

draft-masotta-tftpexts-window-size-opt-06.txt  
Intended status: Informational  
INTERNET-DRAFT  
Expires: Apr 14, 2013

Patrick Masotta/Vercot  
Nov 2012

## TFTP Window Size Option

### Abstract

The Trivial File Transfer Protocol [[1](#)] is a simple, lock-step, file transfer protocol which allows a client to get or put a file onto a remote host. One of its primary uses is the early stages of nodes booting from a Local Area Network. TFTP has been always used because it is very simple to implement. However, the choice of a lock-step schema is not the most efficient for use on a LAN.

This document describes a TFTP option which allows the client and server to negotiate a window size of consecutive blocks to send as an alternative for replacing the single block lock-step schema. The TFTP Option Extension mechanism is described in [[2](#)].

### Legal

This document and the information contained therein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION THEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

### Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Distribution of this draft is unlimited.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at

<http://www.ietf.org/1id-abstracts.html>

The list of Internet-Draft Shadow Directories can be accessed at  
<http://www.ietf.org/shadow.html>

## Introduction

While virtually unused for internet transfers today, TFTP is still massively used in network boot and network installation scenarios. The protocol inherently low transfer rate has been so far partially mitigated by the use of the blocksize negotiated extension [3]. This way the original 512 byte blocks are in practice replaced on Ethernet environments by blocks no larger than 1468 Bytes to avoid IP block fragmentation. The former strategy results insufficient when it comes to transfer big files i.e. the initial ram-disk of live distributions of Linux or the PE images of Microsoft WDS net install scenarios. Considering TFTP looks today far from extinction this draft formally presents a natural extension, already implemented by some companies, that produces TFTP transfer rates comparable to the ones achieved by modern file transfer protocols.

## WindowSize Option Specification

The TFTP Read Request or Write Request packet is modified to include the window size option as follows. Note that all fields except "opc" are NULL-terminated.

```
+-----+-----+-----+-----+-----+-----+-----+-----+
|  opc  |filename| 0  |  mode  | 0  | window size | 0  | #blocks| 0  |
+-----+-----+-----+-----+-----+-----+-----+-----+
```

### opc

The opcode field contains either a 1, for Read Requests, or 2, for Write Requests, as defined in [1].

### filename

The name of the file to be read or written, as defined in [1].

### mode

The mode of the file transfer: "netascii", "octet", or "mail", as defined in [1].

### window size

The Window size option, "window size" (case in-sensitive).

### #blocks

The number of blocks in a window, specified in ASCII. Valid values range between "1" and "65535" blocks, inclusive. The window size refers to the number of consecutive blocks transmitted before stop and wait for the reception of the acknowledgment of the last block transmitted.

For example:

```
+-----+-----+--+-----+--+-----+-----+-----+--+
|  1    | foobar | 0 | octet | 0 | window size | 0 | 16  | 0 |
+-----+-----+--+-----+--+-----+-----+-----+--+
```

is a Read Request, for the file named "foobar", in octet transfer mode, with a window-size of 16 blocks (as option blocksize is not negotiated in this case, the 512 Bytes per block default applies).

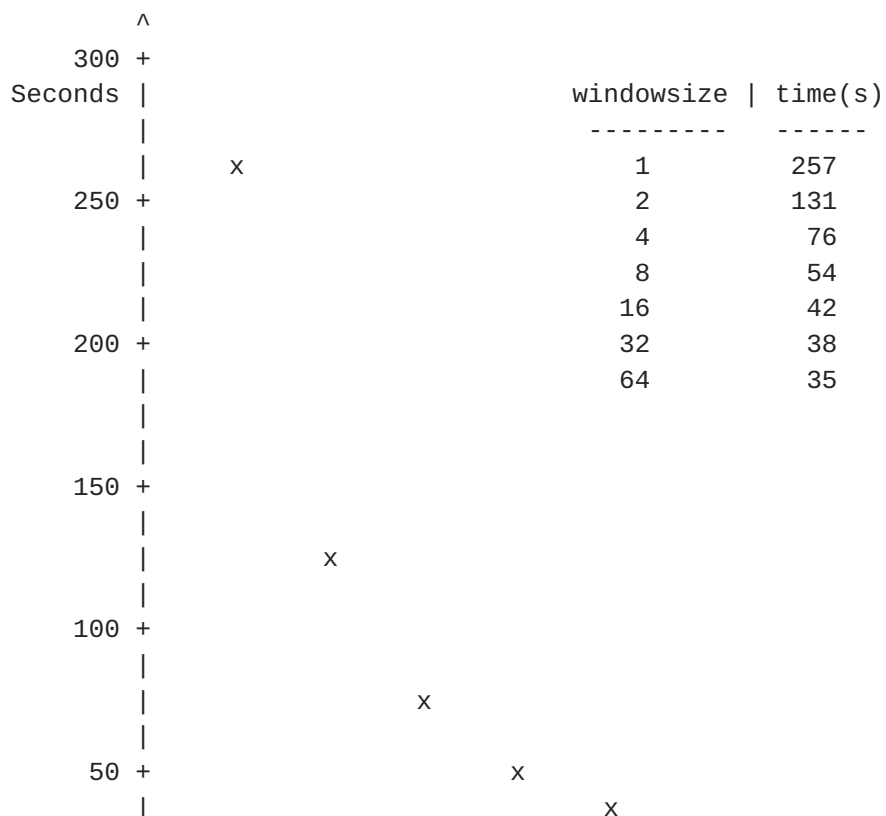
If the server is willing to accept the window size option, it sends an Option Acknowledgment (OACK) to the client. The specified value must be less than or equal to the value specified by the client. The client must then either use the size specified in the OACK, or send an ERROR packet, with error code 8, to terminate the transfer.

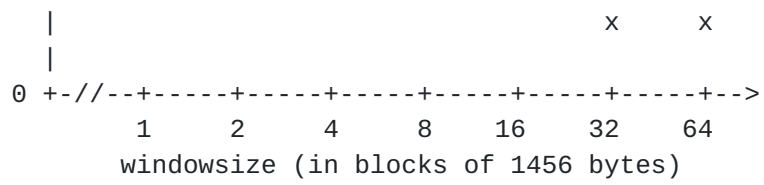
The rules for determining the final packet are unchanged from [\[1\]](#) and [\[3\]](#).

The reception of a data window with a number of blocks less than the negotiated window size is the final window. If the window size is greater than the amount of data to be transferred, the first window is the final window.

#### Proof of Concept

Performance tests were run on the prototype implementation using a variety of window sizes and a fixed blocksize of 1456 bytes. The tests were run on a lightly loaded Gigabit Ethernet, between two Toshiba Tecra Core 2 Duo 2.2 Ghz, in "octet" mode, transferring a 180 MByte file.





The comparison of transfer times (without a gateway) between the standard lock-step schema and the negotiated window sizes are:

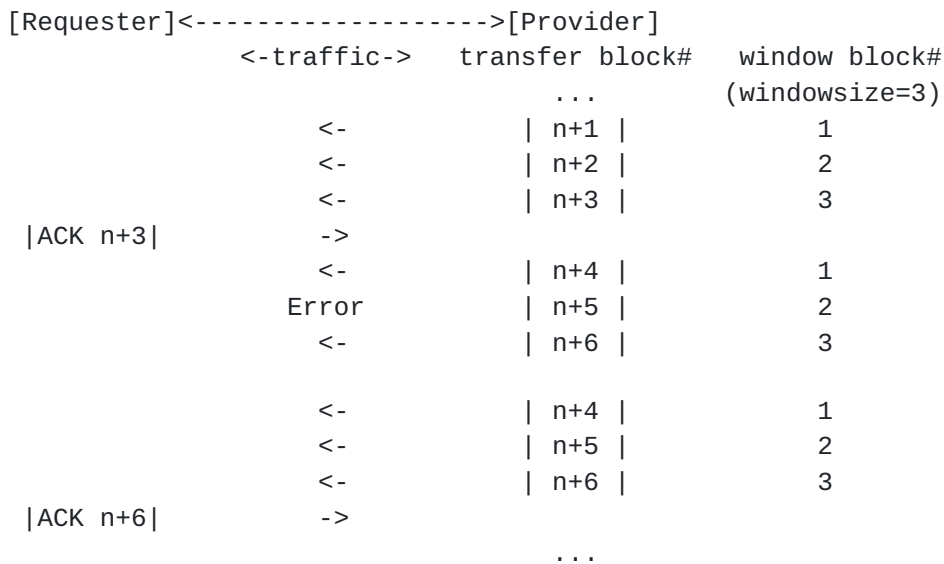
1	-0%
2	-49%
4	-70%
8	-79%
16	-84%
32	-85%
64	-86%

As it was expected, the transfer time decreases with the use of a windowed schema. The reason for the reduction in time is the reduction in the number of the required synchronous acknowledgements exchanged.

Comparatively the same 180 MB transfer performed over an SMB/CIFS mapped drive took 23 seconds.

#### Error Handling

In case of an error detection the whole window size window is retransmitted.



Section of a transfer including error and error recovery

#### Security Considerations

The basic TFTP protocol does not have an explicit security mechanism. However it also does not have list, rename, delete, nor overwrite capabilities either. This document does not add any security to TFTP nor the specified extension adds any additional security risk either.

## IANA Considerations

This document has no actions for IANA.

Patrick Masotta

Expires Apr 14, 2013

[Page 4]



## Normative References

- [1] Sollins, K., "The TFTP Protocol (Revision 2)", [RFC 1350](#) (STD 33), October 1992.
- [2] Malkin, G., Harkin, A., "TFTP Option Extension", [RFC 2347](#) May 1998.
- [3] Malkin, G., Harkin, A., "TFTP Blocksize option", [RFC 2348](#) May 1998.

## Authors' Addresses

Patrick Masotta  
300 W 11th Avenue, #9A  
Denver, CO 80204

EMail: masotta[-at-]vercot[-dot-]com

## Copyright and IPR Provisions

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of [RFC 5378](#). No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under [RFC 5378](#), shall have any effect and shall be null and void, whether

published or posted by such Contributor, or included with or in such Contribution.

Patrick Masotta

Expires Apr 14, 2013

[Page 5]