

Enforce check on all arguments passed to methods and publish functions (audit-argument-checks)

The Meteor package `audit-argument-checks` requires that all arguments in calls to methods and publish functions are `check` ed. Any method that does not pass each one of its arguments to `check` will throw an error. This rule emulates that behavior. Unlike its Meteor counterpart this rule further ensures all `check` 's happen unconditionally.

Rule Details

The following patterns are considered warnings:

```
Meteor.publish("foo", function (bar) {})  
  
Meteor.methods({  
  foo: function (bar) {}  
})  
  
Meteor.methods({  
  foo: function (bar) {  
    if (Math.random() > 0.5) {  
      check(bar, Match.Any)  
    }  
  }  
})
```

The following patterns are not warnings:

```
Meteor.publish("foo", function (bar) {  
  check(bar, Match.Any)  
})  
  
Meteor.methods({  
  foo: function (bar) {  
    check(bar, Match.Any)  
  }  
})  
  
Meteor.methods({  
  foo: function (bar) {  
    var ret;  
    ret = check(bar, Match.Any)  
  }  
})
```

For a check function to be considered "called", it must be called at the top level of the method or publish function (not e.g. within an `if` block), either as a lone expression statement or as an assignment statement where the right-hand side is just the function call (as in the last example above).

Options

If you define your own functions that call `check`, you can provide a list of such functions via the configuration `checkEquivalents`. This rule assumes that these functions effectively check their first argument (an identifier or an array of identifiers).

For example, in `.eslintrc.json`, you can specify the following configuration:

```
"meteor/audit-argument-checks": [  
  "error",  
  {  
    "checkEquivalents": [  
      "checkId",  
      "checkName"  
    ]  
  }  
]
```

When Not To Use It

If you are not using Meteor's `check` package, then you should not use this rule.

Further Reading

- <http://docs.meteor.com/#/full/check>
- <http://docs.meteor.com/#/full/auditargumentchecks>

Possible Improvements

- Emulate behavior of Meteor's `audit-argument-checks` more closely
- Support immediate destructuring of params