**Training Activity: Analysis of 10 Biggest Software Bugs and Tech Fails of 2021**

Source: <https://www.testdevlab.com/blog/2021/12/27/10-biggest-software-bugs-and-tech-fails-of-2021>

**Table 1: Issue Overview and Impact Analysis**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attack/Issue Name** | **Description** | **Impact Analysis 1** | **Impact Analysis 2** |
| T-Mobile Data Breach | Over 50 million customers' personal data was compromised. | Security Breach | Customer Trust |
| Slack Public DM Feature | Public DMs led to spam and harassment before being rolled back. | Usability | Customer Dissatisfaction |
| TikTok Follower Glitch | Users saw follower counts reset to zero. | Customer Dissatisfaction | Loss of Reputation |
| Colonial Pipeline Ransomware Attack | A ransomware attack disrupted fuel supply across the US East Coast. | Financial Loss, Public Disruption | Security Breach |
| Toshiba Ransomware Attack | Toshiba’s European division suffered a ransomware breach. | Security Breach | – |
| Call of Duty Warzone Glitch | A bug allowed unfair loadouts in gameplay. | Usability | Customer Dissatisfaction |
| NHS COVID App Outage | App crashed, affecting access to vaccine travel passes. | Usability | User Dissatisfaction |
| Tesla Full-Self Driving Recall | Cars braked suddenly due to a software bug. | Safety Risk | Brand Reputation |
| GTA Trilogy – Definitive Edition Fail | Bugs and visual issues led to negative feedback and backlash. | User Dissatisfaction | Brand Reputation |
| Log4j Vulnerability | A critical bug exposed millions of systems to potential attacks. | Security Breach | – |

**Table 2: How These Issues Could Have Been Prevented**

|  |  |
| --- | --- |
| **Attack/Issue Name** | **How the Issue Could Have Been Prevented** |
| T-Mobile Data Breach | Stronger security systems and better-quality testing could have stopped the breach. Many users lost trust. |
| Slack Public DM Feature | Better feature planning and stronger privacy controls would have prevented strangers from sending unwanted messages. |
| TikTok Follower Glitch | The issue could have been avoided through proper test planning and data integrity checks. |
| Colonial Pipeline Ransomware Attack | Regular security updates and system monitoring could have protected against the ransomware threat. |
| Toshiba Ransomware Attack | Improved cybersecurity practices, including frequent testing and updates, could have reduced the risk. |
| Call of Duty Warzone Glitch | In-depth testing and gameplay simulation would have caught the glitch before it went live. |
| NHS COVID App Outage | Load testing and performance monitoring would have helped prevent the app crash at a critical moment. |
| Tesla Full-Self Driving Recall | More rigorous real-world testing of the software could have identified braking issues early. |
| GTA Trilogy – Definitive Edition Fail | Thorough quality control and regression testing would have improved the game experience before launch. |
| Log4j Vulnerability | Secure development practices and quicker response to patching would have minimized exposure to this threat. |

**Summary and Key Takeaways**

This analysis highlights the importance of strong planning, thorough testing, and continuous updates in software development. Many of the failures in 2021 could have been avoided with:

* Regular and proactive security measures
* Detailed test plans that consider real-world scenarios
* Focus on usability and user experience
* Monitoring and handling of system performance under load
* Fast responses to vulnerabilities when identified