

Automation Best Practices

This guide outlines the best practices when using Chainlink Automation. These best practices are important for using Chainlink Automation securely and reliably when you [create Automation-compatible contracts](#).

[Use Automation 2.0](#)

To get the best reliability and security guarantees for your upkeep we recommend that you use [Automation 2.0](#). Consider migrating upkeeps to Automation 2.0 if you have upkeeps on an earlier version.

[Use the Forwarder](#)

If your upkeep performs sensitive functions in your protocol, consider using the [Forwarder](#) to lock it down so `performUpkeep` can only be called by the Forwarder. Add other permissible addresses if you need to call it yourself. Note the forwarder is only determined after registration so make this a mutable variable and ensure you add a setter function with permissions for you to set it.

[Verify Data Streams reports fetched with StreamsLookup](#)

If your upkeep uses [StreamsLookup](#), ensure you use the [verification interface](#) to verify your reports onchain.

[Avoid "flickering" custom logic upkeeps](#)

The Automation DON will evaluate your upkeep regularly and will try to perform it when eligible. However, if the state on chain changes and makes your upkeep ineligible before execution happens, the upkeep might not get performed. For best results, ensure `checkUpkeep` remains true until execution.

[Always test your contracts](#)

As with all smart contract testing, it is important to test the boundaries of your smart contract in order to ensure it operates as intended. Similarly, it is important to make sure the compatible contract operates within the parameters of the Registry.

Test all of your mission-critical contracts, and stress-test the contract to confirm the performance and correct operation of your use case under load and adversarial conditions. The Chainlink Automation Network will continue to operate under stress, but so should your contract. For a list of supported testnet blockchains, please review the [supported networks page](#).

[Using ERC-677 tokens](#)

For registration on Mainnet, you need ERC-677 LINK. Many token bridges give you ERC-20 LINK tokens. Use PegSwap to [convert Chainlink tokens \(LINK\) to be ERC-677 compatible](#). To register on a supported testnet, get [LINK](#) for the testnet that you want to use from our faucet.

[Automation v1 upkeep revalidation](#)

If your upkeep is on Automation v1, we recommend that you revalidate the conditions specified in `checkUpkeep` in your `performUpkeep` function. Automation v1 uses a turn taking system where nodes rotate to monitor your upkeep. It does not use consensus.