Here are the 3 steps that you can follow to [minimize risk in blockchain security](https://www.getastra.com/blog/knowledge-base/blockchain-security-issues/). It would also help you prevent data breaches and disruption of your critical operations.

1. **Determine whether you need a public or private blockchain solution:**

[Blockchain application](https://www.blockchain-council.org/blockchain/top-10-real-world-applications-of-blockchain-technology/) types can be described as public or private. In a public blockchain, like Bitcoin, many people have access to the ledger and no one person is responsible for managing it.

This makes the management of transactions very easy as there are multiple validators that compete with each other to make sure all transactions are accurate and trustworthy. Securing this type of solution might require extra work because you need to ensure that every transaction recorded in the blockchain has been verified by at least 51% of participants on a network (otherwise they won’t count).

Securing your application built on top of such an open system requires even more effort due to its distributed nature – if hackers manage to find their way into one node in the network they can still do significant damage.

In a private blockchain, only some people have access to the ledger and it is managed by one or more members within that group of users (usually called “validators”). Securing this type of solution might be easier because you will not need to deal with scalability issues – all transactions are verified ahead of time so there is no need to wait for more than two validators’ approval.

Securing your application built on top of such a system might be trickier if you do not have the right expertise or resources in-house. Also, figuring out the optimum security strategy against blockchain security issues may put your mind in thinking for more time.

**When building blockchain solutions, keep security at the forefront:**

Blockchain technology was designed with security and immutability as key components.

Blockchain applications are built to be resilient from the ground up, however, this does not mean that they cannot be broken.

Securing your blockchain solution will require more than just a single set of encryption keys or user names and passwords – it is important to understand how you can use technology best suited for different requirements throughout the development process.

**2) Find out if you will need an off-chain service provider for parts of your system:**

[Securing a blockchain application](https://www.getastra.com/blog/knowledge-base/blockchain-security/) can be done on-chain or off-chain. On the one hand, you have public blockchains which are decentralized and require no trust in a third party to function properly. These types of solutions offer better security because competitors validate transactions so it is more difficult for data breaches to happen. However, this comes with tradeoffs.

On the other hand, there are private ledgers that rely on centralized providers (sometimes called “oracles”). Securing these systems requires less effort as all parties involved will need to do their own part but they might not always provide immutability. In some cases, attackers could get around them if they make changes at the provider level instead of trying to break into your application directly.

Securing blockchain applications that rely on off-chain services can be more challenging as they might not always provide 100% security. For example, if you use an oracle for data validation and the provider is hacked, attackers will get access to your whole database even though there are no traces of their activity within the ledger itself.

**3)Research what type of consensus mechanism is best suited for your needs; one example would be Proof-of-Work (PoW) :**

Securing a blockchain solution is all about understanding which consensus mechanism to use in a given situation. Secured public blockchains usually rely on Proof-of-Work, whereas private ledgers might have different requirements depending on the application and business model (e.g., if there are many parties involved that need to approve transactions).

If you want your solution to be as secure as possible it’s important not only to choose an appropriate consensus algorithm but also to understand what type of data validation is required for specific use cases. This will ensure uptime of your system and without any security breaches along the way. Securing blockchain applications requires an understanding of which consensus mechanism you are using.

Secured public blockchains usually use Proof-of-Work, while private ledgers have different security requirements depending on the application and business model (e.g., if there are many parties involved that need to approve transactions).

If you want your solution to be as secure as possible it’s important not only to choose an appropriate consensus algorithm but also to understand what type of data validation is required for specific use cases.

**Summing it up**

Blockchain technology is already making a powerful impact on how we run businesses. As more and more people are using it, there need to be security measures in place to make sure they know what they’re doing.

Beyond blockchain, securing the supporting infrastructure of websites and apps which are interfacing with users above the blockchain layer also becomes super important. Websites and apps related to cryptocurrencies or other aspects of crypto-related tech should also be secured with SSL certificates so users can feel safe when logging into their accounts online. Further, regular security audits/pentests should be done too.

### **How to Secure Your Private Blockchain Network**

Here are six steps organizations should consider for securing their private blockchain solution.

1. **Use the privacy-by-design concept in the early design phase.**When using this approach, you will consider data management, retention and deletion in the earlier stages of the design. You will consider regulatory requirements such as GDPR and other privacy laws relevant to the data. It is important to note that taking this approach may affect the type of data that can be stored on the chain due to any regulatory requirements. Still, the earlier you can determine this the better, as you can ensure that you have an optimal design to deploy into production. This approach will identify any off-chain services that your private blockchain will rely on and ensure the appropriate controls are applied to address any risk. For example, if you use a third-party provider for data validation and that provider gets hacked, your private blockchain will be exposed.
2. **Complete a risk assessment before deployment.**Work with relevant business units within and outside of IT to ensure that you have identified the acceptable risk of deploying a private blockchain within the environment. Ensure that controls are in place to protect the data with a level of residual risk acceptable to the business. It is essential to obtain input from all stakeholders.
3. **Periodically perform a third-party risk review of vendors and users on the blockchain.**Don’t trust anyone connecting to your blockchain and ensure that your requirements for connectivity are documented and reviewed periodically. This is very important in a private blockchain since there is a possibility that data can be deleted or modified, and you do not want any insecure sources connected that could exploit this vulnerability.
4. **Have a robust key management process in place.** Implementing a secure, scalable and resilient key management process is extremely important. This would include the backup of keys, automated management/rotation of keys, enforced key management requirements and, possibly, hardware to store them. Protecting these keys is essential to safeguard the data and environment, and any unauthorized access to the keys could break the encryption. This could cause significant issues if the central authority were to have their private key stolen by an attacker.
5. **Continue to apply production-grade security controls to your private blockchain.**For example, ensure that firewall protection, two-factor authentication, file integrity monitoring, endpoint security controls, etc. are applied to your private blockchain. Don’t assume that since it is encrypted standard security controls are unnecessary; because the environment is not a public blockchain, you have to prevent any unauthorized modification of data.
6. **Use a trusted cybersecurity vendor to audit and review your design and controls.** This includes penetration tests, security assessments, smart contract audits, source code reviews and blockchain infrastructure audits. A trusted organization should only do this with experienced resources. It would be best to do this before deploying a private blockchain into the production environment. This can be used periodically to identify any gap in the design and prepare the infrastructure for emerging threats or automated agents.

[Security Risks with Public and Private Blockchains](https://www.cryptyk.com/security-risks-with-public-and-private-blockchains/)

### Public Blockchain Security Risks

* More than half of the network’s hashing power rests in a single country's (China) hands. The concentration of mining power in countries like China is partially due to cheaper electricity prices. This threatens to subvert crypto currency's democratic nature. Giant mining pools and the other massive bitcoin-mining conglomerates can effectively monopolize control over the bitcoin blockchain. This may lead to network centralization and the possibility of collusion and making the network vulnerable to changes in policy on electricity subsidies.
* Cyber criminals are increasingly interested in stealing crypto-currency due to their climb in value. They have recently hacked into DAO and Bitfinex exchange. The DAO lost more than $50m, cutting the value of the currency by a third. Bitfinex lost about $65m in a cyber attack in 2016.
* Blockchain code is still in its infancy and may be subject to currently unknown security vulnerabilities. In particular, the Ethereum smart contract language is relatively new and there may be zero day attacks which hackers can exploit.
* Sometimes, the attacker announces an inaccurate timestamp while connecting to a node for a transaction. The network time counter of node is altered by the attacker and the deceived node may accept an alternate block chain. The serious consequences of this are double-spending and wastage of computational resources during mining process. This also known as a “timejacking attack”.
* The double spending attack is a serious threat for the blockchain transaction in which the attacker successfully makes more than one transaction using a single coin resulting in invalidating the ‘honest’ transaction. This attack is most likely to occur with ‘fast payment’ mode.
* There may be bugs in Bitcoin Core that haven't been discovered yet. However, the implementation of alternative client software is helping to uncover unexpected behavior as the network matures.
* The most popular mode of storage for crypto-currencies may be insecure. Many users store their private keys in internet based, and thus hack-prone, wallets. The best practice is to avoid using these hot wallets.
* The veracity of each entry rests on those in control of the private key of each account.
* Regulations and laws sometimes require the use of certain controls that may not be relevant or possible using blockchain.
* The legal liability for losses resulting from a failure of algorithmic trust is yet to be determined.
* Hackers may employ Blockchain cryptographic algorithms and mechanisms to perform malicious activities without leaving any traces (ex. a sybil attack).
* A vulnerability that allows a pool of sufficient size to obtain revenue larger than its ratio of mining power. In this attack, the colluding group of miners will force the honest miners into performing wasted computations on the stale public branch. In other words, the honest miners spend their cycles on blocks that eventually will not be part of the blockchain and they are forced by selfish miners to do so. The selfish mining group will keep their mined blocks private and will secretly perform bifurcation of the blockchain while the ‘honest’ miners continue to waste their computational power to the public branch. The selfish miners will then reveal the blocks to the public branch and the ‘honest’ miners will switch to the recently mined blocks which will make the selfish miner group earn more revenue. This is also known as "Selfish Mining"

### Private Blockchain Risks

* A node that restricts the transmission of information, or transmits incorrect information, must be identifiable and circumvented to maintain the integrity of the system. Blockchains achieve consensus on their ledger through communication. This communication occurs between nodes, each of which maintains a copy of the ledger and informs the other nodes of new information: newly submitted or newly verified transactions. Private blockchain operators can control who is allowed to operate a node, as well as how those nodes are connected. A node with more connections will receive information faster. Likewise, nodes may be required to maintain a certain number of connections to be considered active.
* Another security concern is the treatment of uncommunicative or intermittently active nodes. Nodes may go offline for innocuous reasons, but the network must be structured to function without the offline nodes, and it must be able to quickly bring these nodes back up to speed if they return.
* In a private blockchain, operators may choose to permit only certain nodes to perform the verification process. These trusted parties would be responsible for communicating newly verified transactions to the rest of the network.