ENDL
TEXAS

Date: 2 November 2005
To: T10 Technical Committee & SNIA OSD TWG
From: Ralph O. Weber
Subject: Possible inconsistencies in integrity check value algorithm field definition/usage

In r0 of this thrilling page-turner, it was noted that the OSD-2 r00 Capability format includes a four-bit INTEGRITY CHECK VALUE ALGORITHM field, but the supported integrity check value algorithm attributes in the Root Policy/ Security attributes page define Vendor Specific integrity check value algorithm values that do not fit in four bits.

Table 6 — Capability format

Bit / Byte	7	6	5	4	3	2	1	0
0	Reserved				CAPABILITY FORMAT (1h)			
1	KEY VERSION				INTEGRITY CHECK VALUE ALGORITHM			
2	Reserved				SECURITY METHOD			
3	Reserved							
	⋮							

Table 114 — Supported integrity check value algorithm codes

Value	Algorithm	Reference
00h	No algorithm supported	
01h	HMAC-SHA1	FIPS 180-1 (1995) and FIPS 198 (2002)
02h - DFh	Reserved	
E0h - FFh	Vendor specific	

In the September CAP working group it was observed that the introduction to table 114 defines the usage perfectly with one level of indirection.

> The low order four bits of the attribute number are the value that appears in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.9.2.2) in each capability (e.g., attribute number 8000 0007h identifies the integrity check value algorithm used if the INTEGRITY CHECK VALUE ALGORITHM field contains seven).

The SNIA OSD TWG later reported that all implementations follow this low-order-four-bits approach.

Therefore, it is proposed to modify every discussion of the INTEGRITY CHECK VALUE ALGORITHM field in OSD-2 to enunciate clearly this rule.

All references are to osd2r00.pdf. Additions are in red. Deletions are in ~~blue strikeout~~. Notes are in dark green.

All changes are new to this revision of this proposal, so they are not identified by change bars.

18484 Preston Road, Suite 102 PMB 178, Dallas, TX 75252 214-912-1373 972-596-2775 Fx ENDL_TX@computer.org

Detailed OSD-2 Changes

4.9.2.2 Capability format

4.9.2.2.1 Introduction

… {2nd paragraph after table 7}

The KEY VERSION field, INTEGRITY CHECK VALUE ALGORITHM field, and SECURITY METHOD field are used by the security manager (see 4.10.3). If capabilities are not coordinated with the security manager, the KEY VERSION field, INTEGRITY CHECK VALUE ALGORITHM field, and SECURITY METHOD field are reserved.

{no other changes in 4.9.2.2.1}

4.10.3 Preparing credentials

…

7) Set the capability INTEGRITY CHECK VALUE ALGORITHM field to the low order four bits of the attribute number of the attribute in the Root Policy/Security attributes page (see 7.1.2.20) that ~~specifies~~ indicates the algorithm used to compute all integrity check values related to this credential (e.g., if attribute number 8000 0003h identifies the integrity check value algorithm used in this credential, then the INTEGRITY CHECK VALUE ALGORITHM field shall contain three)~~. The algorithm shall be one of those identified by the supported integrity check value algorithm attributes in the Root Policy/Security attributes page (see 7.1.2.20)~~;

{no other changes in 4.10.3}

4.10.4.3 The CAPKEY security method

…

a) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) ~~(see 4.9.2.2)~~;

{no other changes in 4.10.4.3}

4.10.4.4 The CMDRSP security method

…

a) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) ~~(see 4.9.2.2)~~;

…

1) Compute an integrity check value for the response data using:
   A) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) ~~(see 4.9.2.2)~~;

{no other changes in 4.10.4.4}

4.10.4.5 The ALLDATA security method

…

a) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) ~~(see 4.9.2.2)~~;

…

The DATA-OUT INTEGRITY CHECK VALUE field contains the data-out integrity check value computed by the application client.

The device server shall validate the data-out integrity check value by:

1) Compute an integrity check value for the response data using:
   A) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) ~~(see 4.9.2.2)~~;

…

The device server shall compute the response integrity check value using the same algorithm specified for the CMDRSP security method (see 4.10.4.4) and the application client validates the response integrity check value using the same algorithm specified for the CMDRSP security method.

The device server shall compute the data-in integrity check value using:

a) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field;

…

After status has been received, the application client validates the data-in integrity check value by:

1) Computing an integrity check value using:
   A) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page whose attribute number is specified in the capability INTEGRITY CHECK VALUE ALGORITHM field;

{no other changes in 4.10.4.5}

4.10.6.1 Credential validation

…

The device server shall validate the credential associated with a CDB by:

1) Reconstructing the credential containing the capability as described in 4.10.6.2;
2) Computing the credential integrity check value for the reconstructed credential using the algorithm, inputs, and secret key specified in 4.10.6.3;
3) Computing the request integrity check value using:
   A) The algorithm ~~specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified by the INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) in the capability;

{no other changes in 4.10.6.1}

4.10.6.3 Computing the credential integrity check value

The credential integrity check value shall be computed using:

a) The algorithm indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified by the INTEGRITY CHECK VALUE ALGORITHM ~~FIELD~~ field (see 4.10.3) in the capability;

{no other changes in 4.10.6.3}

4.10.9.2 Computing updated generation keys and new authentication keys

…

c) The integrity check value algorithm~~, as specified~~ indicated by the attribute in the Root Policy/Security attributes page (see 7.1.2.20) whose attribute number is specified in the INTEGRITY CHECK VALUE ALGORITHM field (see 4.10.3) in the capability in the CDB for the command.

{no other changes in 4.10.9.2}