MatrixSSL 3.6.2 Commercial Source Release Notes

1 BUG FIXES AND IMPROVEMENTS

1.1 ECC public point validation

Previous versions of MatrixSSL were not testing ECC public keys for validity when ECDHE suites were used. Security researchers at Matsano Security crafted a malicious public ECC key that was able to cause a memory read access violation on some platforms and an infinite loop on others.

ECDHE cipher suites are not enabled in the default configuration, however customers that have enabled ECDHE cipher suites should update to the current version of MatrixSSL, or contact INSIDE Secure for information on patching older versions.

1.2 Fixed AES_GCM for big endian platforms

A bug was preventing the AES_GCM tag from being created correctly on big endian platforms

1.3 Additional sanity tests in math module

Some bad keys have shown to cause problem when put through the math operations. Some additional sanity tests have been added to fail on the calculations when unexpected conditions are found.

1.4 X.509 path length for servers that send root CA

Clients were incorrectly calculating the pathLen constraint in X.509 certificate chains when servers sent the root CA as part of the chain. It is not advised servers send the root CA but it is now handled correctly if those servers are encountered.

1.5 Both client and server initiate a re-handshake

A client will now ignore a HELLO_REQUEST message from a server if that client had just sent a CLIENT_HELLO re-handhake request. The assumption is that both peers had decided to re-handshake simultaneously and the client received the request after sending its own request.

2 FRAMEWORKS CHANGES

2.1 Makefile build system creates module libraries

The Makefile framework now generates three module libraries when compiling MatrixSSL. The modules are core, crypto, and matrixssl and correspond to the source code underneath those directories. This change was made to mirror the development environment of the code maintainers to more easily capture build system improvements when generating product packages.

The standalone crypto library is also a commonly requested package for those only wanting to work with the raw crypto APIs.

2.2 Client example application command line parameters

The client example application (./apps/client.c) now accepts getopt style parameters rather than strict ordered parameters. The runClient.sh script has been updated to show the new usage or simply run ./client to see the usage.

2.3 CLIENT_KEY_EXCHANGE added to “postponed” PKA framework

To better support asynchronous hardware operations, the creation of handshake flights that require public key crypto operations use a two-step process. First, the entire plaintext flight is written to the buffer and the location of the public key crypto output is left empty. Second, the public key operations are performed and the results are written to the empty areas that were left for them. This version has added client side support to this framework with the CLIENT_KEY_EXCHANGE message. This has no impact on the public API.