

Network Working Group
Internet-Draft
Intended status: Experimental
Expires: December 23, 2022

B. Squillari
F. Zimbimbakis
G. Budo Berra
ITBA
June 21, 2022

M16 server management protocol
m16-server-management-protocol-05

Abstract

This document specifies a experimental protocol for the exchange of requests with a server in order to perform server behavior modifications or server performance statistics retrieval, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on December 23, 2022.

Copyright Notice

Copyright (c) 2022 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 (<https://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.

Table of Contents

1. Introduction	2
2. Conventions	2
3. Procedure for clients	2
4. Requests	4
5. Replies	6
5.1. Status	6
5.2. Fields	7
6. Security	9
7. Normative References	9
Authors' Addresses	9

1. Introduction

With the emerge of more sophisticated communication systems, there exists a need to provide a standard protocol to transparently and securely communicate with these systems in order to adapt them to different situations.

There exists, also, a need to retrieve performance statistics in order to achieve maximum efficiency through analysis.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119[RFC2119].

Unless otherwise noted, the decimal numbers appearing in packet format diagrams represent the length of the corresponding field, in octets. Where a given octet must take on a specific value, the syntax X'hh' is used to denote the value of the single octet in that field. When the word 'Variable' is used, it indicates that the corresponding field has a variable length defined either by an associated (one or two octet) length field, or by a data type field.

In addition, all fields which are larger than one octet and contain ASCII characters are represented in network byte order. If the field is larger than one octet and does not contain ASCII characters, then it is represented in little endian.

3. Procedure for clients

When a client wishes to establish a connection to a server which provides M16 protocol, it must open a TCP connection to the appropriate port on the server system. The service is conventionally located on TCP port 8080. If the connection request succeeds, the

client enters a negotiation for the authentication method to be used, authenticates with the chosen method, then sends a request. The server evaluates the request, replies with the appropriate reply and closes the connection.

The client connects to the server, and sends a version identifier/method selection message:

VER	NMETHODS	METHODS
X'01'	1	1 to 255

The VER field is set to X'01' for this version of the protocol. The NMETHODS field contains the number of method identifier octets that appear in the METHODS field.

The server selects from one of the methods given in METHODS, and sends a METHOD selection message:

VER	METHOD
X'01'	1

If the selected METHOD is X'FF', none of the methods listed by the client are acceptable, and the client MUST close the connection.

The values currently defined for METHOD are:

- o X'00' NO AUTHENTICATION REQUIRED
- o X'02' USERNAME/PASSWORD
- o X'FF' NO ACCEPTABLE METHODS

The client and server then enter a method-specific sub-negotiation. Descriptions of the method-dependent sub-negotiations appear in separate memos.

Descriptions of the method USERNAME/PASSWORD sub-negotiation appear in RFC 2119[RFC1929].

Compliant implementations SHOULD support USERNAME/PASSWORD authentication methods.

4. Requests

Once the method-dependent subnegotiation has completed, the client sends a request to the server.

The request is formed as follows:

REQUEST INDEX	PARAMETERS
1	Variable

Where:

- o REQUEST INDEX: refers to the selection between the actions and data retrievals the server offer. When REQUEST INDEX is X'00' server MUST reply with a list of supported indexes.
- o PARAMETERS: refers to the needed parameters to accomplish the request. The quantity, size and content of the parameters are directly dependant of the REQUEST INDEX, this information will be detailed below. Parameters have fixed or variable length. In case of a variable length, the parameter MUST follow the next structure:

LENGTH	DATA
1	Variable

Where LENGTH contains the number of octets of the DATA that follow, there is no terminating NUL octet. And DATA contains the parameter information. DATA size is restricted to 255 octets.

The possible REQUEST INDEX, their functionality and the corresponding PARAMETERS with their length will be described below. In the request, PARAMETERS MUST follow the listed order below.

Information retrieval indexes:

- o X'00' A list of all the REQUEST INDEX that the server supports. PARAMETERS: None.
- o X'01' List of active users on the server. PARAMETERS: None.

- o X'02' Amount of historic connections to the server. PARAMETERS: None.
- o X'03' Amount of actual concurrent connections to the server. PARAMETERS: None.
- o X'04' Max concurrent connections to the server. PARAMETERS: None.
- o X'05' Amount of historic byte transferred by the server. PARAMETERS: None.
- o X'06' Amount of historic authentication attempts to the server. PARAMETERS: None.
- o X'07' Amount of historic connections attempts to the server. PARAMETERS: None.
- o X'08' Average bytes per single server read. PARAMETERS: None.
- o X'09' Average bytes per single server write. PARAMETERS: None.

Behavior modification indexes:

- o X'0A' Add a new user. PARAMETERS:
 - 1. Content: Username. Type: ASCII characters. Size: Variable.
 - 2. Content: Password. Type: ASCII characters. Size: Variable.
- o X'0B' Delete a user. PARAMETERS:
 - 1. Content: Username. Type: ASCII characters. Size: Variable.
- o X'0C' Disable server authentication. PARAMETERS:
 - 1. Content: New authentication status. Size: 1 octet. Where X'00' indicates to turn on authentication and any other value means to turn authentication off.
- o X'0D' Disable password dissectors. PARAMETERS:
 - 1. Content: New password dissectors status. Size: 1 octet. Where X'00' indicates to turn on password dissectors and any other value means to turn password dissectors off.

Server MUST support the index X'00'. The rest of the indexes are not compulsory, but RECOMMENDED.

The server will typically evaluate the request based on the REQUEST INDEX and PARAMETERS, and return one reply messages with the appropriate data.

5. Replies

The reply MUST follow:

STATUS	FIELDS
1	Variable

The reply will have a STATUS and fields related to the request of the client. The fields, if any, will contain the main reply information. The size and content of the fields are directly dependant of the REQUEST INDEX, this information will be detailed below. Fields have fixed or variable length. In case of a variable length, it should follow the next structure:

LENGTH	DATA
1	Variable

Where LENGTH contains the number of octets of the DATA that follow, there is no terminating NUL octet. And DATA contains the field information. DATA size is restricted to 255 octets.

The M16 server MUST terminate the TCP connection shortly after sending the reply.

5.1. Status

STATUS describes the return value of the server related to a request. The possible status are:

- o X'00': Success
- o X'01': Server error
- o X'02': Index not supported
- o X'03': Failed to create a new user because server reached the maximum number of users

In case of a failure status, server must reply with the status. One field MAY be used by the server to describe the error via an ASCII error message. If this is not the case, the number of fields MUST be 0.

The M16 server MUST terminate the TCP connection shortly after sending the reply.

5.2. Fields

In case of success status, the fields on the reply directly depend of the REQUEST INDEX. Unless specified below, the possible status for an index are X'00', X'01', X'02'. The next list shows the REQUEST INDEX with their corresponding FIELDS on the reply:

Information retrieval indexes:

- o X'00': Number of fields: 1. FIELDS:
 1. Content: List of supported indexes. Type: byte array Size: Variable. Each byte of the data represents a single supported index

Possible return status: X'00', X'01'.

- o X'01': Number of fields: Variable. This is a special case where the first field will be used to inform the amount of fields that follow. FIELDS:
 1. Content: Amount of fields that follow. Each of these fields represent a user. Type: byte. Size: 1 octet.
 2. Content: Username. Type: ASCII characters. Size: Variable.
 3. ...
- o X'02': Number of fields: 1. FIELDS:
 1. Content: Historic connections to the server. Type: unsigned int. Size: 4 octets.
- o X'03': Number of fields: 1. FIELDS:
 1. Content: Actual concurrent connections. Type: unsigned int. Size: 4 octets.
- o X'04': Number of fields: 1. FIELDS:

1. Content: Max concurrent connections. Type: unsigned int.
Size: 4 octets.
- o X'05': Number of fields: 1. FIELDS:
 1. Content: Historic byte transferred. Type: unsigned int.
Size: 4 octets.
- o X'06': Number of fields: 1. FIELDS:
 1. Content: Historic authentication attempts. Type: unsigned int.
Size: 4 octets.
- o X'07': Number of fields: 1. FIELDS:
 1. Content: Historic connections attempts. Type: unsigned int.
Size: 4 octets.
- o X'08': Number of fields: 1. FIELDS:
 1. Content: Average byte size per server read. Type: unsigned int.
Size: 4 octets.
- o X'09': Number of fields: 1. FIELDS:
 1. Content: Average byte size per server write. Type: unsigned int.
Size: 4 octets.

Behavior modification indexes:

- o X'0A' Number of fields: 0. Possible return status: X'00', X'01', X'02', X'03'.
- o X'0B' Number of fields: 0.
- o X'0C' Number of fields: 0.
- o X'0D' Number of fields: 0.

Developers that come up with new possible standard indexes for this protocol should contact authors for a index number.

Compliant implementations MUST support X'00' index.

6. Security

This document describes a protocol to achieve actions or information retrieval that may be sensitive for the client or the server. The security of both client and server is highly dependent on the particular authentication method provided in a particular implementation, and selected during negotiation between client and server.

Careful consideration should be given by the administrator to the selection of authentication methods.

7. Normative References

- [RFC1929] Leech, M., "Username/Password Authentication for SOCKS V5", RFC 1929, DOI 10.17487/RFC1929, March 1996, <<https://www.rfc-editor.org/info/rfc1929>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

Authors' Addresses

Bruno Omar
ITBA

EMail: bsquillari@itba.edu.ar

Facundo
ITBA

EMail: fzimbimbakis@itba.edu.ar

Gaspar Andres
ITBA

EMail: gbudoberra@itba.edu.ar