Knowledge base

Subject: Reserved RSA key space
Publish date: 13.03.2008
Revision: 1.0

Classifications: RSA keys; PKI; Format; CardOS; Java card

The information in this article applies to: PKCS#11; Token initialization parameters, reserved memory space. eToken format type

Description:

This article describes the method for setting reserved space for RSA key-pairs on the eToken.

The requirement for reserving a specific folder of pre-defined size for RSA keys is related to tokens that are initialized with format type 0 or 4. Please see table below for etoken format information:

Format type	Supported tokens
Format 0	RTE 3.65 format or PKI Client 4.x with legacy format
Format 4	PKI Client 4.x default format (same as 0 regarding RSA behavior)
Format 5	FIPS mode CardOS based tokens and Javacard tokens - includes new RSA behavior that is not controlled by key size. Each key is created in a separate directory.

The information and formulas below are related to CardOS based tokens only of format type 0 and 4.

It is not relevant for Java based eToken or CardOS based tokens formatted for FIPS.

What is Reserved RSA key space?

The space reserved on the eToken in a dedicated folder for storing RSA key-pairs. Once set, this is the maximal space that will be used for storing RSA keys on the token **even if there is still free memory space on the eToken**.

The size of the allocated space is determined at the time of eToken initialization and cannot be modified later on without re-initializing the token.

1. Setting the reserved RSA key space using native PKCS#11 API:

The PKCS#11 native *C_InitToken* function can be controlled by a registry key that manages the RSA directory size. Please see *eToken PKI 4.5 Developers Guide* for more details on the ETCKA_RSA_AREA_SIZE parameter.

ETCKA_RSA_AREA_SIZE – defines the area size in bytes reserved for RSA keys.

CardOS based tokens use this parameter to reserve the place for RSA keys. The RSA keys

are created and stored only within this area. The dedicated folder cannot be used to store any other types of data.
Providing a value size of 0 bytes prevents creation of RSA keys on the token - effectively leaving more place for other user data.

For allocating dedicated area for RSA keys during eToken initialization, use the following Registry entry: *HKEY_LOCAL_MACHINE\SOFTWARE\Aladdin\eToken\MIDDLEWARE\Init*

Create a DWORD value of ETCKA_RSA_AREA_SIZE and set it to the desired number of bytes.

The number of reserved bytes can be calculated using the following formula:

N 1024 key = (RSA-AREA-SIZE - 100) / 500

or RSA-AREA-SIZE = (N * 500) + 100

For example:
8 = (4100 - 100) / 500

Note: the value of 500 bytes in the formula is an approximate value only. It mught be changed from key to key, therefore the calculation provide an estimated value only.

For 2048 keys, replace the value of 500 with 750:

N 2048 key = (RSA-AREA-SIZE - 100) / 750

Important Notes:

1. As stated above, there is no way to determine the **exact** number of keys stored on an eToken, created in the RSA directory. It is possible to verify it with demo creation and deletion of keys.
2. The space allocated for RSA key is deducted from the total eToken available free memory that the user sees.
3. In tokens that are initialized in initialization type 5 (“format 5” - Java Cards or Cards 4.20B formatted as FIPS) the calculation is not relevant since each key saved on different folder and the memory capacity is limited only by the token’s memory size.

2. Setting the reserved RSA key space using eToken API (PKCS#11 Extension):

It is possible to use our PKCS#11 Init Extension function for initializing the eToken device and use RSA size as a parameter of the eToken object.
More details on using the parameter can be found in the eToken PKI 4.5 Developers Guide.