SonicWall® Secure Mobile Access 12.1 Client Extensibility Toolkit

Reference Guide

SONICWALL®

Contents

1

Using This Guide

Welcome to the *SonicWall® Secure Mobile Access 12.1 Connect TunnelClient Extensibility Toolkit Reference Guide.* This guide describes how to use the Connect Tunnel client with the Windows Remote Access Service (RAS) to allow client applications to access network resources secured by a SonicWall SMA appliance.

The content of this guide is applicable to the following firmware versions:

- SonicWall SMA 11.x and higher
- SonicWall E-Class SRA 10.7.x

Organization of This Guide

The *SonicWall® Secure Mobile Access 12.1 Connect TunnelClient Extensibility Toolkit Reference Guide* is structured into these parts:

Using This Guide	This chapter gives a summary of the contents of this manual and direction to basic technical support.
Using Connect Tunnel with Windows RAS API	This chapter provides information on using the Windows Remote Access Service (RAS) including a section about using the NGDIAL command line utility. A section giving error messages is also included.
SonicWall Support	This chapter provides SonicWall Support and Contact information.

2

Using Connect Tunnel with Windows RAS API

The SonicWall Secure Mobile Access Connect Tunnel client supports command-line utilities that can modify the normal run-time behavior of the client and enable you to perform troubleshooting and diagnostic tasks without using the standard graphical user interface. Controlling the functionality of the Connect Tunnel client from a programming or scripting language enables you to integrate it with your applications for a more complete and robust overall solution.

For example, suppose you want to use SonicWall Secure Mobile Access Connect Tunnel to give users secure access to a Windows client/server application hosted on your network. Rather than have the user interact with the Connect Tunnel client directly, you can use the Client Extensibility Toolkit to integrate the tunnel access capabilities in your client application.

Using the Windows Remote Access Service

You can use the Windows Remote Access Service (RAS) to develop client applications that access network resources secured by an SRA EX-Series or Secure Mobile Access appliance through the SonicWall Secure Mobile Access Connect Tunnelclient. The RAS interface to the client enables the application developer to control how the Connect Tunnel client connects to the remote VPN and what, if any, prompting is performed. This section describes the key RAS functions that application developers can use to access the Connect Tunnelclient software.

For more information about the Windows RAS interface, see *Windows Platform SDK: Remote Access Service* on the Microsoft MSDN site:

- https://msdn.microsoft.com/en-us/library/windows/desktop/bb545687%28v=vs.85%29.aspx
- https://msdn.microsoft.com/en-us/library/windows/desktop/aa446739%28v=vs.85%29.aspx

Topics:

NGCUSTOMAUTHDATA Structure

The NGCUSTOMAUTHDATA structure is a SonicWall Secure Mobile Access-defined structure for specifying authentication information specific to a Connect Tunnel client phone book entry.

```
include: ngras.h
#define NG_MAX_REALM_NAME   255
#define NG_MAX_USERNAME_LEN 255
#define NG_MAX_PASSWORD_LEN 255
#define NG_MAX_CONNECTION_NAME 255

typedef struct tagNGCUSTOMAUTHDATA
{
   DWORD dwSize;
   DWORD afMask;

   DWORD dwOptions;

   DWORD dwAuthType;
   WCHAR szRealm[ NG_MAX_REALM_NAME + 1 ];

   WCHAR szConnection[ NG_MAX_CONNECTION_NAME + 1 ];

   WCHAR szProxyUsername[ NG_MAX_USERNAME_LEN + 1 ];
   WCHAR szProxyPassword[ NG_MAX_PASSWORD_LEN + 1 ];

   DWORD nReserved;
   DWORD anReserved[10];

   DWORD nServerCertHashType;
   WCHAR szServerCertHash[41];
   WCHAR szCredentials[ NG_MAX_CREDENTIALS_LEN + 1 ];
   WCHAR szCredArray[NG_MAX_CREDENTIALS][NG_MAX_USERNAME_LEN];

} NGCUSTOMAUTHDATA;
```

dwSize

Specifies the size of this structure, in bytes. Set this member to `sizeof(NGCUSTOMAUTHDATA)`. This indicates the version of the structure.

afMask

Indicates which members of the NGCUSTOMAUTHDATA structure are valid. A member can be any combination of the following values:

Value	Description
NGCAD_MASK_OPTIONS	The `dwOptions` field is valid.
NGCAD_MASK_AUTH_TYPE	The `dwAuthType` field is valid.
NGCAD_MASK_REALM	The `szRealm` field is valid.
NGCAD_MASK_PROXY_CREDENTIALS	The `szProxyUsername` and `szProxyPassword` fields are valid.

dwOptions

A set of bit flags that define function options. The following bit flags are defined; set all undefined bits to zero.

Flag	Description
NGCAD_OPTION_ShowConnectingStatus	Displays a connecting-status dialog box to the user.
NGCAD_OPTION_ShowTaskbarStatusIcon	Displays an icon in the taskbar notification area that shows connection-status information.
NGCAD_OPTION_DisableGUI	Causes the function call to fail if more information is needed to complete the request. For example, a RasDial() function call will fail with an NGRAS_E_GUI_DISABLED error code if the connection request results in the need to display a suspect server certificate, or if the authentication method requires additional information. If this option is not set, a dialog box prompts the user to specify any additional information that is needed to complete the request.

dwAuthType

Specifies the type of authentication credential that will be used when **RasDial** is invoked.

The authentication type specified must match the one defined on the appliance for the specified realm (or login group), or be a valid TEAM credential for the realm. The following authentication types are defined:

Authentication type	Value	Credential type
AUTH_T_UNKNOWN	0	Uses the default authentication type defined for the realm.
AUTH_T_NULL	1	No authentication is required.
AUTH_T_LDAP_PWD	2	LDAP username/password credential.
AUTH_T_LDAP_CERT	3	LDAP certificate credential.
AUTH_T_RADIUS_CRAM	4	RADIUS CRAM credential.
AUTH_T_RADIUS_PWD	5	RADIUS username/password credential.
AUTH_T_UNIX	6	UNIX username/password credential.
AUTH_T_TEAM	7	TEAM credential.
AUTH_T_ACTIVE_DIRECTORY	8	Active Directory username/password credential.

szRealm

Specifies the name of the realm (or login group) to authenticate to. This field is required.

szConnection

Specifies the “friendly name” of the connection from Connection list entries

szProxyUsername

Specifies the username to be used if proxy server credentials are required.

szProxyPassword

Specifies the password to be used if proxy server credentials are required.

nReserved

Must be set to zero.

anReserved

Must be set to zero.

nServerCertHashType

Specifies the szServerCetHash type.

szServerCertHash

Specifies the hash of an acceptable server certificate.

szCredentials

Specifies the credential details.

szCredArray

Specifies the credentials for stacked authorizations.

RasConnectionNotification Function

The **RasConnectionNotification** function specifies an event object that the system sets to the signaled state when a RAS connection is created or terminated.

For more information about the Windows **RasConnectionNotification** API function, see *Windows Platform SDK: Remote Access Service: RasConnectionNotification* on the Microsoft MSDN site.

RasDial Function

The **RasDial** function establishes a RAS connection between a RAS client and a RAS server. The connection data includes callback and user authentication information.

For more information about the Windows **RasDial** API function, see *Windows Platform SDK: Remote Access Service: RasDial* on the Microsoft MSDN site.

Before invoking the **RasDial** function, authentication data can be specified for the phone book entry being dialed. Authentication data can be used to specify the realm, or login group, for which the user's credentials are valid. More information about setting authentication data can be found in RasSetCustomAuthData Function and NGCUSTOMAUTHDATA Structure.

C++ Example

```
#include <windows.h>
#include <ras.h>

#define RASPHONE_PBK
L"C:\\ProgramData\\Microsoft\\Network\\Connections\\Pbk\\rasphone.pbk"
#define SONICWALL_VPNL"SonicWall VPN Connection" if CT > 12.0

#define NG_MAX_REALM_NAME255
#define NG_MAX_USERNAME_LEN255
#define NG_MAX_PASSWORD_LEN255
#define NG_MAX_CREDENTIALS_LEN512
#define NG_MAX_CONNECTION_NAME255
#define NG_MAX_CREDENTIALS10 //max credentials including usernames and passwords

// NGCUSTOMAUTHDATA field masks
#define NGCAD_MASK_OPTIONS0x00000001  // dwOptions is valid
#define NGCAD_MASK_AUTH_TYPE0x00000002  // dwAuthType is valid
#define NGCAD_MASK_REALM0x00000004  // szRealm is valid
#define NGCAD_MASK_PROXY_CREDENTIALS0x00000008  // szProxyUsername & szProxyPassword
fields are valid
#define NGCAD_MASK_CREDENTIALS0x00000010  // szCredentials is valid
#define NGCAD_MASK_SERVER_CERT_HASH0x00000020  // szServerCertHash &
nServerCertHashType fields are valid
#define NGCAD_MASK_CONNECTION_ENTRY0x00000030  // Connection field is valid.

// NGCUSTOMAUTHDATA dwOptions flags
#define NGCAD_OPTION_ShowConnectingStatus0x00000001
#define NGCAD_OPTION_ShowTaskbarStatusIcon0x00000002
#define NGCAD_OPTION_DisableGUI0x00000008//With this option EULA and CT upgrade
won't work
#define NGCAD_OPTION_ExclusiveHangup0x00000010
#define NGCAD_OPTION_HideConnectingStatus0x00000020
#define NGCAD_OPTION_HideTaskbarStatusIcon0x00000040
#define NGCAD_OPTION_IgnoreCertErrors0x00000080
#define NGCAD_OPTION_UseSystemCertStore0x00000100

typedef struct tagNGCUSTOMAUTHDATA
{
   DWORD dwSize;              // size of this structure in bytes
   DWORD afMask;              // NGCAD_MASK_* value indicates which members are valid

   DWORD dwOptions;        // NGCAD_OPTION_* defined above

   DWORD dwAuthType;       // Realm authentication type
   WCHAR szRealm[ NG_MAX_REALM_NAME + 1 ];// realm/login group name

   WCHAR szConnection[ NG_MAX_CONNECTION_NAME + 1 ];// Friendly name of connection
from Connection list entries

   WCHAR szProxyUsername[ NG_MAX_USERNAME_LEN + 1 ];
   WCHAR szProxyPassword[ NG_MAX_PASSWORD_LEN + 1 ];

   DWORD nReserved;        // Must be set to zero
   DWORD anReserved[10]; // Must be set to zero; some room for future options

   DWORD nServerCertHashType;// szServerCetHash type
   WCHAR szServerCertHash[41];// Hash of an accptable server certificate
```

```
   WCHAR szCredentials[ NG_MAX_CREDENTIALS_LEN + 1 ];  //
["]<public>["],[["]<private>["]][;["]<public>["],[["]<private>["]]]...   (example:
"fred figment",test;root,"secret word";test,password )
   WCHAR szCredArray[NG_MAX_CREDENTIALS][NG_MAX_USERNAME_LEN]; //credentials for
stacked auth

} NGCUSTOMAUTHDATA;

struct NG_RASDIALPARAMS
{
   DWORD dwSize;
   WCHAR szEntryName[RAS_MaxEntryName + 1];
   WCHAR szPhoneNumber[RAS_MaxPhoneNumber + 1];
   WCHAR szCallbackNumber[RAS_MaxCallbackNumber + 1];
   WCHAR szUserName[UNLEN + 1];
   WCHAR szPassword[PWLEN + 1];
   WCHAR szDomain[DNLEN + 1];
   DWORD dwSubEntry;
   ULONG_PTR dwCallbackId;
   DWORD dwIfIndex;
};

int main(int argc, char* argv[])
{
   HRASCONN hRasConnection = NULL;

   NGCUSTOMAUTHDATA NgAuthData;
   ZeroMemory( & NgAuthData, sizeof(NgAuthData) );
   NgAuthData.dwSize = sizeof(NGCUSTOMAUTHDATA);
   NgAuthData.dwOptions = NGCAD_OPTION_ShowConnectingStatus |
                          NGCAD_OPTION_IgnoreCertErrors |
                          NGCAD_OPTION_DisableGUI;
   NgAuthData.afMask = NGCAD_MASK_REALM |
                       NGCAD_MASK_OPTIONS |
                       NGCAD_MASK_CONNECTION_ENTRY;
   wcscpy_s(NgAuthData.szRealm, L"odt");//Realm Name

   WCHAR sDirPhoneBooks[MAX_PATH] = L"";
   wcscpy_s(sDirPhoneBooks, MAX_PATH, RASPHONE_PBK);

   DWORD dwRetValue = RasSetCustomAuthData(sDirPhoneBooks, SONICWALL_VPN,
                     (BYTE *)&NgAuthData, NgAuthData.dwSize);

   if ( ERROR_SUCCESS == dwRetValue )
   {
      NG_RASDIALPARAMS RasDialParams;
      ZeroMemory( &RasDialParams, sizeof(RasDialParams) );

      RasDialParams.dwSize = sizeof(RasDialParams);
      wcscpy_s( RasDialParams.szEntryName, SONICWALL_VPN);
      wcscpy_s( RasDialParams.szPhoneNumber, L"10.194.22.150");//Appliance
      lstrcpy(RasDialParams.szUserName,L"user1");//user name
      lstrcpy(RasDialParams.szPassword,L"pass1");//password

      dwRetValue = RasDial(NULL, sDirPhoneBooks, (RASDIALPARAMS *)&RasDialParams,
                  0xFFFFFFFF, NULL, &hRasConnection);

      if(ERROR_SUCCESS == dwRetValue && hRasConnection )
      {
         MessageBox(NULL, L"Hang up ?", SONICWALL_VPN, MB_OK);
```

```
            RasHangUp( hRasConnection );
        }
        else
        {
            MessageBox(NULL, L"Failed at RasDial", SONICWALL_VPN, MB_OK);
        }
    }
    else
    {
        MessageBox(NULL, L"Failed to Initialize Auth Data", SONICWALL_VPN, MB_OK);
    }

    return dwRetValue;
}
```

RasDialDlg Function

The `RasDialDlg` function establishes a RAS connection using a specified phone book entry. The function displays a sequence of dialog boxes that indicates the state of the connection operation.

For more information about the Windows `RasDialDlg` API function, see *Windows Platform SDK: Remote Access Service: RasDialDlg* on the Microsoft MSDN site.

RasGetConnectStatus Function

The **RasGetConnectStatus** function retrieves information about the current status of the specified remote access connection. An application can use this call to determine when an asynchronous **RasDial** call is complete.

For more information about the Windows **RasGetConnectStatus** API function, see *Windows Platform SDK: Remote Access Service: RasGetConnectStatus* on the Microsoft MSDN site.

RasHangUp Function

The **RasHangUp** function terminates a remote access connection. For more information about the Windows **RasHangUp** API function, see *Windows Platform SDK: Remote Access Service: RasHangUp* on the Microsoft MSDN site.

RasSetCustomAuthData Function

You can use the **RasSetCustomAuthData** function to set connection-specific authentication information before calling the **RasDial** function.

The Connect Tunnel client uses this function to define extended authentication information in the form of an `NGCUSTOMAUTHDATA` structure for the phone book entry you are connecting to.

For more information about the Windows **RasSetCustomAuthData** API function, see *Windows Platform SDK: Remote Access Service: RasSetCustomAuthData* on the Microsoft MSDN site.

NGDIAL Command-Line Utility

- The NGDIAL command-line utility establishes a connection to a remote network using Connect Tunnel, much like the Windows RASDIAL utility does with other network connections.
- The NGDIAL command-line utility can also create, delete, and modify network connection phone book entries. Issuing the NGDIAL command without any parameters will list all RAS connections.

Command Syntax

```
ngdial <connection name> <public> [<private>|* [<auth type>]]
         [-phonebook=<phonebook>]
         [-server=<server name>|<server IP>]
         [-login=<login group>]
         [-proxycredential=<username>[,<password>|*]]
         [-status[=enable|disable]] [-icon[=enable|disable]] [-gui]

ngdial <connection name> -disconnect|-d

ngdial <connection name> -prompt
         [-phonebook=<phonebook>]

ngdial <connection name> -create
         [-phonebook=<phonebook>]
         [-server=<server name>|<server IP>]
         [-login=<login group>]
         [-status[=enable|disable]] [-icon[=enable|disable]]

ngdial -help | -?
```

Option	Description
<connection name>	The name of the network connection; if the name includes a space, enclose it in quotes.
<public>	The user's public credential (username) for authentication; if the name includes a space, enclose it in quotes. For example: `ngdial report_server "Jen Bates"` The public and <*private*> portions of the credentials must correspond correctly with the authentication type specified by the authentication realm on the appliance.

Option	Description
[<private>|* [<auth type>]]	The private credentials (password) and authentication type to be used when authenticating the user. If the <*private*> portion of the credential is omitted or an asterisk (*) is specified, the NGDIAL command prompts the user to enter the password. If you do not specify an <*auth type*> when logging in to an appliance, the default authentication type for the realm is used. Values for <*auth type*> are: NULL: No authentication required LDAPUNPW: LDAP username/password credential LDAPCERTIFICATE: LDAP certificate credential RADIUSCRAM: RADIUS token/securID credential RADIUSUNPW: RADIUS username/password credential UNIX: UNIX username/password credential TEAM: TEAM credential ADUNPW: Active Directory username/password credential
-create	Generates a new network connection, or updates an existing network connection, with the information passed on the command line.
-delete	Deletes the specified network connection entry from the specified phone book. You must have system administrator privileges to perform this operation.
-disconnect | -d	Causes the VPN to disconnect from the <*connection name*> remote network.
-gui	If additional information is necessary to establish the VPN network connection, use this parameter to allow RAS to prompt the user with a graphical user interface (GUI). For example, the user could be prompted to accept the appliance's server certificate if there are any problems with the certificate, or the user might need to be notified regarding password expiration or required changes. If the -gui option is not specified in such a case, the NGDIAL utility fails and returns an error code to the caller.
-help | -?	Displays the command-line syntax for the NGDIAL command. When combined with the -gui option, displays the online Help.
[-icon[=enable|disable]]	Controls the display of an icon in the taskbar notification area that allows the user to manage the VPN network connection and receive connection notifications.
-login=<login group>	The name of the login group (authentication realm) used to authenticate the user. If a login is specified without specifying an <auth type> for the credentials, NGDIAL uses an <*auth type*> of ADUNPW.
-phonebook=<phonebook>	Specifies the file name of the phone book where the <*connection name*> is defined. The file name must include the fully qualified path to the phone book file. If a path is not specified, NGDIAL looks in the directory that contains the system phone book (*rasphone.pbk*) for the specified phone book file.
-prompt	Causes the NGDIAL command to prompt the user to connect to the <*connection name*> remote network.

Option	Description
`-proxycredential= <username> [,<password>|*]`	If a proxy server is required for access to the appliance, use this option to specify the username and password credentials for it. If the password is omitted, or entered as an asterisk (`*`), the `NGDIAL` command prompts the user for a proxy password.
`-server=<server name>| <server IP>`	Specifies the appliance name or IP address. If a server is specified, and it is different from the server defined in the phone book entry, the server and login group (if specified) are saved to the phone book entry.
`[-status[=enable|disable]]`	Controls the display of a connection status dialog box when the VPN network connection takes more than two seconds to connect.

Examples

```
NGDIAL "ACME Corp" -create -server=remote.acme.com -icon -status
NGDIAL "ACME Corp" "Jen Bates" * -login="Business Partners" -icon -gui
NGDIAL "ACME Corp" jdoe password
NGDIAL "ACME Corp" -disconnect
```

Notes

Although the `ngdial -help` usage statement indicates that the `-icon=disable` flag is an option without the `-create` flag, in some cases the `-create` flag is necessary to disable the icon.

To disable the icon so that it does not appear on the taskbar, you can use either of the following two methods:

- Set `taskbar=0` in the `ngsetup.ini` file and then type a command such as:

  ```
  ngdial "SonicWall SMA VPN Connection" -server=<server IP address>
  -login="Realm name" username password -icon=disable -gui
  ```

- Type a command using the `-create` option with the `-icon=disable` option to store the icon parameter, and then type the command to connect, such as:

  ```
  ngdial "SonicWall SMA VPN Connection" -create -server=<server IP
  address> -icon=disable -gui
  ```

  ```
  ngdial "SonicWall SMA VPN Connection" -server=<server IP address>
  -login="Realm name" username password -icon=disable -gui
  ```

Connect Tunnel Error Messages

The following error messages may be returned by the Connect Tunnel client, in addition to RAS and WIN32 return codes. The error messages are applicable to SonicWall SMA 11.x and higher and SonicWall E-Class SRA 10.7.x.

Topics:

VPN

Error	Number	Description
NGTUNNEL_E_AUTH_TYPE_MISMATCH	0xE3050009L	The requested authentication method is not supported by the selected login group.
NGTUNNEL_E_INVALID_AUTH_REQ	0xE305000AL	An invalid authentication request was received from the server.
NGTUNNEL_E_INVALID_TEAM	0xE305000BL	An invalid TEAM ID was received from the server.
NGTUNNEL_E_SERVER_AUTH_FAILED	0xE305000CL	The user's authentication credentials have been rejected by the server.
NGTUNNEL_E_SERVER_AUTH_ERROR	0xE305000DL	The server encountered an error communicating with the authentication system.
NGTUNNEL_E_SERVER_PROTOCOL_VERSION_MISMATCH	0xE305000EL	The client protocol version is not supported by the server.
NGTUNNEL_E_SERVER_INVALID_MESSAGE	0xE305000FL	An invalid or unsupported protocol message was sent to the server.
NGTUNNEL_E_SERVER_TIMEDOUT	0xE3050010L	A server operation has exceeded its timeout value.
NGTUNNEL_E_SERVER_CLIENT_VIP	0xE3050011L	The server was unable to allocate a client IP address.
NGTUNNEL_E_SERVER_AUTHORIZATION_FAILED	0xE3050012L	The user is not authorized to access the requested resource.
NGTUNNEL_E_SERVER_AUTHORIZATION_ERROR	0xE3050013L	The server encountered an error communicating with the authorization system.
NGTUNNEL_E_SERVER_SYSTEM	0xE3050014L	General server system failure.
NGTUNNEL_E_SERVER_CLIENT_VERSION_MISMATCH	0xE3050015L	The client version is not supported by the server.
NGTUNNEL_E_SERVER_ADDRESS_CONFLICT	0xE3050016L	The server was unable to assign a client IP address due to a fatal IP address conflict.
NGTUNNEL_E_ICF_NO_PHONEBOOKENTRY	0xE3010017L	Unable to determine if the Microsoft Internet Connection Firewall is running because the phone book entry for this connection could not be determined.
NGTUNNEL_E_ICF_DISABLE	0xE3010018L	Unable to disable the Microsoft Internet Connection Firewall for the %1 connection. The error is %2!u!.
NGTUNNEL_E_REALM_NOT_FOUND	0xE3050019L	The requested login group does not exist.
NGTUNNEL_E_COULD_NOT_CREATE_USER_CACHE_ENTRY	0xE305001AL	The user cache entry could not be created.
NGTUNNEL_E_SERVER_DO_RESUME	0xE305001BL	The server has requested that the client resume the tunnel.

Tunnel

Error	Number	Description
TUNNEL_E_PROTOCOL_HEADER_SIZE	0xE3050001L	Error parsing protocol header: buffer too small.
TUNNEL_E_PROTOCOL_INVALID_VERSION	0xE3050002L	Error parsing protocol header: version mismatch.
TUNNEL_E_PROTOCOL_INVALID_TYPE	0xE3050003L	Error parsing protocol header: invalid message type.
TUNNEL_E_PROTOCOL_MESSAGE_SIZE	0xE3050004L	Error parsing protocol header: message size exceeds buffer size.
TUNNEL_E_PROTOCOL_REASSEMBLY_SIZE	0xE3050005L	Error reassembling extended message: data received exceeds message size.
TUNNEL_E_PROTOCOL_REASSEMBLY_TYPE	0xE3050006L	Error reassembling extended message: invalid message type.
TUNNEL_E_PROTOCOL_EXTENDED_MSG_NOT_FOUND	0xE3050007L	Error reassembling extended message: can't find message ID.
TUNNEL_E_PROTOCOL_INVALID_STATE	0xE3050008L	Error processing protocol message: invalid message for current state.

HTTP PROXY

Error	Number	Description
MSG_HTTPPROXY_IMPERSONATE_FAILED	0xE3030013L	The HTTP proxy client has failed to impersonate the current user. The error is 0x%4!08X!: %1.

PPP

Error	Number	Description
MSG_PPP_BUFF_TO_SMALL	0xE3040001L	PPP parsing error. The buffer is too small. Expected %1!lu!, Actual %2!lu!.
MSG_PPP_BAD_PROTOCOL	0xE3040002L	PPP parsing error. An invalid protocol (%1!lu!) has been specified.
MSG_PPP_BAD_CONTROL	0xE3040003L	PPP parsing error. An invalid control code (%1!lu!) has been specified.
MSG_PPP_BAD_MESSAGE_SIZE	0xE3040004L	PPP parsing error. The message size is too large. Available %1!lu!, Actual %2!lu!.

RAS

Error	Number	Description
NGRAS_E_CONNECTION_FAILURE	0xE3020001L	Unable to establish the VPN connection. The VPN server may be unreachable, or security parameters may not be configured properly for this connection.
NGRAS_E_SERVER_NOT_DEFINED	0xE3020002L	No VPN server has been configured.
NGRAS_E_GUI_DISABLED	0xE3020003L	The graphical user interface (GUI) has been disabled, and a request to display, or prompt, for information was requested. Use the `-gui` parameter to enable the graphical user interface.
NGRAS_E_SOFTWARE_UPDATE_REQUIRED	0xE3020004L	The remote network administrator has indicated that a required update to this software must be installed in order to connect to the remote network.
NGRAS_W_SOFTWARE_UPDATE_IN_PROGRESS	0xA3020005L	A software update has been started or is in progress.
NGRAS_E_REALM_REQUIRED	0xE3020007L	A login group (or realm) must be specified.

NGDIAL

Error	Number	Description
NGDIAL_E_NO_CONNECTION_NAME	0xE3070001L	No connection was specified.

Logon

Error	Number	Description
LOGON_E_NO_LID	0xE3080001L	No logon ID was found.
LOGON_E_MI_CANCELLED	0xE3080002L	Access denied due to local system limitations.
LOGON_E_SI_CANCELLED	0xE3080003L	Access denied. The required system capabilities are not present, enabled, or current.
LOGON_E_DECODING_DATA	0xE3080004L	An error was encountered decoding the data received from the logon server on the appliance.
LOGON_E_ENCODING_DATA	0xE3080005L	An error was encountered encoding data to be sent to the logon server on the appliance.

A

SonicWall Support

Technical support is available to customers who have purchased SonicWall products with a valid maintenance contract and to customers who have trial versions.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. To access the Support Portal, go to https://www.sonicwall.com/support.

The Support Portal enables you to:

- View knowledge base articles and technical documentation
- View video tutorials
- Access MySonicWall
- Learn about SonicWall professional services
- Review SonicWall Support services and warranty information
- Register for training and certification
- Request technical support or customer service

To contact SonicWall Support, visit https://www.sonicwall.com/support/contact-support.

About This Document

Legend

WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

IMPORTANT, NOTE, TIP, MOBILE, or VIDEO: An information icon indicates supporting information.

Connect Tunnel Client Extensibility Toolkit Reference Guide
Updated - December 2017
Software Version - 12.1
232-004161-00 Rev A