Global Cybersecurity Threats Analysis

Contents

1 Setup

1.1 Import Packages

```
import pandas as pd
```

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import FuncFormatter
import seaborn as sns
```

1.2 Load data

```
pd.set_option('display.max_columns', 50, 'display.width', 200)
df = pd.read_csv('data/Global_Cybersecurity_Threats_2015-2024.csv')
df.head()
```

```
|   | Country | Year | Attack Type       | Target Industry    | Financial Loss (in Million $) | Number of Affected Users | Attack Source | Security Vulnerability Type | Defense Mechanism Used | Incident Resolution Time (in Hours) |
|---+---------+------+-------------------+--------------------+-------------------------------+--------------------------+---------------+-----------------------------+------------------------+-------------------------------------|
| 0 | China   | 2019 | Phishing          | Education          |                         80.53 |                   773169 | Hacker Group  | Unpatched Software          | VPN                    |                                  63 |
| 1 | China   | 2019 | Ransomware        | Retail             |                         62.19 |                   295961 | Hacker Group  | Unpatched Software          | Firewall               |                                  71 |
| 2 | India   | 2017 | Man-in-the-Middle | IT                 |                         38.65 |                   605895 | Hacker Group  | Weak Passwords              | VPN                    |                                  20 |
| 3 | UK      | 2024 | Ransomware        | Telecommunications |                         41.44 |                   659320 | Nation-state  | Social Engineering          | AI-based Detection     |                                   7 |
| 4 | Germany | 2018 | Man-in-the-Middle | IT                 |                         74.41 |                   810682 | Insider       | Social Engineering          | VPN                    |                                  68 |
```

2 Trend Analysis

2.1 Financial Losses

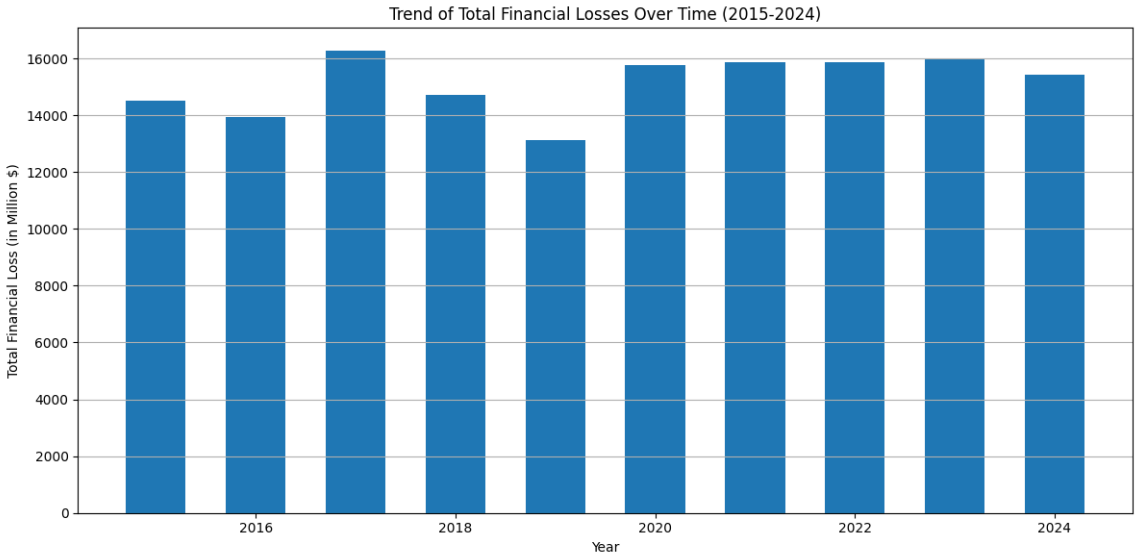
2.1.1 Overall

```
trend_df = df.groupby('Year')['Financial Loss (in Million $)'].sum()

plt.figure(figsize=(12, 6))
plt.bar(trend_df.index, trend_df, width=0.6)

plt.title('Trend of Total Financial Losses Over Time (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Financial Loss (in Million $)')
plt.grid(axis='y')
```

```
plt.tight_layout()
plt.show()
```

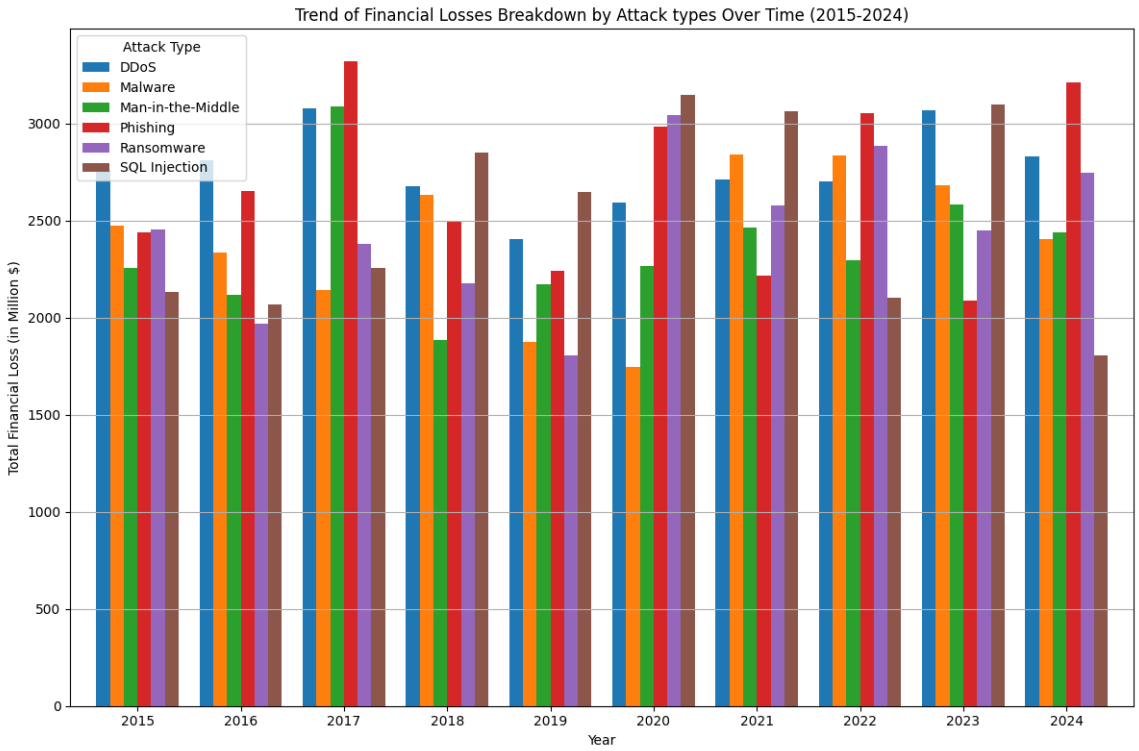


2.1.2 Breakdown by Attack types

```
# Group by Year and Attack Type, summing financial losses
trend_df = df.groupby(['Year', 'Attack Type'])['Financial Loss (in Million
↪  $)'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
trend_df.plot(kind='bar', figsize=(12, 8), width=0.8)

plt.title('Trend of Financial Losses Breakdown by Attack types Over Time (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Financial Loss (in Million $)')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Attack Type')
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```

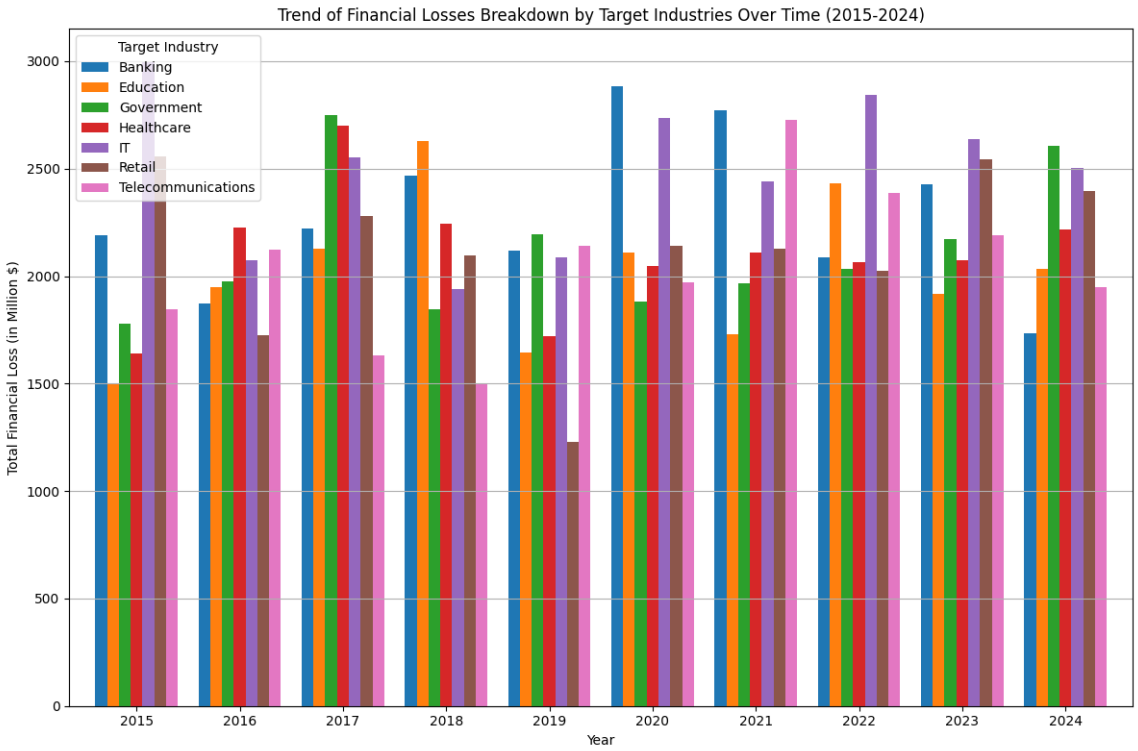


2.1.3 Breakdown by Target Industries

```
# Group by Year and Target Industry, summing financial losses
trend_df = df.groupby(['Year', 'Target Industry'])['Financial Loss (in Million
↪  $)'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
trend_df.plot(kind='bar', figsize=(12, 8), width=0.8)

plt.title('Trend of Financial Losses Breakdown by Target Industries Over Time (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Financial Loss (in Million $)')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Target Industry')
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```

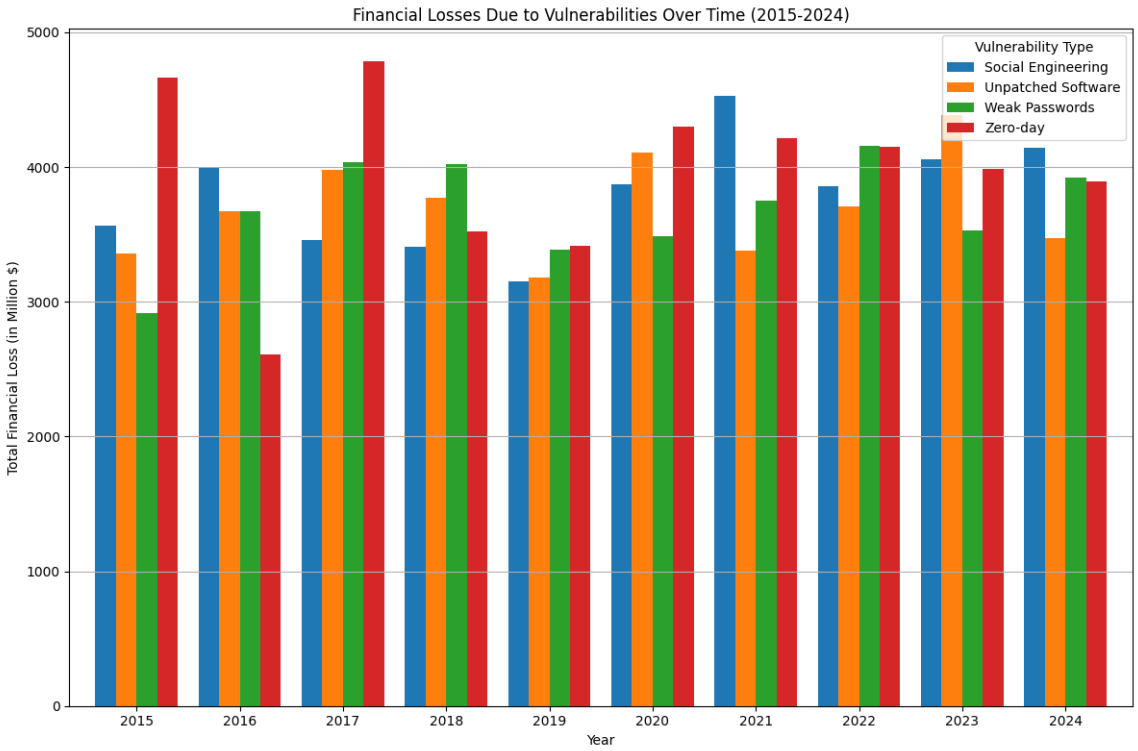


2.1.4 Breakdown by Vulnerabilities

```
# Group by Year and Security Vulnerability Type, summing financial losses
vulnerability_df = df.groupby(['Year', 'Security Vulnerability Type'])['Financial Loss (in
↪  Million $)'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
vulnerability_df.plot(kind='bar', figsize=(12, 8), width=0.8)

plt.title('Financial Losses Due to Vulnerabilities Over Time (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Financial Loss (in Million $)')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Vulnerability Type')
plt.grid(axis='y')
plt.tight_layout()
plt.show()
```



2.2 Number of Affected Users

2.2.1 Overall

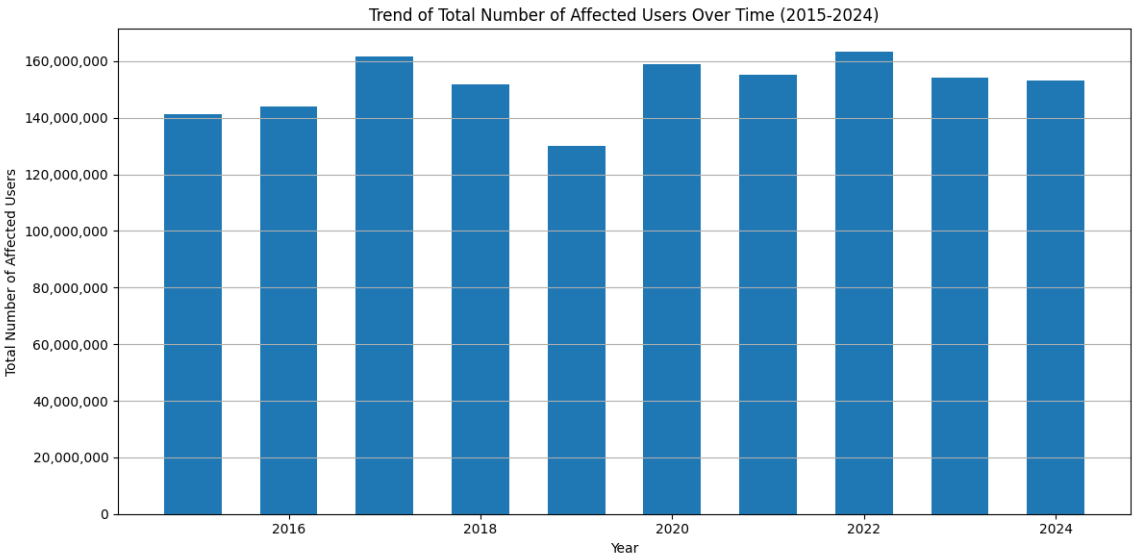
```
trend_df = df.groupby('Year')['Number of Affected Users'].sum()

plt.figure(figsize=(12, 6))
plt.bar(trend_df.index, trend_df, width=0.6)

plt.title('Trend of Total Number of Affected Users Over Time (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Number of Affected Users')
plt.grid(axis='y')

# Apply number formatting
plt.gca().yaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{int(x):,}'))

plt.tight_layout()
plt.show()
```

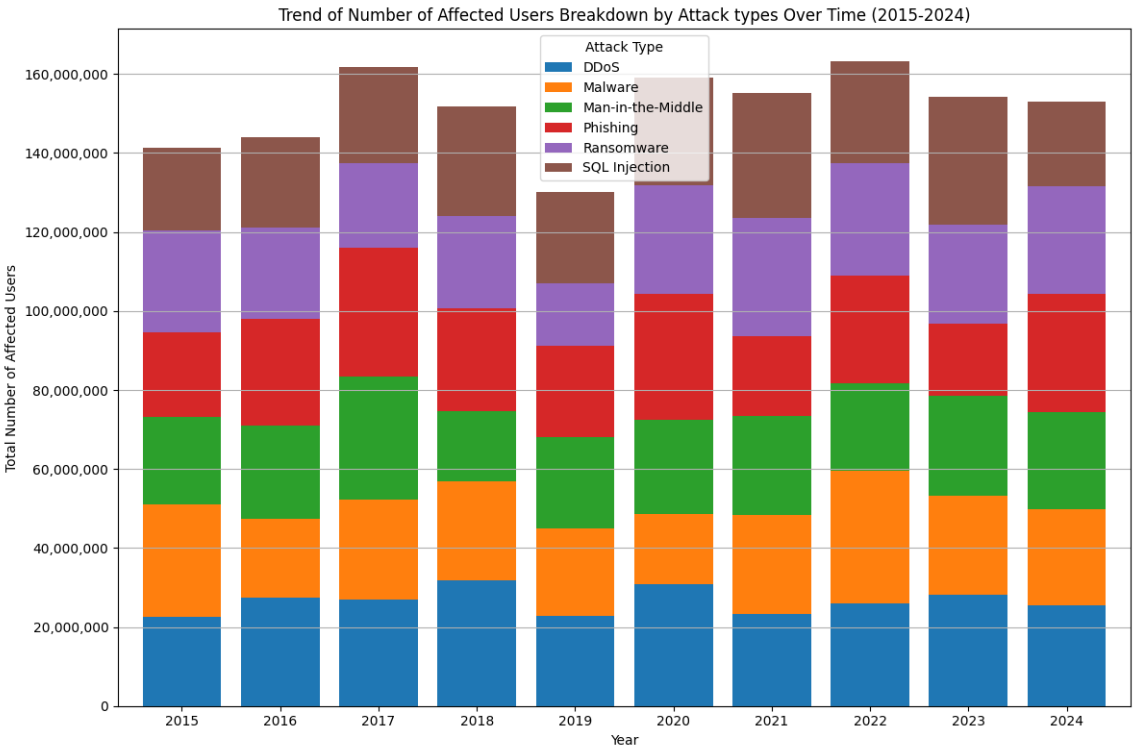


2.2.2 Breakdown by Attack types

```
trend_df = df.groupby(['Year', 'Attack Type'])['Number of Affected
↪  Users'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
trend_df.plot(kind='bar', stacked=True, figsize=(12, 8), width=0.8)

plt.title('Trend of Number of Affected Users Breakdown by Attack types Over Time
↪  (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Number of Affected Users')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Attack Type')
plt.grid(axis='y')
plt.gca().yaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{int(x):,}')) # Apply
↪  number formatting
plt.tight_layout()
plt.show()
```

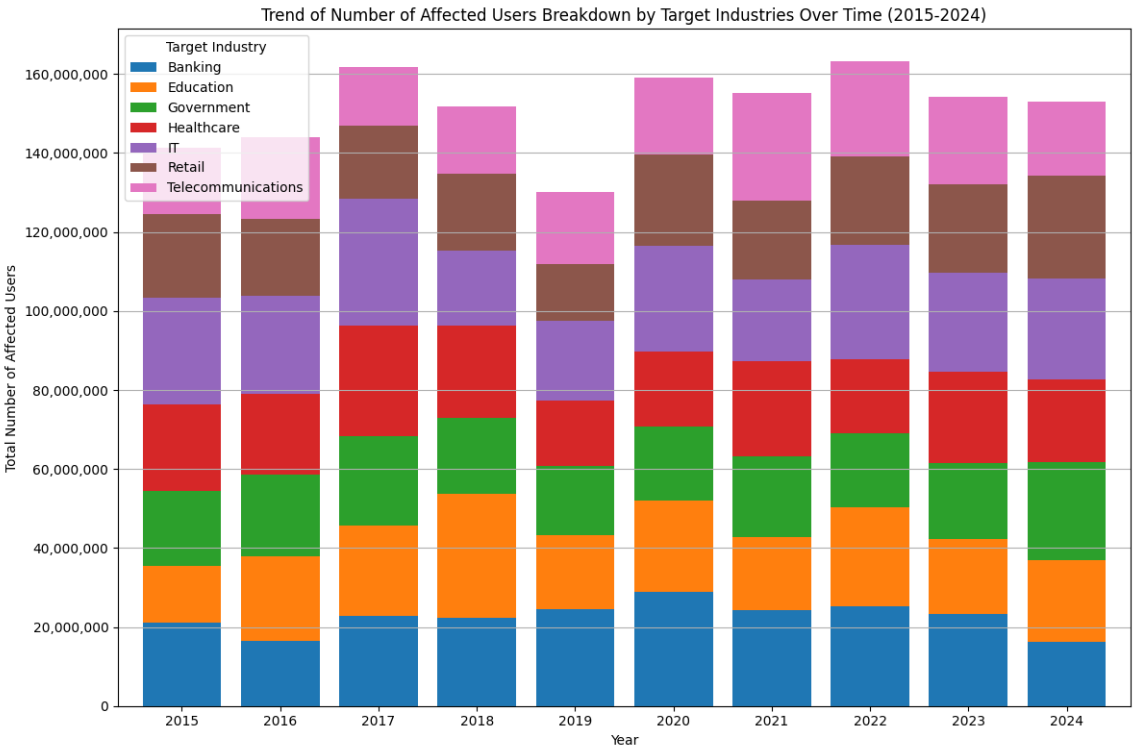


2.2.3 Breakdown by Target Industries

```
trend_df = df.groupby(['Year', 'Target Industry'])['Number of Affected
↪  Users'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
trend_df.plot(kind='bar', stacked=True, figsize=(12, 8), width=0.8)

plt.title('Trend of Number of Affected Users Breakdown by Target Industries Over Time
↪  (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Number of Affected Users')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Target Industry')
plt.grid(axis='y')
plt.gca().yaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{int(x):,}')) # Apply
↪  number formatting
plt.tight_layout()
plt.show()
```

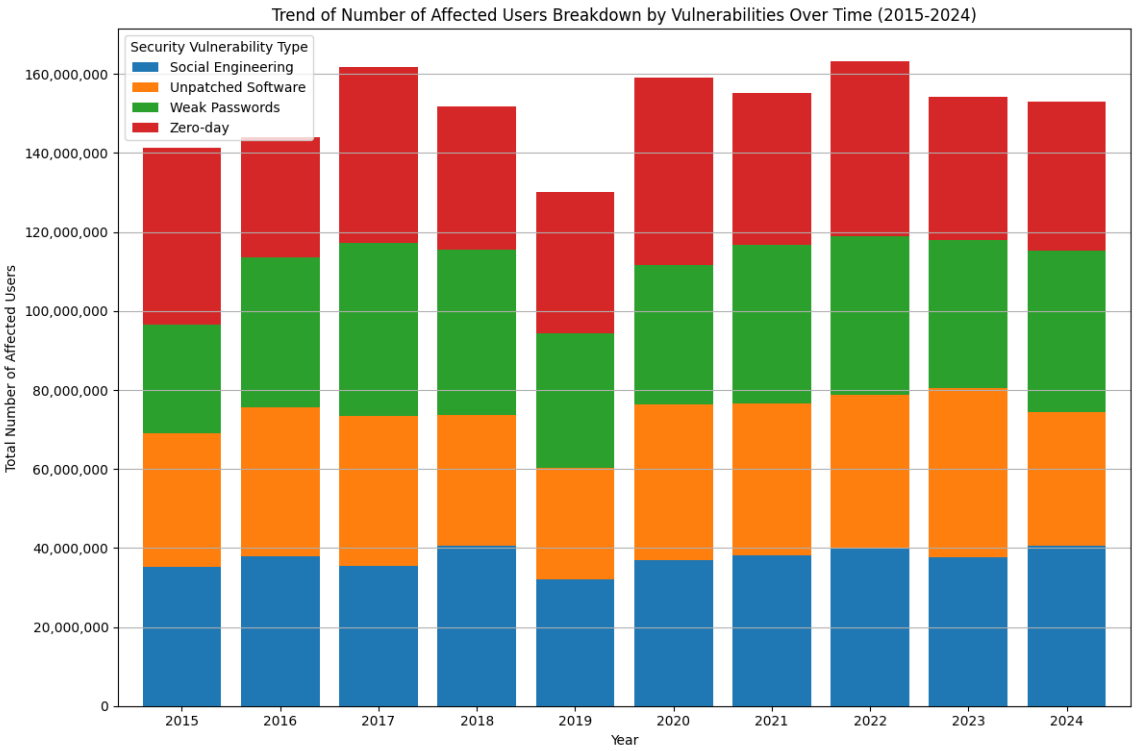


2.2.4 Breakdown by Vulnerabilities

```
trend_df = df.groupby(['Year', 'Security Vulnerability Type'])['Number of Affected
↪  Users'].sum().unstack(fill_value=0)

# Plotting the data as a bar chart
trend_df.plot(kind='bar', stacked=True, figsize=(12, 8), width=0.8)

plt.title('Trend of Number of Affected Users Breakdown by Vulnerabilities Over Time
↪  (2015-2024)')
plt.xlabel('Year')
plt.ylabel('Total Number of Affected Users')
plt.xticks(rotation=0)  # Keep x-axis labels horizontal
plt.legend(title='Security Vulnerability Type')
plt.grid(axis='y')
plt.gca().yaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{int(x):,}')) # Apply
↪  number formatting
plt.tight_layout()
plt.show()
```



3 Geographical Analysis

- Compare the frequency and impact of cyberattacks across different countries.
- Identify which countries are most affected by specific attack types.

4 Financial Impact Analysis

- Assess the total financial losses caused by cyberattacks per year or country.
- Analyze the correlation between attack types and financial losses.

5 Industry Analysis

- Determine which industries are most frequently targeted by cyberattacks.
- Assess the impact of attacks on different sectors, such as healthcare, finance, and education.

6 Vulnerability Analysis

- Identify common security vulnerabilities exploited in attacks.
- Analyze the effectiveness of various defense mechanisms used against attacks.

7 User Impact Analysis

- Assess how many users are affected by different attack types or in different countries.
- Explore the relationship between the number of affected users and financial losses.

8 Response Time Analysis

- Analyze the incident resolution times based on attack types or countries.
- Identify any patterns in response effectiveness.

9 Defensive Mechanism Effectiveness

- Evaluate the success rates of different defense mechanisms against various attack types.