Community Comments on 20131002 DRAFT

This document reflects the community comments gathered from October 2, 2013 to November 8th, 2013.

#	Comment	Submitter	Date	Notes
1	- "4.1.2 Interpretation of IAP requirement, Section 4.2.3.4 - Stored Authentication Secrets" "We interpret this requirement to mean that encryption software that decrypts disk sectors (and not just individual Authentication Secrets) as they are accessed would meet the requirement of "only decrypt(ing) the needed Secret when immediately required for authentication" as spelled out in this section, presuming such software uses Approved Algorithms for the encryption process." As written, this would be overbroad, e.g., decrypting a needed secret for one individual might result in the decryption of MULTIPLE secrets, e.g., the one for that individual AND ones used by others. As such, that would violate the requirement that passwords must "only [be] decrypted when immediately required for authentication" because you're also potentially decrypting OTHER passwords that are not needed at all. This would represent a failure to meet the requirement, at least from my POV. In the extreme case, imagine a person proposing to use boot time whole disk decryption: while off, the disk may be encrypted with an Approved Algorithm, but upon boot, the entire disk is decrypted, including the password store, which is then "immediately" (and intermittently) used until the system is eventually shut down. Would that be satisfactory/sufficient to meet the requirement? I don't think so. Remember that presumably the goal is to limit the exposure of passwords to unauthorized access or misuse. If the passwords are routinely held in non-encrypted form whenever the system is "live", except briefly during boot time when the system is coming up, it isn't clear to me that the encryption protects against any exposure except theft of the disk from a quiescent system. Any attack against the password store while the system is live would not require the attacker to decrypt the password store if the password store is routinely decrypted at boot time. Thus, I explicitly reject the argument advanced in 5.1.1 later in the document. -- "5.1.2 Remove Insecure (LMHASH) Stored Secrets" Good to see you recommend removing LMHASH'd passwords. However, unfortunately, you ALSO insist that NTLM ALSO not be used, consistent with: http://msdn.microsoft.com/en-us/library/cc236715.aspx http://technet.microsoft.com/en-us/library/dd560653%28WS.10%29.aspx Note that you will run into issues if you have an environment that uses antique versions of Windows (Vista, 2008, XP, etc.), but those systems should be getting upgraded or taken off the wire anyhow. If you can't break use of NTLM entirely, at least break NTLMv1, see http://support.microsoft.com/kb/2793313 [Oh! I see that you talk about this in 5.3.2, as well... but you imply that NTLMv2 is "reasonably secure" -- it isn't] -- "5.2.1 Transmission of Authentication Secrets Between Credential Stores" In the bulleted item, the text current reads "select one of the AES options" There are only two options: AES128_HMAC_SHA1 and AES256_HMAC_SHA1 Of the two, AES256_HMAC_SHA1 would be preferable, but it still uses SHA1 which is deprecated/will be deprecated as the document itself notes at 2.3 in bold text. This section also requires use of LDAPS (TLS/SSL), but more specificity is needed when it comes to explaining what constitutes an acceptable version of TLS (e.g., is TLS 1.0 good enough? It shouldn't be treated as such). Require TLS 1.2 with an appropriate cipher suite (that should be a whole section of its own) The Microsoft references in document section 5.3.1 ("Section 4.2.3.6.2 requirements") really don't clear this up, either. -- 5.3.4 How would a "temporarily compromised" account be rehabilitated? If an account is every "temporarily compromised," it would need to have a thorough security audit before being re-enabled, but my worry is that in some cases folks may just require a password change, and that obviously wouldn't be enough to ensure that a "temporarily compromised" account has been restored to a trustworthy state. Trivial example: assume that while "temporarily compromised" a backdoor was installed, or access controls were weakened, allowing persistent access and abuse, even if the password's changed. Also, this doesn't treat the possibility of a privileged account being "temporarily compromised", in which case the entire system (or even multiple systems, in the case of transitive trust relationships) may need to be audited and remediated. -- 6. "Alternate Controls and Alternative Means Statements"	Joe St Sauver	Oct 2, 2013	6. FIXED

	When I try to access the link in this part, I get an access failure. -- 7.1 Repeats the unsatisfactory use of a full disk encryption tool approach. Still not okay. -- 7.3 is the bold text "need to validate algorithm to see if this is good enough" an author's note that was meant to be resolved prior to publication? I also have a concern about the 72 hour window mentioned in the last paragraph of that section. 72 hours is an eternity for an attacker, and might as well be six months if you're going to make it 72 hours. As suspected, too, I note that the "temporarily compromised" account is just required to have credentials reset. That's not enough, as previously discussed. -- 7.4 Practical attacks against NTLMv2 exist. **Example.** Repeats the unacceptable "temporarily compromised" language. (yes, Zack is in the running for one of the top 10 most annoying presenters of all time, but still) -- 7.6 If a persistent password is used, how does it preclude a replay attack? The persistent password is the same thing this time, and next time, and the time after that, etc. A replay-resistent credential would be something like a one-time crypto fob -- you can't replay that credential because it's different every time you use it... -- Appendix A Recommend removal/decommissioning of all Windows XP systems. -- Appendix B Has the Cisco issue been filed with Cisco Security Intelligence Operations? If not, a case should be opened. See http://tools.cisco.com/security/center/home.x			