PROCEEDINGS
OF
LONDON INTERNATIONAL CONFERENCES
eISSN 2977-1870

Secure Data Sharing in Decentralized Networks: Encryption Techniques, Access Control Mechanisms, and Data Integrity Verification

Tensae Laki[1]
Taha Asif[2]
Nahaar Mirza[3]
Mehmet Uzgoren[4]

Abstract

The introduction of decentralized systems in a rapidly changing technological world has introduced many issues and risks regarding privacy and secure data exchange. This paper addresses the complexities of secure data exchange within decentralized systems, focusing on encryption techniques, access control mechanisms, and data integrity verification. We begin by analyzing the impending risks and limitations within decentralized systems, including the potential for unauthorized access and data breaches from vulnerable points present in decentralized system entities. Subsequently, we analyze and explore modern methods of encryption to safeguard confidential data while also ensuring privacy. Furthermore, we examine access control strategies and integrity verification protocols, which play crucial roles in ensuring that data remains secure against unwarranted access and reliable across distributed ecosystems. Our findings emphasize the need for robust and intricate security measures adapted to the unique framework of decentralized systems. This study aims to advance the understanding of privacy and security in decentralized systems while serving as a stepping stone and valuable resource for future research.

Keywords: Distributed systems, Computer network security, Data privacy, Internet of things, Decentralized networks

https://doi.org/10.31039/plic.2024.11.239

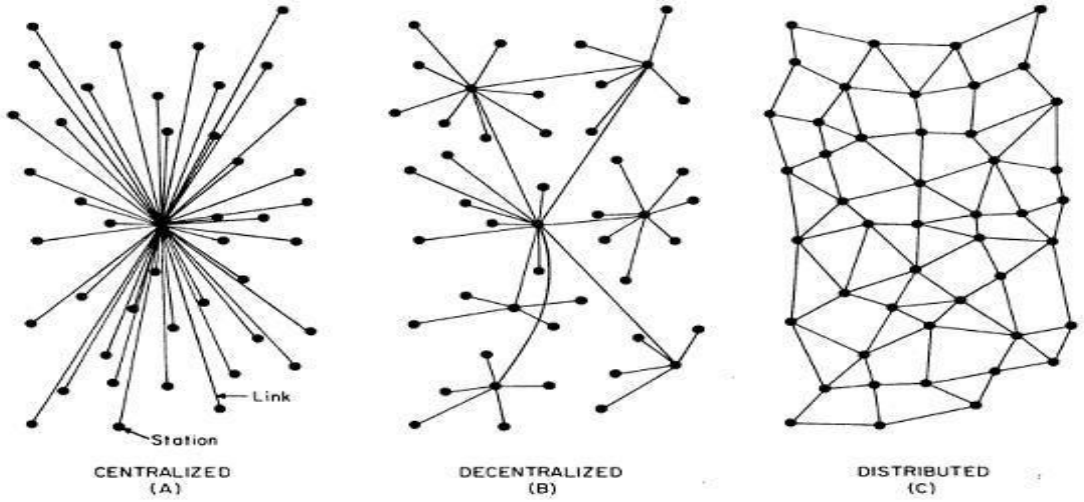
[1] HS Academy, Pflugerville, USA, tensaelaki@gmail.com
[2] HS Academy, Pflugerville, USA, tahaasif999@gmail.com
[3] HS of Innovation, Katy, USA, nahaarmirza@gmail.com
[4] Rice University, USA, . mehmetuzgoren@gmail.com

1. Introduction

Distributed systems provide a means of resource allocation between devices on a network over servers. They exist everywhere in society with unique advantages and limitations. These systems are useful in various applications since an increasingly interconnected digital world necessitates new and more efficient ways to accomplish tasks (Admin, 2020). For example, cloud computing technology is a distributed system since it splits a task into different objectives that each device can accomplish. Moreover, it allows users to access the networks remotely, making the process more easily accessible for completion. Another example of distributed systems commonly known is the World Wide Web (WWW). This is a global system of interconnected hypertext documents and multimedia content, accessible via the Internet. Users navigate the web through web browsers by entering Uniform Resource Locators (URLs) or following hyperlinks. The primary components include web servers, which host websites and serve content to users, and web browsers, which request and display this content. The communication between browsers and servers uses either the Hypertext Transfer Protocol (HTTP) or Hypertext Transfer Protocol Secure (HTTPS) protocol. The infrastructure supporting the web includes routers, switches, and other networking equipment that facilitate data transfer across the Internet. Continuing, another form of common distributed systems are cloud computing platforms, which include names like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform. Cloud computing platforms such as these provide a wide range of services, including computing power (virtual machines, containers), storage (object storage, file storage), databases (SQL, NoSQL), and networking. These services are hosted in data centers distributed globally, allowing for scalability, redundancy, and high availability. Users interact with these services through APIs or web interfaces (Eagar, 2017).

Furthermore, another well-known form of distributed systems is the information of things or IoT. IoT systems consist of interconnected devices equipped with sensors and actuators that collect and transmit data. These devices communicate with gateways, which aggregate and send the data to cloud services for processing and analysis. IoT applications range from smart homes (automated lighting, heating, security) to industrial systems (predictive maintenance, process automation). These could also be devices that use Bluetooth technology or Ethernet. Lastly, what many call the pinnacle of distributed application systems, are email systems. This is because email systems are distributed by nature, involving multiple servers and protocols to handle email delivery, storage, and retrieval. SMTP (Simple Mail Transfer Protocol) is used to send emails, while IMAP (Internet Message Access Protocol) and POP3 (Post Office Protocol) are used to retrieve emails. Email clients (e.g., Outlook, Gmail) interact with email servers to manage user messages, which are stored and replicated across server infrastructures to ensure reliability and accessibility. As we explore the intricacies and benefits of distributed systems, we must understand the closely related topic of decentralized networks, which further distribute control and bolster the independence of each participating entity.

Figure 1



Source: Centralized vs Decentralized vs Distributed Systems by Berty Technologies, n.d., *Berty Technologies*

2. Limitations of Decentralized Networks

Decentralized networks are systems in which no machine or single entity has control of the entire network. This means that each entity or machine has equal power in the system and can independently operate even if another entity fails. A key characteristic of decentralized networks is that operations and transactions can be available to all other participants. An example of a decentralized network is blockchain, which is a digital ledger technology where transactions are recorded across many computers so that the record may not be altered retroactively without interfering with the system and all other subsequent blocks. An example of blockchain technology is cryptocurrency, where we see many types of currencies traded online and over secure networks. This is also sometimes referred to as decentralized finance (DeFi). A popular example of this is BitCoin, the first and most valuable cryptocurrency that is worth tens of thousands of dollars. This currency does not have a central organization or entity controlling the trade and is done online over a user's network. Another example of a decentralized network is a Peer-to-Peer (P2P) Network, which is a system where computers share resources directly with each other without using a main server. Examples of this type are clients such as BitTorrent and Limewire, where users can download and upload files directly from and to other users. Each user contributes to the network's bandwidth by downloading and uploading simultaneously. Now that we understand decentralized networks and how they work, let's go into some limitations this type of distributed system has. One major limitation of decentralized networks is their requirement of high maintenance and increasing complexity as the systems are scaled. This is because decentralized networks require more machines and infrastructure compared to centralized systems, therefore being high maintenance. Each node in the network must also be properly configured and maintained, leading to higher operational costs and a greater burden on information technology (IT) resources. Another important risk that must be considered when using

decentralized networks is the security risks and data inconsistency. This is because while decentralization can improve security by removing single points of failure, it introduces new risks by creating entire system failures. Ensuring data consistency across all nodes is also challenging, furthering the burden. Errors or interruptions in the replication process can also cause security vulnerabilities and data inconsistencies. As more nodes are added, the complexity of managing the network increases. Additionally, the performance may degrade because each node must validate all data being added to the system, which can slow down the overall network performance. Furthermore, decentralized networks often lack a central authority, making coordination and governance very challenging. This can lead to difficulties in implementing consistent policies and managing the network efficiently. It can also be harder to address issues quickly since there is no single point of command. These limitations highlight the need to carefully consider the specific use case and requirements before opting for a decentralized network architecture. While they offer benefits such as improved fault tolerance and security, these must be balanced against the increased complexity, costs, and potential performance issues.

3. Risks of Unauthorized Access and Data Breaches

While offering numerous benefits such as enhanced fault tolerance, improved resource utilization, and reduced dependency on central authorities, decentralized networks also introduce many significant security challenges (AWS, 2023). These challenges are quite precarious in the context of unwarranted access and data compromises. The framework of decentralized networks expands the potential grounds for any type of attack due to the distribution of nodes and their open nature (N-Able, 2018). Each node operates as a possible point access point to attackers, this increases the risks of unwarranted access and data compromises, especially if any single node is compromised. Since there is no single point of control in decentralized systems, comprehensive strategies to safeguard against diverse vulnerabilities are necessary (ZPE Systems, 2021).

Phishing and Social Engineering

These attacks are used to gain valuable information by exploiting human vulnerabilities which result in gaining unauthorized access. Phishing attacks involve attackers tricking users into revealing sensitive information such as usernames, passwords, or cryptographic keys through deceptive emails or websites (APWG, 2020). Social Engineering attacks are executed by extracting information from individuals by manipulating them by leveraging trust and social connections (Hadnagy & Fincher, 2018). This type of attack is conducted through techniques that include pretexting, baiting, and impersonation. In decentralized networks, the lack of centralized oversight can result in significant security breaches particularly if attackers gain access to critical nodes or accounts, making phishing and social engineering particularly effective (Brook, 2023).

Man-in-the-Middle (MitM) Attacks

MitM attacks occur when attackers intercept communications between nodes, allowing them to monitor data, while also enabling them to alter or insert malicious data. Additionally, attackers will impersonate one of the communicating parties, tricking the other party into

believing that they are communicating with a reliable and trusted party (Magnusson, 2024). This allows the attacker to extract sensitive information or issue malicious commands.

Real World Examples

1. Ethereum DAO Hack: In 2016, attackers exploited a vulnerability in the Decentralized Autonomous Organization (DAO)'s smart contract, resulting in the theft of ether valued at 50 million Dollars (Atzei, Bartoletti, & Cimoli, 2017). This incident emphasizes the importance of rigorous security auditing for smart contracts and decentralized applications.

2. Mt. Gox Incident: Mt. Gox was a centralized exchange for bitcoin located in Shibuya, Tokyo, Japan, responsible for handling over 70% of all bitcoin trades worldwide, the incident regarding the company highlights potential risks relevant to decentralized systems. Multiple hacks executed resulted in the loss of 850,000 Bitcoins, eventually leading to the company declaring bankruptcy (McMillan, 2014). This incident demonstrates the need for robust security protocols and proper auditing mechanisms.

Consequences of Data Breaches

Data breaches in decentralized networks can lead to financial losses as shown in the Mt. Gox Incident, and can cause reputational damage, and legal ramifications. Unwarranted access to sensitive and confidential data can result in identity theft, fraud, and other malicious activities. For organizations, breaches can incur substantial financial penalties, loss of customer trust, and long-term reputational damage.

4. Enhanced Encryption Techniques for Secure Data Sharing

Since we've now covered all of the problems and issues surrounding the topic of decentralized networks, it's clear that something must be done to address them. One such method could be introducing more enhanced ways to encrypt data when it is being sent from one device to another. Currently, some of the most commonly used encryption algorithms are AES, Triple DES, RSA, Blowfish, and Twofish. These algorithms are widely implemented due to their efficiency, security, reliability, and flexibility. These systems generally operate by separating messages into 64- or 128-bit blocks and individually encrypting the blocks multiple times. However, each has its own unique characteristics to differentiate from each other. For example, Triple DES applies the older DES algorithm three times, RSA creates gibberish to attempt to stop hackers, and Twofish always encrypts data in 16 rounds, regardless of key size (Simplilearn, 2024).

Technology is always advancing, so it makes sense that methods of encryption also advance over time. With the implementation of a new encryption algorithm, decentralized networks can be made more secure than before, managing the risks that come with data sharing. In this case, we propose the development of a new algorithm, CipherGuard, combining robust lattice-based cryptography with elements of the One-Time Pad (OTP) for enhanced security against both quantum and conventional attacks. CipherGuard leverages lattice-based schemes like NTRUEncrypt to resist quantum computing threats while integrating OTP principles for critical encryption phases, ensuring near-perfect secrecy (Nejatollahi, 2017). This algorithm prioritizes efficiency, scalability, and adaptability across diverse data types and environments,

supported by advanced key management and resistance to side-channel attacks. Designed with future-proofing in mind, CipherGuard represents a significant advancement toward achieving superior encryption standards in the era of evolving cybersecurity threats.

Another potential solution to enhance the security of decentralized networks is the adoption of blockchain technology for secure data transactions. Blockchain, by design, offers a decentralized and tamper-resistant ledger, ensuring that all data exchanges are recorded in an immutable manner. This technology can be integrated with decentralized networks to create a more transparent and secure data flow (*What Is Blockchain on AWS?*, n.d.). By ensuring that all data transfers are verifiable and irreversible, blockchain can help mitigate the risks associated with unauthorized data tampering and breaches. Furthermore, the consensus mechanisms inherent in blockchain, such as Proof of Work (PoW) or Proof of Stake (PoS), can provide additional layers of security, ensuring that any changes to the data are validated by multiple nodes before being accepted into the network (Hayes, 2024).

In addition to encryption and blockchain, the implementation of zero-trust architecture (ZTA) can significantly enhance the security of decentralized networks. ZTA operates on the principle of "never trust, always verify," which means that no user or device, whether inside or outside the network, is trusted by default. Every access request is thoroughly authenticated, authorized, and encrypted before access is granted, regardless of the user's location. This approach can be particularly effective in decentralized networks, where multiple devices and users interact with each other across different nodes (Rose et al., 2020). By continuously monitoring and validating every interaction within the network, ZTA can prevent unauthorized access and limit the potential damage caused by compromised nodes or malicious actors.

Another promising approach is the development of secure multi-party computation (SMPC) techniques, which allow multiple parties to jointly compute a function over their inputs while keeping those inputs private. In decentralized networks, where data is often distributed across multiple nodes, SMPC can be used to perform computations on encrypted data without exposing the underlying information. This method can be particularly useful in scenarios where sensitive data needs to be processed collaboratively without compromising privacy. By ensuring that no single node has access to the complete dataset, SMPC can reduce the risk of data breaches and enhance the overall security of the network (Lindell, 2020).

Finally, enhancing the security of decentralized networks also requires a focus on user education and awareness. Even the most advanced encryption algorithms and security protocols can be rendered ineffective if users do not follow best practices for data protection. Educating users about the importance of strong passwords, regular software updates, and secure data-sharing practices is crucial for maintaining the integrity of decentralized networks. Additionally, organizations should provide training on recognizing phishing attacks, social engineering tactics, and other common cyber threats. By fostering a culture of cybersecurity awareness, users can become the first line of defense in protecting decentralized networks from potential threats.

5. Access Control Mechanisms and Data Integrity Verification

Access control mechanisms are vital in decentralized networks to manage who can access and modify resources, ensuring secure data sharing and preventing unauthorized access. This distribution enhances security and resilience and introduces complexity in managing access controls. Access control mechanisms, such as Discretionary Access Control (DAC), where resource owners set access permissions, are essential for the confidentiality of data. They ensure that only authorized users can access sensitive information or perform important operations, therefore protecting the network from potential breaches. Effective access control mechanisms are vital in maintaining trust and security in decentralized networks, allowing for secure and efficient data sharing.

Role-Based Access Control (RBAC)

Role-Based Access Control (RBAC) is a widely used approach to managing access permissions in a network by assigning roles to users and granting permissions based on those roles. In decentralized networks, RBAC provides a structured and scalable way to enforce access policies without the need for a central authority. By categorizing users into roles with predefined permissions, RBAC simplifies the process of access management, enhancing security and efficiency (Ferraiolo et al., 1995).

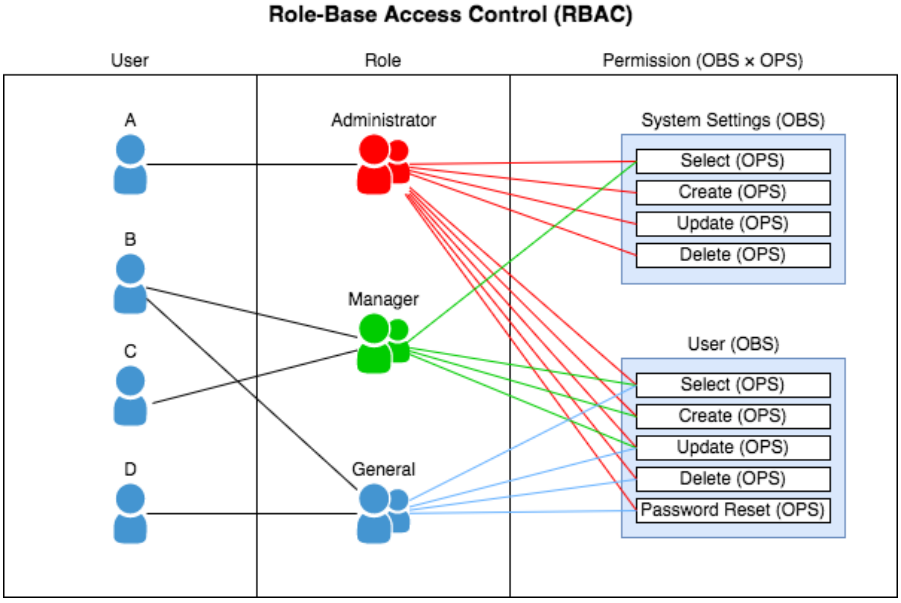
RBAC operates by defining roles that correspond to specific job functions or responsibilities within the network. Each role has a set of associated permissions, which determine what actions a user in that role can perform. Users are then assigned roles based on their responsibilities and needs. This role-based approach streamlines the management of access rights, as administrators only need to manage roles rather than individual user permissions.

Examples of RBAC Implementations

1. Blockchain Networks: In blockchain systems, RBAC can be used to manage permissions for different types of participants, such as miners, validators, and regular users. For instance, miners might have permission to validate and add new transactions, while regular users have permission to initiate transactions and view the blockchain.

2. Decentralized Applications (DApps): DApps often use RBAC to control access to various functionalities. For example, an e-commerce DApp might have roles such as "buyer," "seller," and "admin." Buyers can browse and purchase products, sellers can list products and manage inventory, and admins can oversee the platform and handle disputes.

Figure 2



Source: "Role-based access control overview," by D. Sonoda, 2020, *Medium*

Attribute-Based Access Control (ABAC)

Attribute-Based Access Control (ABAC) is an advanced access control model that determines access permissions based on a combination of attributes associated with users, resources, and environmental conditions. Unlike Role-Based Access Control (RBAC), which assigns permissions based on predefined roles, ABAC allows for more dynamic access control by evaluating a wide range of attributes. This flexibility allows ABAC to provide more precise and context-aware access decisions.

ABAC operates by evaluating attributes that can be categorized into three main types:

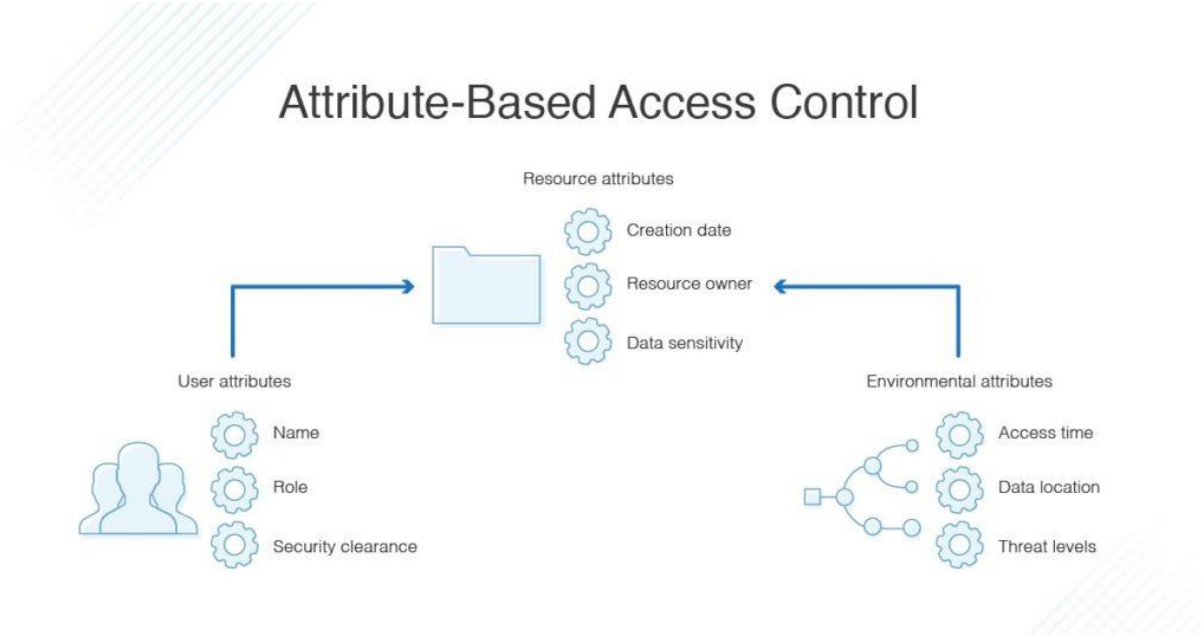
1. User Attributes: Characteristics of the user requesting access, such as their role, department, security clearance, or any other user-specific information.

2. Resource Attributes: Properties of the resource being accessed, such as the type of data, classification level, or ownership.

3. Environmental Attributes: Contextual factors that can influence access decisions, such as the time of day, location of the access request, or current threat level.

Access policies in ABAC are defined using these attributes, creating rules that specify which combinations of attributes are required for access to be granted. For instance, a policy might state that only users with a certain security clearance can access sensitive documents, but only during working hours and from secure locations (Am, 2024).

Figure 3



Source: "RBAC vs. ABAC: What's the Difference?," by Staff Contributor, 2019, *DNSstuff*

Data Integrity Verification Techniques

In decentralized networks, data integrity is crucial to ensure that information remains accurate, consistent, and trustworthy. Without a central authority to manage and verify data, decentralized networks rely on mechanisms to prevent tampering and ensure that data has not been altered. Ensuring data integrity is fundamental for maintaining trust among participants, securing transactions, and preserving reliability (Goswami et al., 2024).

Techniques for Data Integrity Verification:

1. Cryptographic hash functions are mathematical algorithms that take an input (or 'message') and produce a fixed-size string of characters, which typically appear as a sequence of numbers and letters. This output is known as the hash value or digest. Hash functions are designed to be deterministic, meaning the same input will always produce the same hash value. They are also designed to be fast to compute, infeasible to invert (one cannot deduce the input from the hash value), and resistant to collisions (it is extremely unlikely that two different inputs will produce the same hash value).

2. Digital signatures are cryptographic mechanisms that provide data integrity, authentication, and non-repudiation. A digital signature is created using a private key to generate a unique signature for a specific piece of data. This signature can be verified by anyone with the corresponding public key. For example, in blockchain transactions, the sender signs the transaction data with their private key. Other participants can use the sender's public key to verify the signature, confirming both the integrity of the data and the authenticity of the

sender. If the data is modified in any way, the signature verification will fail, indicating a breach of integrity.

6. Discussion and Conclusion

Decentralized networks have emerged as a revolutionary technology in recent years, offering new possibilities in data sharing and management. Unlike traditional centralized systems, decentralized networks distribute data across multiple nodes, eliminating the need for a central authority. This structure provides resilience against single points of failure, enhances transparency, and promotes user autonomy. However, as with any innovative technology, the shift to decentralized systems brings about a unique set of challenges, particularly in the realm of cybersecurity.

The decentralized nature of these networks creates an environment where data is more accessible to participants across the network. While this can improve information sharing and empower users, it also increases the potential for cyber attacks. Hackers can exploit vulnerabilities in the system, gaining unauthorized access to sensitive information or even manipulating data for malicious purposes. The open and distributed architecture that is fundamental to decentralized networks also makes them more susceptible to sophisticated cyber threats, including Distributed Denial of Service (DDoS) attacks, phishing, and data breaches.

Moreover, the anonymity and privacy features that are often touted as advantages of decentralized networks can be double-edged swords. While these features protect user identities and data from unauthorized scrutiny, they can also be exploited by malicious actors to conceal their activities. The ability to operate anonymously within a decentralized network can make it difficult to track and identify cybercriminals, complicating efforts to prevent and respond to cyber-attacks. This underscores the need for a careful balance between maintaining user privacy and ensuring the security of the network.

Given these challenges, it is important that robust security measures are developed and implemented to protect decentralized networks from emerging cyber threats. This includes the adoption of advanced encryption techniques, comprehensive access control mechanisms, and continuous monitoring of network activity. By addressing the inherent vulnerabilities in decentralized systems, we can create a more secure and resilient framework that enables the safe and efficient use of this transformative technology.

In summary, decentralized networks, while providing a means of global data sharing, are inherently dangerous due to the prevalent risks of sending data over the Internet. The findings from our research highlight the need for advanced security measures in order to balance the intricate architecture of decentralized systems. The unique characteristics of decentralized networks, such as the distribution of control and the elimination of a single point of failure offer significant advantages but also introduce numerous security risks. The increased risk of unauthorized access and potential data breaches is quite concerning, along with the open nature and distribution of nodes within these fragile networks. These vulnerabilities necessitate comprehensive strategies to ensure robust security across the entire network and to mitigate any unauthorized access or data breaches.

Real world incidents, such as the Ethereum DAO hack and the Mt. Gox incident, underscore the importance of an intricate deterrent framework and the need to develop enhanced security protocols. Enhanced encryption techniques, such as the proposed CipherGuard algorithm, offer promising solutions to safeguard data against evolving threats. Access control mechanisms like RBAC and ABAC, along with data integrity verification techniques, are crucial in managing permissions and ensuring the reliability of data across decentralized systems.

This research advances upon the understanding of privacy and security in decentralized networks, providing valuable insights on the current architecture of decentralized networks and potential solutions to mitigate risks. This research acts as an aide to future research in decentralized networks and strongly advocates for further exploration in innovative security measures and encryption techniques to address the ever-evolving landscape of cyber threats in decentralized environments. By implementing the techniques discussed and further exploring robust security measures, the framework of decentralized networks can be well adapted to imminent security threats. Through this we can harness the full potential and use of decentralized networks while ensuring the privacy and security of data.

References

Admin. (2018, September 14). What is Distributed Computing, its Pros and Cons? Open Cirrus; Open Cirrus. https://opencirrus.org/distributed-computing-pros-cons/#:~:text=In%20a%20nutshell%2C%20distributed%20computing

Am. (2024, February 20). Understanding Attribute-Based Access Control (ABAC). Medium. https://medium.com/@alokemajumder/understanding-attribute-based-access-control-abac-6eda89301860

APWG. (2020). Phishing Activity Trends Report, 4th Quarter 2020. Anti-Phishing Working Group. Retrieved from https://apwg.org/trendsreports/.

Atzei, N., Bartoletti, M., & Cimoli, T. (2017). A Survey of Attacks on Ethereum Smart Contracts (SoK). Lecture Notes in Computer Science, 10323, 164-186.

AWS. (2023). What is Decentralization? Amazon Web Services, Inc. https://aws.amazon.com/blockchain/decentralization-in-blockchain/

Brook, C. (2023, June 8). Attribute-Based Access Control: Pros, Cons & Use Cases. Www.digitalguardian.com. https://www.digitalguardian.com/blog/attribute-based-access-control

Eagar, M. (2017, November 4). What is the difference between decentralized and distributed systems? Medium; Positive Mavericks—Blockchain Leadership Development. https://medium.com/distributed-economy/what-is-the-difference-between-decentralized-and-distributed-systems-f4190a5c6462

Ferraiolo, D., Cugini, J., & Kuhn, D. (1995, January). Role-Based Access Control (RBAC): Features and Motivations. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=916537

Goswami, P., Neetu Faujdar, Debnath, S., Khan, A., & Singh, G. (2024). Investigation on storage level data integrity strategies in cloud computing: classification, security obstructions, challenges and vulnerability. Journal of Cloud Computing, 13(1). https://doi.org/10.1186/s13677-024-00605-z

Hadnagy, C., & Fincher, M. (2018). Social Engineering: The Science of Human Hacking. John Wiley & Sons.

Hayes, A. (2024, June 29). Blockchain Facts: What is it, how it works, and how it can be used. Investopedia. https://www.investopedia.com/terms/b/blockchain.asp

Lindell, Y. (2020). Secure multiparty computation. Communications of the ACM, 64(1), 86–96. https://doi.org/10.1145/3387108

Magnusson, A. (2024, January 29). Man-in-the-Middle (MITM) Attack: Definition, Examples & More | StrongDM. Discover.strongdm.com. https://www.strongdm.com/blog/man-in-the-middle-attack

McMillan, R. (2014). The Inside Story of Mt. Gox, Bitcoin's $460 Million Disaster. Wired. Retrieved from https://www.wired.com/2014/03/bitcoin-exchange/.

N-Able. (2018, November 30). The Difference between Centralized and Decentralized Networks. N-Able. https://www.n-able.com/blog/centralized-vs-decentralized-network

Nejatollahi, H., Dutt, N., Ray, S., Regazzoni, F., Banerjee, I., & Cammarota, R. (2017). Software and Hardware Implementation of Lattice-based Cryptography Schemes. Retrieved July 17, 2024, from https://www.cecs.uci.edu/files/2018/06/2017-tr-1.pdf

Rose, S., Borchert, O., Mitchell, S., & Connelly, S. (2020). Zero trust architecture. https://doi.org/10.6028/nist.sp.800-207

Simplilearn. (2024, July 2). *What Is Data Encryption: Types, Algorithms, Techniques and Methods*. Simplilearn.com. https://www.simplilearn.com/data-encryption-methods-article

Sonoda, D. (2020, May 30). Role-based access control overview. Medium. https://dsonoda.medium.com/role-based-access-control-overview-257de64534c

Staff Contributor. (2019, October 31). RBAC vs. ABAC Access Control: What's the Difference? DNSstuff. https://www.dnsstuff.com/rbac-vs-abac-access-control

Systems, Z. P. E. (2021, September 1). Centralized vs. Distributed Network Management: Which One to Choose? ZPE Systems. https://zpesystems.com/centralized-vs-distributed-network-management-zs/

Technologies, B., & Technologies, B. (n.d.). *Centralized vs Decentralized vs Distributed Systems · Berty Technologies*. Berty Technologies. https://berty.tech/blog/decentralized-distributed-centralized

What is Blockchain on AWS? (n.d.). [Video]. Amazon Web Services, Inc. https://aws.amazon.com/what-is/blockchain/