[MS-NLMP]: NT LAN Manager (NTLM) Authentication Protocol

This topic lists the Errata found in [MS-NLMP] since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.

Errata are subject to the same terms as the Open Specifications documentation referenced.

RSS

Atom

Errata below are for Protocol Document Version V27.0 – 2015/10/16.

Errata Published*	Description

2016/06/27	In various sections, indicated that ClientChallenge is defined in section 3.3.2 and updated code snippets with ChallengeFromClient.
	In Section 2.2.2.4, LMv2_RESPONSE, changed from:
	ChallengeFromClient (8 bytes): An 8-byte array of unsigned char that contains the client's ClientChallenge, as defined in section 3.1.5.1.2.
	Changed to:
	ChallengeFromClient (8 bytes): An 8-byte array of unsigned char that contains the client's ClientChallenge (as defined in section 3.3.2). See section 3.1.5.1.2 for details.
	In Section 2.2.2.7, LMv2_RESPONSE, changed from:
	ChallengeFromClient (8 bytes): An 8-byte array of unsigned char that contains the client's ClientChallenge (section 3.1.5.1.2).
	Changed to:
	ChallengeFromClient (8 bytes): An 8-byte array of unsigned char that contains the client's ClientChallenge (as defined in section 3.3.2). See section 3.1.5.1.2 for details.
	In Section 3.1.5.1.2, Client Receives a CHALLENGE_MESSAGE from the Server, changed from:
	`ComputeResponse(CHALLENGE_MESSAGE.NegotiateFlags, ResponseKeyNT,`
	`    ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge,`
	`    AUTHENTICATE_MESSAGE.ClientChallenge, Time,`
	`    CHALLENGE_MESSAGE.TargetInfo)`
	Changed to:
	`ComputeResponse(CHALLENGE_MESSAGE.NegotiateFlags, ResponseKeyNT,`
	`    ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge,`
	`    ChallengeFromClient, Time,`
	`    CHALLENGE_MESSAGE.TargetInfo)`
	In Section 3.2.5.1.2, Server Receives an AUTHENTICATE_MESSAGE from the Client, changed from:

Errata Published*	Description

```
 --    Time - Temporary variable used to hold the 64-bit current
time in
       the AUTHENTICATE_MESSAGE.ClientChallenge, in the format of a
       FILETIME as defined in [MS-DTYP] section 2.3.1.
```

Changed to:

```
 --    Time - Temporary variable used to hold the 64-bit current
time from the
       NTLMv2_CLIENT_CHALLENGE.Timestamp, in the format of a
       FILETIME as defined in [MS-DTYP] section 2.3.1.
 --    ChallengeFromClient – Temporary variable to hold the client's
8-byte
       challenge, if used.
```

Changed from:

```
     Set ExpectedNtChallengeResponse, ExpectedLmChallengeResponse,
      SessionBaseKey to ComputeResponse(NegFlg, ResponseKeyNT,
      ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge,
      AUTHENTICATE_MESSAGE.ClientChallenge, Time, ServerName)
     Set KeyExchangeKey to KXKEY(SessionBaseKey,
      AUTHENTICATE_MESSAGE.LmChallengeResponse,
CHALLENGE_MESSAGE.ServerChallenge)
     If (AUTHENTICATE_MESSAGE.NtChallengeResponse !=
      ExpectedNtChallengeResponse)
        If (AUTHENTICATE_MESSAGE.LmChallengeResponse !=
          ExpectedLmChallengeResponse)
           Retry using NIL for the domain name: Retrieve the ResponseKeyNT
            and ResponseKeyLM from the local user account database using
            the UserName specified in the AUTHENTICATE_MESSAGE and
            NIL for the DomainName.
           Set ExpectedNtChallengeResponse, ExpectedLmChallengeResponse,
            SessionBaseKey to ComputeResponse(NegFlg, ResponseKeyNT,
            ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge,
            AUTHENTICATE_MESSAGE.ClientChallenge, Time, ServerName)
```

Changed to:

```
     If
AUTHENTICATE_MESSAGE.NtChallengeResponseFields.NtChallengeResponseLen >
0x0018
       Set ChallengeFromClient to
NTLMv2_RESPONSE.NTLMv2_CLIENT_CHALLENGE.ChallengeFromClient
     ElseIf NTLMSSP_NEGOTIATE_EXTENDED_SESSIONSECURITY is set in NegFlg
       Set ChallengeFromClient to LM_RESPONSE.Response[0..7]
     Else
       Set ChallengeFromClient to NIL
     EndIf
     Set ExpectedNtChallengeResponse, ExpectedLmChallengeResponse,
      SessionBaseKey to ComputeResponse(NegFlg, ResponseKeyNT,
      ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge,
      ChallengeFromClient, Time, ServerName)
     Set KeyExchangeKey to KXKEY(SessionBaseKey,
      AUTHENTICATE_MESSAGE.LmChallengeResponse,
CHALLENGE_MESSAGE.ServerChallenge)
     If (AUTHENTICATE_MESSAGE.NtChallengeResponse !=
      ExpectedNtChallengeResponse)
        If (AUTHENTICATE_MESSAGE.LmChallengeResponse !=
          ExpectedLmChallengeResponse)
```

Errata Published*	Description
	Retry using NIL for the domain name: Retrieve the ResponseKeyNT and ResponseKeyLM from the local user account database using the UserName specified in the AUTHENTICATE_MESSAGE and NIL for the DomainName. Set ExpectedNtChallengeResponse, ExpectedLmChallengeResponse, SessionBaseKey to ComputeResponse(NegFlg, ResponseKeyNT, ResponseKeyLM, CHALLENGE_MESSAGE.ServerChallenge, ChallengeFromClient, Time, ServerName)
2016/06/27	In Section 2.2.1.2, CHALLENGE_MESSAGE, added a product behavior note to describe product specific behavior for the TargetInfoFields field. Changed from: TargetInfoFields (8 bytes): A field containing TargetInfo information. The field diagram for TargetInfoFields is as follows. … If the NTLMSSP_NEGOTIATE_TARGET_INFO flag is not clear in NegotiateFlags, indicating that TargetInfo is required, the fields are set to the following values: … Changed to: TargetInfoFields (8 bytes): A field containing TargetInfo information. The field diagram for TargetInfoFields is as follows. … If the NTLMSSP_NEGOTIATE_TARGET_INFO flag is not clear in NegotiateFlags, indicating that TargetInfo is required, the fields are set to the following values:<7> <7> Section 2.2.1.2: In Windows Vista and subsequent versions of Windows according to the applicability list at the beginning of this section, the TargetInfo field is always sent. …
2016/01/25	In three sections, updated references and links. In Section 3.4.2, Message Integrity, changed from: -- MAC() - Defined in section 3.4.3. Changed to: -- MAC() - Defined in sections 3.4.4.1 and 3.4.4.2. In Section 3.4.6.1, Signature Creation for GSS_WrapEx(), changed from: Section 3.4.3 describes the algorithm used by GSS_WrapEx() to create the signature. Changed to: Section 3.4.2 describes the algorithm used by GSS_WrapEx() to create the signature. In Section 4.2.3.4, GSS_WrapEx Examples, changed from:

Errata Published*	Description
	The output message data and signature is created using SEAL() specified in section 3.4.4. Changed to: The output message data and signature is created using SEAL() specified in section 3.4.3.
2015/11/09	In various sections, corrected text that implies MaxLifetime applies to versions later than Windows XP. In Section 3.1.1.1, Variables Internal to the Protocol, changed from: MaxLifetime: An integer that indicates the maximum lifetime for challenge/response pairs <35> <35> In Windows NT 4.0 and Windows 2000, the maximum lifetime for the challenge is 30 minutes. In Windows XP and subsequent versions of Windows, according to the applicability list at the beginning of this section, the maximum lifetime is 36 hours. Changed to: MaxLifetime: An integer that indicates the maximum lifetime for challenge/response pairs <35> <35> In Windows NT 4.0 and Windows 2000, the maximum lifetime for the challenge is 30 minutes. In Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2, the maximum lifetime is 36 hours. In Section 3.2.5.1.2, Server Receives an AUTHENTICATE_MESSAGE from the Client, changed from: If NTLM v2 authentication is used and the AUTHENTICATE_MESSAGE.NtChallengeResponse.TimeStamp (section 2.2.2.7) is more than MaxLifetime (section 3.1.1.1) difference from the server time, then the server SHOULD return a failure.<64> <64> Supported by Windows NT, Windows 2000, and Windows XP. Changed to: If NTLM v2 authentication is used and the AUTHENTICATE_MESSAGE.NtChallengeResponse.TimeStamp (section 2.2.2.7) is more than MaxLifetime (section 3.1.1.1) difference from the server time, then the server SHOULD return a failure.<64> <64> Supported by Windows NT 4.0, Windows 2000, Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

Errata Published*	Description
	In Section 3.2.5.2.2, Server Response Checking, changed from: If NTLM v2 authentication is used and the AUTHENTICATE_MESSAGE.NtChallengeResponse.TimeStamp (section 2.2.2.7) is more than MaxLifetime (section 3.1.1.1) difference from the server time, then the server SHOULD return a failure.<69> <69> Supported by Windows NT, Windows 2000 and Windows XP. Changed to: If NTLM v2 authentication is used and the AUTHENTICATE_MESSAGE.NtChallengeResponse.TimeStamp (section 2.2.2.7) is more than MaxLifetime (section 3.1.1.1) difference from the server time, then the server SHOULD return a failure.<69> <69> Supported by Windows NT 4.0, windows_2000, Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

* Date format: YYYY/MM/DD