

The whole Trust+ / Internet Sehat blacklist database, now in one regular expression;

reinhardt1010.id – 7 January 2023

From <https://reinhardt1010.id/blog/2023/01/07/trustpositif-regex>. Scan the QR Code to view the article on your device or web browser.



Your Internet access provided by



Empowering You!

Oops!

Maaf, akses ke situs ini diblokir sehubungan dengan Peraturan Menteri KomInfo No. 19/ 2014 tentang Internet Sehat. Terima kasih atas pengertian Anda.

Sorry, access to this site is blocked in relation to the Ministry of Communication and Informatics Regulation No. 19/ 2014 related to Safe Internet. Thank you for your understanding.

Untuk informasi lebih lanjut untuk situs yang Anda tuju, Anda dapat mengakses <http://trustpositif.kominfo.go.id>. For further information on the site you try to access, please refer to <http://trustpositif.kominfo.go.id>



Content may subject to copyright. Visit the original website to view copyright and licensing information about this content. QR Code is a registered trademark of DENSO WAVE, Inc. in Japan and other countries. Generated on 2025-04-19 14:59:17.

(#_)!

At the end of 2022, I decided to experiment on building a *lightweight* Indonesian internet blacklist database, which can be consumed offline.

No network connections to servers of Kominfo, Telkom Indonesia, and community-run services like indi.wtf. Because all you need is a freakin' huge [regular expression](#).

Research methods

We wrote a simple Go script to compile the official Indonesian internet blacklist, found on <https://trustpositif.kominfo.go.id>, and convert it into a freakin' huge trie. Then that trie is then converted into regular expressions.

And to test whether the regex is effective, we decided to test the generated regex back against the original list of blocked domains.

Results

The experiment grew a 20MB-ish regex file, representing the freakin' huge trie I have mentioned earlier. That said, there's always many ways to improve, including reversing the original domain's arrangement of characters (e.g. "alterine0101.id" ↔ "di.1010eniretla") to yield more compact results (because there are more domains ending with ".com" instead of those starting with "www.").

Unfortunately, these gigantic regex files cannot be parsed by Go's own **regexp** system library, hence we decided to use the [regexp2](#) library instead, which is based on Microsoft's regex parses implementation for .NET.

And even if I switch to regexp2, only the reversed version of the regex would work well. I feel confident that the generated regex is 99.9% accurate, tested on Reinhart's M1 MacBook Air with no issues.

You can see my GitHub repo [here](#) for the code and the results. Feel free to use that as a benchmark tool for PCRE regex engines out there. We may eventually update the blocked domains list, eventually, to ensure the freshness of these regex-based blocklists.

That's all and (#_)!