Notes on RFC 4120 superseding RFC 1510

dnickull@adobe.com

High Level notes:

RFC 4120's primary reason for existing is to provide additional clarity where ambiguity existed in RFC 1510. It should be noted that both RFC's give an overview and specification of Version 5 of the protocol for the Kerberos network authentication system.

Implementers cannot automatically assume forward compatibility. Section 1.5.1 of RFC 4210 outlines the basis for backwards compatibility. Note that existing Kerberos message formats cannot readily be extended by adding fields to the ASN.1 types. Sending additional fields often results in the entire message being discarded without an error indication. Future versions of this specification will provide guidelines to ensure that ASN.1 fields can be added without creating an interoperability problem. There are no timetables established for this deliverable by the authors.

Accordingly, this TC *may* have to expand the kerebos token profile to be able to distinguish between tokens compliant with 4120 vs 1510. This would probably be best done in section 3.3. The @valueType is probably a logical place to assert the nature of the token if the TC feels it is warranted. Since 4120 replaces and supersedes 1510, it is probably logical to assume all new implementations will only support 4120. Question for TC to decide is if we want to explicitly support both?

In section 3.3 of our specification, we state clearly that Name MUST NOT be used. This is perhaps ambiguous and possibly in conflict with RFC 4120 since section 3.3.3.2 specifies that the name of the realm WILL be used. It was unclear to me if this is the same NAME mentioned in 3.3.

Security Considerations:

The authors of 4120 noted "Many RFC 1510 implementations ignore unknown authorization data elements. Depending on these implementations to honor authorization data restrictions may create a security weakness."

Functional considerations:

With the exception of the INVALID flag, clients MUST ignore ticket flags that are not recognized. KDCs MUST ignore KDC options that are not recognized. Some implementations of RFC 1510 are known to reject unknown KDC options, so clients may need to resend a request without new KDC options if the request was rejected when sent with options added since RFC 1510. Because new KDCs will ignore unknown options, clients MUST confirm that the ticket returned by the KDC meets their needs.

AP-REQ

There are some new additions to the AP-REQ message in RFC 4210 since RFC 1510:

1. DISABLE-TRANSITED-CHECK By default the KDC will check the transited field of a TGT against the policy of the local realm before it will issue derivative tickets based on the TGT. If this flag is set in the request, checking of the transited field is disabled. Tickets issued without the performance of this check will be noted by the reset (0) value of the TRANSITED-POLICY-CHECKED flag, indicating to the application server that the transited field must be checked locally. KDCs are encouraged but not required to honor the DISABLE-TRANSITED-CHECK option.

Section 3.2.1 seems to have some additional clarify about how to prevent replay type attacks without having to rely on other mechanisms like NONCE. I didn't see any equivalent to this in 1510.

Changes to Kerebos Token Profile document (non exhaustive)

Line 227 – reference to RFC 1510 to be changed to RFC 4120

Line 101 – I am not sure if we need to allow for multiple attributes (@valueType) to represent Kerebos tokens compliant with the RFC 1510 vs. the RFC 4120.