HIGH SCHOOL EDITION
Journal of Student Research
Volume 11 Issue 4 (2022)

Analysis on Blockchain Effectiveness Towards Protecting Renewable-Based Smart Power Grids

Brendan McQuilkin[1] and Samuel Liburd Jr.[#]

[1]Conestoga High School, Berwyn, PA, USA
[#]Advisor

Introduction & Background

Society's push for replacing unsustainable and destructive non-renewable energies has been quickly accelerating. To replace non-renewable sources, researchers have turned to various forms of renewable energies ranging from biofuels to wind and solar energy. However, while renewable energy may be the best candidate for the environment's safety, renewable energies may not be able to satisfy the increasing demands of energy using current infrastructures (Jabeen, 2021). Smart Electrical Grids (SEGs) are being developed to fulfill extensive energy needs with algorithms to manage, distribute, and store renewable energy efficiently. SEGs will implement Battery Management Systems (BMSs) and Battery Monitoring Systems (BMoSs) to optimize energy output and closely monitor stored energy (Friansa et al., 2017). Since SEGs will control a vital component of society, guaranteeing that they are safe is necessary. The potential security threats to these systems must be thoroughly analyzed, and an efficient solution must be constructed before the large-scale deployment of these grids is initiated. Network, firmware, data storage, and hardware vulnerabilities must be considered, and a reliable solution that will not impact the efficiency of SEGs must be chosen (Kim et al., 2020). Since researchers have already identified the most dangerous attacks and potential solutions, the remaining question is: which security protocols are the most efficient and effective to integrate into a renewable-energy-based SEG?

Renewables, Batteries, and Smart Grids

Renewable energies have been increasingly focused on due to increased attention towards the environment's health. Non-renewable energies, such as the burning of fossil fuels, are leaving devastating impacts on the environment. The most common forms of renewable energies include biofuels, solar, hydropower, wind, and nuclear energy.

While these energies sound like an easy solution at a glance, more profound studies reveal potential drawbacks that would limit their effectiveness. Some energies, like solar and wind, are not reliable enough to outright replace non-renewable energies. Solar energy is only available while the sun is out, and wind energy is not predictable enough to be reliable (Boretti, 2021). Nuclear energy, while reliable, creates radioactive waste that cannot be efficiently destroyed and needs ample time to decay (Frankel et al., 2021). For renewable energies to be efficiently utilized, storage systems will need to be integrated into power grids to supply energy when energies like solar and wind are unavailable or unpredictable. SEGs and BMSs can be combined with multiple forms of renewable energies to store and distribute energy while minimizing waste efficiently.

Battery Management Systems work with Battery Energy Storage Systems (BESSs) to efficiently and safely distribute power. BESSs consist of numerous individual battery cells. A BMS controls the individual cells to adjust output to meet power demand. Additionally, more advanced BMSs can reduce degradation and increase efficiency by reducing the power output from the batteries when there is little demand (Rahimi-Eichi et al., 2013). BMoSs are required to maintain safe operation, as they track performance and battery health.

BMoSs also measure various statistics such as voltage and temperature to prevent issues such as overheating (Friansa et al., 2017).

Battery Management Systems and BESSs are crucial for allowing renewable-generated energy to be stored when production exceeds demand. Excess energy generated by renewable sources can be stored in a BESS and later retrieved and used when energy production is unpredictable or unavailable. For example, excess solar energy during the day can be stored in a BESS and later accessed at night when the sun is unavailable to generate energy. Since some resources such as solar and wind energy are not always accessible on demand, being able to store and access excess energy is critical to the feasibility of SEGs.

Battery technology is very diverse, and there is ongoing research to decide which technologies would be most effective in smart grid BESSs. Lithium-ion (LIB), lead-acid, sodium-sulfur (SSB), and potassium-ion (PIB) batteries are various forms of battery technologies that have been researched, with lithium-ion being the most commonly used today (Stecca et al., 2020). LIBs are a dominating battery technology, but their hazards to the environment and their environmentally-unfriendly production process do not fit in with the goal of an SEG (Piatek et al., 2021). PIBs have been observed to have worse performance than LIBs. However, PIBs and SSBs are safer, more abundant, and easier-to-obtain materials than lithium-ion (Min et al., 2021). Lead-acid batteries contain dangerous materials, have worse performance, and shorter life spans (Min et al., 2021). SSBs offer a much higher energy density and a much lower cost of materials (Eng et al., 2021). With this information, SSBs and PIBs can be viewed as the best potential alternatives to use in SEGs instead of LIBs.

Smart Electrical Grids are a type of energy management and distribution system that will be crucial in replacing current power grid technology. SEGs utilize advanced monitoring and measuring systems to distribute energy across an extensive network efficiently. Additionally, SEGs can quickly provide large-scale analytics from data that spreads across multiple grid sectors that current electric grids cannot produce efficiently. These grids can also provide advanced features such as self-repair during errors and efficient and predictive distribution methods. (Rabie et al., 2021). When combined with renewable energies, SEGs can efficiently access and distribute stored energy in a non-wasteful manner, which is especially important considering stored renewable-based energy must be used conservatively at certain times. Additionally, SEGs can dynamically change load management to match varying power consumption at different times to prevent distribution and waste of excess energy. For example, SEGs will understand that there is higher energy usage during the day than at night so they will distribute energy more conservatively during late hours. These grids are conceptually the future of power grids and will increase the efficiency of energy distribution and reliability while also reducing operational costs (Escobar et al., 2021).

Relevance

In recent years, society has become more environmentally cautious and is driving the development of green energy. The desire for large-scale renewable energy has only been increasing. The conversion to clean energy can be seen in the switch to electric vehicles, the adoption of solar energy in houses, and many other cases. However, replacing simple gasoline-powered devices such as cars can only reduce humanity's carbon impact by so much. Converting infrastructure to function entirely on carbon-neutral green energy is one of the most effective ways to reduce environmental pollution significantly. Without efficient and secure SEGs, this concept will cease to become a reality and it will not be feasible to cater to Earth's rapidly growing population. Therefore, research on selecting the most efficient and secure method to serve the electrical needs of society is a must. Current security protocols require increased computational power and a large pool of available memory, which prove too complicated to be efficient (Kim et al., 2020). A secure SEG is necessary as any vulnerabilities could compromise the reliability of these grids, and lead to invasions of privacy and critical losses of power.

Vulnerabilities

Network Vulnerabilities

Since large quantities of data will need to be distributed between various grids, systems, and other devices, an SEG must utilize some form of networking. Firewalls are a standard line of defense against malicious network attacks. Firewalls work by filtering out malicious requests and blocking any potential threats. Firewalls are generally good at fulfilling their job, but there are still ways to circumvent their protections. A Distributed Denial of Service (DDOS) attack is one of the most threatening and hardest-to-prevent attacks. DDoS attacks work by disrupting a Firewall so severely that it will be unable to continue operations due to a lack of computational resources or bandwidth (Douligeris & Mitrokotsa, 2004). A DDoS attack sends massive amounts of heavy requests to a network to drain the Firewall of resources to process any further requests. The wave of requests causes a considerable delay in processing standard requests or, worst-case scenario, a complete shutdown of operations. Another function of modern-day Firewalls is to inspect requests for malicious data. Firewalls that utilize packet inspection functionality analyze incoming data and perform actions based on the integrity and validation of the requests (Dubrawsky, 2003). Flaws or poor maintenance in the algorithms used to inspect data open a potential vulnerability of mistaking malicious data as legitimate. Without proper maintenance and strict testing, analysis methods may be vulnerable to lesser-known exploits. Poor configuration of Firewalls may also lead to these vulnerabilities happening even with proper maintenance and testing.

Another area of attack via networking is in the form of a Remote Code Execution (RCE) attack. RCE attacks allow attackers to execute malicious code remotely onto a server. Methods such as SQL Injection (SQLi), Cross Site Scripting (XSS), and Buffer Overflow (BO) attacks are just the top few of many methods used to inject malicious code into servers (Biswas et al., 2018). XSS attacks, for example, allow an attacker to execute malicious code on a victim's device, which could further allow them to steal or corrupt sensitive data, among other possibilities (Gupta & Gupta, 2017). While some attacks like SQLi and XSS can be easily prevented using data sanitization and other attempts to filter malicious data, vulnerabilities caused by the exploitation of unintended behavior like BO attacks are very uncertain. They can be equally harmful, if not more dangerous. Preventing attacks like BO has been experimented with but has been observed to cause performance loss (Xu et al., 2018).

Finally, vulnerabilities in networking protocols can lead to data compromise and malicious attacks. Commonly used protocols designed to be secure and efficient, such as the HyperText Transfer Protocol Secure (HTTPS) and Secure Shell (SSH) protocols, all have specific vulnerabilities that can be abused when not appropriately addressed. Most of these attacks involve a Man-In-The-Middle (MITM) approach, which involves a malicious third-party intercepting a request and finding a way to reverse any encryption applied to the requests (4). For example, the HTTPS DROWN attack involves an attacker observing multiple requests and connections from a protected server and using complex algorithms to decrypt any encrypted data (Aviram et al., 2016). SSH attacks involve a MITM redirecting a request to an unauthorized and malicious connection where it can later be abused to access session contents (Gutmann, 2011). While both attacks are complex, lengthy, or difficult to achieve, they are feasible under the right conditions. They could lead to sensitive data being compromised or tampered with harmfully. However, since the encryption algorithms utilized by HTTPS and other modern protocols are powerful, the decryption process may take very long depending on the number of requests observed and the attack vector utilized. Weaker protocols that prioritize efficiency might be more vulnerable due to lightweight encryption algorithms.

Data Storage Vulnerabilities

Sensitive data stored on the servers of SEGs may be vulnerable to attacks. The most evident and easy-to-prevent vulnerability is leaving data unencrypted. Any attacker with data-reading access can easily access all raw data by allowing data to be stored unprotected on a server. On the other hand, encrypted data needs to be decrypted first and may become difficult for an attacker depending on the encryption used. However, decrypting encrypted data is still possible, and the ability to access data by an attack must be entirely prevented for data to be secure. As previously mentioned, SQLi attacks can allow attackers to execute malicious inputs and access sensitive data. SQLi attacks are most notoriously known for being able to allow attackers to get unrestricted access to data stores. Structured Query Language (SQL) is a programming language used to manage databases and, when abused by an attacker, can also be used to alter and compromise data maliciously. (Halfond et al., 2006)

Authorization Vulnerabilities

Improperly written or configured authorization systems may lead to various forms of damage for SEGs. Weak authorization may allow attackers to impersonate administrators and other users with permissions, thus granting them elevated access and control. For example, the MQ Telemetry Transport (MQTT) protocol is a lightweight protocol that prioritizes efficiency over proper encryption and security. Protocols designed for efficiency like MQTT may suffer from little to no validation. On MQTT, any client with network details can post requests to any actions (Harsha et al., 2018). Weak authorization may not be the only issue for protocols. Improper or broken handling of sessions may also allow attackers to impersonate verified users. Broken Authentication and Session Management vulnerabilities are caused by poor implementations, misconfigurations, and other potential areas of oversight such as lack of encryption, improper storage of data, and weak verification effort (Hassan et al., 2018).

Current Methods of Security

Protecting Data

One of the essential components of data security is encryption. Many encryption algorithms are available, but few are considered very strong. One of the most robust encryption algorithms commonly used by networking protocols is the Advanced Encryption Standard (AES) algorithm. Because of the algorithms' structure, attackers cannot access data without cracking the keys used in the encryption. The algorithm generally uses 128-bit, 192-bit, or 256-bit keys and is considered efficient and secure (Abdullah, 2017). A 256-bit key means that an attacker will have to try 2^256, a number that is 78 digits long, combinations in a time-consuming and impractical process for modern devices to crack. Another standard and powerful algorithm used to encrypt data is the Rivest-Shamir-Adleman (RSA) encryption algorithm. The RSA algorithm is generally efficient and does an excellent job at encrypting data so that it cannot be reversed without finding the keys (Meneses et al., 2016). Using powerful encryption methods such as RSA and AES on transferred data is already strong enough to prevent most attacks. However, different methods of handling the keys to decrypt the data may also affect security. Symmetric Encryption involves using a single public key that is used to both encrypt and decrypt the data. Asymmetric Encryption involves using two different keys, a public and private key, to encrypt and decrypt the data and are transferred separately. Asymmetric Encryption methods are considered much more secure than Symmetric Encryption algorithms. AES functions as a symmetric algorithm, while RSA is asymmetric. (Suguna et al., 2016)

While data must be encrypted, it also must be verified securely. Hashing provides a way for data to be verified and proven that a malicious attacker has not modified it. A hash of a string of data can best be described as a unique signature of that data. Hashing is a one-way function, meaning it is almost impossible to reverse the

result of a hash to the source. However, unlike encryption, hashing is not meant to be reversible. The Secure Hash Algorithm (SHA) is a widespread and secure algorithm used to verify data. Various forms of SHA, such as SHA-256, SHA-384, and SHA-512, named after their differing hash values, are considered efficient and secure algorithms. Hashing is static, meaning that the same algorithm used on the data will produce the same result every time. This behavior means data can be verified by hashing it to see if it produces the intended result. Hash algorithms can produce the same result for two different strings of data, but it is challenging to find overlapping hashes on good algorithms. (Sumagita et al., 2018)

Protect ing Networks

As previously stated, Firewalls are a standard and usually the first line of defense for networks. Firewalls usually include protections such as blocking malicious requests and forwarding connections to their correct locations. In addition, firewalls can include data monitoring activities such as packet inspection that will attempt to scan incoming connections for malicious data. Any requests deemed harmful will be rejected.

Additionally, different networking protocols have become much more secure than past protocols. For example, the HTTP protocol was replaced by the improved HTTPS protocol. HTTPS uses Transport Layer Security (TLS) to encrypt data and thus provide higher security and integrity (Amann et al., 2017). In addition, TLS provides end-to-end encryption using various algorithms to prevent attackers from intercepting and viewing confidential information. Other protocols have also been relaunched with upgraded encryption, such as MQTT upgrading to SMQTT.

Protecting Software

Software to identify and quarantine malicious software has generally used the same identification methods in various forms: Signature Detections, Behavior Detections, and Heuristic Detections. Signature-based detection involves searching for various identified forms of malicious code in programs. The patterns used to identify the malicious code are created uniquely and are common in most defense systems. However, using patterns requires that the threats have already been identified and fingerprinted, so unknown threats are not detected. Behavior-based detections watch the actions and logic of programs for suspicious or malicious activities. Heuristic-based methods utilize large amounts of centralized data and machine learning to identify potential threats. Behavior deemed malicious by the AI is then searched for in other programs and blocked if detected. (Bazrafshan et al., 2013) While these methods are generally effective, they have some difficulties against certain viruses that utilize advanced concealment strategies. Obfuscation techniques such as metamorphic code involve the viruses transforming themselves into code that appears different in a signature but functions the same way. This allows them to evade signature-based detections and sometimes even weaker behavior and heuristic detections. (Konstantinou & Wolthusen, 2008)

Proposed Methods of Security

Blockchain Technology

Due to its frequent usage in cryptocurrencies which have recently become heavily popularized, blockchain technology has become a talking point as a potential aide in many security and communication applications. The first large-scale usage of blockchain technology was in 2009, when Satoshi Nakamoto deployed the Bitcoin blockchain (12). A blockchain represents a leger of all communications (requests, transactions, or other types

of information) that have been shared across a peer-to-peer network. A *peer-to-peer network* is a network in which two or more devices share information directly rather than through a centralized server. Blockchain technology always uses a peer-to-peer architecture and focuses on decentralization. This architecture means that Blockchain networking revolves around multiple individual peers to process and verify requests instead of relying on a single server as many protocols do.

Because additions to the blockchain are verified across the network, the technology allows for a potential database that requires immense difficulty to tamper with compared to current technologies (12). Furthermore, since most clients must verify all actions on the Blockchain ledger, a single attacker will not be able to have a change published unless they control a majority of devices. If utilized in SEGs, this will help prevent attackers from tampering with sensitive information stored in the databases of the grids.

Additionally, using the blockchain to verify requests and connections may be feasible since it can be used to prevent impersonation (12). In theory, if requests are verified, malicious requests attempting to impersonate the grid will be identified and blocked. This concept is similar to Namecoin technology, which allows for a secure way of identifying accounts and interacting with peers while preventing impersonation (12).
Blockchain's cryptographic technologies protect sensitive data much more thoroughly than previous technologies (such as MQTT and other protocols) (Zhang et al., 2019). Signatures of data (also known as 'hashes' discussed earlier) can be used to detect if data has been tampered with, and tampering with the blockchain ledger itself is nearly impossible due to the technology (Zhang et al., 2019).

Artificial Intelligence

Blockchain technology is not the only proposed method for the security of SEGs. Artificial intelligence (AI) and machine learning have been studied in cybersecurity as a heuristic method to see if their usage would be practical. Models can be trained to defend against malware, malicious code, and other threats and respond accordingly (Veiga, 2018). If properly trained to an effective and efficient extent, these models used by the AI could be utilized in incredibly effective ways unmatched by other methods.

Artificial Intelligence is the process of using algorithms and trained networks to simulate intelligent behavior (Zhang & Lu, 2021). Machine learning is the ability of AI to learn from specific training and develop intelligent functionality from data sets (Janiesch et al., 2021). A model is a trained data set that can be loaded to give an AI the ability to make decisions based on patterns learned in the model. *Machine learning* is a long process requiring large amounts of data for consistent and proper results to be outputted by an AI. However, devices are becoming powerful enough to handle the demanding requirements of training and running neural networks. Additionally, there is already an abundance of patterns and other data on the subject of malware and other attacks for an AI to learn from.

Artificial Intelligence can block threats in real time and be used to discover unidentified threats as they are happening. *Zero-day attacks* are exploits that an attacker and only that attacker discover. Since nobody else knows about the exploit, most defense systems cannot defend against that exploit. In most cases, the exploit is not patched until after the damage is done. With AI, models can be trained to identify abnormal conditions that zero-day attacks may cause. Since the AI will be using its intelligence to stop threats, attacks attempting to disguise themselves will be much more prone to detection due to the AI not using traditional methods (such as signature analysis or behavioral analysis) to detect threats.

Analysis of Proposed Methods

Blockchain Technology

Blockchain technology is a strong candidate for security and should be implemented into future systems. While blockchain technology does not apply to all areas of defense, it will help most in encrypting important information, verifying and certifying transactions and requests, and other miscellaneous cryptographic applications. It may not be able to defend against malware actively and identify all attacks immediately, but it will play a crucial role in protecting sensitive data. Additionally, it will prove helpful as a way of secure authentication and identification so that attackers cannot access information or privileges that can be used maliciously. By itself, blockchain technology could not serve as the only defense component. However, combined with other rising technology such as AI-based security protocols, it would provide a robust layer of additional defense where applicable.

Artificial Intelligence

Artificial intelligence proves to be another strong candidate to integrate into SEGs. Its potential ability to be trained to detect unknown threats as they occur is a unique ability that no other solutions may offer. Preventing zero-day exploits before they can be used maliciously is critical, considering SEGs and BMSs will be responsible for continuously fueling the lives of billions of people. Additionally, artificial intelligence can be used beyond preventing unknown attacks and can be utilized against other dangerous threats as well. While not unique to AI, the technology could be trained to identify other attacks, such as ransomware attacks. Ransomware attacks involve attackers stealing, corrupting, or encrypting sensitive data and threatening malicious action unless given a ransom to stop the attack. Ransomware attacks such as NotPeyta and WannaCry have cost billions of dollars to governments, businesses, and people (Greenberg, 2018). Being able to prevent these attacks at a higher efficiency would prove AI to be a very worthy candidate considering attackers could theoretically hold SEGs hostage for a ransom.

However, AI is not without its limitations. For one, AI is not perfect and will not be able to detect every attack consistently. Neural network models are trained off existing data, so if an incredibly unique attack surfaces, there is a chance the AI might not discover it at all without manual intervention. Similarly, there is a chance that a poorly trained model could falsely detect normal operations as an attack. If there is a slight change in condition due to unknown circumstances, the AI could perceive it as an attack and slow or stop operations. While this is better than not stopping a potential threat, misinterpretation is a possibility. Artificial intelligence is still quite different from an actual thought process, and errors or miscalculations can have a significant impact (Zohuri & Moghaddam, 2020).

Furthermore, AI needs large quantities of data to learn to function accurately and efficiently (18). While there is a large quantity of data available online to train these models, gathering the resources to collect all this data and supply it to the neural networks may be challenging. Finally, training the neural networks and running them in real-time may prove to be a challenging task. Machine learning is not lightweight and does require a considerable amount of time and computational resources (Hwang, 2018). Recently, some processor manufacturers, such as Apple, have begun incorporating machine learning accelerators into their processors (Banerjee, 2018). This practice may apply to SEGs so that performance might be less of an issue. However, these accelerators are not magic, and a considerable amount of processing power is still required to run the neural networks. Because of these reasons, AI seems to be a less-considerable candidate due to its uncertainty and its need for computational power that may impact the efficiency of SEGs and BMSs.

Traditional Methods

Even with the addition of modern and powerful methods like the Blockchain and AI, using some traditional methods would not be a bad idea. Certain concepts such as the Firewall are still essential lines of defense and should be included with any proposed methods. Data encryption, provided by robust encryption methods such

as RSA and AES, is also necessary on top of added technology. Verifying data and requests using hashing is another form of existing technology that will continue adding needed security layers to an SEG.

Limitations

This review has multiple limitations that can be improved upon by further research. First, blockchain technology is still relatively new compared to other security protocols and methods. For example, Blockchain technology was first used in around 2008, while the HTTPS protocol was first used in around 1995. While this technology has still existed for over a decade, its uses have been relatively saturated in the areas of cryptocurrencies and other financial transactions. Research on Blockchain's abilities and potential outside these areas is still limited. New research may further support Blockchain's use in SEGs or suggest an even better alternative. Currently, the claims about Blockchain's use in SEGs are theoretically based on how the technology works and the research about its alternative uses.

Additionally, the efficacy of SEGs themselves is very theoretical and not entirely sure. Because there have been no large-scale deployments of completely renewable-based SEGs yet, no claims about them can be guaranteed. Until SEGs are tested in realistic scenarios, the claims cannot be marked as factual. However, the research for this technology is well supported and generally viewed as accurate and credible.

Finally, because SEGs do not exist at the level discussed in this paper, the effectiveness of the methods discussed is theoretical based on past research and their effectiveness in other scenarios. Due to this, their actual effectiveness may differ in the case of an SEG. Future research may change this by simulating these methods in the case of an SEG and offering quantitative data supporting or rejecting their use.

Conclusion

After analysis of blockchain technology and other candidates for SEG security, blockchain technology in combination with AI-based monitoring seems like the best route for efficiency and effectiveness. Blockchain technology would help filter out malicious requests and provide enhanced security for protected data and other valuable resources. Additionally, AI trained specifically for SEGs and BMSs will provide additional monitoring to prevent unknown attacks and other threats that blockchain technology may not cover, including attacks outside of accessing data and privacy. With processors becoming more efficient and unique technology being allocated for machine learning, the computation power required for AI and blockchain technology should not be an issue by the time SEGs are implemented into society. Therefore, blockchain technology and artificial intelligence are most likely the best candidates to replace outdated security protocols in smart electrical grids.

Acknowledgments

I would like to thank my advisor for the valuable insight provided to me on this topic.

References

Zhang, C., & Lu, Y. (2021). Study on artificial intelligence: The state of the art and future prospects. Journal of Industrial Information Integration, 23, 100224, DOI: https://doi.org/10.1016/j.jii.2021.100224
Jabeen, S. (2021). A comparative systematic literature review and bibliometric analysis on sustainability of renewable energy sources. 670216917, DOI: https://doi.org/10.32479/ijeep.10759

Friansa, K., Haq, I. N., Santi, B. M., Kurniadi, D., Leksono, E., & Yuliarto, B. (2017). Development of battery monitoring system in smart microgrid based on internet of things (IoT). Procedia engineering, 170, 482-487, DOI: https://doi.org/10.1016/j.proeng.2017.03.077
Kim, T., Ochoa, J., Faika, T., Mantooth, A., Di, J., Li, Q., & Lee, Y. (2020). An overview of cyber-physical security of battery management systems and adoption of blockchain technology. IEEE Journal of Emerging and Selected Topics in Power Electronics, DOI: https://doi.org/10.1109/JESTPE.2020.2968490
Boretti, A. (2021). Integration of solar thermal and photovoltaic, wind, and battery energy storage through AI in NEOM city. Energy and AI, 3, 100038, DOI: https://doi.org/10.1016/j.egyai.2020.100038
Frankel, G. S., Vienna, J. D., Lian, J., Guo, X., Gin, S., Kim, S. H., ... & Scully, J. R. (2021). Recent advances in corrosion science applicable to disposal of high-level nuclear waste. Chemical Reviews, 121(20), 12327-12383, DOI: https://doi.org/10.1021/acs.chemrev.0c00990
Stecca, M., Elizondo, L. R., Soeiro, T. B., Bauer, P., & Palensky, P. (2020). A comprehensive review of the integration of battery energy storage systems into distribution networks. IEEE Open Journal of the Industrial Electronics Society, 1, 46-65, DOI: https://doi.org/10.1109/OJIES.2020.2981832
Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. ACM Computing Surveys (CSUR), 52(3), 1-34, DOI: https://doi.org/10.1145/3316481
Veiga, A. P. (2018). Applications of artificial intelligence to network security. arXiv preprint arXiv:1803.09992, DOI: https://doi.org/10.48550/arXiv.1803.09992
Greenberg, A. (2018). The untold story of NotPetya, the most devastating cyberattack in history. Wired, August, 22, Referenced from https://www.wired.com/story/notpetya-cyberattack-ukraine-russia-code-crashed-the-world/
Hwang, T. (2018). Computational power and the social impact of artificial intelligence. arXiv preprint arXiv:1803.08971, DOI: https://doi.org/10.48550/arXiv.1803.08971
Banerjee, D. (2018). A microarchitectural study on apple's a11 bionic processor. Arkansas State University: Jonesboro, AR, USA, Referenced from https://www.researchgate.net/publication/325053646_A_Microarchitectural_Study_on_Apple%27s_A11_Bionic_Processor
Zohuri, B., & Moghaddam, M. (2020). Deep learning limitations and flaws. Mod. Approaches Mater. Sci, 2, 241-250, DOI: http://dx.doi.org/10.32474/MAMS.2020.02.000138
Douligeris, C., & Mitrokotsa, A. (2004). DDoS attacks and defense mechanisms: classification and state-of-the-art. Computer networks, 44(5), 643-666, DOI: https://doi.org/10.1016/j.comnet.2003.10.003
Dubrawsky, I. (2003). Firewall evolution-deep packet inspection. Security Focus, 29, 21, Referenced from https://www.academia.edu/download/33609188/Firewall-Evolution-deep-packet-inspection.pdf
Biswas, S., Sohel, M., Sajal, M. M., Afrin, T., Bhuiyan, T., & Hassan, M. M. (2018, October). A study on remote code execution vulnerability in web applications. In International Conference on Cyber Security and Computer Science (ICONCS 2018), Referenced from https://www.researchgate.net/publication/328956499_A_Study_on_Remote_Code_Execution_Vulnerability_in_Web_Applications
Gupta, S., & Gupta, B. B. (2017). Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. International Journal of System Assurance Engineering and Management, 8(1), 512-530, DOI: https://doi.org/10.1007/s13198-015-0376-0
Xu, B., Wang, W., Hao, Q., Zhang, Z., Du, P., Xia, T., ... & Wang, X. (2018). A security design for the detecting of buffer overflow attacks in IoT device. IEEE Access, 6, 72862-72869, DOI: https://doi.org/10.1109/ACCESS.2018.2881447
Aviram, N., Schinzel, S., Somorovsky, J., Heninger, N., Dankel, M., Steube, J., ... & Shavitt, Y. (2016). {DROWN}: Breaking {TLS} Using {SSLv2}. In 25th USENIX Security Symposium (USENIX Security 16)

(pp. 689-706), Referenced from https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/aviram
Gutmann, P. (2011). Do users verify SSH keys. Login, 36, 35-36, Referenced from https://www.usenix.org/system/files/login/articles/105484-Gutmann.pdf
Halfond, W. G., Viegas, J., & Orso, A. (2006, March). A classification of SQL-injection attacks and countermeasures. In Proceedings of the IEEE international symposium on secure software engineering (Vol. 1, pp. 13-15). IEEE, Referenced from https://www.cc.gatech.edu/fac/Alex.Orso/papers/halfond.viegas.orso.ISSSE06.pdf
Harsha, M. S., Bhavani, B. M., & Kundhavai, K. R. (2018, September). Analysis of vulnerabilities in MQTT security using Shodan API and implementation of its countermeasures via authentication and ACLs. In 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (pp. 2244-2250). IEEE, DOI: https://doi.org/10.1109/ICACCI.2018.8554472
Hassan, M. M., Nipa, S. S., Akter, M., Haque, R., Deepa, F. N., Rahman, M., ... & Sharif, M. H. (2018). Broken authentication and session management vulnerability: a case study of web application. Int. J. Simul. Syst. Sci. Technol, 19(2), 1-11, Referenced from https://ijssst.info/Vol-19/No-2/paper6.pdf
Abdullah, A. M. (2017). Advanced encryption standard (AES) algorithm to encrypt and decrypt data. Cryptography and Network Security, 16, 1-11, Referenced from https://www.researchgate.net/profile/Ako-Abdullah/publication/317615794_Advanced_Encryption_Standard_AES_Algorithm_to_Encrypt_and_Decrypt_Data/links/59437cd8a6fdccb93ab28a48/Advanced-Encryption-Standard-AES-Algorithm-to-Encrypt-and-Decrypt-Data.pdf
Meneses, F., Fuertes, W., Sancho, J., Salvador, S., Flores, D., Aules, H., ... & Nuela, D. (2016). RSA encryption algorithm optimization to improve performance and security level of network messages. IJCSNS, 16(8), 55, Referenced from https://www.academia.edu/download/51917545/20160809.pdf
Suguna, S., Dhanakoti, V., & Manjupriya, R. (2016). A study on symmetric and asymmetric key encryption algorithms. Int Res J Eng Technol (IRJET), 3(4), 27-31, Referenced from https://www.academia.edu/download/54558849/IRJET-V3I407.pdf
Sumagita, M., Riadi, I., Sh, J. P. D. S., & Warungboto, U. (2018). Analysis of secure hash algorithm (SHA) 512 for encryption process on web based application. International Journal of Cyber-Security and Digital Forensics (IJCSDF), 7(4), 373-381, Referenced from https://www.researchgate.net/profile/Imam-Riadi-2/publication/327392778_Analysis_of_Secure_Hash_Algorithm_SHA_512_for_Encryption_Process_on_Web_Based_Application/links/5b8cbe5e4585151fd1447946/Analysis-of-Secure-Hash-Algorithm-SHA-512-for-Encryption-Process-on-Web-Based-Application.pdf
Amann, J., Gasser, O., Scheitle, Q., Brent, L., Carle, G., & Holz, R. (2017, November). Mission accomplished? HTTPS security after DigiNotar. In Proceedings of the 2017 Internet Measurement Conference (pp. 325-340), DOI: https://doi.org/10.1145/3131365.3131401
Bazrafshan, Z., Hashemi, H., Fard, S. M. H., & Hamzeh, A. (2013, May). A survey on heuristic malware detection techniques. In The 5th Conference on Information and Knowledge Technology (pp. 113-120). IEEE, Referenced from https://www.researchgate.net/profile/Zahra-Bazrafshan-2/publication/260729684_A_survey_on_heuristic_malware_detection_techniques/links/54df00e60cf2953c22b0d005/A-survey-on-heuristic-malware-detection-techniques.pdf
Konstantinou, E., & Wolthusen, S. (2008). Metamorphic virus: Analysis and detection. Royal Holloway University of London, 15, 15, Referenced from https://www.csee.umbc.edu/courses/undergraduate/426/fall14/lectures/l07/RHUL-MA-2008-02.pdf
Rahimi-Eichi, H., Ojha, U., Baronti, F., & Chow, M. Y. (2013). Battery management system: An overview of its application in the smart grid and electric vehicles. IEEE industrial electronics magazine, 7(2), 4-16, DOI: https://doi.org/10.1109/MIE.2013.2250351

Piątek, J., Afyon, S., Budnyak, T. M., Budnyk, S., Sipponen, M. H., & Slabon, A. (2021). Sustainable Li-Ion Batteries: Chemistry and Recycling. Advanced Energy Materials, 11(43), 2003456, DOI: https://doi.org/10.1002/aenm.202003456

Min, X., Xiao, J., Fang, M., Wang, W. A., Zhao, Y., Liu, Y., ... & Huang, Z. (2021). Potassium-ion batteries: outlook on present and future technologies. Energy & Environmental Science, 14(4), 2186-2243, Referenced from http://eprints.bournemouth.ac.uk/35513/1/PIBs%20review%202021.pdf

Eng, A. Y. S., Kumar, V., Zhang, Y., Luo, J., Wang, W., Sun, Y., ... & Seh, Z. W. (2021). Room-temperature sodium–sulfur batteries and beyond: realizing practical high energy systems through anode, cathode, and electrolyte engineering. Advanced Energy Materials, 11(14), 2003493, DOI: https://doi.org/10.1002/aenm.202003493

Rabie, A. H., Saleh, A. I., & Ali, H. A. (2021). Smart electrical grids based on cloud, IoT, and big data technologies: state of the art. Journal of Ambient Intelligence and Humanized Computing, 12(10), 9449-9480, DOI: https://doi.org/10.1007/s12652-020-02685-6

Moreno Escobar, J. J., Morales Matamoros, O., Tejeida Padilla, R., Lina Reyes, I., & Quintana Espinosa, H. (2021). A comprehensive review on smart grids: Challenges and opportunities. Sensors, 21(21), 6978, DOI: https://doi.org/10.3390/s21216978

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. Electronic Markets, 31(3), 685-695, DOI: https://doi.org/10.1007/s12525-021-00475-2