Security

Misconfiguration

Nishanth P
CS20B025

November 4, 2023

Introduction

Imagine a castle with thick walls, a sturdy moat, and a watchful guard. This castle is well-protected from attack, right? But what if the castle gate is left unlocked? Or the guard falls asleep on the job? Then, even the strongest castle is vulnerable.

2

Security misconfigurations are like unlocked gates and sleeping guards. They are security weaknesses that can make even the most well-protected systems vulnerable to attack.

Here is a narrative example of how a security misconfiguration can lead to a data breach:

- A company has a cloud-based database that stores customer data. The database is configured with default permissions, which allow anyone with an internet connection to access the data.
- An attacker discovers the company's misconfigured database and uses it to steal customer data, including names, addresses, and credit card numbers.
- The attacker then sells the stolen data on the dark web.
- The company's customers are now at risk of identity theft and fraud.

This is just one example of how a security misconfiguration can have a serious impact on an organization. Security misconfigurations can also lead to financial losses, regulatory compliance violations, and reputational damage.

That's why it's so important for organizations to take steps to prevent security misconfigurations. Here is a story about how one organization did just that:

- A company called Acme Corporation had a history of security misconfigurations. This had led to several data breaches over the years, costing the company millions of dollars.
- Acme decided to take action to address the problem. They hired a security consultant to review their IT infrastructure and identify any misconfigurations.
- The consultant found a number of misconfigurations, including default passwords, open ports, and unnecessary services. The consultant worked with Acme to remediate these misconfigurations.

- Acme also implemented a security configuration management process to help prevent future misconfigurations. This process included regular reviews of security settings and automated scans for misconfigurations.

As a result of these efforts, Acme significantly reduced the number of security misconfigurations in its IT infrastructure. This led to a decrease in data breaches and other security incidents.

Acme's story shows that security misconfigurations can be prevented. By taking the necessary steps, organizations can reduce their risk of exposure and protect their systems and data.

Description

Security Misconfiguration occurs when system or the application configuration settings are missing or implemented erroneously. It can also due to

- Security gaps
- Exposing the application more than necessary
- Not implementing security policies
- Non removal of default configuration

Attacks can happen at any layer of the application stack. Few most prominent examples are listed below

- Cloud containers
- Exposing unused port
- Default configuration in database
- Misconfigured Network services
- Lack of security policies enforced in development code
- Forget to remove testing components in production environment

Causes

Unpatched flaws

All software has flaws, but most software vendors soon issue patches to repair those flaws. When you don't install the patch, attackers who already know the flaws will be able to penetrate your systems.

Unused pages and unnecessary service

Unused web pages and unnecessary features or services also allow attackers to gain unauthorized access to an enterprise application or device. These issues may result in cyberattacks such as command injections, brute force attacks, and credential stuffing exploits if left unchecked.

Inadequate access controls

Threat actors can gain entry into the network infrastructure by using default passwords, abandoned user accounts, or unused access permissions that admins did not

update or remove. Overly permissive access rules also allow adversaries to cause chaos, including malware attacks and data compromise.

Unprotected files and directories

Files and directories not protected by strong security controls are vulnerable to cyberattacks. Hackers can identify platforms and applications that use easy-to-guess names and locations to garner valuable system information and orchestrate targeted attacks.

Predictable file names and locations can also expose admin interfaces and allow the adversary to get privileged access, configuration details, or business logic and even add, remove, or modify application functionality.

Poor coding practices and using vulnerable XML files

Many security misconfigurations can occur in Java web.xml files. Custom error pages or SSL may not be configured, or the code may be missing web-based access controls.

Coding errors may allow attackers to access parts of web applications via non-SSL and launch session hijacking attacks. Using URL parameters for session tracking or not setting a session timeout may also result in these attacks. Similarly, cookies without the HttpOnly flag can increase the possibility of cross-site scripting (XSS) attacks.

Disabled antivirus

Sometimes users temporarily turn off the antivirus if the antivirus overrides a particular action, such as running an installer. Once the user completes the installation, if he or she forgets to reactivate the antivirus, that leaves the organization vulnerable to hacks and data breaches.

Inadequate hardware management

Hackers use devices such as routers, switches, and endpoints to access enterprise applications and data by exploiting unsecured ports, overly permissive network traffic rules, and inadequately patched and maintained hardware.

Excess privilege

Excess privilege happens if an employee or a contractor is given more administrative rights or access than what is required for their job. For example, someone could be given excessive data access permissions for cloud storage containers. This often happens when an employee has moved roles within the organization, is a new hire but had privileges mirrored from an incorrect account, or has left the organization but didn't have their access revoked in a timely manner.

Real life cases

Atlassian [Permission Settings]

The 2019 JIRA security misconfiguration breach was discovered by Avinash Jain, a lead infrastructure security engineer at Grofers. Jain found that a misconfiguration in JIRA's Global Permissions settings allowed anyone to view dashboards, filters, and user pickers that had been shared publicly.

Jain was able to find links to these publicly accessible dashboards, filters, and user pickers by using Google dorks. Google dorks are special search queries that can be used to find specific types of information on the web. In this case, Jain used Google dorks to find links to publicly accessible JIRA dashboards, filters, and user pickers that contained sensitive information about employees and projects of many large companies, including NASA, Google, Yahoo, and Zendesk.

Jain reported the issue to Atlassian and to the affected companies. Atlassian released a security patch to fix the misconfiguration, and the affected companies took steps to secure their JIRA instances.

However, the breach highlights the importance of security awareness and training for employees. Even if a company has a strong security policy and procedures in place, employees need to be trained on how to identify and report potential security risks.

Here are some additional details about the breach:

- The breach was discovered in August 2019.
- At least 1,972 JIRA instances were affected.

The affected companies included a wide range of organizations, including Fortune 500 companies, government agencies, and nonprofits.

The exposed data included usernames and email addresses, project details, upcoming milestones, and internal company information.

Source: Link to blog

Nissan [Default credentials Git server]

In 2021, Nissan, a Japanese multinational automobile manufacturer, had some source code leaked online. A Swiss security researcher discovered that it was due to misconfiguration of a company Git server. The Git server was left exposed online with a default username and password of admin/admin

Source: Link to blog

Mercedes-Benz [Exposed unnecessary information]

Earlier, in May 2020, Mercedes-Benz experienced a similar breach. The Git server of Daimler AG, the company that owns the Mercedes-Benz brand, was compromised by a straightforward Google Dorking operation, an attacker technique that involves using search engines to find security flaws in publicly accessible servers.

Shopify [Excess Permission]

Two employees of Shopify's support team collaborated to steal transaction records containing information related to customer emails, names, addresses, and orders. The workers had permission to access its internal network to service customers but weren't authorized to access the network for any other purpose. One co-conspirator stole merchant and customer data by taking screenshots of the data and by uploading the data to cloud storage.

Amazon S3 Misconfiguration Attacks

When	Casualty	Leak
Nov 2017	Australian Broadcasting Corporation	Hashed passwords, internal resources, and keys were leaked.
Nov 2017	United States Army Intelligence and Security Command	Several files, including Oracle Virtual Appliance (.ova). volumes with portions marked top secret.
Sept 2017	Accenture	Authentication information, which included certificates, plaintext passwords, keys, and sensitive customer information.

Mirai

Mirai is a type of malware that infects network devices. After devices are infected they can be remotely controlled by the operator, which uses them as bots that extend the power of a botnet. Mirai targeted mainly IoT devices, and managed to execute several high

profile attacks even after it was discovered in August 2016. Eventually, the creator released the code as open source (Anna-senpai), and the technique has since been used in other malware projects. Mirai managed to infect and run on CCTV cameras, home routers, and DVRs. It succeeded by trying commonly used passwords. This simple method enabled the mirai botnet to produce 280 Gbps and 130 Mpps in DDoS capability and attack the DNS provider Dyn. Mirai also rendered several notable sites inaccessible, including GitHub, Reddit, Airbnb, Netflix, and Twitter.

Source: Link to source code

Effects

Default accounts / passwords are enable

Using vendor-supplied defaults for system accounts and passwords is a common security misconfiguration, and may allow attackers to gain unauthorized access to the system.

Secure password policy is not implemented

Failure to implement a password policy may allow attackers to gain unauthorized access to the system by methods such as using lists of common username and passwords to brute force a username and/or password field until successful authentication.

Software is out of date and flaws are unpatched

Failure to update software patches as part of the software management process may allow attackers to use techniques such as code injection to inject malicious code that the application then executes.

Files and directories are unprotected

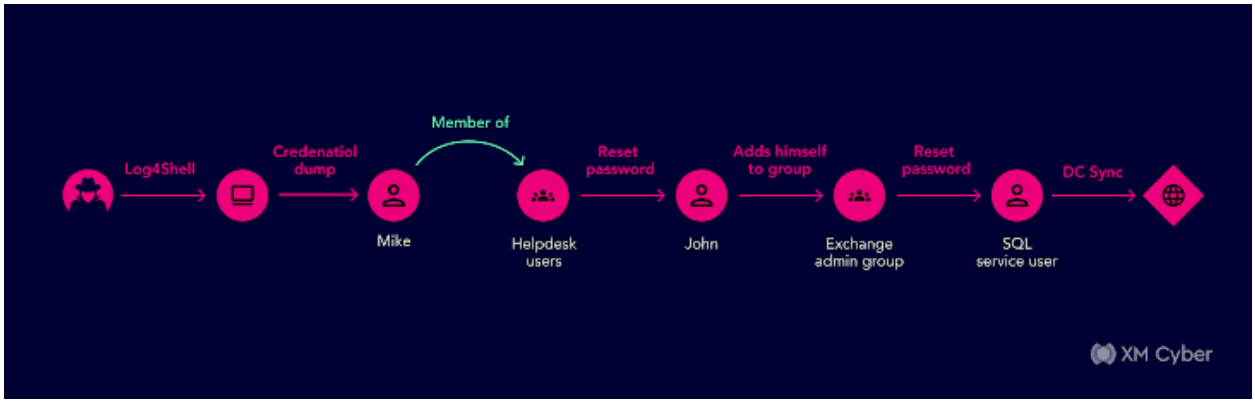
Leaving files and directories unprotected may allow attackers to use techniques such as forceful browsing to gain access to restricted files or areas in the server directory.

Unused features are enabled or installed

Failure to remove unnecessary features, components, documentation, and samples makes the application susceptible to misconfiguration vulnerabilities, and may allow attackers to use techniques such as code injection to inject malicious code that the application then executes.

Security features not maintained or configured properly

Failure to properly configure and maintain security features makes the application vulnerable to misconfiguration attacks.



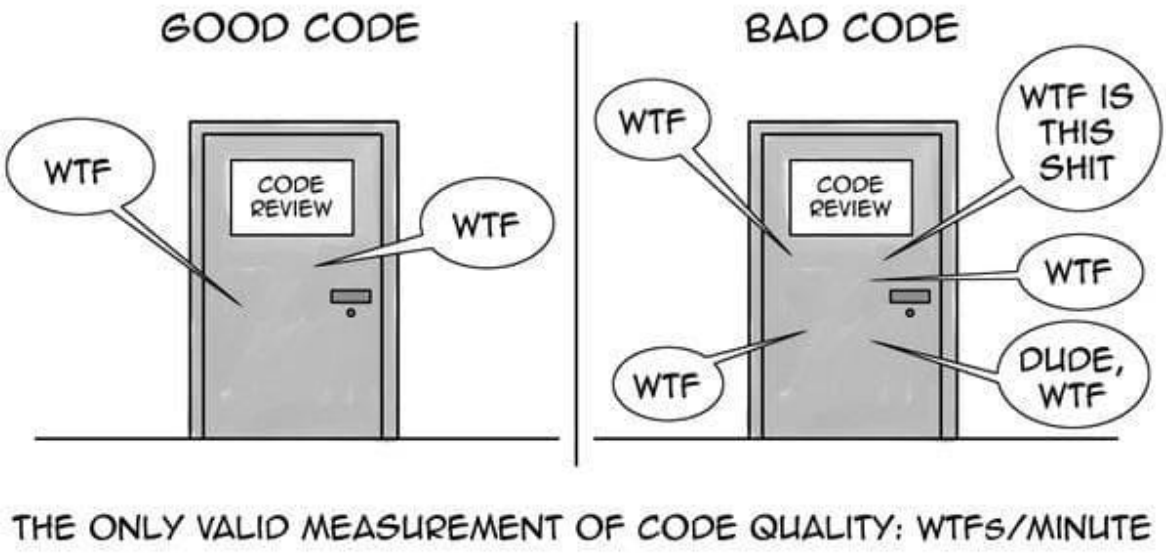
This example is taken from thehackernews website. Here, the security is compromised using phishing mail.

Unpublished URLs are not blocked from receiving traffic from ordinary users

Unpublished URLs, accessed by those who maintain applications, are not intended to receive traffic from ordinary users. Failure to block these URLs can pose a significant risk when attackers scan for them.

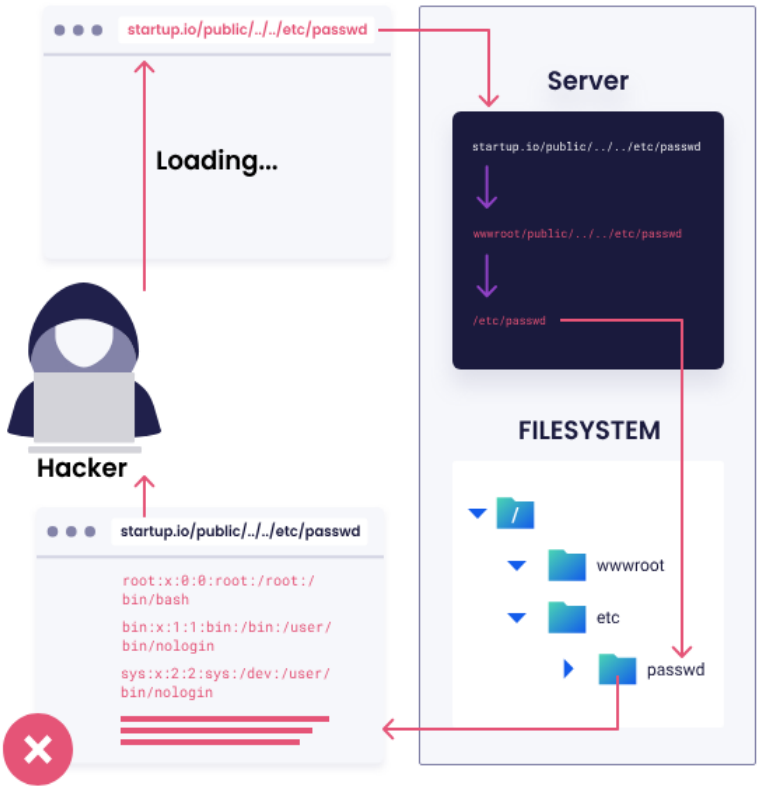
Improper / poor application coding practices

Improper coding practices can lead to security misconfiguration attacks. For example, the lack of proper input/output data validation may lead to code injection attacks which work by injecting code that the application executes.



Directory traversal

Allows an attacker to access directories, files, and commands that are outside the root directory. Armed with access to application source code or configuration and critical system files, a cybercriminal can change a URL in such a way that the application could execute or display the contents of arbitrary files on the server. Any device or application that reveals an HTTP-based interface is possibly vulnerable to a directory traversal attack



Prevention

Education and Training

One of the most effective means of preventing security misconfiguration is training and educating your staff members about the latest security trends. This allows them to make smarter decisions and adhere to best practices.

Encryption

Data exfiltration is a concern for many organizations. Sensitive or proprietary data in the hands of individuals with ill intent can lead to dramatic losses or embarrassment for an organization, both in relation to personnel and financially. Data can often be an organization's most essential asset.

Utilizing data-at-rest encryption schemes might assist with the protection of files from data exfiltration. You can also apply appropriate access controls to directories and files. These measures offset the vulnerability of susceptible directories and files.

Scanning

Conducting security scans on systems is an automated method of isolating vulnerabilities. Running such scans on a regular schedule, after creating architectural changes, is a significant step in improving the overall vulnerability.

If implementing custom-written code, you should also make use of a static code security scanner. This must come prior to implementing that code in the production environment.

Least Privilege

Only provide users with access to information they absolutely require to do their jobs. You will need strong access controls, including a strong password and username, and establish two-factor authentication.

You should also compartmentalize data. Ensure that administrators hold unique accounts for when they are making use of their administrative rights as opposed to when they are behaving as a regular user of the system.

Updating Software

The use of outdated software remains one of the most prevalent security vulnerabilities. Many companies don't appreciate the need to invest in the newest and latest. They may feel it is more cost-effective to continue making use of legacy software. However, using outdated software can actually place an organization at risk of losing assets—as well as the trust of their investors and customers.

Establishing a consistent patch schedule, and maintaining updated software, is essential to reducing an organization's threat vectors.

Security Checklist

To ensure you've covered all your configuration security requirements, implement a checklist that incorporates the different measures you want to put in place. Based on the recommendations of security experts, a checklist as follows may help prevent security misconfiguration:

- Create a patching schedule and encrypt your data
- Ensure software is up-to-date and disable default accounts
- Implement reliable access controls
- Give administration a routine process to so they don't overlook items
- Establish security settings in development frameworks to safeguard value
- Undertake system audits periodically and launch security scanners

Code based examples

Default account: