Prebid bug introduced starting version 8.15

Created by	① Julian Rassolov
Created time	@December 4, 2023 4:13 PM
∷ Tags	

Disclaimer: We recognize this feature as a bug, as it has the potential to result in different errors being triggered on the client and potentially on the server side (specifically, the server side of adapters)

Bug description

We have identified a flaw in prebid.js versions 8.15 and above that could significantly impact performance and potentially affect revenue on a page utilizing it, given certain configurations.

Conditions:

For this bug to manifest, two specific conditions must be met:

- 1. An identifier must be added to the pubprovided ids prebid module, with the source generated by an identity module.
- 2. An identity module (capable of producing an id with the source from point 1) must be initialized and generate an id.

Result:

Upon initiation of an auction and preparation of requests, the number of ids in the corresponding array for the previously mentioned source will increase.

Consequences:

As eids are included in the majority of outgoing requests:

- 1. The size of these requests will gradually grow over time, assuming the page continues to initialize additional auctions.
- 2. Requests may exceed the maximum allowed size.

Our case study

We have tested this on a sample test page that has configuration for p[ubprovided id and unified id module added manually:

```
function myf(){
return [
    {
        "uids": [
            {
                "ext": {
                    "stype": "ppuid"
                "id": "4a958e6e-de2c-45f1-bd66-836890a2e78a"
        ],
        "source": "adserver.org"
]
}
pbjs.setConfig({
     userSync: {
         userIds: [{
             name: "pubProvidedId",
             params: {
                 eidsFunction: myf // any function that exists in the page
        }]
     }
 });
pbjs.refreshUserIds()
pbjs.setConfig({
     userSync: {
         userIds: [{
             name: "unifiedId",
             value: {"tdid": "D6885E90-2A7A-4E0F-87CB-7734ED1B99A3"}
        }]
     }
});
pbjs.refreshUserIds()
pbjs.setConfig({
          rubicon: {
            rendererConfig: {
              align: "center", // player placement: left|center|right (default is center)
              position: "append", // position relative to ad unit: append|prepend|before|after (default is after)
             closeButton: true, // display 'Close' button (default is false)
             label: "Advertisement", // custom text to display above the player (default is '-')
              collapse: true, // remove the player from the page after ad playback (default is true)
           },
          },
          debug: true,
          cache: {
           url: "https://prebid.adnxs.com/pbc/v1/cache",
          },
          userSync: {
            userIds: [
              {
               name: "unifiedId",
                value: { tdid: "XXXXXXXX-XXXX-XXXX-7734ED1B99A3" },
```

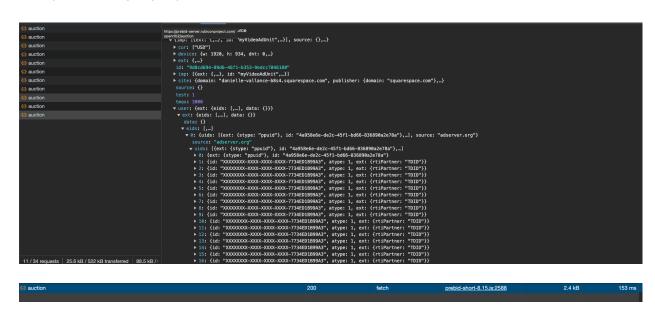
```
name: "id5Id",
       params: {
         partner: 173, // change to the Partner Number you received from ID5
         // pd: 'MT1iNTBjY...', // optional, see table above for a link to how to generate this
         abTesting: {
           // optional
           enabled: true, // false by default
           controlGroupPct: 0.1, // valid values are 0.0 - 1.0 (inclusive)
         },
       },
       storage: {
         type: "html5", // "html5" is the required storage type
         name: "id5id", // "id5id" is the required storage name
         expires: 90, // storage lasts for 90 days
         refreshInSeconds: 8 * 3600, // refresh ID every 8 hours to ensure it's fresh
       },
     },
    ],
    auctionDelay: 50, // 50ms maximum auction delay, applies to all userId modules
 },
});
```

In this case (when using prebid version 8.15) immidiatly after page initialization we have the following id in prebid stream:

After running additional auction:

After running 10 additional auctions:

Example of an outgoing request:



Possible code causing this issue

According to our analysis was introduced here:

https://github.com/prebid/Prebid.js/commit/425cf20b55c8b1cccae5cc66157ce7154d091c94

 $(\underline{https://github.com/prebid/Prebid.js/commit/425cf20b55c8b1cccae5cc66157ce7154d091c94})$