Network Working Group
Category: INTERNET-DRAFT
Expires April 2000

A. Malpani (ValiCert)

October 1999

The NULL Public Key Algorithm (NPKA)
<draft-malpani-npka-00.txt>

Status of this Memo

Copyright Notice

Abstract

This memo defines the NULL public key algorithm. The original goal of this effort was to be able to allow people to optionally sign data, without needing the make the signature optional in the ASN.1. While we were at it, we decided to, for completeness, also specify the method for NULL public key encryption.

1. Introduction

This memo defines the NULL public key algorithm. It explains how NPKA NULL algorithm should be used both for digital signatures and encryption/key exchange.

Despite the fact that we are not lawyers, we are relatively confident that it is quite safe to use this algorithm for export for any key length size. It is also quite impossible for people to discover your private key via timing, power analysis or other cryptographic methods, as long as you are only using this algorithm.

2. Algorithm Details

2.1 Algorithm Definition

In this section, we will show how NPKA can be used for both digital signatures and key exchange. We use the following notation:

```
 B represents the puBlic key
 V represents the priVate key
 C is the Clear text message
 E is the Encrypted message
 S is the Signing algorithm
 G is the siGnature verification algorithm
 Y is the key/data encrYption algorithm
 D is the key/data decryption algorithm
 F{x, y} is the function F on data elements x and y
```

2.1.1 Digital Signatures

This section shows how a private key is used to create a digital signature and a public key used to verify the digital signature.

For signatures, the holder of the private key uses the message and the private key to produce a digital signature, which can be verified by anyone with the holder's public key.

```
S{C, V} => C
G{E, B} => E
```

Note: This satisfies the property required by all public key signature algorithms - G{S{C, V}, B} => C

2.1.2 Encryption/Key Exchange

This section shows how a public key is used to encrypt data/keys and a public key used to decrypt the data.

```
Y{C, B} => C
D{E, V} => E
```

Note: This satisfies the property required by all public key encryption algorithms - D{Y{C, B}, V} => C

2.2 Keying Material

Like other modern ciphers, e.g., RC5 [RFC-2040], NPKA can make use of keys of varying lengths. However, no measurable increase in security is afforded by the use of longer key lengths.

2.3 Padding

NULL has a block size of 1 byte, thus padding is not necessary.

2.4. Performance

The NULL encryption algorithm is significantly faster than other commonly used symmetric encryption algorithms and implementations of the base algorithm are available for all commonly used hardware and OS platforms.

2.5 Test Vectors

[TBD]
We should also show what a cert with an NPKA signature looks like

3. Object Identifiers

[TBD]
We need to create the OIDs for sha1withNPKA, md4withNPKA, ...

4. Operational Requirements

[TBD]

5. Security Considerations

If you do implement this algorithm, please make sure that signatures using that algorithm are only accepted in places where you do not need signatures. Similarly, encryption with this algorithm is only performed where you do not want encryption.

6. Algorithm properties

In this section, we try to outline the main properties of NPKA.

- Very, very high performance for both encryption and decryption, for key exchange and signing.
- No export restrictions (for any key length).
- No risk of exposing your private key to any potential attacks.
- Short key sizes are as strong as keys twice as long.
- Small algorithm footprint - excellent for smart card support or other low memory devices.
- Support for any sized key.
- Can easily be used in both a block or streaming mode.
- Great synchronization properties - loss of a single bit in transmission

results in only a single bit loss at the receiver (?)

7. Intellectual Property Rights

[TBD]

8. Acknowledgments

Spiritual and textual guidance for this document we provided by [RFC2410].

9. References

[RFC-2410] Glenn R., and Kent, S., "The NULL Encryption Algorithm and Its Use With IPsec", RFC 2410, November 1998.

10. Editors' Addresses

Ambarish Malpani
ValiCert, Inc.
1215 Terra Bella,
Mountain View, CA 94043

EMail: ambarish@valicert.com
Phone: 650.567.5457

11. Full Copyright Statement

BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Expires April 2000