**Fundamentals of Cybersecurity for Databases**

Ishan Prabhune (A20538828)

[iprabhune@hawk.iit.edu](mailto:iprabhune@hawk.iit.edu)

College of Computing,

Illinois Institute of Technology

ITMS 528: Database Security

Prof. Maurice Dawson

August 29, 2023

**Introduction to Cybersecurity for Databases**

The importance of cybersecurity for databases cannot be emphasized in the current digital environment, when data has become both extremely valuable and insecure. Among the many domains that demand stringent cybersecurity measures, databases stand out as repositories of invaluable information. Protecting sensitive data contained in databases has become essential due to the ever-evolving sophistication of cyber-attacks. The foundation of this protection is built on the core ideas of cybersecurity as they apply to databases.

Database security encompasses a diverse range of concerns that stem from the dynamic interplay between technology, law, and policy. This multifaceted landscape addresses several crucial issues that underscore the importance of safeguarding sensitive information.

* Legal and ethical considerations are paramount, with the recognition that certain data demands restricted access. The notion of privacy takes centre stage, as laws and regulations govern access to private information, thus protecting individuals and organizations from unauthorized intrusions. This is exemplified in the United States by an intricate web of privacy laws.
* The creation of policies plays a significant role in database security. Governments, organizations, and businesses must decide what information should be kept private and off-limits to the general public. Among the areas deemed deserving of increased protection are credit scores and personal medical details.
* Another crucial aspect of the security framework is its hierarchy, which raises concerns regarding the ideal level at which security measures are to be implemented. To ensure thorough security against potential threats, it's crucial to strike the correct balance, whether at the hardware, operating system, or database management system level.
* Organizations working with classified data must build numerous security tiers as a matter of urgency. It takes careful execution to classify individuals and data into categories like top secret, secret, confidential, and unclassified. A significant obstacle that needs to be overcome is the implementation of a security policy that restricts access based on these classifications.

In essence, database security concerns cross legal, policy, system, and hierarchical lines, forming a thorough framework that safeguards the confidentiality and integrity of vital data. Due to the importance of database security in the current digital environment, the convergence of these elements highlights how urgent it is to protect databases from potential intrusions.

Access control, inference control, flow control, and encryption are four primary control mechanisms used to safeguard databases from these kinds of risks.

In a multiuser database system, the DBMS must offer methods that permit particular users or user groups to access specific database areas while preventing access to the entire database. This is crucial when numerous users inside the same firm will be using a sizable integrated database.

Sensitive information, such as employee wages or performance reports, for instance, should be kept private from the majority of database system users. A DBMS often has a database security and authorization subsystem that is in charge of protecting certain database areas from unauthorized access. The two types of database security mechanisms are Discretionary security mechanisms & Mandatory security mechanisms.

**Importance of Database Security**

1. **Threat Landscape**

The realm of databases is not immune to a multitude of threats that can severely compromise their integrity, availability, and confidentiality – the three foundational pillars of security.

* Integrity, one of the pillars of database security, comprises protecting data against unauthorized changes. Information's accuracy and dependability can be compromised by unauthorized changes, whether they are deliberate or unintentional. When ignored, weakened integrity can result in mistakes, inaccuracies, and even fraudulent acts, undermining the accuracy and reliability of the data and the judgments drawn from it.
* The second important factor, availability, ensures that the database will always be accessible to applications and authorized users. Databases may become inaccessible as a result of threats aimed at availability, causing a halt to crucial operations and interruptions. Such hiccups can have an impact on an entire organization, reducing productivity and sometimes even costing money.
* The third aspect, confidentiality, prevents the disclosure of sensitive information without permission. Such breaches have grave consequences, which can range from legal action for violation of data privacy to jeopardizing national security. This can have penalties which include severe legal repercussions along with loss of public trust and reputational harm.

Examples of Recent Data Breaches:

* Neopets : A virtual pet game

July 2022

Impact on 69 Million Users

Description -  Hackers breached Neopets’s database and [stole the personal data of potentially 69 million users](https://www.cpomagazine.com/cyber-security/data-breach-on-virtual-pet-website-neopets-affected-69-million-users-and-leaked-source-code/) (current and former) and 460 MB of source code. The hackers had access to this database from January 3, 2022, to July 19, 2022. The attackers attempted to sell this data for four bitcoins, which alerted Neopets to the breach.

The data stolen included users’ personal information such as names, usernames, email addresses, Ip addresses, gender, date of birth, PIN for Neopets, hashed passwords, and information about the users’ pets and gameplay.

* Facebook :

March 2021

Impact on 533 million users

Description - Hackers scraped Facebook due to a security gap that the company had patched back in 2019. As a result, [533,000,000 user records from 106 countries](https://www.businessinsider.com/stolen-data-of-533-million-facebook-users-leaked-online-2021-4?r=US&IR=T) were posted on a hacking forum. The leaked information included user locations, full names, biographical information, phone numbers, and email addresses. This information was discovered when a user in the hacking forum promoted an automated scraping bot that could extract phone numbers for hundreds of millions of Facebook users.

* Aadhaar Data Breach (Indian ID Proof) :

Impact on 1.1 Billion people

Description - In March of 2018, it became public that the personal details of more than a billion citizens in India stored in the world’s largest [biometric](https://www.clearscope.io/blog/biometrics) database could be bought online. This massive data breach was due to a [data leak](https://www.upguard.com/blog/data-leak) on a system run by a state-owned utility company. This breach gave access to the private information of [Aadhaar](https://www.upguard.com/security-report/uidau" \t "_blank) holders, exposing their names, their unique 12-digit identity numbers, and their bank details along with this type of information the photographs, thumbprints, retina scans and other identifying details were also exposed of nearly every Indian citizen.

In this complex digital landscape, threats to databases come in various forms, endangering the core principles of security. Mitigating these threats demands a comprehensive and proactive approach, integrating robust security measures to preserve data integrity, ensure uninterrupted access, and protect against unauthorized data exposure. As databases continue to be repositories of critical information, addressing these threats becomes not only a technical imperative but also a strategic necessity.

1. **Data Sensitivity and Regulatory Compliance**

Data sensitivity, which establishes the degree of protection necessary for various forms of information, is crucial in determining the parameters of database security procedures. Data sensitivity is a qualitative evaluation of how important and private data is, indicating the degree to which it requires protection. Designing and putting into practice effective security methods requires an understanding of the significance of data sensitivity.

Sensitivity classification serves as a guideline for database security. Databases can be anywhere along a spectrum, from being devoid of sensitive data to having all sensitive data. Through access controls, these two extremes can be managed rather easily. Complexities develop, though, when databases contain both sensitive and non-sensitive data. For a number of reasons, data may be considered sensitive:

* Inherent Sensitivity: Due to their very nature, some data are inherently sensitive. Examples include a person's income or a medical record.
* Source Sensitivity: Information with clandestine or private origins requires privacy, such as the identity of an informant.
* Declared Sensitivity: Data owners have the option of formally designating a subset of their data as sensitive.
* Record or Attribute Sensitivity: A database's sensitive attributes or records, such as a worker's wage history, may be marked.
* Contextual Awareness: Although data may not by itself be sensitive, it might become sensitive when paired with additional data. For example, the context of a classified incident may make geographical coordinates sensitive even though they may not be alone.

Maintaining an organization's security rules depends heavily on the work of a database administrator and security administrator. They choose access rights while weighing the demands of data accessibility, access restriction, and authenticity assurance.

1. Data Availability: Accurate data must be available and accessible. During changes, data integrity is ensured through concurrency control methods.

2. Access Acceptability: Only authorized users should be given access to data. If it can unintentionally divulge sensitive information, even non-sensitive material may be withheld.

3. Authenticity Assurance: Data disclosure is influenced by user attributes, query history, and time of access. These elements shield unapproved data exposure.

A trade-off frequently occurs when precision and security are sought after. Sensitive data must be protected, frequently at the expense of non-sensitive data. Contrarily, precision aims to protect sensitive information while disclosing as much non-sensitive data as possible. It's critical to find the correct balance between data protection and accessibility by balancing these factors.

Determining the proper security measures for databases ultimately depends on having a clear understanding of how sensitive the data is. Security measures establish what needs to be secured, who will have access to it, and how its veracity can be confirmed. These factors together provide the framework for a robust database security system that will safeguard all crucial data from threats.

The Overall Information Assurance Guideline (GDPR) and the Medical coverage Transportability and Responsibility Act (HIPAA) are the pertinent administrative structures that decisively influence the scene of data set security by laying out unambiguous rules that organizations should comply with. These structures ensure the protection, security, and safe treatment of delicate information with regards to individual and clinical data.

* Due to its extraterritorial reach, the European Union's major rule, the GDPR, has an effect on the entire world. It emphasizes safeguarding the personal information of residents and citizens of the EU. The GDPR requires strict safeguards for data, making it important for database security. Organizations are required to establish strong security measures to guard against breaches and unauthorized access and get explicit consent before collecting and processing personal data. Organizations are obligated to notify data breaches within 72 hours. The right to be forgotten and the right of access to one's own data highlight data subject control even more.
* HIPAA, on the other hand, is unique to the American healthcare industry. It establishes requirements for protecting protected health information (PHI). Database security is crucial because the HIPAA Security Rule requires administrative, physical, and technical safeguards for electronic PHI (ePHI). Access controls, encryption, audits, and contingency plans must be put in place by covered entities, such as healthcare providers and insurers, to guarantee the security, integrity, and availability of ePHI. Business Associate Agreements are also required by HIPAA with third-party service providers to guarantee that they follow security protocols.

To evaluate and reduce potential privacy issues, both models demand a Privacy Impact Assessment (PIA). Compliance with the GDPR and HIPAA requires ongoing audits, monitoring, and documentation.

In conclusion, legal frameworks like GDPR and HIPAA play a key role in promoting strict security procedures for databases. They place a strong emphasis on data protection while highlighting consent, responsibility, transparency, and data subject rights. In order to ensure compliance, avoid penalties, and, most importantly, safeguard the confidentiality and integrity of the data they manage, organizations subject to these standards must carefully implement the necessary security measures.

**Fundamentals of Database Security**

1. **Access Control and Authentication**

* Access Control and Authentication in Database Security:

**The process of controlling who has access to particular resources within a system is known as access control. Access control in relation to database security makes ensuring that only users who have been given permission can interact with the database and its contents. This requires determining client jobs, authorizations, and cutoff points to protect the security, respectability, and accessibility of information.** **Verification, then again, is the most common way of checking a client's or an element's character. It gives confirmation that the singular utilizing the data set is who they guarantee to be. In order to give or restrict access, this frequently needs credentials like usernames and passwords that are validated against stored information.**

* Role-Based Access Control (RBAC) and Least Privilege Principle:

**Users are given rights based on their positions and responsibilities inside an organization using the role-based access control (RBAC) concept. Specific roles with predetermined access rights are assigned to users. Due to the fact that permissions are linked to roles rather than specific users, access management is made simpler. According to the Least Privilege Principle, which is a security best practice, users should only be given the minimal amount of access required to do their tasks. By doing this, the potential harm from a security breach or misuse is reduced. For instance, a manager may have read and write access while a clerk simply has read access.**

* Multi-Factor Authentication (MFA) and Its Significance:

**Multifaceted validation, or MFA, is the cycle through which clients should at first give numerous types of personality to get to a framework. It offers extra layer of assurance on top of the ordinary username and secret phrase mixes. MFA frequently combines the user's knowledge (password), possession (a tangible object like a smartphone), and identity (biometric information like fingerprints or facial recognition). An example would be for a user to input a password, get a verification code on their phone, and then scan their fingerprint. MFA dramatically improves security by reducing the dangers posed by compromised or stolen credentials. Even if one element is weak, it prohibits illegal entry.**

**Examples of Multi-Factor Authentication:**

* SMS or Email Verification:

Users receive a one-time code by SMS or email after inputting their password, which they must enter to complete the authentication process.

* Mobile App-Based Authentication:

Users download a mobile app that creates OTPs based on events or on time. To access the database, they must type in the OTP that is displayed on the app. Similar to Microsoft Authenticator and Okta Verify.

* Security Questions and Answers:

Users must successfully respond to specified security questions in addition to their passwords in order to access the database.

**In conclusion, access control and authentication are essential parts of database security, making sure that only people who are supposed to be there can interact with critical information. The Least Privilege Principle and Role-Based Access Control simplify permissions administration, while Multi-Factor Authentication increases security by demanding various forms of identification. Together, these ideas support preserving databases' confidentiality, integrity, and accessibility while protecting sensitive data from potential dangers.**

1. **Encryption and Data Masking**

* Role of Encryption in Data Security:

Sensitive data can be protected by encryption both at rest and while being transmitted. When data is at rest, lying in databases or storage systems, encryption uses sophisticated algorithms to convert the data into an unreadable format. This makes sure that even in the event of unwanted access, the data will still be impossible to decrypt without the associated decryption key. Encryption scrambles the data while it is in motion, during transmission over networks or between computers, guarding against unauthorized access. Encryption efficiently prevents eavesdropping and data breaches by ensuring that only authorized parties with the proper decryption key can access and interpret the information.

* Concept of Data Masking and its Relevance:

A strategy for protecting privacy called data masking includes replacing the actual sensitive data with fake but accurate values. By doing this, genuine data is kept secret while preserving its format and properties. This is especially important when testing, analysing, or training needs realistic data in non-production contexts. Data masking ensures that even if unauthorized individuals obtain access, the exposed data is useless and poses no risk, preventing unwanted access to critical information. For instance, to conceal the actual data, a credit card number might be disguised as a similar-looking string of characters.

* Difference between Encryption and Data Masking:

In the field of database security, encryption and data masking are two separate yet complimentary strategies. Through the application of sophisticated algorithms, encryption aims to make data incoherent while maintaining its original content and making it unreadable without the right decryption keys. Data masking, in contrast, entails modifying the original data with made-up but contextually appropriate information, preserving the data's natural structure while hiding its true value.

Data masking is very helpful in non-production settings where genuine data is required for testing, analytics, or training but its sensitive nature must be masked. While encryption is primarily meant to secure data from unauthorized access during storage and transfer, data masking is particularly effective in those situations. Data masking is particularly effective against insider threats or unintentional exposures in non-production environments with comparison to encryption, which protects data from both internal and external threats.

In terms of database protection, encryption and data masking each have certain security goals. Data integrity is protected during storage and transmission by encryption, and even in restricted, non-production situations, critical information is kept private through data masking. Both methods make a substantial contribution to an all-encompassing database security strategy and offer defence against a variety of potential dangers.

**Security Measures and Best Practices**

1. **Patch Management and Vulnerability Assessment**

* Patch Management in Database Security:

**Finding, applying, and managing software upgrades, or "patches," to correct bugs and security problems in databases and other software applications as well as computer systems is the process of patch management. Because it guarantees that vulnerabilities are patched as soon as they are found, patch management is crucial to database security. Coding mistakes or defects might lead to vulnerabilities that could be abused by bad actors. Organizations can lower their risk of cyberattacks and unauthorized access by applying updates on a regular basis. In addition to enhancing database security, patch management also protects the entire IT infrastructure from potential hacks and data compromise.**

* Importance of Regular Vulnerability Assessments:

**A strong database security posture must be regularly maintained, which means conducting regular vulnerability assessments. In these evaluations, databases and related systems are methodically scanned to find any potential security flaws, errors, or vulnerabilities. The necessity of routinely carrying out such evaluations stems from the fact that as attack methods and cyber threats evolve, new vulnerabilities appear. Regular assessments enable organizations to proactively resolve vulnerabilities before they can be exploited, allowing them to stay ahead of possible dangers. Early vulnerability detection allows companies to put in place the right security controls, such installing patches, changing settings, or improving access controls, to reduce possible threats.**

* Examples of Common Database Vulnerabilities and Impact

1. SQL Injection: When malicious code is introduced into input fields to alter database queries, this happens. The result can compromise data integrity and confidentiality by allowing unwanted data access, altering, or deleting data.
2. Weak Authentication and Authorization: Unauthorized access may result from weak authentication systems. The manipulation or extraction of sensitive data by intruders could result in data breaches.
3. Lack of Encryption: Data exchanged between programs and databases can be intercepted if it is not properly encrypted. Data leakage or unwanted access during transmission could result from this.

**Data confidentiality, integrity, and availability breaches are among the potential effects of these vulnerabilities. If these vulnerabilities are used to an organization's advantage, it could result in data breaches, financial losses, legal troubles, and reputational harm.**

1. **Auditing and Monitoring**

* Significance of Database Auditing and Monitoring:

A solid database security plan must include database auditing and monitoring. While monitoring comprises seeing these events in real time, auditing involves tracking and recording activities within a database system. These procedures are important because they allow for the prompt detection and mitigation of security risks, illegal access, and potential breaches. By establishing an audit trail of actions, assisting in forensic investigations, and strengthening accountability, auditing assists companies in maintaining compliance with regulatory standards. On the other side, monitoring enables security teams to quickly respond to potential hazards by enabling the detection of anomalies, strange patterns, and unauthorized activity in real time.

* Utilizing Audit Logs for Incident Detection and Investigation:

In a database context, audit logs are essential for spotting and looking into security breaches. These logs record a history of user and application actions, such as access attempts, modifications, and queries. Security teams can find anomalous or unauthorized actions by looking through audit logs. Alerts may be set off by things like recurrently unsuccessful login attempts, unauthorized users accessing private information, or modifications to crucial database configurations. When an issue happens, audit logs offer a thorough timeline of what happened, helping to recreate the series of events that led up to the occurrence. This data is crucial for conducting forensic analysis, determining the extent of the occurrence, and putting the right precautions in place.

* Examples of Real-Time Monitoring Tools for Database Security:

Several tools offer real-time monitoring capabilities to enhance database security:

* IBM Guardium: This application offers ongoing monitoring and notifications for illegal access, data breaches, and suspicious activity. It provides in-the-moment analysis of database operations, facilitating quick incident response.
* McAfee Database Security: The McAfee solution provides database activity auditing and real-time monitoring. It has methods for threat identification and attack prevention, assisting organizations in their defense against assaults.
* Oracle Database Vault: Real-time monitoring features in Oracle's product enable to enforce separation of roles, keep an eye on privileged activity, and uncover unauthorized access to sensitive data.

These technologies actively keep an eye on database operations, sending out alarms and offering information about potential security issues as they emerge. They enable businesses to quickly respond to risks as they materialize and guarantee the security and integrity of their database settings.

**Risk Management and Disaster Recovery**

1. **Risk Assessment and Mitigation**

* Risk Assessment in Database Security:

The systematic evaluation of potential risks and vulnerabilities that could endanger the confidentiality, integrity, and accessibility of sensitive data within a database system is what risk assessment includes in the context of database security. It is a methodical process that seeks to identify and evaluate risks so that companies may select the best security measures and risk-reduction strategies. Organizations may efficiently prioritize and deploy resources to protect against potential threats by using risk assessment to acquire a thorough picture of their security posture.

* Risk Assessment in Identifying Vulnerabilities and Threats:

A database environment's vulnerabilities and dangers can be found by doing a risk assessment. Analysing the possible effects of threats and the chance that those threats will take advantage of current vulnerabilities is part of this process. Organizations can learn about the areas where security measures need to be improved by methodically examining vulnerabilities including configuration errors, lax access controls, and out-of-date software. Additionally, risk assessment helps in understanding the potential repercussions of a security breach, assisting companies in planning and putting into practice the proper preventive and mitigation methods.

* Strategies for Mitigating Identified Risks:

**An extensive risk assessment will reveal hazards that need to be mitigated, and best practices will be used in tandem with strategies to do so.**

* Access Controls: Implement strong access controls to make sure users have the bare minimum of authority to complete their tasks. This limits unauthorized access and reduces the attack surface.
* Encryption: To protect data while it is in transit and at rest, use encryption techniques. To guarantee that regardless of whether information is gotten, it can't be decoded without the legitimate unscrambling keys, information is protected by means of encryption.
* Regular Patching: Apply security patches as soon as they become available to keep database and software systems current. This stops the use of known vulnerabilities.
* Database Monitoring: To spot anomalies, unwanted access, and suspicious activity, use real-time monitoring technologies. This enables quick reaction to new threats.
* User Training: Inform users on security best practices, such as how to manage strong passwords and spot phishing attacks. Users that are knowledgeable are better able to prevent inadvertent security breaches.
* Incident Response Planning: Create a thorough incident response plan to deal with security incidents quickly and successfully. This lessens the effects of breaches and speeds up the healing process.
* Backup and Recovery: Practice database recovery processes and regularly backup your databases. This guarantees that information can be reestablished in case of a break or information misfortune.
* Vendor Risk Management: When utilizing third-party software, evaluate the security procedures used by vendors and make sure they follow security guidelines.

**Organizations use risk assessment to help them decide how to allocate resources and spend money on security. Organizations can improve their overall database security posture by proactively reducing vulnerabilities and successfully managing possible threats by employing these mitigation methods.**

1. **Disaster Recovery and Business Continuity**

* Importance of Disaster Recovery Planning for Databases:

Database security requires careful planning for disaster recovery. It is vital to create strategies, methods, and procedures to ensure the speedy and effective restoration of databases and linked systems in the event of a catastrophic event, such as a natural disaster, cyberattack, or hardware failure. The core of company operations, databases store vital organizational data. In addition to protecting this priceless data, a solid disaster recovery plan also guarantees business continuity by limiting downtime and lowering financial losses. Organizations may retain operations even in the face of unanticipated disruptions by having a clearly defined recovery strategy in place, protecting their brand and guaranteeing ongoing service to clients and consumers.

* Concept of RPO (Recovery Point Objective) and RTO (Recovery Time Objective):

Critical elements of disaster recovery planning are RPO and RTO:

Recovery Point Objective (RPO):

RPO identifies the maximum amount of data loss that is allowable in the event of an emergency. It denotes the moment at which data must be recovered to ensure the least amount of disturbance. The recovery method must get data back to the state it was in an hour prior to the disaster, for example, if the RPO is one hour.

Recovery Time Objective (RTO):

RTO represents the maximum acceptable downtime for systems and applications after a disaster. It indicates the time it takes to restore operations and resume normal functionality. A shorter RTO implies minimal disruption and quicker recovery.

* Examples of Database Backup and Replication as Disaster Recovery Measures:

Backup and replication of databases are essential elements of disaster recovery plans:

* Database Backup:

Backups that are routinely planned make copies of the database at various times in time. The most recent backup can be restored in a disaster to reduce data loss. In order to decrease the backup window and storage needs, incremental or differential backups are frequently used.

1. Full Backup:

An exact time-and-date snapshot of the complete database is captured by a full backup. It acts as a starting point for healing. An e-commerce database complete backup performed each night is an example.

1. Differential Backup:

Differential backups keep track of database modifications made since the last complete backup. As a result, backup takes less time and money. An email server's daily differential backups, for instance.

1. Cloud Based Backup:

Database backups are automatically copied and stored in a remote cloud environment via cloud-based backup systems, ensuring data redundancy and off-site storage. Use of AWS S3 as an example for off-site backups.

1. Incremental Backup:

Whether it was a full or incremental backup, incremental backups only store changes that have occurred since the previous backup. Instance: Financial database incremental backups performed every hour.

* Database Replication:

Replication entails continuously building and keeping copies of the database on several systems or locations. The standby system can be swiftly activated to take over operations in the event of a primary system failure. Reduced downtime and data redundancy are both provided by replication.

1. Master-Slave Replication:

Changes made to the master database are replicated to one or more slave databases in a master-slave arrangement. This offers fault tolerance and read scalability. An e-commerce website with several read-only clones is an example.

1. Multi-Master Replication:

Updates can be made on several nodes, with changes propagating to all other nodes, thanks to multi-master replication. For applications that are deployed geographically, this is helpful. An illustration would be a global CRM system with local data centers.

1. Synchronous Replication:

Changes are instantly synced across nodes thanks to synchronous replication. Applications where data consistency is important can use it. For instance, real-time transaction processing in banking systems.

1. Asynchronous Replication:

Asynchronous replication offers extra performance flexibility by allowing a delay between changes made on the master and their replication to slave nodes. Consider a news website with a global audience.

* Off-Site Storage:

Data is preserved even if the original data center is compromised by storing backups and replicas off-site, preferably in remote places. This improves catastrophe recovery and data availability.

Organizations can maintain up-to-date copies of their databases and drastically lower both RPO and RTO by putting backup and replication procedures into practice. In the case of disruptions, these procedures allow for quick data recovery and flawless business continuity, reiterating the significance of disaster recovery planning for protecting crucial data and preserving operational resilience.

**References**

1. Elmasri, R., & Navathe, S. (2010). Fundamentals of Database Systems. Addison-Wesley Publishing Company.
2. Mathew, Alex, and Chiang Mai. “Study of Various Data Recovery and Data Back Up Techniques in Cloud Computing & Their Comparison.” 2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2018, pp. 2021–24. IEEE Xplore, <https://doi.org/10.1109/RTEICT42901.2018.9012485>.
3. <https://termly.io/resources/articles/biggest-data-breaches/>
4. <https://www.upguard.com/blog/biggest-data-breaches>
5. Altaf, Insha, et al. “Vulnerability Assessment and Patching Management.” 2015 International Conference on Soft Computing Techniques and Implementations (ICSCTI), 2015, pp. 16–21. IEEE Xplore, <https://doi.org/10.1109/ICSCTI.2015.7489631>.
6. Samarati, Pierangela, and Sabrina Capitani de Vimercati. “Access Control: Policies, Models, and Mechanisms.” Foundations of Security Analysis and Design, edited by Riccardo Focardi and Roberto Gorrieri, Springer, 2001, pp. 137–96. Springer Link, <https://doi.org/10.1007/3-540-45608-2_3>.
7. Qiu, Guowei, et al. “Privacy-Preserving Linear Regression on Distributed Data by Homomorphic Encryption and Data Masking.” IEEE Access, vol. 8, 2020, pp. 107601–13. IEEE Xplore, <https://doi.org/10.1109/ACCESS.2020.3000764>.
8. Jiang, Keyuan, and Xiang Cao. “Design and Implementation of an Audit Trail in Compliance with US Regulations.” Clinical Trials, vol. 8, no. 5, Oct. 2011, pp. 624–33. DOI.org (Crossref), <https://doi.org/10.1177/1740774511413943>.
9. Jiang, Keyuan, and Xiang Cao. “Design and Implementation of an Audit Trail in Compliance with US Regulations.” Clinical Trials, vol. 8, no. 5, Oct. 2011, pp. 624–33. DOI.org (Crossref), <https://doi.org/10.1177/1740774511413943>.
10. Ometov, Aleksandr, et al. “Multi-Factor Authentication: A Survey.” Cryptography, vol. 2, no. 1, Mar. 2018, p. 1. [www.mdpi.com](http://www.mdpi.com/), <https://doi.org/10.3390/cryptography2010001>.