Migrating from VRF v1

You are viewing the VRF v2 guide - Subscription method

To learn how to request random numbers without a subscription, see the direct funding method guide.

Security Considerations

Be sure to review your contracts with the security considerations in mind.

Comparing VRF v1 to the VRF v2 subscription method

Chainlink VRF v2 includes several improvements and changes to the way you fund and request randomness for your smart contracts.

- Subscription management:Chainlink VRF v2 introduces a Subscription Manager application that allows smart contract applications to pre-fund multiple requests for randomness using a single LINK token balance. This reduces the gas fees for VRF requests by eliminating the need to transfer LINK tokens for each individual request. You transfer LINK tokens to the subscription balance only when it requires additional funding. Read the Subscription Manager page to learn more.
- Variable Callback Gas Limit:Chainlink VRF v2 lets you adjust the callback gas limit when your smart contract application receives verifiable randomness. Consuming contracts can execute more complex logic in the callback request function that receives the random values. Tasks involving the delivered randomness are handled during the response process. The new gas limits are higher than the VRF V1 limit, and vary depending on the underlying blockchain you use. See the gas limits on the VRF Supported Networks page.
- More configuration capability:You can define how many block confirmations must pass before verifiable randomness is generated and delivered onchain when your application makes a request transaction. The range is from 3 to 200 blocks. VRF V1 always waited 10 blocks on Ethereum before delivering onchain randomness. Select a value that protects your application from block re-organizations while still providing sufficiently low latency from request to response. See the Security Considerations page to learn more.
- Multiple Random Outputs in a Single Request:The VRF Coordinator contracts in VRF v2 allow you to request multiple random numbers (multi-word) in a single onchain transaction, which reduces gas costs. The fulfillment is also a single transaction, which reduces the latency of responses.
- Unified Billing - Delegate Subscription Balance to Multiple Addresses:Chainlink VRF v2 allows up to 100 smart contract addresses to fund their requests for verifiable randomness from a single LINK subscription balance, which is managed by the subscription owner.

Read the Chainlink VRF v2 blog post for a detailed explanation about the benefits and use cases for VRF v2.

Updating your applications to use VRF v2

Subscription Manager

Read the Subscription Manager UI page to learn how to use all the features of the VRF v2 user interface. To learn how to troubleshoot your VRF requests, read the pending and failed requests sections.

Go to vrf.chain.link to open the Subscription Manager.

To modify your existing smart contract code to work with VRF v2, complete the following changes. See the Get a Random Number guide for an example.

1. Set up and fund a subscription in the Subscription Manager at vrf.chain.link .

Open the Subscription Manager 2. Import the new VRFConsumerBaseV2.solcontract and remove the v1VRFConsumerBase.solimport. This contract includes thefulfillRandomWordsfunction. 3. Import the VRFCoordinatorV2Interface.solinterface . This interface includes the newrequestRandomWordsfunction. 4. Add aVRFConsumerBaseV2constructor as shown in the Get a Random Number example. 5. ChangerequestRandomnessfunction calls torequestRandomWords. TherequestRandomWordsfunction requires several additional parameters. 6. ChangefulfillRandomnessfunction calls tofulfillRandomWords. Update the call to handle the returneduint256[]array instead of the singleuint256variable.