Document:	T10/00-355r0	Date:	September 13, 2000
To:	T10 Committee Membership		
From:	Edward A. Gardner, Ophidian Designs		
Subject:	Reporting Supported Request Types in SVP		

When discussing SVP with the IBTA Application Working Group, one concern that was expressed was that the environment while booting be simplified as much as possible. One piece of this is that boot code should only have to deal with a minimal set of IU types and formats. The AWG requested that I raise this with T10.

SVP IUs consist of requests and corresponding responses. With this proposal each end of an SVP connection (Initiator or Target) would identify the request types it supports and is able to receive. Claiming or denying support for a request type can be thought of as enabling or disabling the other end to issue that request. Note that support for responses need not be reported. If one does not support a response type, then one does not issue the corresponding request.

SVP IU type codes are presently encoded using the following convention:

00h – 07h	Requests sent by the Initiator to the Target
10h – 17h	Corresponding responses
08h – 0Fh	Requests sent by the Target to the Initiator
18h – 1Fh	Corresponding responses

This proposal defines two one-byte quantities.

The Target Supported Request Bit Mask indicates which request type codes in the range 00h – 07h the Target supports. A bit set to one indicates that the Target is able to receive and process the corresponding request, implying the request may be issued by the Initiator. A bit set to zero indicates that the corresponding request shall not be issued by the Initiator. Note that bit 1, corresponding to SVP_CMD, had better be set or the Target is not functional.

The Initiator Supported Request Bit Mask indicates which request type codes in the range 08h – 0Fh the Initiator supports. A bit set to one indicates that the Initiator is able to receive and process the corresponding request, implying the request may be issued by the Target. A bit set to zero indicates that the corresponding request shall not be issued by the Target.

The current SVP specification defines three request types from Initiator to Target, four if you include login which is necessarily unique and not controlled by these flags. It defines two request types from Target to Initiator. I cannot conceive of our ever defining more than 8 IU types in each direction. If anyone disagrees, please speak now.

These bit masks would be added to the current login and parameter negotiation request and response IUs near the beginning, either byte 1 or 2. As with other parameters, they may be changed during the life of a connection. I don't consider this particularly valuable, but it is the way all other parameters work and I see no reason to make it a special case.

One consequence of this change is that I will remove all mention of security challenge requests and responses. Those were intended as place holders, to ensure graceful migration if such functions were necessary in the future. The mechanism in this proposal is superior.