MatrixSSL 3.3.1 Release Notes

AuthenTec, Inc.
100 Rialto Place, Suite 100
Melbourne, Florida 32901
321.308.1300
authentec . com

AuthenTec welcomes your input. We try to make our publications useful, interesting, and informative, and we hope you will take the time to help us improve them. Please send any comments or suggestions by mail or e-mail.

AuthenTec, Inc.
100 Rialto Place, Suite 100
Melbourne, Florida 32901
321-308-1300
www.authentec.com
apps@authentec.com

MatrixSSL 3.3.1 Release Notes

Table of Contents

FEATURE ADDITIONS 3

MAXIMUM FRAGMENT LENGTH TLS EXTENSION 3
DISABLE SPECIFIC TLS PROTOCOL VERSIONS 3

IMPROVEMENTS 3

RC4 CIPHER SUITE BUG FIX 3
CORE/WIN32/OPDEP.C 3
CLEANUP AFTER FAILED KEY LOADING 3
TLS 1.1 EXPLICIT IV HANDLING 4

CHANGES 4

RAW RSA CRYPTO FUNCTIONS 4

Feature Additions

This section highlights the new features that have been added since MatrixSSL 3.3.0

Maximum Fragment Length TLS Extension

The max_fragment_length extension defined in RFC 4366 has been added to MatrixSSL. This extension allows TLS clients to suggest the maximum record size that can be used in communications with a server. Support for this extension has been added to both MatrixSSL clients and servers. The new define REQUESTED_MAX_PLAINTEXT_RECORD_LEN in matrixsslConfig.h has been added to control this feature.

See the Developer's Guide for more information on this feature.

Disable Specific TLS Protocol Versions

Previous versions of MatrixSSL automatically enabled TLS 1.0 if TLS 1.1 support was compiled into the library (USE_TLS_1_1). A new set of DISABLE_TLS defines is now available in matrixsslConfig.h to disable specific protocol versions. This disabling of protocol versions is handled at runtime when the SSL peers are negotiating.

The USE_TLS_ set of defines still must "stack" for compilation purposes (i.e. can not enable USE_TLS_1_1 without enabling USE_TLS). See the code comments surrounding these defines in matrixsslConfig.h for more information.

Improvements

This section highlights under-the-hood changes since MatrixSSL 3.3.0

RC4 Cipher Suite Bug Fix

An issue was discovered in connections that negotiated to RC4 cipher suites. If multiple SSL records were received in a single data buffer the second record would not be correctly processed. This issue is fixed in MatrixSSL 3.3.1

core/WIN32/opdep.c

The psGetFileBuf API is now correctly wrapped in the MATRIX_USE_FILE_SYSTEM define and the Windows ReadFile call has been improved to only ask for the remaining bytes in the file if fewer than 512.

Cleanup After Failed Key Loading

Some code paths from failed calls to matrixSslLoad family of APIs were attempting double memory frees causing crashes. These paths have been eliminated. This issue was confined to initialization of the SSL application so generally would have been caught at integration time.

TLS 1.1 Explicit IV Handling

The explicit IV portion of encrypted TLS 1.1 records are now handled in the matrixssl module rather than the crypto module. Previous MatrixSSL versions implemented a mechanism where the cipher 'ate' the additional block. This change improves compatibility with other crypto providers.

Changes

This section highlights user interface or configuration changes since MatrixSSL 3.3.0

Raw RSA Crypto Functions

The function prototypes for the semi-public psRsa family of crypto APIs have changed. There is now an additional void* parameter as a final argument to these functions for internal purposes. Existing consumers of these functions should add a NULL argument to their calls when upgrading to MatrixSSL 3.3.1.

AuthenTec, Inc.
100 Rialto Place, Suite 100
Melbourne, Florida 32901
321-308-1300 (voice)
321-308-1430 (fax)
www.authentec.com
apps@authentec.com