# debugbar.ini [DebugBarSettings]

The settings of [DebugBarSettings] in debugbar.ini: ReadAccess, Settings, Groups, GroupNames, DiscoverBlocks, ExtensionDiscovery, DiscoveryPattern, DiscoveryExclude ...

Source file: `doc/eng-us/examples/ini-settings/debugbar/DebugBarSettings.md` Last changed 2026-11-04
https://doc.exponential.earth/exphelp/viewer/examples/ini-settings/debugbar/DebugBarSettings.md

The block `[DebugBarSettings]` of `settings/debugbar.ini`, with 18 setting(s).

Never edit `settings/debugbar.ini` itself: an update replaces it. Put the settings you change in an override, which wins over the shipped file:

- `settings/override/debugbar.ini.append.php` for every siteaccess,
- `settings/siteaccess/<siteaccess>/debugbar.ini.append.php` for one siteaccess,
- `extension/<extension>/settings/debugbar.ini.append.php` in an extension (read when the extension is active).

## ReadAccess

> Who reads what the bar shows. debug (the default): whoever gets the debug output (DebugByIP, DebugByUser), logged in
> or not, reads everything: the settings, their log and the IP test, the cache list, the policies of their roles, the
> User and the Rendering tab, the versions of PHP and Exponential, OPcache and APCu. privileged: only users with
> setup/setup (settings, log, policies) or setup/managecache (cache list) read those, and the User and the Rendering
> tab show to setup/setup and to an anonymous visitor with debug output (the stricter rule). Changing settings and
> clearing caches always needs setup/setup or setup/managecache. A reader without setup/setup gets nobody else's data:
> the change log without who made each change, the IP and the user list masked, the IP test for their own address.
> Debug output on a public address shows every visitor their own session: set privileged in production.

Default: `debug`.

**INI**

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
ReadAccess=debug
*/ ?>
```

## Settings

> The settings, in the order the bar lists them inside a group

Default: a list of 37: `debug_output`, `debug_mode`, `debug_redirection`, `display_debug_warnings`, `debug_log_only`, `display_included_files`, `always_log`, `deprecated_log`, `script_debug_output`, `debug_toolbar`, `debug_bar_theme`, `debug_by_ip` ….

**INI**

```ini
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "Settings[]" first replaces the shipped list; without it the items are added
Settings[]
Settings[]=debug_output
Settings[]=debug_mode
Settings[]=debug_redirection
Settings[]=display_debug_warnings
Settings[]=debug_log_only
Settings[]=display_included_files
Settings[]=always_log
Settings[]=deprecated_log
Settings[]=script_debug_output
Settings[]=debug_toolbar
Settings[]=debug_bar_theme
Settings[]=debug_by_ip
Settings[]=debug_ip_list
Settings[]=debug_by_user
Settings[]=debug_user_list
Settings[]=template_debug
Settings[]=template_show_xhtml_code
Settings[]=template_show_used_templates
Settings[]=template_path_comments
Settings[]=template_show_method_debug
# ... 17 more in the shipped file
*/ ?>
```

## Groups

> Groups, in the order the bar shows them

Default: a list of 9: `output`, `access`, `templates`, `sql`, `regional`, `mail`, `conditions`, `caching`, `extensions`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "Groups[]" first replaces the shipped list; without it the items are added
Groups[]
Groups[]=output
Groups[]=access
Groups[]=templates
Groups[]=sql
Groups[]=regional
Groups[]=mail
Groups[]=conditions
Groups[]=caching
Groups[]=extensions
*/ ?>
```

## GroupNames

Default: a hash of 9: `output=Debug output`, `access=Who gets debug`, `templates=Templates`, `sql=Database`, `regional=Translations`, `mail=Mail`, `conditions=Debug conditions`, `caching=Caching`, `extensions=Extensions`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "GroupNames[]" first replaces the shipped list; without it the items are added
GroupNames[]
GroupNames[output]=Debug output
GroupNames[access]=Who gets debug
GroupNames[templates]=Templates
GroupNames[sql]=Database
GroupNames[regional]=Translations
GroupNames[mail]=Mail
GroupNames[conditions]=Debug conditions
GroupNames[caching]=Caching
GroupNames[extensions]=Extensions
*/ ?>
```

## DiscoverBlocks

> Blocks whose every variable is a bool setting of the bar: <file>;<block>;<group>

Default: a list of 1: `debug.ini;GeneralCondition;conditions`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "DiscoverBlocks[]" first replaces the shipped list; without it the items are added
DiscoverBlocks[]
DiscoverBlocks[]=debug.ini;GeneralCondition;conditions
*/ ?>
```

## ExtensionDiscovery

> Extension debug switches: in every active extension's settings/*.ini and *.ini.append.php, a variable whose name matches DiscoveryPattern and whose value is enabled/disabled or true/false
> becomes a bool setting of the group extensions, unless the registry already has it.

Default: `enabled`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
ExtensionDiscovery=enabled
*/ ?>
```

## DiscoveryPattern

Default: `/debug/i`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
DiscoveryPattern=/debug/i
*/ ?>
```

## DiscoveryExclude

> <extension> or <extension>;<file> (e.g. ezjscore;ezjscore.ini) to leave out

Default: an empty list.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "DiscoveryExclude[]" first replaces the shipped list; without it the items are added
DiscoveryExclude[]
*/ ?>
```

## Presets

> Presets, applied from the bar (each value a separate, undoable write)

Default: a list of 4: `template_work`, `sql_tuning`, `everything`, `off`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "Presets[]" first replaces the shipped list; without it the items are added
Presets[]
Presets[]=template_work
Presets[]=sql_tuning
Presets[]=everything
Presets[]=off
*/ ?>
```

## Thresholds

> The colours of the live summary: <warn>;<high>
> SQL query cache hit rate of the request in percent: amber BELOW the first, red below the second
> (also amber when a write made every result stale, red in shared mode without APCu)
> Memory of OPcache and APCu in use, in percent (shown to who reads everything, see ReadAccess; APCu is shared by
> every site of the PHP pool, and the query cache stops adding results below 25 % free, its ApcuReserve)

Default: a hash of 8: `time_ms=500;1500`, `sql_count=100;300`, `sql_ms=200;800`, `memory_mb=64;192`, `templates=150;400`, `querycache_hit_rate=50;10`, `opcache_memory=80;95`, `apcu_memory=75;90`.

INI

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "Thresholds[]" first replaces the shipped list; without it the items are added
Thresholds[]
Thresholds[time_ms]=500;1500
Thresholds[sql_count]=100;300
Thresholds[sql_ms]=200;800
Thresholds[memory_mb]=64;192
Thresholds[templates]=150;400
Thresholds[querycache_hit_rate]=50;10
Thresholds[opcache_memory]=80;95
Thresholds[apcu_memory]=75;90
*/ ?>
```

## SessionMask

> The section Session of the User tab (the user's own PHP session, loaded when opened): keys whose name matches one
> of these masks (a part of the name, a regular expression without delimiters, case-insensitive) show only their type
> and length, never the value; everything below a masked key is masked too. Empty: token, csrf, nonce, secret, passw,
> hash, key$, credential, auth, jwt, bearer, otp, saml, assertion, ticket, sid$. An extension adds its own in its

> settings/debugbar.ini.append.php. Masks hide secrets, not personal data: names or addresses in the session are shown.

Default: a list of 16: `token`, `csrf`, `nonce`, `secret`, `passw`, `hash`, `key$`, `credential`, `auth`, `jwt`, `bearer`, `otp` ....

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "SessionMask[]" first replaces the shipped list; without it the items are added
SessionMask[]
SessionMask[]=token
SessionMask[]=csrf
SessionMask[]=nonce
SessionMask[]=secret
SessionMask[]=passw
SessionMask[]=hash
SessionMask[]=key$
SessionMask[]=credential
SessionMask[]=auth
SessionMask[]=jwt
SessionMask[]=bearer
SessionMask[]=otp
SessionMask[]=saml
SessionMask[]=assertion
SessionMask[]=ticket
SessionMask[]=sid$
*/ ?>
```

## SessionValueMask

> Values that are masked whatever their key is called, in addition to the patterns that always apply (a JSON Web
> Token anywhere in the text, the credentials of an Authorization header, a PEM private key, well-known API key
> prefixes, 32 or more hex digits, and a long random text of base64 characters): regular expressions without
> delimiters, case-sensitive (start one with (?i) for case-insensitive), matched against the first 4096 bytes of
> every text and every key. Example: SessionValueMask[]=^tok_[a-z0-9]{20,}$

Default: an empty list.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "SessionValueMask[]" first replaces the shipped list; without it the items are added
SessionValueMask[]
*/ ?>
```

## SessionWarnBytes

> A session larger than this many bytes (serialized) is marked as large in the User tab: every request of the user
> reads the whole session, and writes it when it changed.

Default: `65536`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
SessionWarnBytes=65536
*/ ?>
```

## Panels

> Panels of extensions: a tab each after Memory, a chip in the summary and data in the page's JSON. An
> extension adds an id here and names its class (expDebugBarPanelInterface) in [Panel_<id>] Class=; the
> class decides who may see the panel. Guide: doc/eng-us/bc/6.0/debug-bar.md, "Panels of extensions".
> Panels[]=mypanel
> [Panel_mypanel]
> Class=myExtensionDebugBarPanel

Default: an empty list.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
# an empty "Panels[]" first replaces the shipped list; without it the items are added
Panels[]
*/ ?>
```

## RenderTree

> The template tree of the Rendering tab: every template the page rendered, nested, with the file the overrides chose
> and its time (at most 2000). Recorded only while debug output is on; disabled leaves it out.

Default: `enabled`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
RenderTree=enabled
*/ ?>
```

## Theme

> The looks of the bar: default (light chips) or toolbar (a dark, compact toolbar with an icon in each block, the
> request's HTTP status and module/view in the first block, and a block's details in a popup above it). The panel
> above the bar is the same in both. Any other value is default.

Default: `default`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
Theme=default
*/ ?>
```

## LogFile

> The write log, in var/<site>/log/ (one JSON object per line)

Default: `debugbar.log`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
LogFile=debugbar.log
*/ ?>
```

## LogListLimit

> Entries the bar lists

Default: `20`.

**INI**

```
<?php /* #?ini charset="utf-8"?
# settings/override/debugbar.ini.append.php
[DebugBarSettings]
LogListLimit=20
*/ ?>
```

## Related

- All blocks of debugbar.ini
- All INI files
- The settings page

---

Generated by `./console exp:help:examples` from `settings/debugbar.ini`. Do not edit this

page: change the source and generate again (how). Free documentation: GFDL 1.3 or later.

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.