

# Guarded Publisher ID

03/14/2025 8:37 am EDT

## Introduction to Guarded Publisher ID (GPID)

ID5 is offering partners a license to the Guarded Publisher ID, a publisher specific first-party cross-domain identifier. The GPID is a partner specific version of the ID5 ID which remains unique for a user across a owned and operated properties within a given browser environment. It's quality and stability is implicitly linked to the provision of signals such as hashed emails and TrueLink signals. It is especially useful as a first party user identifier for media owners or brands with multiple owned and operated properties with a limited amount of authenticated traffic.

Since the GPID is derived from the ID5 ID but it will not have cross-publisher user reconciliation properties. This means that the same user will have a different GPID on other publishers' domains or apps. The GPID will expire after 90 days in order to protect user's privacy.

The GPID can be used to facilitate use cases such as

- Cross domain audience building and activation
- Campaign optimisation and frequency capping
- Measurement
- As a Publisher Provided ID (PPID) in an ad server such as Google Ad Manager, Xandr.

The GPID is at its most accurate and stable when publishers:

- Share additional signals like hashed email as described [here](#)
- Implement an [ID5 TrueLink integration](#) alongside their standard ID5 deployment

## Publishers with a Prebid.js integration

### Using GPID as PPID in Prebid v9.31 or higher.

Guarded publisher ID is provided as PPID and can be natively used in Prebid after version 9.31. To use it, the `gp.id5-sync.com` needs to be provided as a ppid source in prebid userSync configuration.



This implementation will only work if the `externalModuleUrl` is configured in our UserIds Prebid module.

```
pbjs.setConfig({
  userSync: {
    ppid: 'gp.id5-sync.com',
    userIds: [], //userIds modules should be configured here
  }
});
```

## Publishers with an ID5 JS API Integration

This is sample code only and may need some modification in your production environment.



```
var id5Callback = function (id5Status) {  
  // ... existing callback code  
  
  // pick up the gpid  
  var ppid = id5Status.getGpid();  
  if (ppid) {  
    // pass the gpid as PPID to GPT tag  
    window.googletag.pubads().setPublisherProvidedId(ppid);  
  }  
};  
var id5Status = ID5.init({ partnerId: 173 }).onUpdate(id5Callback);
```