



Blosent Protocol

v1.0.0

by Bakhankov Anton

blosent.com

November 2024

Abstract

The Blosent Protocol provides a decentralized solution for recurring payments in the Web3 ecosystem. Traditional financial systems support recurring payments seamlessly, but blockchain technology creates unique challenges that hinder such functionality. Using smart contracts and a network of bots, the Blosent Protocol automates the execution of recurring payments without requiring the sender to sign off on each transaction.

What are recurring payments?

A recurring payment is an automatic transaction scheduled on a regular basis. Common examples include: payments for services such as streaming platforms, magazines or software subscriptions, electricity, water, internet and other utilities.

What is Web3?

Web3 refers to a decentralized online ecosystem based on blockchain. The ecosystem infrastructure, as well as applications built on it, are owned by the users of the ecosystem without a centralized entity.

What about recurring payments in Web3?

Blockchain does not natively support recurring payments in the same way that traditional financial systems do.

A recipient cannot initiate a transaction to transfer money from a user's balance. In blockchain transactions, only the owner of the private key associated with the wallet can sign and initiate a transaction to send funds.

Therefore, in the Web3 field, you will need to sign each transaction manually before sending.

Cryptocurrency recurring payments

Some payment systems that provide payments using cryptocurrency allow merchants to create recurring subscriptions. The provider sends customers emails with invoices according to a specified schedule. Users will need to manually sign transactions to pay each invoice.

Current state of on-chain recurring payments

As we can see, there is a clear need for solutions for recurring payments in permissionless blockchain networks.

Blosent protocol overview

Blosent offers a solution built on smart contracts and a network of executors to automate the execution of smart contracts.

Because the solution is built on smart contracts, it can be considered permissionless and Web3 compliant.

The network of executors eliminates the need for the sender to sign off on every scheduled transaction.

The Blosent protocol ticks the boxes to be decentralized and permissionless, and allows users to schedule recurring payments without having to sign off on every ongoing transaction.

Blosent protocol elements

Schedule smart contract: A smart contract that contains the logic for recurring payments.

Factory smart contract: A smart contract that contains the bytecode of the Schedule smart contract for on-demand deployment.

Executors Network: A network of bots interested in the reward paid for executing a payment under the Schedule smart contract.

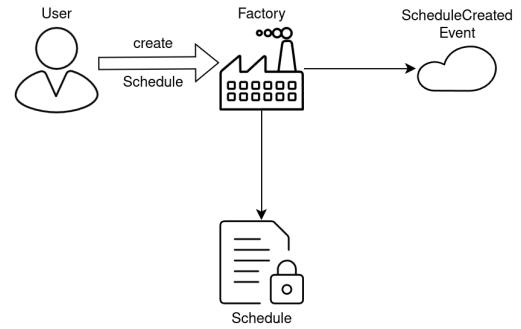
dApp: A user interface for smart contracts that allows users to conveniently interact with the protocol.

How does the Blosent protocol work?

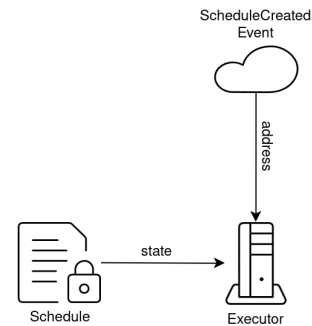
Factory smart contract is deployed by the protocol administrator. Factory smart contract address is added to the dApp and also broadcast to the Executors network.

Users using the dApp deploy a Schedule smart contract by providing the recipient address, payment amount, payment interval, and token smart contract address.

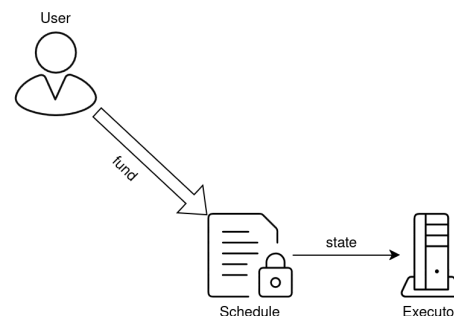
The Factory smart contract emits a ScheduleCreated event.



Executor bots are subscribed to ScheduleCreated events and can immediately receive notifications of newly deployed Schedule smart contracts.

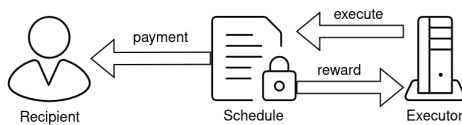


A user deposits funds into the Smart Contract Schedule. When the Smart Contract Schedule has enough funds for a single payment and has reached nextPaymentTime, it becomes ready to execute.



Executor bots constantly monitor known Schedule smart contracts to prepare for execution. If the bot finds one, it sends a transaction to initiate a transfer of funds from the Schedule smart contract balance to the recipient.

As a reward, the bot will receive an incentive from the Schedule smart contract.



Executing the Schedule smart contract will set the `nextPaymentTime` variable to a future point in time within the predefined interval. This means that the next time the Schedule smart contract will be executed, the time interval has passed and the balance of the Schedule smart contract will be sufficient to pay.

Source code

The source code for the smart contracts can be found on [github](#).

Some considerations

The execution transactions of the Schedule smart contract can be targeted by MEV bots. To prevent losses for the executor bots, the Blostent protocol uses an intermediate state pattern. The execution transaction of the Schedule does not send a reward as part of the transaction, but updates the internal state. At any time later, the executor bots can withdraw the incentive by calling a separate method in the Schedule smart contract.

For user convenience, the Schedule smart contract has a `withdraw` method that allows users to withdraw the token and coin held by the smart contract. The `withdraw` method should be used when the execution of the Schedule is no longer required.

The Schedule smart contract contains an enabled variable that represents whether the smart contract can be executed. The Schedule smart contract can be enabled or disabled by the user, and can be used to pause execution without withdrawing.

Use Cases

Subscription Services:

Platforms that offer subscription-based services can use the protocol to automate recurring payments.

Charitable Giving:

Non-profit organizations can create recurring donation systems for their supporters.

Loan Payments:

Financial institutions can use the protocol to automate loan repayment schedules.

Insurance Premiums:

Insurance companies can automate the collection of recurring premiums.

Future of Blostent protocol

Recurring payments are common in traditional payment systems. It is expected that the Blostent protocol can be used in the future as a component of the Web3 payment infrastructure. At the moment, the most promising way to gain support is the adoption and integration of the Blostent protocol by crypto payment providers and wallets.

Conclusion

The Blostent protocol addresses a critical gap in the Web3 ecosystem by enabling decentralized, automated recurring payments. Using smart contracts and a network of bot executors, it enables recurring transactions to be executed efficiently and securely without manual intervention. This innovation has the potential to become a foundational component of Web3 payment systems, facilitating wider adoption and integration by cryptocurrency payment providers and wallets. As the protocol matures, it promises to bring the convenience and reliability of traditional recurring payments to the decentralized world of blockchain.