Backward Compatibility Policy

As the Solana developer ecosystem grows, so does the need for clear expectations around breaking API and behavior changes affecting applications and tooling built for Solana. In a perfect world, Solana development could continue at a very fast pace without ever causing issues for existing developers. However, some compromises will need to be made and so this document attempts to clarify and codify the process for new releases.

Expectations

- Solana software releases include APIs, SDKs, and CLI tooling (with a fewexceptions
-).
- Solana software releases follow semantic versioning, more details below.
- Software for aMINOR
- version release will be compatible across all software on the
- sameMAJOR
- version.

Deprecation Process

1. In anyPATCH
2. orMINOR
3. release, a feature, API, endpoint, etc. could be marked as deprecated.
4. According to code upgrade difficulty, some features will be remain deprecated for a few release
5. cycles.
6. In a futureMAJOR
7. release, deprecated features will be removed in an incompatible way.

Release Cadence

The Solana RPC API, Rust SDK, CLI tooling, and SBF Program SDK are all updated and shipped along with each Solana software release and should always be compatible betweenPATCH updates of a particularMINOR version release.

Release Channels

- edge
- software that contains cutting-edge features with no backward compatibility policy
- beta
- software that runs on the Solana Testnet cluster
- stable
- software that run on the Solana Mainnet Beta and Devnet clusters

Major Releases (x.0.0)

MAJOR version releases (e.g. 2.0.0) may contain breaking changes and removal of previously deprecated features. Client SDKs and tooling will begin using new features and endpoints that were enabled in the previousMAJOR version.

Minor Releases (1.x.0)

New features and proposal implementations are added tonew MINOR version releases (e.g. 1.4.0) and are first run on Solana's Testnet cluster. While running on the testnet,MINOR versions are considered to be in thebeta release channel. After those changes have been patched as needed and proven to be reliable, theMINOR version will be upgraded to thestable release channel and deployed to the Mainnet Beta cluster.

Patch Releases (1.0.x)

Low risk features, non-breaking changes, and security and bug fixes are shipped as part ofPATCH version releases (e.g. 1.0.11). Patches may be applied to bothbeta andstable release channels.

RPC API

Patch releases:

- Bug fixes
- Security fixes
- Endpoint / feature deprecation

Minor releases:

- New RPC endpoints and features

Major releases:

- Removal of deprecated features

Rust Crates

- solana-sdk
  - Rust SDK for creating transactions and parsing account state
- solana-program
  - Rust SDK for writing programs
- solana-client
  - Rust client for connecting to RPC API
- solana-cli-config
  - Rust client for managing Solana CLI config files
- solana-geyser-plugin-interface
  - Rust interface for developing Solana Geyser plugins.

Patch releases:

- Bug fixes
- Security fixes
- Performance improvements

Minor releases:

- New APIs

Major releases

- Removal of deprecated APIs
- Backwards incompatible behavior changes

CLI Tools

Patch releases:

- Bug and security fixes
- Performance improvements
- Subcommand / argument deprecation

Minor releases:

- New subcommands

Major releases:

- Switch to new RPC API endpoints / configuration introduced in the previous major version.
- Removal of deprecated features

Runtime Features

New Solana runtime features are feature-switched and manually activated. Runtime features include: the introduction of new native programs, sysvars, and syscalls; and changes to their behavior. Feature activation is cluster agnostic, allowing confidence to be built on Testnet before activation on Mainnet-beta.

The release process is as follows:

1. New runtime feature is included in a new release, deactivated by default
2. Once sufficient staked validators upgrade to the new release, the runtime feature switch
3. is activated manually with an instruction
4. The feature takes effect at the beginning of the next epoch

Infrastructure Changes

Public API Nodes

Solana provides publicly available RPC API nodes for all developers to use. The Solana team will make their best effort to communicate any changes to the host, port, rate-limiting behavior, availability, etc. However, we recommend that developers rely on their own validator nodes to discourage dependence upon Solana operated nodes.

Local cluster scripts and Docker images

Breaking changes will be limited toMAJOR version updates.MINOR andPATCH updates should always be backwards compatible.

Exceptions

Web3 JavaScript SDK

The Web3.JS SDK also follows semantic versioning specifications but is shipped separately from Solana software releases.

Attack Vectors

If a new attack vector is discovered in existing code, the above processes may be circumvented in order to rapidly deploy a fix, depending on the severity of the issue.

CLI Tooling Output

CLI tooling json output (output --json) compatibility will be preserved; however, output directed for a human reader is subject to change. This includes output as well as potential help, warning, or error messages. Previous Frequently Asked Questions
Next Solana CLI Tool Suite