data (such as system settings, user accounts, sales data, customer contact information, etc.) Besides the possibility to obtain user keycodes, this also revealed the sensitive credentials for the online signature creation unit for signing receipts according to the Austrian RKSV[3] and fields that may contain credentials for the backend system of the Austrian card payment acquirer Hobex (browsing the public demo instance revealed that these are credentials for the service at https://online.hobex.at/). Since the latter was not used by the production system for which we had permission to test on, we were unable to estimate the impact of access to such credentials.

[3] https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=20009390

CVE-2023-3655

Title	Unauthenticated remote database exfiltration
Description	The application includes unauthenticated, remotely accessible HTTP endpoints that allow exporting the entire database. This includes system settings, user accounts (with plaintext passwords), sales data, customer contact information, credentials for third-party services, etc.
Vendor	PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH
Product	cashIT! - serving solutions.
Homepage	https://www.cashit.at/
Vulnerable versions	$\leq$ 03.A06rks 2023.02.37
Type	CWE-749: Exposed Dangerous Method or Function
CVSS score	7.5
CVSS vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

10.4 Unauthenticated arbitrary remote code execution with administrative privileges

Similarly, the implementation of the API that we previously identified to allow remote code execution exposed an unauthenticated version of the endpoint by bypassing the login status check through injecting user-defined variables:

```
curl -d "login_status=OK&cmd_mode=2&dosline=whoami" -X POST \
     "http://>HOST>/page/plugin/datalink/start.mv?NOFRAMES+0+0+0+0+0+SETUP+DOSCMD+"
```

This eventually leads to unauthenticated arbitrary remote code execution with administrative privileges on the cash register host system. Again, a malicious user could abuse this to remotely take over the entire cash register host system without prior authentication.

CVE-2023-3656

Title	Unauthenticated remote code execution
Description	The application includes an unauthenticated, remotely accessible HTTP endpoint that allow executing arbitrary code as administrative user on the underlying Windows server.
Vendor	PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH
Product	cashIT! - serving solutions.
Homepage	https://www.cashit.at/
Vulnerable versions	$\leq$ 03.A06rks 2023.02.37
Type	CWE-749: Exposed Dangerous Method or Function
CVSS score	9.8
CVSS vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H

10.5 Vendor response

We reported these findings to the vendor on 2023-06-27 (explaining that we would aim for responsible disclosure after 90 days with an option to extend if

necessary) and received a response on the next day. The vendor explained that these issues will be investigated in a timely manner and will be fixed if deemed necessary. Despite approaching the vendor again (partially with assistance of CERT.at), we did not receive any information about mitigation plans or timelines.

With regard to remote access in general, the vendor explained that remote access through the cash register web interface is an optional functionality provided to customers free of charge, and that is only available if explicitly activated through appropriate port-forwarding by the customer.

10.6 Access to cloud backups

Cloud backups, when performed by the cash register, are stored to an FTP server at ftp://update.posdev.eu/. The credentials for the upload account (username "backup") are hard-coded into the application source code and, thus, the same for all installations. Files are named "PosNNNNN.zip" and contain the data files (all databases, credentials of the online signature unit, signed receipt history, etc.) of the installation. A malicious user may be able to manipulate such backups by replacing the (unversioned?) backup file with a modified version on the backup server.

Interestingly, cloud backups seem to be performed regardless of the cloud backup status indicated in the web user interface of the cash register.

Obviously, we asked ourselves how these cloud backups could be downloaded. We immediately guessed that they would be available at the FTP server for download as well. This assumption turned out to be wrong though. Also, http://update.posdev.eu/PosNNNNN.zip did not work. Given that the username of the FTP server is backup, we also tried http://update.posdev.eu/backup/PosNNNNN.zip and were finally successful. Hence, backups of all cash registers are available without user authentication at an easily guessable location (verified by downloading the backups of ===REDACTED===). This permits full disclosure of sensitive configuration (including user accounts), sales data, and potentially even contact data of natural persons (functionality to store such data exists; however, it was not actively used in the data that we were permitted to use for verification purposes).

10.6.1 Vendor response

We reported this finding to the vendor on 2023-06-27 (explaining that we would aim for responsible disclosure after 90 days with an option to extend if necessary) and received a response on the next day. The vendor confirmed that cloud backups were publicly available at the mentioned endpoint and that access to them has been disabled until a different access logic is implemented.

11. Outdated software versions

Several outdated versions of software and device firmware seem to be used in the deployed infrastructure that, given the list of products on the vendor's website, looks as if it was part of the overall cash register bundle.

11.1 Miva Empresa runtime framework

The web application is based on the MivaScript markup/scripting language[4]. All cash registers (and several backend services) use the Miva Empresa NT 3.97 interpreter released in August 2003 and adapted with several patches.

This version is no longer maintained. The current MivaScript version is 5.38 (released in January 2023). However, this version is no longer compatible with code written for version 3.97.

11.2 Windows Server and IIS

Systems seem to run rather old and potentially out-of-support versions of Windows Server and IIS. From the discovered systems, 95 were running IIS 7.5, 1 was running IIS 8.5, and 125 were running IIS 10.0.

11.3 MikroTik RouterOS

Several installations use a MikroTik hEX PoE lite[5] (RB750UPr2) as gateway. Some of them seem to use outdated firmware versions potentially susceptible to CVE-2018-14847[6].

11.3.1 Vendor response

We reported these findings to the vendor on 2023-06-27. We received a clarifying response on 2023-08-22 where the vendor explained that updates of the cashIT! cash register software are provided free-of-charge and that it is up to the customer to install these software updates. The vendor further explained that the same would apply to software on customer-side routers and network components.

12. Use of deployment-independent default admin passwords

The analysis of the production system revealed that standard passwords used for the Windows user account "cmd" and the MikroTik hEX PoE lite. The structure of both passwords suggests that these are default credentials used across all installations. Further tests would be necessary to confirm this.

13. Information disclosure and potential unprotected write access in customer and license management service

A publicly accessible dashboard at http://update.posdev.eu/License/ exposes data about all(?) installations of the cashIT! cash register including data about

[4]https://www.mivascript.com/
[5]https://mikrotik.com/product/RB750UPr2
[6]https://nvd.nist.gov/vuln/detail/cve-2018-14847

the customers, subscription status (in support, end-of-support, locked), cloud backup status, etc. The user interface suggests that data can also be edited without authentication (though we did not verify this). Moreover, the dashboard pointed to another publicly accessible dashboard by the retailer that reveals information about customer payment history and potentially other business data that we believe should typically not be made publicly available).

Appendix A. Version distribution

Figure 8 shows the version distribution at the time of our evaluation. Table 1 lists the observed counts for each version/revision.

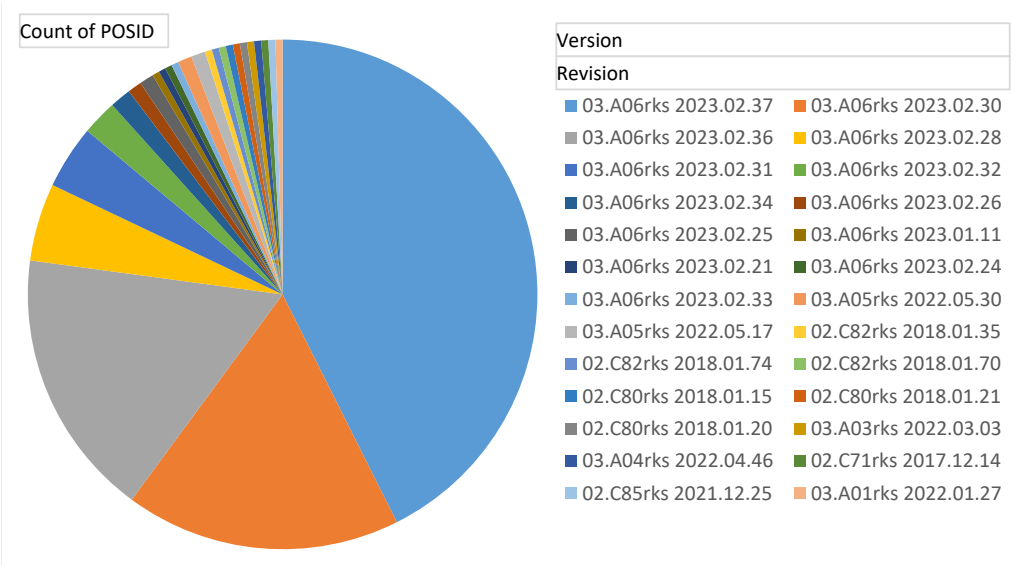


Figure 8: Version distribution (based on data from 2023-06-21)

Table 1: Version distribution (based on data from 2023-06-21)

Version	Revision	Count
03.A06rks	2023.02.37	95
03.A06rks	2023.02.36	38
03.A06rks	2023.02.34	3
03.A06rks	2023.02.33	1
03.A06rks	2023.02.32	5
03.A06rks	2023.02.31	9
03.A06rks	2023.02.30	39
03.A06rks	2023.02.28	11
03.A06rks	2023.02.26	2
03.A06rks	2023.02.25	2
03.A06rks	2023.02.24	1
03.A06rks	2023.02.21	1
03.A06rks	2023.01.11	1

Version	Revision	Count
03.A05rks	2022.05.30	2
03.A05rks	2022.05.17	2
03.A04rks	2022.04.46	1
03.A03rks	2022.03.03	1
03.A01rks	2022.01.27	1
02.C85rks	2021.12.25	1
02.C82rks	2018.01.35	1
02.C82rks	2018.01.74	1
02.C82rks	2018.01.70	1
02.C80rks	2018.01.15	1
02.C80rks	2018.01.21	1
02.C80rks	2018.01.20	1
02.C71rks	2017.12.14	1

Appendix B. Availability of cloud backups

Figure 9 shows the age distribution of cloud backups publicily available for download at the time of our evaluation. Table 2 lists the observed counts for each bin.

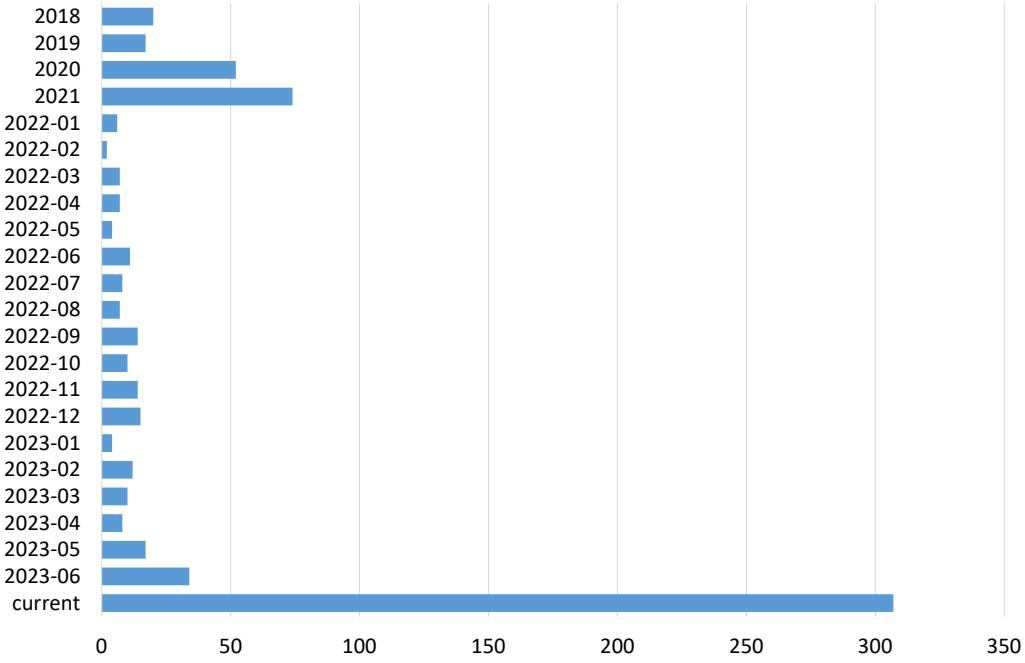


Figure 9: Cloud backup distribution (based on data from 2023-06-24)

Table 2: Cloud backup distribution (based on data from 2023-06-24)

Backup Creation Time	Count
current (2023-06-24±12h)	307
2023-06	34
2023-05	17
2023-04	8
2023-03	10
2023-02	12
2023-01	4
2022-12	15
2022-11	14
2022-10	10
2022-09	14
2022-08	7

Backup Creation Time	Count
2022-07	8
2022-06	11
2022-05	4
2022-04	7
2022-03	7
2022-02	2
2022-01	6
2021	74
2020	52
2019	17
2018	20
before 2018	0

Appendix C. Vendor communication

C.1 Vendor notification via e-mail on 2022-12-07

```
Subject: Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Wed, 7 Dec 2022 16:12:41 +0100
From: Michael Roland <===REDACTED===@ins.jku.at>
To: support@cashit.at, support@posdev.eu, reports@cert.at
CC: Tobias Höller <===REDACTED===@ins.jku.at>

Sehr geehrte Damen und Herren,

im Zuge eine routinemäßigen Analyse unseres universitätsinternen Netzwerks sind
wir durch Zufall auf eines Ihrer Kassensysteme (PoS/ cash/IT Kassa, Version
03.A03rks (2022.03.15), verwendet ===REDACTED=== am Uni-Campus) gestoßen. Dabei
mussten wir feststellen, dass dieses System frei zugänglich im Internet hängt
und ein Login mit dem Keycode "admin" möglich ist. Wir haben diesen Umstand
auch umgehend dem zuständigen Betreiber, ===REDACTED===, Ansprechpartner
===REDACTED===, mitgeteilt.

Während wir ursprünglich davon ausgegangen sind, dass das Kassensystem
versehentlich über eine öffentliche IP-Adresse erreichbar ist, hat unsere weitere
Recherche ergeben, dass Sie dafür explizit einen Service anbieten, der ihre
Kassensystem unter Hostnamen der Form *.posdev.online erreichbar macht. Wir
nehmen daher an, dass dies auch so für Ihr Produkt vorgesehen ist.

Wir haben jedoch diesbezüglich einige Sicherheitsbedenken die potentielle
Schwachstellen in Ihrem System darstellen und über die wir Sie informieren
möchten:

- Die Kassensysteme nutzen keine TLS-Verschlüsselung. Beim Login werden die
  Zugangsdaten daher im Klartext übertragen.

- Zumindest einige Ihrer Kunden erlauben den Login mit dem Keycode "admin".

- Über den Admin-Zugang könne auch die Keycodes aller am System gespeicherten
  Nutzer ausgelesen werden, weil diese im Klartext an den Client (Webbrowser)
  gesendet werden.

- Neben dem Zugang über den Keycode "admin", scheint uns auch die
  Passwortkomplexität zu gering um einen unautorisierten Zugriff ausreichend
  abzuwehren (16 mögliche Zeichen, Keycode suggieriert die Nutzung von PINs ->
  gewählte Zeichenlängen werden sehr begrenzt sein). Uns ist auch nicht ganz
  klar, wofür Benutzernamen in Ihrem System gespeichert werden, wenn für den
  Login lediglich der Keycode (= Passwort) benötigt wird.

- Dies sehen wir insbesondere deshalb problematisch, weil die Kassensysteme
  sehr leicht über die Adressen http://*.posdev.online/ auffindbar sind.
  Dieser Endpunkt ermöglicht ein Auffinden anhand von Teilen der gewählten
  Kassennamen, also ist z.B. ===REDACTED=== unter
  http://===REDACTED===.posdev.online/ aber auch unter http://hwer.posdev.online/
  erreichbar. Damit ist eine relativ rasche Enumeration aller Kassensysteme,
  welche diesen Service nutzen, möglich.

Wir sehen aufgrund obiger Sicherheitsbedenken kritische Abrechnungsdaten vieler
```

Ihrer Kunden gefährdet und würden Sie um rasche Reaktion und ggf. Information Ihrer Kunden bitten. Parallel erfolgt mit dieser Nachricht auch eine Meldung an CERT.at über diesen Sicherheitsvorfall.

Wir stehen Ihnen gerne für Rückfragen zur Verfügung.

Mit freundlichen Grüßen
Tobias Höller und Michael Roland

--
Dr. Michael Roland
Post-doc Researcher
Institute of Networks and Security

Johannes Kepler University Linz
===(SIGNATURE TRUNCATED)===

C.2 Response via e-mail on 2022-12-07

Subject: Re: Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Wed, 7 Dec 2022 22:01:58 +0100
From: ===REDACTED=== <===REDACTED===@posdev.eu>
To: Michael Roland <===REDACTED===@ins.jku.at>, reports@cert.at
CC: Tobias Höller <===REDACTED===@ins.jku.at>

Danke für ihre Ausführungen, unsere Anmerkungen finden Sie unten!

Herzliche Grüße
===REDACTED===

_____
PoS/
Dienstleistung,
Entwicklung &
Vertrieb GmbH

Von: Michael Roland <===REDACTED===@ins.jku.at>
An: <support@cashit.at>, <support@posdev.eu>, <reports@cert.at>
Kopie: Tobias Höller <===REDACTED===@ins.jku.at>
Gesendet: 07.12.2022 16:12
Betreff: Sicherheitsprobleme im Kassensystem PoS/ cash/IT

> Sehr geehrte Damen und Herren,
>
> im Zuge eine routinemäßigen Analyse unseres universitätsinternen Netzwerks sind wir durch Zufall auf eines Ihrer Kassensysteme (PoS/cash/IT Kassa, Version 03.A03rks (2022.03.15), verwendet ===REDACTED=== am Uni-Campus) gestoßen. Dabei mussten wir feststellen, dass dieses System frei zugänglich im Internet hängt und ein Login mit dem Keycode "admin" möglich ist. Wir haben diesen Umstand auch umgehend dem zuständigen Betreiber, ===REDACTED===, Ansprechpartner ===REDACTED===, mitgeteilt.
>
> Während wir ursprünglich davon ausgegangen sind, dass das Kassensystem versehentlich über eine öffentliche IP-Adresse erreichbar ist, hat unsere weitere Recherche ergeben, dass Sie dafür explizit einen Service anbieten, der

ihre Kassensystem unter Hostnamen der Form *.posdev.online erreichbar macht. Wir nehmen daher an, dass dies auch so für Ihr Produkt vorgesehen ist.

Wir haben jedoch diesbezüglich einige Sicherheitsbedenken die potentielle Schwachstellen in Ihrem System darstellen und über die wir Sie informieren möchten:

- Die Kassensysteme nutzen keine TLS-Verschlüsselung. Beim Login werden die Zugangsdaten daher im Klartext übertragen.

-> Die Software basiert auf einem Webserver welcher optional auch eine TLS-Verschlüsslung unterstützt, Wir bieten unseren Kunden (entgeltlich) auch diese Möglichkeit - entweder mit privaten oder öffentlichen Zertifikaten - zwingen unsere Kunden aber nicht diese mit Kosten verbundene Variante zu nutzen. Da die Daten in der Regel keine DSGVO-Relevanz haben kann der Kunde das selbst entscheiden. Die Nutzung ist außerdem nur möglich wenn seitens des Kunden AKTIV ein entsprechendes Port-Forwarding veranlasst wurde.

- Zumindest einige Ihrer Kunden erlauben den Login mit dem Keycode "admin".

-> Grundsätzlich obliegt es dem Kunden dieses "Standard-Passwort" zu ändern - dies kann der Kunde natürlich jederzeit selbst ändern. Wir haben aber diesen berechtigten Hinweis unmittelbar nach Erhalt aufgenommen und seit Version .16 (also unmittelbar nach ihrer initialen Prüfung) im Update eine Routine integriert, welche alle Standard-Passwörter entfernt. Im gegenständlichen Fall hatte der Kunde das verfügbare und aktiv angebotene Update aber nicht installiert (das Update ist natürlich kostenlos). Die Installation unserer Updates bedeutet für unsere Kunden nur einen einzigen Knopfdruck auf den am Einstiegsschirm verfügbaren Button "Update!" und ist in weniger als 10 Sekunden abgeschlossen. Nach unserer heutigen Prüfung haben wir festgestellt, dass das Update nicht ausgeführt wurde - daher haben wir das ausnahmsweise für den Kunden erledigt.

- Über den Admin-Zugang könne auch die Keycodes aller am System gespeicherten Nutzer ausgelesen werden, weil diese im Klartext an den Client (Webbrowser) gesendet werden.

-> Ja, der ADMIN-User hat derzeit diese Berechtigung. Es kann das Profil aber so konfiguriert werden, dass diese Funktion nur im lokalen Netz vor Ort verfügbar ist.

- Neben dem Zugang über den Keycode "admin", scheint uns auch die Passwortkomplexität zu gering um einen unautorisierten Zugriff ausreichend abzuwehren (16 mögliche Zeichen, Keycode suggieriert die Nutzung von PINs -> gewählte Zeichenlängen werden sehr begrenzt sein). Uns ist auch nicht ganz klar, wofür Benutzernamen in Ihrem System gespeichert werden, wenn für den Login lediglich der Keycode (= Passwort) benötigt wird.

-> Wie die meisten Kassensysteme beruht auch unser System auf der Logik von "Kellnerschlüsseln" welche durch einen Keycode einen Benutzer referenzieren. Die Mindestlänge dieses Keycodes - und damit auch für den Keycode eines Außenzugriffes kann vom Benutzer selbst festgelegt werden - die nicht unterschreitbare Mindestlänge sind 5 Zeichen. Die Keycodes eines Kellnerschlüssels sind üblicherweise 16 Stellen (hex). Das Keypad dient zur Nutzung als Schlüsselersatz - die Zeichen sind jedoch nicht auf diese

vorgeschlagenen Zahlen und Buchstaben begrenzt (sonst hätte "admin" nicht funktioniert).

    - Dies sehen wir insbesondere deshalb problematisch, weil die Kassensysteme
      sehr leicht über die Adressen http://*.posdev.online/ auffindbar sind.
      Dieser Endpunkt ermöglicht ein Auffinden anhand von Teilen der gewählten
      Kassennamen, also ist z.B. ===REDACTED=== unter
      http://===REDACTED===.posdev.online/ aber auch unter
      http://hwer.posdev.online/ erreichbar. Damit ist eine relativ rasche
      Enumeration aller Kassensysteme, welche diesen Service nutzen, möglich.

-> Diese Anregung haben wir aufgenommen und bereits umgesetzt - ab sofort kann der Kunde -nur mehr mit einem dem selbst hinterlegtem Keyword (z.B. '===REDACTED===') zugreifen. Eine darüber hinausgehende Auflösung erfolgt nicht mehr. Der Kunde kann darüber hinaus den Fernzugriff jederzeit selbst im Setup des Programms deaktivieren (diese Funktion stand auch bisher bereits zur Verfügung) - unabhängig von der Freigabe auf seiner Firewall. Unser Service ersetzt hier lediglich ein oft kostenpflichtiges DynDNS-Service, welches wir unseren Kunden ausschließlich auf deren Wunsch kostenlos zur Verfügung stellen um die Kassa auch bei dynamischen IP-Adressen erreichen zu können (natürlich nur, sofern er in seinem Netz ein Port-Forwarding eingerichtet hat).

    Wir sehen aufgrund obiger Sicherheitsbedenken kritische Abrechnungsdaten
    vieler Ihrer Kunden gefährdet und würden Sie um rasche Reaktion und ggf.
    Information Ihrer Kunden bitten. Parallel erfolgt mit dieser Nachricht
    auch eine Meldung an CERT.at über diesen Sicherheitsvorfall.

-> Wir planen eine Information an unsere Kunden in der wir auf die von ihnen dargestellten Sicherheitsaspekte eingehen und die entsprechenden Konfigurationsoptionen in Erinnerung rufen.

    Wir stehen Ihnen gerne für Rückfragen zur Verfügung.

    Mit freundlichen Grüßen
    Tobias Höller und Michael Roland

    --
    Dr. Michael Roland
    Post-doc Researcher
    Institute of Networks and Security

    Johannes Kepler University Linz
    ===(SIGNATURE TRUNCATED)===

C.3 Vendor notification via e-mail on 2023-06-27

Subject: Re: Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Tue, 27 Jun 2023 23:21:23 +0200
From: Michael Roland <===REDACTED===@ins.jku.at>
To: ===REDACTED=== <===REDACTED===@posdev.eu>, reports@cert.at
CC: Tobias Höller <===REDACTED===@ins.jku.at>,
    René Mayrhofer <===REDACTED===@ins.jku.at>

Sehr geehrter Herr ===REDACTED===,

vielen Dank für Ihre rasche Rückmeldung im Dezember und die Behebung der damals aufgezeigten Probleme bzw. Entkräftigung unserer Annahmen.

Wir hatten eigentlich geplant unsere Erkenntnisse nach erfolgreicher Behebung, als Teil unseres Forschungsauftrags, zu publizieren. Dazu haben wir nun, mit Erlaubnis des Kasseninhabers auf unserem Campus, einige weiterführende Analysen durchgeführt um sicherzustellen, dass das Kassensystem gegen die von uns im Dezember vermuteten Schwachstellen ausreichend abgesichert ist. Leider haben wir dabei eine Reihe weiterer mutmaßlicher Schwachstellen entdeckt, über die wir sie hiermit informieren möchten.

Zunächst mussten wir feststellen, dass neben priviligierten Benutzern auch unprivilegierte Nutzer mit Fernzugriff die Passwörter aller Nutzer auslesen, modifizieren, und sogar neue Nutzerkonten anlegen können. Damit kann ein nicht priviligierter Nutzer seine Berechtigungen zu denen eines Admin ausdehnen. Als Administrator steht ein Endpunkt zur Verfügung, über den ein entfernter Angreifer beliebiger Code mit Administratorrechten auf dem Windows-System der Registrierkassa ausführen kann.

Damit war es uns mögich die Kassenapplikation weiter zu analysieren. Unsere Analyse bezieht sich auf Version 03.A06rks 2023.02.37, welche zum Zeitpunkt der Analyse die aktuellste verfügbare Version gewesen ist. Wir haben dadurch eine Unzahl weiterer möglicher Einfallstore, durch mangelde oder fehlende Benutzerauthentifizierung/-autorisierungsprüfung, gefunden. Unter anderem ist es damit möglich komplett ohne Authentifizierung beliebigen Programmcode mit Administratorrechten auf dem Hostsystem der Registrierkasse auszuführen. Ebenso ist es möglich die gesamten Datenbanken eines Kassensystems auszulesen (und ev. sogar zu verändern, wenngleich wir dies nicht getestet haben um das getestete Produktivsystem nicht zu verändern), ebenso komplett ohne Authentifizierung.

Weiters haben wir entdeckt, dass möglicherweise alle Kassensysteme tägliche Cloud-Backups auf die Server ihres Unternehmens (update.posdev.eu) machen, sogar dann, wenn das Kassensystem anzeigt, dass vor mehreren Jahren zuletzt ein Cloud-Backup durchgeführt wurde. Wir konnten dies jedoch vorläufig lediglich anhand der Ergebnisse für jene Kassensysteme verifizieren, für die wir eine Erlaubnis durch den Inhaber eingeholt hatten. Besonders kritisch sehen wir dabei, dass diese Backups nicht nur unverschlüsselt zu ihren Servern (Standort und damit verknüpfter Rechtsraum unbekannt) übertragen werden, sondern auch ohne jegliche Authentifizierung von überall frei lesbar von diesen heruntergeladen werden können.

Diese Backups enthalten detaillierte Umsatzdaten ihrer Kunden, ggf. persönliche Benutzeraccounts von (ehemaligen) Mitarbeiter*innen ihrer Kunden, ggf. Kundenkontakte ihrer Kunden (wobei die von uns untersuchten Systeme diese Funktion nicht nutzten), und weitere sensible Daten. Insbesondere ist uns dabei aufgefallen, dass auch die unternehmensspezifischen Zugangsdaten zur Online-Signaturerstellungseinheit der A-Trust in den Backups enthalten sind. Ebenso sind, sofern Hobex verwendet wird, die unternehmensspezifischen Zugangsdaten zu deren Backend https://online.hobex.at/ darin enthalten. Die daraus resultierenden Implikationen für ihre Kunden sind für uns derzeit noch nicht abschätzbar.

Daneben haben wir noch eine Reihe weiterer potentieller Probleme entdeckt. Z.B. scheinen Sie, sowie ihr Vertriebspartner, öffentlich abrufbare Dashboards zu betreiben, mit denen weitere Details ihrer (auch ehemaligen) Kunden (u.a. Kontakt- und Abrechnungsdaten) und über deren Kassensysteminstallationen frei

lesbar im Internet stehen. Es sieht auch so aus, als wären diese Daten frei editierbar, von einem Test haben wir jedoch aus ethischen Bedenken abgesehen. Auch die in Bundles mit ausgelieferten Systeme scheinen teils veraltete Software und Firmwareversionen einzusetzten (u.a. MikroTik Router mit bekannten kritischen Schwachstellen) und generische Standardpasswörter zu verwenden (u.a. MikroTik und auch der Windows-Rechner des Kassensystems).

Eine ausführliche Zusammenfassung unserer Erkenntnisse finden Sie im Anhang.

Bitte teilen Sie uns mit, bis wann Sie planen die betroffenen Unternehmen und Personen zu informieren (neben ihren Kunden vermutlich auch Hobex, A-Trust und das Finanzamt) und mutmaßliche Schwachstellen zu beheben bzw. ggf. zu entkräften. Wir haben die Publikation der bereits gemeldeten Schwachstellen aufgrund der neuen Erkenntnisse vorerst auf Eis gelegt, planen jedoch weiterhin die Veröffentlichung unserer gesammelten Erkenntnisse nach einer angemessenen Frist (üblich sind hier zumindest 90 Tage). Weiters möchten wir CVE-Nummern für einige der gefunden Schwachstellen beantragen. Haben Sie hierfür bereits einen Prozess? Falls nein, kann CERT.at hier weiterhelfen? Oder sollen wir uns direkt an MITRE (als CNA-LR) wenden?

Wir würden uns über eine rasche Rückmeldung freuen und stehen für weitere Detailfragen gerne zur Verfügung.

Mit freundlichen Grüßen
Michael Roland und Tobias Höller

===(QUOTED E-MAIL TRAIL TRUNCATED)===

C.4 Response via e-mail on 2023-06-28

Verbatim vendor response redacted from public disclosure.

C.5 Clarification request via e-mail on 2023-06-28

Subject: Re: Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Wed, 28 Jun 2023 15:32:17 +0200
From: Michael Roland <===REDACTED===@ins.jku.at>
To: ===REDACTED=== <===REDACTED===@posdev.eu>
CC: Tobias Höller <===REDACTED===@ins.jku.at>,
    René Mayrhofer <===REDACTED===@ins.jku.at>, reports@cert.at

Sehr geehrter Herr ===REDACTED===,

vielen Danke für die erneut sehr rasche Rückmeldung.

Wir können verstehen, dass unsere Meldung bei Ihnen einigen Aufwand verursacht hat. Dennoch würden wir Sie um die Beantwortung folgender Fragen bitten:

1. Haben Sie einen Prozess für die Beantragung von CVE-Nummern für die eingemeldeten Schwachstellen? Falls nein, würden wir uns selbst um die Beantragung kümmern.

2. Können Sie uns bitte bestätigen, dass Sie die vorgesehene Meldung lt. DSGVO

an die Datenschutzbehörde durchgeführt haben bzw. bescheid geben, sobald diese erfolgt ist?

3. Planen Sie, die betroffenen Unternehmen und Personen (neben Ihren Kunden vermutlich Hobex, A-Trust und das Finanzamt) zu informieren?

Wir würden Sie zudem bitten, uns zu informieren, sobald wir das System nochmals auf Wirksamkeit der umgesetzten Maßnahmen überprüfen können.

Mit freundlichen Grüßen
Michael Roland und Tobias Höller

===(QUOTED E-MAIL TRAIL TRUNCATED)===

C.6 CERT.at notification via e-mail on 2023-07-03

Subject: Re: Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Mon, 3 Jul 2023 16:41:29 +0200
From: Michael Roland <===REDACTED===@ins.jku.at>
To: reports@cert.at
CC: Tobias Höller <===REDACTED===@ins.jku.at>,
René Mayrhofer <===REDACTED===@ins.jku.at>

Sehr geehrtes CERT.at Team,
Sehr geehrter Herr ===REDACTED===,

nachdem wir bzgl. der Beantragung von CVE-Nummern für die Schwachstellen keine Rückmeldung von ===REDACTED=== (PoS/ Dienstleistung, Entwicklung & Vertrieb GmbH) erhalten haben, würden wir den Prozess nun gerne über Sie anstoßen. Können wir direkt über CERT.at CVE-Nummern beantragen?

Weiters sind wir uns nicht sicher, ob die Firma weiter auf unsere Anfragen reagiert (bisher kam innerhalb weniger Stunden eine Rückmeldung, auf die letzte Nachricht (siehe unten) kam jedoch keine Antwort mehr). Die bisherige Reaktion des Softwareherstellers lässt uns jedoch aktuell daran zweifeln, dass Verständnis für eine nachhaltige Beseitigung der Schwachstellen besteht. So wurde beispielsweise die Remote-Code-Execution lediglich hinter einem zusätzlichen GET-Parameter versteckt. Wir würden daher auf das Angebot von ===REDACTED=== vom Dezember zurück kommen, ggf. beim Hersteller zu urgieren.

Wir planen weiterhin eine Veröffentlichung unserer Erkenntnisse nach Ablauf von 90 Tagen. Sollte der Hersteller die aktuelle Strategie fortsetzen, würde die Veröffentlichung jedoch zu einer einer massiven Bedrohung für die Kunden des CashIT!-Produkts werden (sofern nicht auch schon andere die Schwachstellen gefunden haben und aktiv ausnutzen).

Mit freundlichen Grüßen
Michael Roland und Tobias Höller

===(QUOTED E-MAIL TRAIL TRUNCATED)===

C.7 CERT.at response via e-mail on 2023-07-03

```
Subject: [CERT.at #1597859] Sicherheitsprobleme im Kassensystem PoS/ cash/IT
Date: Mon, 03 Jul 2023 17:33:20 +0200
From: ===REDACTED=== via RT <team@cert.at>
To: ===REDACTED===@posdev.eu, ===REDACTED===@ins.jku.at

On Mon, 03 Jul 2023 16:41:51 +0200, ===REDACTED===@ins.jku.at wrote:
> Sehr geehrtes CERT.at Team,
> Sehr geehrter Herr ===REDACTED===,

Sehr geehrter Herr Michael Roland,
Sehr geehrter Herr Tobias Höller,

vielen Dank, dass Sie uns in dem Fall am Laufenden halten und miteinbeziehen.

===(QUOTED E-MAIL TEXT REMOVED)===

Wir werden die Sache intern absprechen und sich alsbald bei Ihnen melden um so
schnell wie möglich zu einem zufriedenstellenden Ergebnis zu kommen.

> Mit freundlichen Grüßen
> Michael Roland und Tobias Höller

===(QUOTED E-MAIL TRAIL TRUNCATED)===

Mit freundlichen Grüßen,
===REDACTED===

--
// CERT Austria <team@cert.at> - T: +43 1 5056416 78
// CERT.at GmbH - https://www.cert.at/
// Firmenbuchnummer 561772k, HG Wien
```

C.8 Vendor response via e-mail on 2023-07-03

Verbatim vendor response redacted from public disclosure.

C.9 Vendor notification via e-mail on 2023-08-21

```
Subject: Re: Sicherheitsprobleme im Kassensystem PoS/ cash/IT [CERT.at #1597859]
Date: Mon, 21 Aug 2023 16:47:42 +0200
From: Michael Roland <===REDACTED===@ins.jku.at>
To: ===REDACTED=== <===REDACTED===@posdev.eu>
CC: team@cert.at, Tobias Höller <===REDACTED===@ins.jku.at>,
    René Mayrhofer <===REDACTED===@ins.jku.at>

Sehr geehrter Herr ===REDACTED===,

wir möchten Sie hiermit über unsere geplanten Aktivitäten zur Veröffentlichung
der von uns gemeldeten Schwachstellen im Sinne der Responsible Disclosure am
Laufenden halten.
```

```
Für die drei wesentlichsten Schwachstellen wurden CVE-Nummern reserviert (siehe
aktualisierter Report im Anhang):

CVE-2023-3654: Bypass of origin check
CVE-2023-3655: Unauthenticated remote database exfiltration
CVE-2023-3656: Unauthenticated remote code execution

Diese CVE-Nummern wurden mit Unterstüzung von CERT.at über eine Partner CNA
reserviert und sind aktuell noch gesperrt (d.h. man kann noch keine Information
über die betroffenen Sicherheitslücken einsehen).

Desweiteren haben wir einen Vortragsvorschlag für die IKT-Sicherheitskonferenz [1]
eingereicht. Dieser wurde nun akzeptiert und ist für 3. Oktober eingeplant. In
dem Vortrag werden wir über die gefundenen Schwachstellen und Probleme öffentlich
sprechen. Ab diesem Tag werden auch die CVEs und unser Report öffentlich abrufbar
sein.

[1] https://seminar.bundesheer.at/

Mit Freundlichen Grüßen
Michael Roland & Tobias Höller

--
Dr. Michael Roland
Post-doc Researcher
Institute of Networks and Security

*Johannes Kepler University Linz*
===(SIGNATURE TRUNCATED)===
```

C.10 Vendor response via e-mail on 2023-08-22

Verbatim vendor response redacted from public disclosure.