Scientific Research Publishing

Open Journal of Optimization, 2023, 12, 25-33
https://www.scirp.org/journal/ojop
ISSN Online: 2325-7091
ISSN Print: 2325-7105

Data Integrity and Risk

Sasidhar Duggineni

PPD Part of Thermo Fisher Scientific, Mason, Warren, OH, USA
Email: Sasidhar.duggineni@gmail.com

How to cite this paper: Duggineni, S. (2023) Data Integrity and Risk. *Open Journal of Optimization*, **12**, 25-33.
https://doi.org/10.4236/ojop.2023.122003

Received: April 14, 2023
Accepted: June 4, 2023
Published: June 7, 2023

Abstract

Data Integrity is a critical component of Data lifecycle management. Its importance increases even more in a complex and dynamic landscape. Actions like unauthorized access, unauthorized modifications, data manipulations, audit tampering, data backdating, data falsification, phishing and spoofing are no longer restricted to rogue individuals but in fact also prevalent in systematic organizations and states as well. Therefore, data security requires strong data integrity measures and associated technical controls in place. Without proper customized framework in place, organizations are prone to high risk of financial, reputational, revenue losses, bankruptcies, and legal penalties which we shall discuss further throughout this paper. We will also explore some of the improvised and innovative techniques in product development to better tackle the challenges and requirements of data security and integrity.

Keywords

Data Governance, Data Integrity, Data Management, Data Security, Technical Controls, Regulations

1. Introduction

Data integrity fundamentally refers to the assurance of data reliability and authenticity over its life cycle. It also further refers to the completeness, accuracy, and consistency of data. To summarize, it can be thought of as trustworthiness of the data throughout its life cycle, starting from the moment it was generated. A data integrity control is a control that is placed to ensure data integrity. These controls include but not limited to preventing unauthorized data inputs/outputs, rejecting invalid data inputs, and protecting data and programs against tampering either accidental or malicious. Bad and unreliable data can potentially cause companies millions of dollars in revenue loss and can have far reaching consequences beyond financial damage for companies. According to a survey con-

ducted by the research firm Gartner, corporations estimate poor data integrity to be the cause for an average of $15 million per year in financial losses. Data integrity is critical for industry leaders, corporations, businesses, and governments alike. To implement data integrity, institutions commonly encrypt their data, implement frequent data backups, maintain audit trails to track sources of any data manipulation and most importantly, implement access controls. But the extent of above-mentioned controls or any other controls varies from application to application and from industry to industry. An important and strategic component of any business or organization is the data they own or have custody for. It is a core component that is used to derive company strategies and business decisions. Having good, reliable, and fool proof data is extremely important, regardless of company size. After all data helps make informed decisions on products, clients, and customer engagement impacts. Data integrity requires that data be reliable, authentic, attributable, consistent, and valid. It also requires data to be complete, and accurate. However, it must be noted that data integrity is not synonymous with data security, although both usually go hand in hand. Data security is concerned with preventing unauthorized access, securing infrastructure, and preventing data corruption. It can include systems, methods and procedures set up to protect data from security threats.

2. Importance of Data Integrity

Every day, an ever-increasing amount of data is generated, captured, and used by organizations worldwide to make business critical decisions. Data is a crucial differentiator in companies as more and more data are generated, used, transmitted, and stored. Reliable and meaningful data provides companies with a performance edge and lets them compete with their competitors. The potential for errors increases as the amount of data increases. The consequences of such errors can have a huge impact on consumers and can potentially affect vast number of people.

Given the importance of data and ever-increasing reliance on data to make decisions, organizations need to be able to trust the decisions they are making based on their data. Data integrity is an important component to be able to trust the insights and crucial decisions being made based of data. Rigorous data integrity would save time, effort, and financial toll it would take to make crucial decisions based on data with inadequate data integrity standards.

Data integrity ensures data is discoverable, reliable, consistent, and recoverable. Protecting data's integrity can improve optimization of resources as it increases reusability. Thus, when data integrity standards are strong and vigorous, the data remains untampered and dependable, regardless of how often it is accessed, transmitted or how long it has been stored.

Furthermore, data integrity inherently ensures confidentiality where it is essential. For instance, companies might collect their customer credit card information or their social security numbers which can be used to process financial

transactions specific to individuals. However, any accidental or malicious manipulation can cause clients to not only be misrepresented but can also lead to identity theft, loss of customer trust and financial losses. The risks and costs of identity theft are high and can severely affect businesses and customers.

Data Integrity can be adversely impacted due to various reasons. One of those reasons can also be human error. Data Integrity can be compromised when stakeholders or employees fail to follow procedures or policies or instructions which were designed to ensure data integrity. To give an example on technology related failures: data transfer errors can also cause data integrity failures when data was not transferred accurately and effectively from source to destination. In relational databases, these data integrity compromises can occur when there are data discrepancies between destination and source tables due to inadequate correlation between source and destination data elements.

Bugs, viruses, and malware can also contribute to the compromises of data integrity as they can facilitate unauthorized access to data, manipulation, and theft of valuable data. Lastly, poor hardware can also compromise data integrity. Frequent server, device, or computer crashes as well as low performance of electronic devices or computers can potentially indicate compromised hardware. Such devices can render data incomplete or incorrect or they can even remove or reduce data access. [1]

3. Technical Controls

Technical controls refer to hardware or software components that are designed to protect systems against unauthorized access, unauthorized modifications, data manipulations, audit log tampering, data backdating, data falsification and cyberattacks. These controls if implemented properly can keep most of access control compromises at bay, detect, inhibit variety of data integrity and data security violations. Technical controls help maintain data integrity by providing protection for data stored in physical databases as well as when data is being transferred and accessed across networks. Technical controls can be implemented in a variety of ways including setting up system architecture, network architecture, procedures, or even through programming language code. [1] [2] [3]

Controls can be of different types based on their objectives. The first type of data integrity goal is preventive control, which is designed to prevent security and integrity incidents. Other type of controls called as detective controls are designed to detect threats and incidents either immediately or within an established threshold time frame. Corrective controls aim to reduce or eliminate the adverse effect of the threat or violation after it has occurred. Other types of controls include deterrent controls which attempt to deter actions from personnel or bad actors causing data integrity or security compromises. Firewalls, antivirus, IDS or intrusion detection systems, and encryptions are some common examples of technical controls. In later sections, we will go into various types of forementioned controls in a more detailed manner.

Encryption is an example for a protective control which implements security by scrambling or masking data. Unauthorized users cannot access and read the actual information or data since it appears to be random masked characters. On the back end, encryption uses patterns and algorithms to mask data. With the right key, which only authorized users have, the data is unmasked, and those users are allowed to view it in a meaningful way. [1]

Firewalls work by monitoring the incoming and outgoing network traffic. It is designed to block unwanted traffic from accessing the network. They typically exist between a private network and the internet and essentially act as a network border screening tools for network packets. Firewalls detect threats and then prevent them from accessing the network, thus they can be considered as both detective and preventive type of technical control. [2] [3]

Antivirus software constantly monitors a device for threats such as viruses, bugs, and other malware on end point level. They can also scan new and open files for malware. Typically, antivirus runs in the background and periodically it may run a comprehensive scan of the device and notify the user if any threat(s) are detected. It can also alert the user of potential actions to take. It thus acts as both a detective and preventive control. These are just a few examples. The National Institute of Standards and Technology has an extensive framework of these controls. [1]

Technical controls are needed to protect the data both while it is being created, at rest, use and while it is in motion across the network. They are essential to implement an effective data security and data integrity.

4. Data Integrity as a Code (A Novel Approach)

A range of secure coding standards such as OWASP, ISO 27001, SEI CERT C, NIST, PCI DSS have been created over the years based on demand, encompassing different domains and industries. All of them integrate security as code to ensure systems and databases remain safe. The secure coding standards are a set of rules and guidelines that help mitigate security threats by detecting and proactively mitigating security risks and vulnerabilities. Security as Code embeds security in the foundation of the system and forces developers to always reinforce code from a security standpoint. Secure coding best practices include the following:

- Input validation
- Design architecture that complies with security policies
- Simple code
- Deny access as default.
- Principle of least privilege.
- Sanitize Data
- Défense in depth
- Modelling threats

A newly evolved and improvised framework of data integrity as code (DIaC) can be implemented in a similar fashion. Data validations, audit logging, data

transmission thresholding, data pipeline monitoring change controls, backups, archiving, and built-in access controls can be implemented at the code level to ensure data integrity is preserved. This design methodology along with traditional software development methodology of choice is a very effective way of preventing malicious actors from forging data, manipulating logs, or executing unauthorized data edits or deletions. Data integrity as Code (DIaC) is increasingly finding its way as a standard in code compiling and code reviews. The need for DIaC arises from the fact that data integrity and security has become a staple in today's challenging product development landscape. DIaC is no longer seen as a good to have but in fact as a must have when it comes to application development. Businesses and institutions need to actively deploy DIaC frameworks that will help mitigate bulk of data integrity challenges. Developers should revise their stock coding standards by incorporating data integrity at the code level like compilers, binaries, libraries, and IDE's. This DIac Framework was derived from the results of a case study along with a comparison analysis which will follow in further sections. Please see **Figure 1** below.

Data integrity is also increasingly becoming a focal point for governmental regulations and legislations. The situation is evident with the number of warning letters over data integrity issued by the FDA grew nearly six times over the span of six years (2013-2018). Lachman Consultants reported FDA fines doubled since 2015 where 50% were data integrity breaches. Companies can expect fines up to $1 Million per incident or even more depending on the severity of breach and type of data involved in the breach. [4] [5]

Please see **Figure 2** showing the trends in data integrity warnings from FDA.

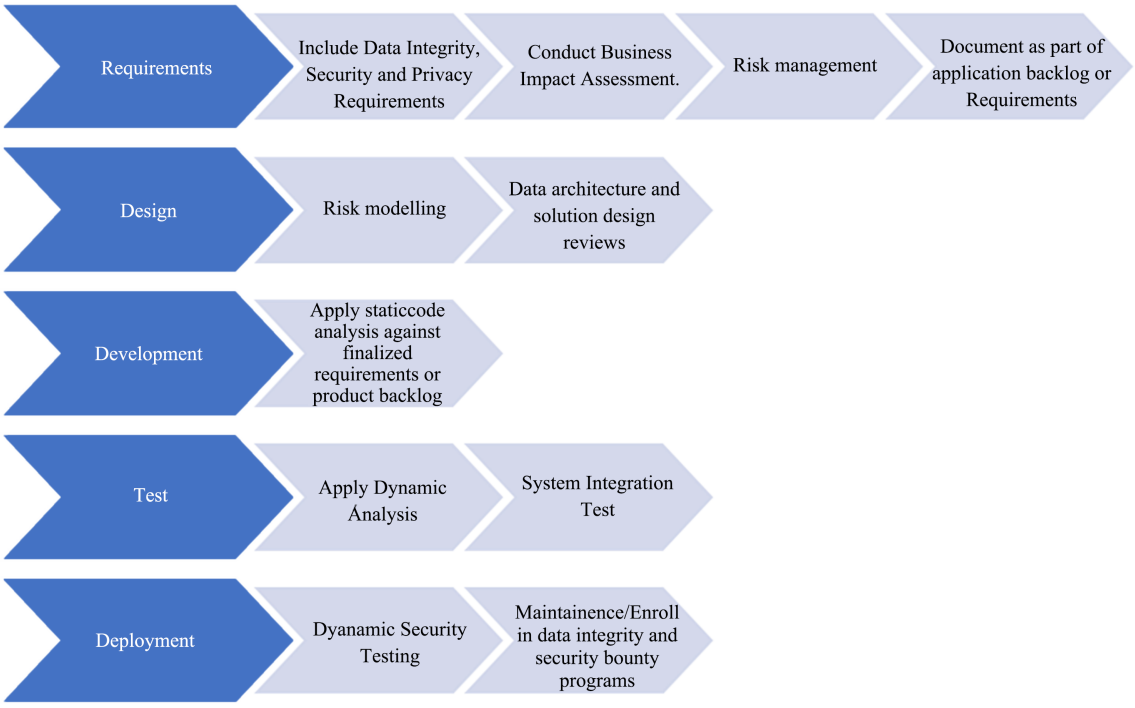

Figure 1. Basic DIaC framework that can be used as a starting ground for product development teams.

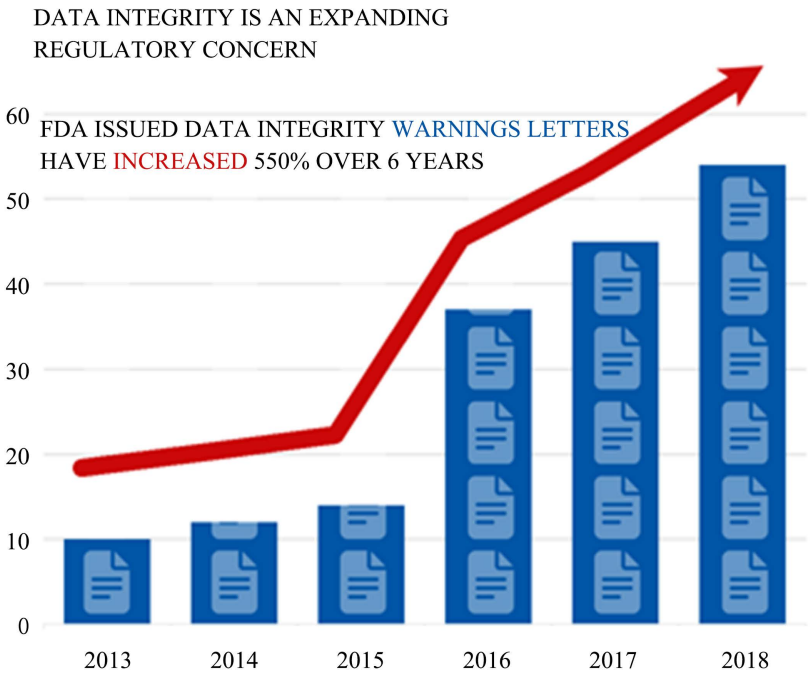

Figure 2. Data integrity warnings issued by FDA from 2013-2018.

4.1. Regulations and Laws Governing Data Integrity in Industries

The US Food and Drug Administration (FDA) ensures data integrity in the pharmaceutical industry by enforcing the 21 Code of Federal Regulations (CFR). 21 CFR Part 11 went into effect in 1997 to extend data integrity regulations into the modern era with electronic records and electronic signature standards. The World Health Organization, European Medical Agency and United Kingdom's Medicines and Healthcare Products Regulatory Agency each introduced their own guidelines on data integrity compliance. [6] [7] [8] [9]

The US Sarbanes-Oxley Act (SOX) is a US federal law that sets strict standards for safeguarding the integrity of financial data. The US Health Insurance Portability and Accountability Act (HIPAA) provides federal protection of patients' health data against misuse or exposure and requires technical and administrative controls to ensure compliance. [10]

Non-Compliance with the above-mentioned laws and regulations can lead to serious consequences like heavy fines, loss of reputation, banning products from markets, financial losses, and lawsuits. For 21 CFR Part 11, U.S Food and Drug Administration can issue direct and indirect penalties, warning letters which can affect company's stock values and revenue generation. Penalties for non-compliance with SOX include fines, removal from delisting's in public stock exchanges, punishment to CEOs and CFOs who willfully submit incorrect certification to SOX compliance ranging up to $5 million and 20 years in jail. HIPAA violation fines and penalties result from failing to comply with HIPAA rules, resulting in civil and criminal penalties, depending on the type and severity of the violation. Fines for HIPAA violations range between minimum and maximum

amounts and have a calendar-year cap of $1,919,173 for multiple violations of an identical HIPAA provision. Penalties for non-compliance with the data protection rules contained within the GDPR can be harsh, including GDPR fines reaching millions of Euros. [6] [7] [8] [9] [10]

4.2. Case Study

A case study conducted on an already productionized application from the year 2016 shows the gradual increase in quality, reduction in data integrity violations and significant elimination of regulatory risks over the span of seven years after hybridizing the existing software methodology by incorporating both data integrity and data security as part of coding standards (**Figure 3**).

4.3. Comparative Analysis

Organizations should retrospect their product development methodologies from time to time by evaluating the scope for inclusion of Data Integrity as Code (DIaC) to fully meet the compliance requirements with rapidly evolving regulations and operational requirements on data integrity and security. A comparative study between two applications was conducted at a medium-sized healthcare company. One of the applications was a Clinical Trial Management System (CTMS) without the Data Integrity as a Code (DIaC) framework, the other application being a Patient Monitoring Application (PMA) with the DIaC framework (**Figure 4**).

Application developers will need to rethink data integrity and adopt improvised and customized approaches. Hybrid SDLC methodologies that integrate data risk analysis results into code development is a method that can help fill gaps in data integrity compliance. Below **Table 1** depicts a comparison between the traditional and hybrid SDLC methodologies.

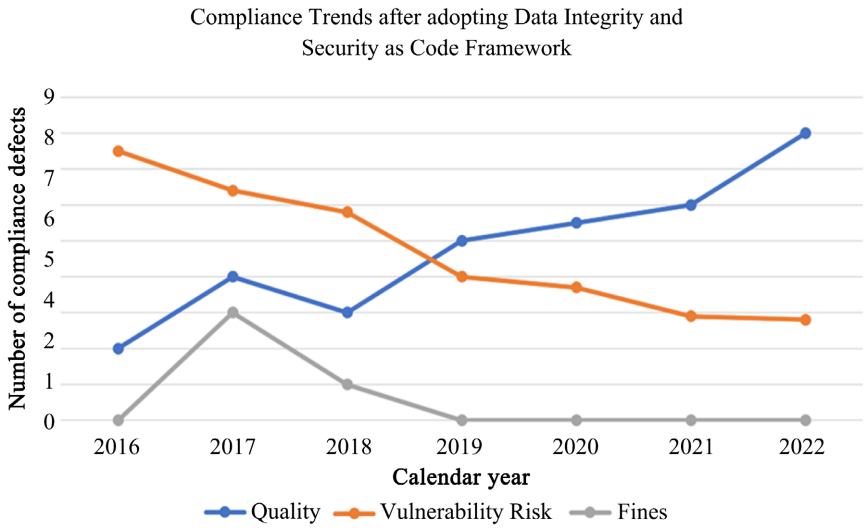

Figure 3. Compliance trends after adopting data integrity and security as code framework (Timeline of 7 years).

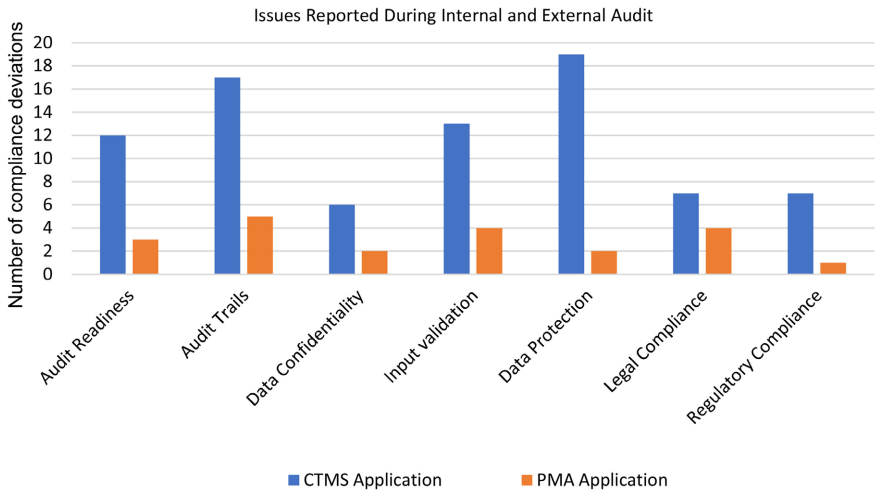

Figure 4. The comparative results between the two applications (CTMS & PMA) highlight the stark difference in compliance and security issues found during internal and external audits.

Table 1. Compliance compatibility comparisions between Traditional SDLC Methodology vs Data Integrity as code infused SDLC Methodology.

Compliance Requirement	Data Integrity as a code infused SDLC Methodology	Traditional SDLC Methodology
Privacy	Fully Supports	Partially Supports
Audit Trails		
Data Storage Protections		
Regulatory Compliance		
Digital Forensics suitability		
Data Confidentiality		Does not support
Audit Readiness		
Legal Compliance		

The case studies discussed earlier are derived from the first-hand research and analysis conducted by the author. As the digital landscape heats up, it becomes even more critical for businesses to ensure their data and systems remain safe. As emphasized prior, implementing data integrity and technical controls is a crucial step when dealing with data.

5. Conclusion

We discussed data integrity, an innovative and improvised framework to better tackle it, along with its standard technical and process controls. The importance of data integrity is undeniable. Companies with bad data evidently suffer great reputational losses and lose out in revenue and customers to their competitors. By putting effective data integrity measures into place, a business can protect it-

self from data breaches, unnecessary financial expenditures, loss of public trust, potential threats to brand reputation, and loss of future revenue. In a nutshell, it is viable to confirm that hybridization of software methodology of choice with Data Integrity as a Code (DIaC) framework can immensely help enterprises across all industries to remain proactive in terms of compliance and data integrity controls on a comprehensive scale, while also fulfilling the relevant functional and performance requirements of the system.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Security and Privacy Controls for Information Systems and Organizations (2020) https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final

[2] These Real-World Data Breach Examples Will Make You Rethink Your Data Strategy. https://www.doherty.co.uk/blog/data-breach-examples-rethink-your-data-strategy/

[3] OWASP Security Knowledge Framework. https://owasp.org/www-project-security-knowledge-framework/#:~:text=The%20OWASP%20Security%20Knowledge%20Framework,that%20are%20secure%20by%20design

[4] Tunggal, A.T. (2023) The 70 Biggest Data Breaches of All Time. https://www.upguard.com/blog/biggest-data-breaches

[5] Todd, D. (2022) Top 10 Data Breaches of All Time. https://www.secureworld.io/industry-news/top-10-data-breaches-of-all-time

[6] Code of Federal Regulations, Part 11—Electronic Records; Electronic Signatures. USA, 20 March 1997. https://www.ecfr.gov/current/title-21/chapter-I/subchapter-A/part-11

[7] European Medicines Agency (2016) Data Integrity: Key to Public Health Protection. https://www.ema.europa.eu/en/news/data-integrity-key-public-health-protection

[8] World Health Organization (2020) Guideline on Data Integrity. Switzerland. https://www.who.int/docs/default-source/medicines/norms-and-standards/current-projects/qas19-819-rev1-guideline-on-data-integrity.pdf

[9] Medicines and Healthcare Products Regulatory Agency (2016) MHRA GxP Data Integrity Definitions and Guidance for Industry. https://www.gov.uk/government/news/mhra-gxp-data-integrity-definitions-and-guidance-for-industry

[10] H.R.3763 Sarbanes-Oxley Act of 2002 (2002) https://www.congress.gov/bill/107th-congress/house-bill/3763