

Telnet Environment Option Interoperability Issues

Status of this Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

1. Overview

RFC 1408 [1], the specification for the Telnet Environment Option, specifies definitions for VAR and VALUE that are reversed from the BSD implementation of the Telnet Environment option. Since the BSD implementation was the reference implementation that the RFC was supposed to document, and is the base for many existing implementations, there exists an interoperability problem between implementations with conflicting definitions for VAR and VALUE.

This document describes a method for allowing implementors to ensure that their implementation of the Environment option will be interoperable with as many other implementations as possible, by providing a set of heuristics that can be used to help identify which definitions for VAR and VALUE are being used by the other side of the connection.

2. Client Telnet: Parsing a SEND

The SEND suboption should only contain VAR and USERVAR commands. It should never contain a VALUE. If while parsing a SEND suboption, a VALUE is encountered, the client should assume that the server has reversed values for VAR and VALUE, and from that point on the client should reverse those values, both in parsing the rest of the SEND suboption, and when generating an IS or INFO suboption. If both VALUE and VAR commands are encountered, the SEND command is not well formed, and it is implementation dependent as to what will happen.

If there are not VAR or VALUE commands in the SEND suboption, then the client cannot know what values the server is expecting for VAR and VALUE. In this case, it should just assume that the server has the correct definitions, and use the correct values for VAR and VALUE.

3. Server Telnet: Parsing an IS or INFO

The IS or INFO in a suboption can only be legally followed by either a VAR or a USERVAR. If an IS or INFO is immediately followed by a VAR, then it can be assumed that the client has the correct definitions for VAR and VALUE. If the IS or INFO is immediately followed by a VALUE, then it can be assumed that the client has reversed definitions for VAR and VALUE, and from that point on the server should assume that the VALUE and VAR definitions are reversed.

If the IS or INFO is immediately followed by a USERVAR, further heuristics must be applied to determine what are the client definitions for VAR and VALUE. This is because it is legal for a USERVAR to be followed by either a VAR or a VALUE. A VALUE after a USERVAR gives the value for the USERVAR. A VAR after a USERVAR implies that the USERVAR is undefined.

The next thing to do is to scan the entire suboption, looking for two consecutive instances of VAR or VALUE, or for a VAR or VALUE that is empty. It is not legal for a suboption to contain two VALUES without an intervening VAR or USERVAR, and it is also not legal for a suboption to contain an empty VAR. Thus, if two consecutive VARs or an empty VALUE can be found, it can be assumed that the client that generated the suboption uses the correct definitions for VAR and VALUE. If two consecutive VALUES or an empty VAR can be found, then it can be assumed that the client that generated the suboption has reversed definitions for VAR and VALUE, and from that point on the server should assume that the VAR and VALUE definitions are reversed.

If things are still in doubt, the next test that can be applied is to count up how many VARs, USERVARs and VALUES were received. (Consecutive USERVARs without an intervening VALUE or VAR should only be counted as one.) Because a VALUE can only follow a VAR or a USERVAR, there can never be more VALUES than the sum of VARs and USERVARs, and if all VARs and USERVARs have values, then there will be exactly as many VALUES as there are VARs and USERVARs. If the number of VARs and USERVARs is the same as the total number of VALUES, then the client has correct definitions for VAR and VALUE. If the number of VALUES and USERVARs is the same as the total number of VARs, then the client has reversed definitions for VAR and VALUE.

If the number of VALUES is more than the sum of VARs and USERVARs, it can be assumed that the client has reversed definitions of VAR and VALUE, and if there are more VARs than USERVARs and VALUES, then it can be assumed that the client has the correct definitions for VAR and VALUE. However, in order to get to this step, it has already been determined that there are no consecutive VARs and VALUES. A little math will show that this means that the number of VALUES will

never exceed the sum of VARs and USERVARs, and the number of VARs will never exceed the sum of VALUEs and USERVARs. Hence, this check is redundant and can be skipped.

If things are still in doubt, the values of the VAR commands can be checked to see if they do indeed specify well known variables. If any of them do, then the client is probably using the correct definitions for VAR and VALUE. Otherwise, if any of the VALUEs contain well known variable names, then the client probably has reversed definitions for VAR and VALUE.

If all the above heuristics fail, then the server has done all it can to determine what type of client it is, and it should just be assumed that the client is using the correct definitions for VAR and VALUE.

4. Client Summary

The SEND suboption contains only VAR and USERVAR commands.
The server is ok.
The SEND suboption contains VALUE commands.
The server is reversed.
No VAR or VALUE commands are found.
Assume the server is ok.

5. Server Summary

IS/INFO is followed by VAR.
The client is ok.
IS/INFO is followed by VALUE.
The client is reversed.
There are two consecutive VARs.
The client is ok.
There are consecutive VALUEs.
The client is reversed.
There is an empty VALUE.
The client is ok.
There is an empty VAR.
The client is reversed.
The number of USERVARs and VARs is equal to the number of VALUEs.
Assume the client is ok.
The number of USERVARs and VALUEs is equal to the number of VARs.
Assume the client is reversed.
There are VARs with names that are well known.
Assume the client is ok.
There are VALUEs with names that are well known.
Assume the client is reversed.
Anything else.
Assume the client is ok.

References

- [1] Borman, D., Editor, "Telnet Environment Option", RFC 1408, Cray Research, Inc., January 1993.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

David A. Borman
Cray Research, Inc.
655F Lone Oak Drive
Eagan, MN 55123

Phone: (612) 452-6650
EMail: dab@CRAY.COM

Telnet Working Group Mailing List: telnet-ietf@CRAY.COM