是否使用插件进行错误上报服务器（windows.onerror） fundebug JSTracker - 淘宝前端监控平台

https://fed.taobao.org/blog/taofed/do71ct/jstracker-how-to-collect-data/?spm=taofed.homepage.header.14.7eab5ac8EqHgzM

CORS，所以很多的错误信息被替换为script error:解决方案： JavaScript 请求的 http 返回头上需要加上一个 Access-Control-Allow-Origin 头。 script 标签上添加 crossorigin 属性

错误很多：ISP在页面注入的脚本报错、插件问题（注入）

另类思路：<https://cloud.tencent.com/developer/article/1367170?fromSource=waitui>

const originAddEventListener = EventTarget.prototype.addEventListener; EventTarget.prototype.addEventListener = function (type, listener, options) { const wrappedListener = function (...args) { try { return listener.apply(this, args); } catch (err) { throw err; } } return originAddEventListener.call(this, type, wrappedListener, options); }

自己try catch主动上报 在 try 语句块中不要定义太多的变量，最好是只写一个函数调用，避免 try 运行中变量拷贝造成的性能损耗 上报的内容，浏览器的判断，代码的堆栈？

window.onerror = function(){return true;} 可以忽略错误提示。如果 js 发生错误，仍然会阻挡下面的 js 执行

try/catch可以捕获可预料的异常情况，那未知的异常怎么发生了怎么处理，浏览器提供onerror函数作为全局的异常捕获接口 监听error事件捕获资源加载错误 reject的Promise需要添加catch函数，如果没有添加，会抛出全局的unhandledrejection

总结

代码做必要的业务错误处理

可预料的异常用try/catch包含处理

同域下的脚本注册全局onerror函数

同域下的promise添加unhandledrejection事件监听

同域下的资源加载错误监听error事件

性能监控用performanc API

跨域脚本内错误、页面卡顿和崩溃需要特殊处理

vue错误捕获使用全局errorHandler接口

Vue中的异常处理：<https://segmentfault.com/a/1190000019497128>

ErrorHandler

warnHandler

renderError

errorCaptured

Window.onerror

<http://jartto.wang/2018/11/20/js-exception-handling/>

<https://fed.taobao.org/blog/taofed/do71ct/try-catch-runing-problem/?spm=taofed.homepage.header.31.7eab5ac8sl05so>