feat(validation): support BigInt in props type validation #11191

**What kind of change does this PR introduce? (check at least one)**

* ** Bugfix**
* ** Feature**
* ** Code style update**
* ** Refactor**
* ** Build-related changes**
* ** Other, please describe:**

**Does this PR introduce a breaking change? (check one)**

* ** Yes**
* ** No**

**If yes, please describe the impact and migration path for existing applications:**

**The PR fulfills these requirements:**

* ** It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch**
* ** When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)**
* ** All tests are passing:**[**https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup**](https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup)
* ** New/updated tests are included**

**If adding a new feature, the PR's description includes:**

* ** A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)**

**Other information:  
Just a simple addition to make BigInt validation work.**

types(v-node-directive): changed expression type to optional string(fix #10871) #11189

I changed expression type to optional string

fix [#10871](https://github.com/vuejs/vue/issues/10871)

it seems stopped since December ... so i try it 🙌  
please point out if i have any deficiencies 🙏

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe: types

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

fix(keep-alive): should not cache destroyed instance #11178

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

reproduce link : <https://jsfiddle.net/mwpeodL8>

After keep-alive child component destroys itself,  
component instance is being created every time when it re-activated

Please check reproduce link and test code

docs(grid component example): Removed autocomplete to avoid boring history #11166

Removed autocomplete to avoid boring history from the browser.

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe: Documentation

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

fix(vdom): check old handlers are defined before remove (fix #11151) #11159

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

feat(compiler): Allow BigInt usage in templates (issue #11126) #11152

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
* [ x ] Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
* [ x ] No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

* [ x ] It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
* [ x ] When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
* [ x ] All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
* [ x ] New/updated tests are included

If adding a **new feature**, the PR's description includes:

* [ x ] A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**  
Hello!  
This pull request is related to issue [#11126](https://github.com/vuejs/vue/issues/11126) . Now we can use BigInt in "Mustache" syntax.  
I had already added test for this feature in "test/unit/features/filter/filter.spec.js".  
And all tests are passing.

$http.delete shows warning "avoid using JavaScript unary operator as property name" (fix #11090) #11143

Update error-detector to look for . prefix

fix [#11090](https://github.com/vuejs/vue/issues/11090)

**What kind of change does this PR introduce?** (check at least one)

* [ X] Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
* [X ] No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

* [ X] It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
* [X ] When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
* [X ] All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
* [X ] New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

$mount can be called on comment node #11139

The typescript definition currently specifies that you can only mount on an Element (or a selector string evaluating to an element), however the result of a server-rendered component can be an empty comment node (e.g. if the component has a v-if on the root node). $mount actually supports this, but the type definition does not reflect this currently.

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe: Type Definition

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

fix(compiler): Better whitespace regex #11138

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

Related to [#11065](https://github.com/vuejs/vue/pull/11065)

fix(transition-group): handle non-visible list items, fix #8516 #11128

Handle the case, when a component inside a transition-group doesn't yield any HTML, e.g. when there is a v-if on the root-element.

Before this fix, transition-group would fail with the errror c$1.elm.getBoundingClientRect is not a function, when moving items, if there was such an item.

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

types: $refs update #11112

Added undefined as a return type of $refs as it was quite hard to convince typescript that something COULD be undefined if you type in a ref that doesn't exist,  
I also changed the two array statements from Element[] | Vue[] to (Element | Vue)[] because it's not guaranteed that they are ALL Elements or they are ALL Vue components. (Very likely, yes, but not always)

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe: Typings

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

fix(ssr): ignore empty text vnode when hydrating (fix #11109) #11111

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**  
Closes [#11109](https://github.com/vuejs/vue/issues/11109)

fix(core): fix sameVnode for async component #11107

Fix sameVNode to return false when asyncPlaceholder and comment are compared.

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

chore: upgrade to ESLint v6 #11099

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

I'm a maintainer of ESLint and am working on [getting our regression build for ESLint](https://github.com/eslint/eslint-canary/pull/27) updated to ESLint v6. I figured I could make a quick PR to get the Vue codebase updated. This is my first PR, so please let me know if I missed something!

feat(ssr): support custom attributes in render scripts #11094

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

Resolve [nuxt/nuxt.js#6635](https://github.com/nuxt/nuxt.js/pull/6635)  
Resolve [nuxt/nuxt.js#6634](https://github.com/nuxt/nuxt.js/issues/6634)

Add getPrefetchLinkAttrs, getPreloadLinkAttrs and getScriptAttrs in ssr renderer options for adding custom attributes in ssr.

Update README.md #11083

added full stop.

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

fix(compiler): avoid converting &nbps; to spaces (fix #11059) #11065

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

Do not warn when binding native modifier on dynamic components, fixes #10939 #11052

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

Removes confusing warning

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

Fix [#10939](https://github.com/vuejs/vue/issues/10939)

Export public types from types/options.d.ts #11047

**What kind of change does this PR introduce?** (check at least one)

[ ]x] Bugfix  
[x] ] Feature

*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

<https://github.com/vuejs/vue/blob/dev/types/umd.d.ts#L3-L7>

types/umd.d.ts imports DefaultData (and others) from the options.d.ts file, but those type definitions are never actually exported. It seems like TypeScript doesn't fail on this (which is surprising, but I've seen them allow bugs in .d.ts files before) but other tooling gets confused and breaks when it can't find this export.

fix(events): rule of calling binded event handler (close #11014) #11031

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

The checkbox 'Refactor' is checked because of rules ordering was changed. The primary rule should have bigger priority than crutches.

fix(build): uprgade http-server (#11027) #11030

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

Fix wording in hydration error message #11024

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe: Fix error message

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

feat(ssr): add integrity and crossorigin support #11021

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

Currently subresource integrity isn't supported by SSR. This PR implements it (along with crossorigin support) by putting the SRI hashes into the client manifest so that they can be added during server rendering. The feature is opt-in when creating the renderer.

fix: Cannot read property 'isString' of undefined (it happend that ch… #10996

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

fix: clear extra styles when hydrating/mounting element #10993

**What kind of change does this PR introduce?** (check at least one)

*  Bugfix
*  Feature
*  Code style update
*  Refactor
*  Build-related changes
*  Other, please describe:

**Does this PR introduce a breaking change?** (check one)

*  Yes
*  No

If yes, please describe the impact and migration path for existing applications:

Inline styles already existing on an element which Vue is mounted on, or server-side rendered DOM, may be removed if the component does not render those styles. Developers should not rely on extra inline styles being left untouched.

**The PR fulfills these requirements:**

*  It's submitted to the dev branch for v2.x (or to a previous version branch), *not* the master branch
*  When resolving a specific issue, it's referenced in the PR's title (e.g. fix #xxx[,#xxx], where "xxx" is the issue number)
*  All tests are passing: <https://github.com/vuejs/vue/blob/dev/.github/CONTRIBUTING.md#development-setup>
*  New/updated tests are included

If adding a **new feature**, the PR's description includes:

*  A convincing reason for adding this feature (to avoid wasting your time, it's best to open a suggestion issue first and wait for approval before working on it)

**Other information:**

There's currently a somewhat sharp gotcha with SSR where an inline style rendered for an element on the server, but not rendered on the client, will end up stuck on the element, leading to partially broken markup.

The styles may differ between the SSR DOM and the client DOM due to practices such as using the screen size as a prop value, which causes a style to be applied on the server which is not applied on the client when loaded on mobile clients. See [vuetifyjs/vuetify#10163](https://github.com/vuetifyjs/vuetify/pull/10163) for a real-world example of this.

I tried to keep this PR concise and focused, but it's worth discussing potential side effects of this change. Ideally it would be scoped just to hydration to limit the surface area for side effects, but that wasn't as simple as limiting it to creation, so I opted for the simpler route for the PR.

Some users may be relying on the current behavior (perhaps without realizing it), so this would break behavior for them. However, considering the current gotcha for otherwise sane looking code, it seems to be a reasonable trade-off to potentially break a tiny amount of code to remove a hard to debug issue which can lead to messed up markup.

An argument could be made that much as the DOM structure needing to match when hydrating, perhaps styles should also be required to match, which would avoid this issue. However, it seems like a tall order to ensure all styles are rendered identically between server and client since more style-oriented information is available on the client (screen size, device capabilities, etc). So, that seems like an overly burdensome requirement to put on developers for SSR. However, if that's preferred, this PR could be turned into issuing a warning on style-mismatch instead.