Vulnerability Assessment

**December 5, 2020**



**Brick Wall Cyber**

**Security Assessment Team**

Justin Neft Principal Analyst

Karin Sannomiya Security Analyst

Eric Ford Security Analyst

Bennett Moore Security Analyst

Division of Responsibilities

| **Student** | **Expected Contributions** |
| --- | --- |
| Justin Neft | Project Management, Scheduling, Other work |
| Karin Sannomiya | Vulnerability research and analysis |
| Eric Ford | Vulnerability research and analysis |
| Bennett Moore | Vulnerability research and analysis |
|  |  |
| **Communication Plan** | |
| Slack / Discord | |
| **Meeting Schedule** | |
| Mondays / Fridays / as needed | |

[**1. EXECUTIVE SUMMARY**](#_yspy8tt3f0xe) **5**

[**2. THREATS AND RISK**](#_1yssrot6jju1) **6**

[2.a Threat Assessment](#_omko7tmd8gqy) 6

[2.a.1 Threat Actor Motivations](#_a7dnmtyqkgno) 6

[2.a.2 Threat Model](#_dnnoakwbjrc1) 7

[2.b Risk Matrix](#_qtc51anu4pvv) 8

[2.c Prioritization Categories](#_ievzmkg8ce0u) 9

[**3. SUMMARY OF RESULTS**](#_uiwxo2x6jzb9) **10**

[3.a Key Findings](#_hjzukj352bu1) 10

[3.a.1 Out of Date Software Causing Vulnerabilities](#_wfwwnqgz326d) 10

[3.a.2 Missing Necessary Security Components](#_egcnjuxuadwn) 10

[3.a.3 Infrastructure Vulnerabilities](#_ek2hkpq5ufcm) 10

[3.b. Key Recommendations](#_qpegg82d2qw3) 11

[3.b.1 Out of Date Software](#_v5vhxjo08zjl) 11

[3.b.2 Missing Security Measures](#_27tabozgw2rw) 11

[3.b.3 Infrastructure Vulnerabilities](#_6bc6rte41cuj) 11

[**3.c. Response Plan**](#_mcjk91ew3uc8) **12**

[**4. VULNERABILITIES**](#_wwo4i4itmd68) **13**

[4.a SSL and TLS Maximum Compatibility with Old Browsers](#_sfc1wpum8fo3) 13

[4.b Lack of DNS protection](#_reub0yrj38je) 13

[4.c Weak Password Requirement](#_g7ll3ma111ce) 14

[4.d Docker Container with Externally Facing Ports](#_majcqqfx0523) 14

[4.e Employee Hosts Have no Standardized Security Software](#_gbwe8uyaxnmu) 15

[4.f Client Router Lacks Firewall Rules for other BWC Subnets](#_8eyntjlm5mo4) 16

[4.g Single Authentication Server](#_h7vavia7szi) 16

[4.h Logs Stored on Single Machine](#_cpl0wmywvrvi) 17

[4.i Infrequent Vulnerability Scans through OpenVAS](#_5ogw5eotqt27) 17

[4.j Kanboard Security Issues](#_nn5dh98mlzwz) 18

[4.k OpenVAS Scans Performed In Client Subnetwork](#_7tfwmi85r40l) 19

[4.l Ansible](#_6510wk2bjfql) 19

[4.m Company-Wide File Share](#_2439vmmb6ocv) 20

[4.n Lack of Mail Filtering](#_7hstzwlq6ukr) 20

[4.o Vulnerability in Outdated ElasticSearch](#_43dsu15qghm) 21

[4.p Kibana Console Code Execution](#_9kzerwkcu6nb) 22

# 1. EXECUTIVE SUMMARY

The goal of this assessment is to see where in Brick Wall Cyber’s structures and software there are critical security issues that could be exploited to halt BWC’s work, erode client trust or cost money through recovering from malicious attacks.

Through our investigations, we found 1 critical vulnerability, 7 high-impact vulnerabilities and 7 medium-impact vulnerabilities. The critical vulnerability could compromise our entire infrastructure and business, while the high-impact vulnerabilities would damage large parts of our networks and expose ours and clients’ data to varying degrees. Finally the medium-impact vulnerabilities can be exploited to reveal our architecture to further attacks and data-theft.

From our key findings, we have determined that a large-portion of BWC’s software needs significant updating, its network-architecture needs significant remodelling and finally the security measures on BWC’s networks must be greatly expanded.

Our recommendations are for BWC to completely overhaul their network infrastructures to reduce risk of malicious attacks from software, architecture and phishing vulnerabilities. While this overhaul will likely be expensive, in the long-run it will save money through ensuring client trust and not having to spend money on system recovery after any potential/likely successful attacks on BWC’s systems.

# 2. THREATS AND RISK

## 2.a Threat Assessment

### 2.a.1 Threat Actor Motivations

| **Motivation** | **Relevance to Brick Wall Cyber** |
| --- | --- |
| Money | The client information and access to the client network may be used for ransom or black mail. |
| Ideology | Stolen employee data of the client organization may be used to spread a particular ideology. |
| Coercion | Successful penetration to BWC may allow the attacker to access multiple networks of the client organization and commit many attacks such as DoS attack and data breach. |
| Ego | The attacker may be motivated to test their hacking ability by attacking an organization that specializes in cybersecurity. |

| **Motivation** | **Relevance to Brick Wall Cyber** |
| --- | --- |
| Reciprocation | The attacker could have been found out in other companies that BWC evaluated and wants to retaliate |
| Authority | Administrative accounts have access to large amounts of sensitive client and organizational information |
| Scarcity | There are few organizations that have the potential to spread a virus outside their organization in daily functions |
| Commitment / Consistency | Attack could be designed to go undetected in normal work if the attacker is familiar with standard operations |
| Liking | The attacker may think BWC does well and wants to “pen test” their servers |
| Social Proof | May be seeking acclaim for “hacking the hackers” |

### 

### 2.a.2 Threat Model

| **Threat** | **High-level Mitigation** | **Importance for Brick Wall Cyber**  **(Low/Medium/High)** |
| --- | --- | --- |
| Spoofing | Using a zero-trust approach would help mitigate spoofing attempts. | By acting as a trusted source, attackers could compromise BWC’s internal servers and gain access to clients’ data.  HIGH |
| Tampering | Encryption would be the best way to mitigate data tampering. | By tampering with BWC’s data, attackers can disrupt normal operations and inhibit work efficiency and trust.  HIGH |
| Repudiation | Implement logs that track all users’ interactions on the server. | Keeping track of peoples’ actions on the servers reduces the risk of actions going unnoticed or unaccounted for, meaning it’s more difficult for an attacker to move around unnoticed.  MEDIUM |
| Information Disclosure | Make sure that critical information is only available to those who need it, and that it is otherwise impossible to access. | Since BWC works with a variety of clients, and have access to their systems, it would be devastating to BWC’s reputation, and possibly their client’s business, if information was leaked.  HIGH |
| Denial of Service | A strong firewall would help quell Denial of Service attacks. | Although a Denial of Service attack would definitely hinder BWC, it wouldn’t affect them any more than a regular company.  MEDIUM |
| Elevation of Privilege | Extra authentication to make sure a user’s privileges are legitimate would stop hackers from abusing faked permissions. | Depending on what the hacker was able to gain access to, they could theoretically have unfettered access to the systems of BWC, which would be devastating to their business.  HIGH |

## 2.b Risk Matrix

| **RISK MATRIX** | | **THREAT IMPACT** | | | |
| --- | --- | --- | --- | --- | --- |
| **LIKELIHOOD** |  | **LOW** | **MEDIUM** | **HIGH** | **CRITICAL** |
| **RARE** | **Low** | **Low** | **Medium** | **Medium** |
| **UNLIKELY** | **Low** | **Medium** | **High** | **High** |
| **LIKELY** | **Low** | **Medium** | **High** | **Critical** |
| **VERY LIKELY** | **Low** | **Medium** | **Critical** | **Critical** |

## 

## 2.c Prioritization Categories

| **Mitigation Priority** | **Description** |
| --- | --- |
| **Immediate (Imme.)** | Finding has a critical business impact, likelihood, and risk. It damages the operation of the client.  Finding causes a direct violation of regulation, law, or compliance that applies to the client.  Finding leaks Personally Identifiable Information, Sensitive Information, or information that can lead to further access to sensitive data.  Finding is related to previous indicators of compromise and suggests the occurrence of past cyberattacks. |
| **Short-term**  **(Short.)** | Finding has a high business impact, likelihood, and risk. It partially damages the operation of the client and has the potential for further exploitation.  Finding gives attackers direct access to a system or a service.  Finding allows the attackers to violate Confidentiality, Integrity, Availability of a system. |
| **Long-term**  **(Long.)** | Finding has a medium business impact, likelihood, and risk.  Finding is related to security misconfigurations which can lead to further potential attacks.  Finding allows attackers to partially violate Confidentiality, Integrity, Availability of a system. |
| **Eventual**  **(Evetl.)** | Finding has a low business impact, likelihood, and risk.  Finding is not following the best security practices.  Finding is a bug or an unintentional mistake that has little to no security implication. |

# 

# 3. SUMMARY OF RESULTS

## 3.a Key Findings

### 3.a.1 Out of Date Software Causing Vulnerabilities

There were many out of date software and protocols that were installed onto the hosts. For example, the version of Kanboard and Elasticsearch were out of date. The SSL and TlS protocols were also out of date. It is critical to update these softwares to prevent security issues and other matters that could be caused from the incompatibility with newer systems.

### 3.a.2 Missing Necessary Security Components

We found that the network as it stands is lacking a few important security factors in both the network and security policy. For example, we found that the employee account’s password requirements were weak and would make information more vulnerable to attacks, as well as that the network was missing things like firewalls and DNS protection measures.

### 3.a.3 Infrastructure Vulnerabilities

From our findings, it is clear that there are a few serious vulnerabilities within the actual infrastructure of BWC. The Docker ports, the authentication server, the lack of client firewalls, the single log server, the universal file share, all of these point to a serious lack of secure infrastructure within the company itself that needs to be addressed.

## 3.b. Key Recommendations

### 3.b.1 Out of Date Software

Updating the Software version to a newer version could prevent many future security flaws and revise any holes the previous version had. Updates often include security patches, so it is important that the version of software or protocols installed onto the systems are up to date.

### 3.b.2 Missing Security Measures

We recommend that BWC review their security policies, for example requiring stronger passwords. Additionally, we feel that there are some key areas of weakness in the network that harm the overall security of Brick Wall Cyber, especially in the lack of a subnet firewall, DNS protection, and vulnerability scans.

### 3.b.3 Infrastructure Vulnerabilities

It is vital that BWC look closely at some of their practices and systems within their infrastructure, and determine whether something could be done more securely. For instance, instead of having a massive company-wide file share, have a system which would allow users to request permission to create or edit specific files or directories, so that the information is only accessible to employees who need it, and viruses such as computer worms cannot spread freely throughout the entire system.

# 

# 3.c. Response Plan

| **Mitigation Prioritization** | **Vulnerability** |
| --- | --- |
| **Immediate (Imme.)** | * 4.k * 4.m * 4.p * 4.b * 4.g * 4.h |
| **Short-term (Short.)** | * 4.f * 4.o * 4.a * 4.d * 4.l |
| **Long-term (Long.)** | * 4.e * 4.i * 4.n * 4.c * 4.j |
| **Eventual (Evetl.)** |  |

# 

# 4. VULNERABILITIES

## 4.a SSL and TLS Maximum Compatibility with Old Browsers

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Information disclosure, Tampering** | **7.6 <High>** | **Short.** |
| **Impact** | **High** |
| **Likelihood** | **Likely** |
| **Hosts Impacted** | WWW, Mail | | |

| **Description** |
| --- |
| WWW and Mail only supports SSL 2.0, SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2, TLS 1.3 for maximum compatibility with old browsers, so it will not have sufficient encryption for the newer browsers. The confidentiality of data will not be guaranteed. |

| **External References** |
| --- |
| <https://www.godaddy.com/garage/browser-support-tls-10-11/> |

## 4.b Lack of DNS protection

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **DDoS attack, DNS spoofing** | **8.0 <High>** | **Imme.** |
| **Impact** | **High** |
| **Likelihood** | **Likely** |
| **Hosts Impacted** | All systems that are connected to the DNS. | | |

| **Description** |
| --- |
| The DNS itself lacks security and will not guarantee a secure DNS service. Additional protection such as DNSSEC(DNS Security Extension), DNS Filtering, and add-on that monitors the DNS activity is critical to keeping the DNS away from threats. |

| **External References** |
| --- |
| <https://heimdalsecurity.com/blog/dns-security-for-business/> |

## 4.c Weak Password Requirement

| **Vulnerability Name** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Ransomware, Phishing** | **5.3 <Medium>** | **Long.** |
| **Impact** | **Low** |
| **Likelihood** | **Likely** |
| **Hosts Impacted** | Mail | | |

| **Description** |
| --- |
| The current requirement of the email account password is a minimum of six letters and upper/lower case, which is too simple and weak. The password should be at least eight characters or longer and should have mixed case letters, numbers, and preferably symbols. |

| **External References** |
| --- |
|  |

## 4.d Docker Container with Externally Facing Ports

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Information disclosure, Tampering** | **7.6 <High>** | **Short.** |
| **Impact** | **High** |
| **Likelihood** | **Likely** |
| **Hosts Impacted** | Wordpress | | |

| **Description** |
| --- |
| The host uses publicly accessible opened ports 80, 443, and 3306 which may be used by the attacker for infiltration and exploit more vulnerabilities. Docker containers by default allows all network traffic between containers under the same host. The open ports need to be monitored and communications between containers should be restricted and encrypted through TLS to protect confidentiality and integrity of the network. |

| **External References** |
| --- |
| <https://www.cimcor.com/blog/the-top-5-security-risks-in-docker-container-deployment#:~:text=Running%20an%20older%20version%20of%20Docker%20containers%20can,fixes%20are%20often%20included%20in%20the%20update%20packages>.  <https://www.upguard.com/blog/open-port> |

## 4.e Employee Hosts Have no Standardized Security Software

| **Vulnerability Name** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Medium (Varies)** | **6.5 Medium** | **Long-Term** |
| **Impact** | **High** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | Corporate | | |

| **Description** |
| --- |
| Individual hosts in the corporate branch of Brickwall Cyber’s infrastructure have no standardized security measures, meaning that the threat of vulnerabilities could vary wildly from system-to-system. One computer could have full hard-drive encryption and a VPN, while another only has basic Windows OS security, making it easier for certain hosts to be compromised. |

| **External References** |
| --- |
|  |

## 4.f Client Router Lacks Firewall Rules for other BWC Subnets

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Medium** | **7.3 <High>** | **Short-Term** |
| **Impact** | **High** |
| **Likelihood** | **Rare** |
| **Hosts Impacted** | Client Server | | |

| **Description** |
| --- |
| Should an attacker manage to infiltrate/compromise a client’s server, they could then easily move-into other BWC subnets since there is no firewall acting between them. As such, it is recommended that BWC has some firewall/protection in-place when traffic from a client’s server moves to other subnets, to avoid multiple systems being compromised at once. |

| **External References** |
| --- |
|  |

## 4.g Single Authentication Server

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Elevation of privilege, DDOS** | **7.6 <High>** | **Imme.** |
| **Impact** | **High** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | Internal IT | | |

| **Description** |
| --- |
| There is no protection in place for backup in an attack on the Active Directory in the IT network, which is the only system that controls access to administrative privileges. If this system goes down, either no administrators would be able to log on or any user could gain administrative privileges. |

| **External References** |
| --- |
|  |

## 4.h Logs Stored on Single Machine

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **DDOS** | **6.7 <Medium>** | **Short.** |
| **Impact** | **Medium / High** |
| **Likelihood** | **Rare / Unlikely** |
| **Hosts Impacted** | ELK | | |

| **Description** |
| --- |
| This is not a vulnerability that is likely to be attacked alone, but as part of a larger attack on the network. Since this is the only place logs are stored in the network, any attacker able to disable or destroy the logs on ELK would make recovery and investigation of the incident significantly more difficult. |

| **External References** |
| --- |
|  |

## 4.i Infrequent Vulnerability Scans through OpenVAS

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Medium** | **N/A N/A** | **Long-Term** |
| **Impact** | **High** |
| **Likelihood** | **Rare** |
| **Hosts Impacted** | Client Server, potentially corporate server. | | |

| **Description** |
| --- |
| OpenVAS is installed on all client networks, and performs quarterly scans. While a good start, vulnerabilities can pop-up and be exploited extremely-quickly in the modern age. As such, it is recommended that OpenVAS’s scanning tools be used once-per-month at minimum to reduce the time a vulnerability can be active in the system. For even more-secure systems, do the security scan once per week. |

| **External References** |
| --- |
|  |

## 4.j Kanboard Security Issues

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Remote code execution** | **4.3 <Medium>** | **Long.** |
| **Impact** | **Medium** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | Kanboard | | |

| **Description** |
| --- |
| The version of Kanboard used on the network (1.2.7) is known to have vulnerabilities that allowed for remote execution of possibly malicious code on the system. However, the host computer does not really have any other function besides running Kanboard, so there is not very much data at risk from this threat, and it would not be enough to stop business operations if exploited. |

| **External References** |
| --- |
| <https://packetstormsecurity.com/files/151792/Kanboard-1.2.7-Code-Execution-Cross-Site-Request-Forgery.html> |

## 4.k OpenVAS Scans Performed In Client Subnetwork

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Low** | **6.7 <Medium>** | **Imme.** |
| **Impact** | **High** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | Client, Potentially Internal IT/Ops | | |

| **Description** |
| --- |
| By having the scanning-software on the same network as the client information and data, it is entirely possible that the scanning software (OpenVAS in this case) could be attacked and compromised to send phony reports back to the main BWC network, hiding potential security risks. To mitigate this, scans should be performed from hosts outside of the client’s network to make tampering of the scanning software less-likely. |

| **External References** |
| --- |
|  |

## 4.l Ansible

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Remote code execution** | **7.2 <High>** | **Short.** |
| **Impact** | **High** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | Any new host in the network that is set up with the Ansible configuration | | |

| **Description** |
| --- |
| A potential attack that injects code into the Ansible system could be spread into new systems throughout the network when they are set up using Ansible. In doing so, the attacker could execute code in machines throughout various parts of the organization, leading to widespread effects. |

| **External References** |
| --- |
|  |

## 4.m Company-Wide File Share

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Information Disclosure, Tampering** | **10.0 <Critical>** | **Imme.** |
| **Impact** | **Critical** |
| **Likelihood** | **Unlikely/Likely** |
| **Hosts Impacted** | Theoretically every company computer inside BWC | | |

| **Description** |
| --- |
| Because there is a company-wide share that any user can read or write to, regardless of privilege, if a hacker were to gain access to this massive share, they could easily spread a worm to every computer connected to the share (AKA every single company computer). |

| **External References** |
| --- |
|  |

## 4.n Lack of Mail Filtering

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Spoofing, Information Disclosure** | **5.8 <Medium>** | **Long.** |
| **Impact** | **Medium** |
| **Likelihood** | **Very Likely** |
| **Hosts Impacted** | Mail | | |

| **Description** |
| --- |
| Because there aren’t any documented spam/scam filters in place, it would be exceedingly easy for someone to send a scam email to an employee of BWC. Even with training, all it would take is for one employee to fall victim to a scam and its possible that the entire system could become infected (especially with the universal file share mentioned earlier). |

| **External References** |
| --- |
|  |

## 4.o Vulnerability in Outdated ElasticSearch

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Information Disclosure** | **5.0 <Medium>** | **Short.** |
| **Impact** | **Medium** |
| **Likelihood** | **Likely** |
| **Hosts Impacted** | ELK | | |

| **Description** |
| --- |
| Due to a vulnerability in ElasticSearch 1.6.0, hackers are able to easily read certain files by using snapshot API calls. |

| **External References** |
| --- |
| <https://www.cvedetails.com/cve/CVE-2015-5531/> |

## 4.p Kibana Console Code Execution

| **Risk Analysis** | | **CVSS** | **Prioritization** |
| --- | --- | --- | --- |
| **Risk** | **Tampering, Information Disclosure, Denial of Service, Elevation of Privilege** | **7.5 <High>** | **Imme.** |
| **Impact** | **High** |
| **Likelihood** | **Unlikely** |
| **Hosts Impacted** | ELK | | |

| **Description** |
| --- |
| If a hacker can gain access to the Kibana console API, they can execute Javascript code as if it is Kibana itself attempting to execute it, bypassing any security protocols that would normally stop the code from being run. |

| **External References** |
| --- |
| <https://www.cvedetails.com/cve/CVE-2018-17246/> |