## NAME

vfs_preopen - Hide read latencies for applications reading numbered files

## SYNOPSIS

vfs objects = preopen

## DESCRIPTION

This VFS module is part of the **samba**(7) suite.

This module assists applications that want to read numbered files in sequence with very strict latency requirements. One area where this happens in video streaming applications that want to read one file per frame.

When you use this module, a number of helper processes is started that speculatively open files and read a number of bytes to prime the file system cache, so that later on when the real application's request comes along, no disk access is necessary.

This module is stackable.

## OPTIONS

preopen:posix-basic-regex = BOOL (default: no)

preopen:posix-basic-regex = yes changes the meaning of the preopen:names option. Further details are described there.

preopen:names = /pattern1/pattern2/

preopen:names specifies the file name pattern(s) which should trigger the preopen helpers to do their work. We assume that the files are numbered incrementally. So if your file names are numbered FRAME00000.frm FRAME00001.frm and so on you would list them as preopen:names=/FRAME*.frm/. The default algorithm uses the first (at least 3) digits it finds in order to calculate the name of the next frames.

preopen:posix-basic-regex = yes changes the meaning of the preopen:names option. It means 'POSIX Basic Regular Expression' strings are used as patterns. The key is each pattern requires exactly one 'subexpression' starting with '\(' and ending with '\)' in order to specify the position of the digits representing the incrementing frame numbers. Given a file names like Movie7599Frame0v1234.txt, Movie7599Frame1v1234.txt, Movie7599Frame2v1234.txt up to Movie7599Frame9v1234.txt you can use preopen:names = /.*Frame\([0-9]\).*\.txt/ in order to match just a single digits, this might not be a real world example, but it shows the flexiblity that is possible here.

preopen:num_bytes = BYTES

Specifies the number of bytes the helpers should speculatively read, defaults to 1.

preopen:helpers = NUM-PROCS

Number of forked helper processes, defaults to 1.

preopen:queuelen = NUM-FILES

Number of files that should be speculatively opened. Defaults to the 10 subsequent files.

preopen:nomatch_log_level = LOGLEVEL

In order to debug or audit the usage of the preopen logic you can use this option to specify at what log level details about filenames not matching any pattern from ’preopen:names’ are logged.

Defaults to the log level 5. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:match_log_level = LOGLEVEL

In order to debug or audit the usage of the preopen logic you can use this option to specify at what log level details about filenames actually matching a pattern from ’preopen:names’ are logged.

See also ’preopen:founddigits_log_level’ and ’preopen:push_log_level’.

Defaults to the log level 5. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:nodigits_log_level = LOGLEVEL

In order to debug or audit problems with the preopen configuration you can use this option to specify at what log level details about filenames actually matching a pattern from ’preopen:names’, but at the same time don’t contain the expected digits, are logged. This is typically something the administrator wants to notice and adjust the configuration in order to define more precisely where to find the digits in the filename.

Defaults to the log level 1. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:founddigits_log_level = LOGLEVEL

In order to debug or audit the usage of the preopen logic you can use this option to specify at what log level details about filenames actually matching a pattern from ’preopen:names’, and at the same time having valid expected digits, are logged. This means enough information is available in order to queue preopens.

Defaults to the log level 3. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:reset_log_level = LOGLEVEL

If a matching filename belongs to a different pattern from ’preopen:names’, a different parent directory or differs in a significant way from the last filename that was found before, the module needs to reset it’s internal queue state. This means that no more preopens will be pushed to helper processes belonging to the former queue state. In order to debug or audit such queue resets you can use this option to specify at what log level details are logged.

Defaults to the log level 5. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:push_log_level = LOGLEVEL

In order to debug or audit the usage of the preopen logic you can use this option to specify at what log level details about filenames actually pushed to preopen helper processes are logged. This means they will actually be preopened soon.

Defaults to the log level 3. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

preopen:queue_log_level = LOGLEVEL

In order to debug details about internal queue processing you can use this option to specify at what log level the details are logged.

Defaults to the log level 10. See also **smb.conf**(5) in the ’**log level**’ section for special handling of the ’preopen’ debug class.

## VERSION

This man page is part of version 4.16.11 of the Samba suite.

## AUTHOR

The original Samba software and related utilities were created by Andrew Tridgell. Samba is now developed by the Samba Team as an Open Source project similar to the way the Linux kernel is developed.

The PREOPEN VFS module was created with contributions from Volker Lendecke and the developers at IBM.