Proposed changes to Web Crypto to address the portion of Bug #25839 to add support for the NUMS curves as defined in [MSRECC].

Based on W3C Editor’s Draft 16 June 2014

# General Comments

The current draft references the ANSI standards X9.62 and X9.63 as normative references for ECDSA and ECDH-related functions. Neither X9.62 nor X9.63 are freely available (you have to buy them from ANSI). I don’t believe it is appropriate for a W3C standard to depend upon non-free standards documents. As X9.62 and X9.63 are not freely available, I have not checked the validity of the various references in Web Crypto to steps in X9.62 and X9.63, nor have I checked to see whether X9.62 and X9.63 have similar problems that would block extensibility as RFC 5480 does. For example, if X9.62 and X9.63 describe only ECC operations on Weierstrass-form curves, they won’t work for either Montgomery-form (like Curve25519) or twisted Edwards-form (like the numsp\*t1 curves).

# Changes to Section 25: ECDSA

In Section 25.4 EcKeyGenParams, in the set of values recognized for NamedCurve, after P-521 add the following:

“numsp256d1”  
 Curve numsp256d1 as specified in [draft-black-numscurves].

“numsp256t1”  
 Curve numsp256t1 as specified in [draft-black-numscurves].

“numsp384d1”  
 Curve numsp384d1 as specified in [draft-black-numscurves].

“numsp384t1”  
 Curve numsp384t1 as specified in [draft-black-numscurves].

“numsp512d1”  
 Curve numsp512d1 as specified in [draft-black-numscurves].

“numsp512t1”  
 Curve numsp512t1 as specified in [draft-black-numscurves].

In Section 25.7 Operations, Import Key, Step 2, if format is “spki”

Step 6 needs to be removed – it forces the set of acceptable namedCurves to be only those listed in RFC 5480. RFC 5480 has more than just the NIST curves (it has all the secp curves) but no extensibility.

In Step 8, after the clause for secp521r1 and before the “Otherwise” clause, add the following:

If params is equivalent to the numsp256d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256d1”.

If params is equivalent to the numsp256t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256t1”.

If params is equivalent to the numsp384d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384d1”.

If params is equivalent to the numsp384t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384t1”.

If params is equivalent to the numsp512d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512d1”.

If params is equivalent to the numsp512t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512t1”.

In Section 25.7 Operations, Import Key, Step 2, if format is “pkcs8”

Remove Step 6 for same reason as above.

In Step 9, remove the clause “is not an instance of the namedCurve ASN.1 type defined in RFC 5480, or”

In Step 11, after the clause for secp521r1 and before the “Otherwise” clause, add the following:

If params is equivalent to the numsp256d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256d1”.

If params is equivalent to the numsp256t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256t1”.

If params is equivalent to the numsp384d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384d1”.

If params is equivalent to the numsp384t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384t1”.

If params is equivalent to the numsp512d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512d1”.

If params is equivalent to the numsp512t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512t1”.

In Section 25.7 Operations, Import Key, Step 2, if format is “jwk”:

NOTE: No changes in this section at this time because there are not jwk crv definitions for the NUMS curves at this time. Once such definitions exist this section will need to be update to match.

In Section 25.7 Operations, Export Key, Step 2, if format is “spki”

In substep 2, in the clause starting “Set the parameters field…”, add the following if statements after the if statement for P-521:

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256d1”:

Let the namedCurve be the object identifier numsp256d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256t1”:

Let the namedCurve be the object identifier numsp256t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384d1”:

Let the namedCurve be the object identifier numsp384d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384t1”:

Let the namedCurve be the object identifier numsp384t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512d1”:

Let the namedCurve be the object identifier numsp512d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512t1”:

Let the namedCurve be the object identifier numsp512t1 defined in [draft-black-numscurve]

In Section 25.7 Operations, Export Key, Step 2, if format is “pkcs8”

In substep 2, in the clause starting “Set the parameters field…”, add the following if statements after the if statement for P-521:

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256d1”:

Let the namedCurve be the object identifier numsp256d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256t1”:

Let the namedCurve be the object identifier numsp256t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384d1”:

Let the namedCurve be the object identifier numsp384d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384t1”:

Let the namedCurve be the object identifier numsp384t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512d1”:

Let the namedCurve be the object identifier numsp512d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512t1”:

Let the namedCurve be the object identifier numsp512t1 defined in [draft-black-numscurve]

In Section 25.7 Operations, Export Key, Step 2, if format is “jwk”

NOTE: No changes in this section at this time because there are not jwk crv definitions for the NUMS curves at this time. Once such definitions exist this section will need to be update to match.

# Changes to Section 26: ECDH

In Section 26.4 Operations, Import Key, Step 2, if format is “spki”

Step 6 needs to be removed for same reasons as above in ECDSA (it forces the set of acceptable namedCurves to be only those listed in RFC 5480).

In Step 8, after the clause for secp521r1 and before the “Otherwise” clause, add the following:

If params is equivalent to the numsp256d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256d1”.

If params is equivalent to the numsp256t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256t1”.

If params is equivalent to the numsp384d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384d1”.

If params is equivalent to the numsp384t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384t1”.

If params is equivalent to the numsp512d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512d1”.

If params is equivalent to the numsp512t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512t1”.

In Section 26.4 Operations, Import Key, Step 2, if format is “pkcs8”

Remove Step 6 for same reason as above.

In Step 9, remove the clause “is not an instance of the namedCurve ASN.1 type defined in RFC 5480, or”

In Step 11, after the clause for secp521r1 and before the “Otherwise” clause, add the following:

If params is equivalent to the numsp256d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256d1”.

If params is equivalent to the numsp256t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp256t1”.

If params is equivalent to the numsp384d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384d1”.

If params is equivalent to the numsp384t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp384t1”.

If params is equivalent to the numsp512d1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512d1”.

If params is equivalent to the numsp512t1 object identifier defined in [draft-black-numscurves]:

Set namedCurve “numsp512t1”.

In Section 26.4 Operations, Import Key, Step 2, if format is “jwk”:

NOTE: No changes in this section at this time because there are not jwk crv definitions for the NUMS curves at this time. Once such definitions exist this section will need to be update to match.

In Section 26.4 Operations, Export Key, Step 2, if format is “spki”

In substep 2, in the clause starting “Set the parameters field…”, add the following if statements after the if statement for P-521:

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256d1”:

Let the namedCurve be the object identifier numsp256d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256t1”:

Let the namedCurve be the object identifier numsp256t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384d1”:

Let the namedCurve be the object identifier numsp384d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384t1”:

Let the namedCurve be the object identifier numsp384t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512d1”:

Let the namedCurve be the object identifier numsp512d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512t1”:

Let the namedCurve be the object identifier numsp512t1 defined in [draft-black-numscurve]

In Section 26.4 Operations, Export Key, Step 2, if format is “pkcs8”

In substep 2, in the clause starting “Set the parameters field…”, add the following if statements after the if statement for P-521:

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256d1”:

Let the namedCurve be the object identifier numsp256d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp256t1”:

Let the namedCurve be the object identifier numsp256t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384d1”:

Let the namedCurve be the object identifier numsp384d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp384t1”:

Let the namedCurve be the object identifier numsp384t1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512d1”:

Let the namedCurve be the object identifier numsp512d1 defined in [draft-black-numscurve]

If the namedCurve attribute of the [[algorithm]] internal slot of key is “numsp512t1”:

Let the namedCurve be the object identifier numsp512t1 defined in [draft-black-numscurve]

In Section 26.4 Operations, Export Key, Step 2, if format is “jwk”

NOTE: No changes in this section at this time because there are not jwk crv definitions for the NUMS curves at this time. Once such definitions exist this section will need to be update to match.

# Changes to Section 42: References

In Section 42.1, Normative References, add:

[draft-black-numscurves] Elliptic Curve Cryptography (ECC) Nothing Up My Sleeve (NUMS) Curves and Curve Generation. Work in progress, currently available as IETF Internet-Draft draft-black-numscurves-01 at <https://datatracker.ietf.org/doc/draft-black-numscurves/>. B. Black, J. Bos, C.Costello, P. Longa, M. Naerhig.

In Section 42.2, Informative References, add:

[draft-black-tls-numscurves] Nothing Up My Sleeve (NUMS) Curves for Ephemeral Key Exchange in Transport Layer Security (TLS). Work in progress, currently available as IETF Internet-Draft draft-black-tls-numscurves-00 at <https://datatracker.ietf.org/doc/draft-black-tls-numscurves/>. B. Black, T. Acar, M. Ray.

[MSRECC] Bos, J., Costello, C., Longa, P., and M. Naehrig, "Selecting Elliptic Curves for Cryptography: An Efficiency and Security Analysis", February 2014, <http://eprint.iacr.org/2014/130.pdf>.