Compiling as3crypto

The Flash client uses the as3crypto library for TLS communication with the server. Should the need arise to build as3crypto.swc from source again, you might save yourself a bit of pain by changing the function pkcs1pad() in src\com\hurlant\crypto\rsa\RSAKey.as to the following:

/\*\*

\* PKCS#1 pad. type 1 (0xff) or 2, random.

\* puts as much data from src into it, leaves what doesn't fit alone.

\*/

private function pkcs1pad(src:ByteArray, end:int, n:uint, type:uint = 0x02):ByteArray {

var out:ByteArray = new ByteArray;

var p:uint = src.position;

var nLocal:uint = n;

end = Math.min(end, src.length, p+nLocal-11);

src.position = end;

var i:int = end-1;

while ((i != -1) && (i >= p) && (nLocal > 11))

{

nLocal = nLocal - 1;

if (i >= 0)

{

out[nLocal] = src[i--];

}

else

{

var nHuh:uint = 4;

}

}

nLocal = nLocal - 1;

out[nLocal] = 0;

if (type==0x02) { // type 2

var rng:Random = new Random;

var x:int = 0;

while (nLocal>2) {

do {

x = rng.nextByte();

} while (x==0);

nLocal = nLocal- 1;

out[nLocal] = x;

}

} else { // type 1

while (nLocal>2) {

nLocal = nLocal - 1;

out[nLocal] = 0xFF;

}

}

nLocal = nLocal - 1;

out[nLocal] = type;

nLocal = nLocal - 1;

out[nLocal] = 0;

return out;

}

For some reason the above function works properly on the iPad while the previous version did not. This may be an Actionscript to Objective-C translation bug because the two functions are logically equivalent.