Decentralized trust between banks on Azure

09/09/2018 • 5 minutes to read • Contributors all

In this article

This example scenario is useful for banks or any other institutions that want to establish a trusted environment for information sharing without resorting to a centralized database. For the purpose of this example, we will describe the scenario in the context of maintaining credit score information between banks, but the architecture can be applied to any scenario where a consortium of organizations want to share validated information with one another without resorting to the use of a central system ran by one single party.

Traditionally, banks within a financial system rely on centralized sources such as credit bureaus for information on an individual's credit score and history. A centralized approach presents a concentration of operational risk and sometimes an unnecessary third party.

With DLTs (distributed ledger technology), a consortium of banks can establish a decentralized system that can be more efficient, less susceptible to attack, and serve as a new platform where innovative structures can be implemented to solve traditional challenges with privacy, speed, and cost.

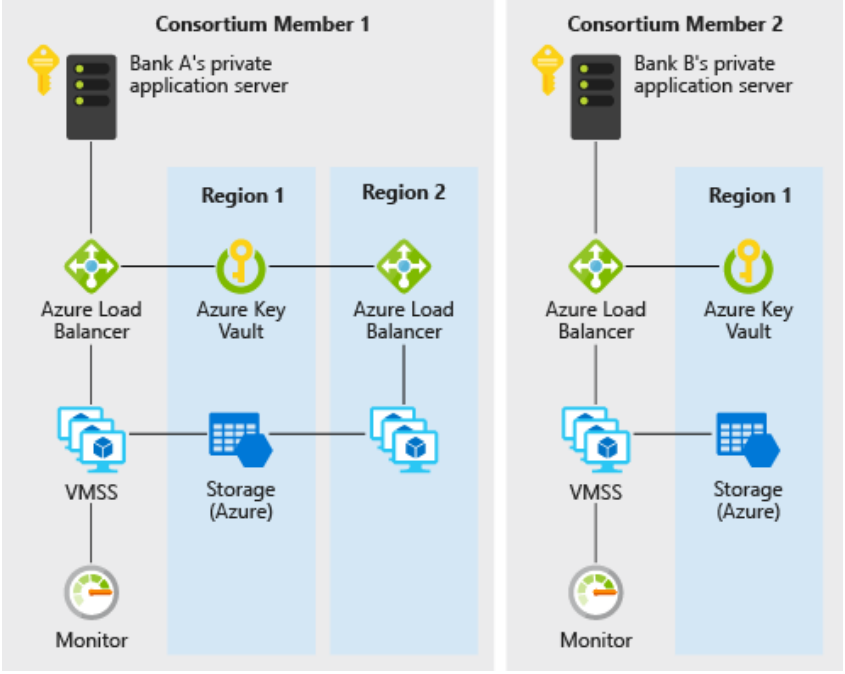
This example will show you how Azure services such as virtual machine scale sets, Virtual Network, Key Vault, Storage, Load Balancer, and Monitor can be quickly provisioned for the deployment of an efficient private Ethereum PoA blockchain where member banks can establish their own nodes.

Relevant use cases

Other relevant use cases include:

- Movement of allocated budgets between different business units of a multinational corporation
- Cross-border payments
- Trade finance scenarios
- Loyalty systems involving different companies
- Supply chain ecosystems

Architecture



This scenario covers the back-end components that are necessary to create a scalable, secure, and monitored private, enterprise blockchain network within a consortium of two or more members. Details of how these components are provisioned (that is, within different subscriptions and resource groups) as well as the connectivity requirements (that is, VPN or ExpressRoute) are left for your consideration based on your organization's policy requirements. Here's how data flows:

1. Bank A creates/updates an individual's credit record by sending a transaction to the blockchain network via JSON-RPC.
2. Data flows from Bank A's private application server to the Azure load balancer and subsequently to a validating node VM on the virtual machine scale set.
3. The Ethereum PoA network creates a block at a preset time (2 seconds for this scenario).
4. The transaction is bundled into the created block and validated across the blockchain network.
5. Bank B can read the credit record created by bank A by communicating with its own node similarly via JSON-RPC.

Components

- Virtual machines within virtual machine scale sets provides the on-demand compute facility to host the validator processes for the blockchain
- Key Vault is used as the secure storage facility for the private keys of each validator
- Load Balancer spreads the RPC, peering, and Governance DApp requests
- Storage hosting persistent network information and coordinating leasing
- Operations Management Suite (a bundling of a few Azure services) provides insight into available nodes, transactions per minute and consortium members

Alternatives

The Ethereum PoA approach is chosen for this example because it is a good entry point for a consortium of organizations that want to create an environment where information can be exchanged and shared with one another easily in a trusted, decentralized, and easy to understand way. The available Azure solution templates also provide a fast and convenient way not just for a consortium leader to start an Ethereum PoA blockchain, but also for member organizations in the consortium to spin up their own Azure resources within their own resource group and subscription to join an existing network.

For other extended or different scenarios, concerns such as transaction privacy may arise. For example, in a securities transfer scenario, members in a consortium may not want their transactions to be visible even to other members. Other alternatives to Ethereum PoA exist that addresses these concerns in their own way:

- Corda
- Quorum
- Hyperledger

Considerations

Availability

Azure Monitor is used to continuously monitor the blockchain network for issues to ensure availability. A link to a custom monitoring dashboard based on Azure Monitor will be sent to you on successful deployment of the blockchain solution template used in this scenario. The dashboard shows nodes that are reporting heartbeats in the past 30 minutes as well as other useful statistics.

Scalability

A popular concern for blockchain is the number of transactions that a blockchain can include within a preset amount of time. This scenario uses Proof-of-Authority where such scalability can be better managed than Proof-of-Work. In Proof-of-Authority–based networks, consensus participants are known and managed, making it more suitable for private blockchain for a consortium of organization that knows one another. Parameters such as average block time, transactions per minute and compute resource consumption can be easily monitored via the custom dashboard. Resources can then be adjusted accordingly based on scale requirements.

For general guidance on designing scalable solutions, see the scalability checklist in the Azure Architecture Center.

Security

Azure Key Vault is used to easily store and manage the private keys of validators. The default deployment in this example creates a blockchain network that is accessible via the internet. For production scenario where a private network is desired, members can be connected to each other via VNet-to-VNet VPN gateway connections. The steps for configuring a VPN are included in the related resources section below.

For general guidance on designing secure solutions, see the Azure Security Documentation.

Resiliency

The Ethereum PoA blockchain can itself provide some degree of resilience as the validator nodes can be deployed in different regions. Azure has options for deployments in over 54 regions worldwide. A blockchain such as the one in this scenario provides unique and refreshing possibilities of cooperation to increase resilience. The resilience of the network is not just provided for by a single centralized party but all members of the consortium. A proof-of-authority–based blockchain allows network resilience to be even more planned and deliberate.

For general guidance on designing resilient solutions, see Designing reliable Azure applications.

Pricing

To explore the cost of running this scenario, all of the services are pre-configured in the cost calculator. To see how the pricing would change for your particular use case, change the appropriate variables to match your expected performance and availability requirements.

We have provided three sample cost profiles based on the number of scale set VM instances that run your applications (the instances can reside in different regions).

- Small: this pricing example correlates to 2 VMs per month with monitoring turned off
- Medium: this pricing example correlates to 7 VMs per month with monitoring turned on

- Large: this pricing example correlates to 15 VMs per month with monitoring turned on

The above pricing is for one consortium member to start or join a blockchain network. Typically in a consortium where there are multiple companies or organizations involved, each member will get their own Azure subscription.

Next Steps

To see an example of this scenario, deploy the Ethereum PoA blockchain demo application on Azure. Then review the README of the scenario source code.

Related resources

For more information on using the Ethereum Proof-of-Authority solution template for Azure, review this usage guide.