# $\pi$*–Cipher* [1]

Designers: Danilo Gligoroski[2] and Hristina Mihajloska[3] and Simona Samardjiska[23] and Håkon Jacobsen[2] and Mohamed El-Hadedy[2] and Rune Erlend Jensen[4]

Submitter: Hristina Mihajloska
`hristina.mihajloska@finki.ukim.mk`

Update to $\pi$-Cipher v2, November 2014

[1] Since, the name of the cipher contains the Greek letter $\pi$, in the software implementations we will use the name `PiCipher`. More precisely in this document we propose the following six variants of the cipher: `Pi16Cipher096v2`, `Pi16Cipher128v2`, `Pi32Cipher128v2`, `Pi32Cipher256v2`, `Pi64Cipher128v2`, `Pi64Cipher256v2`

[2] ITEM, Norwegian University of Science and Technology, Trondheim, Norway

[3] FCSE, "Ss Cyril and Methodius" University, Skopje, Republic of Macedonia

[4] IDI, Norwegian University of Science and Technology, Trondheim, Norway

# Update from $\pi$-Cipher v1

This document is the summary of the updates from v1 to v2 of the $\pi$-Cipher documentation.

## Update in Chapter 1. Specification

This chapter contains correction on the padding rule that leads to easy forgery strategy as it was pointed out by Gaëtan Leurent and Thomas Fuhr [1].

In Version 1, the padding rule for the last block of the $AD$ is the following:

$$AD_a \leftarrow \begin{cases} AD_a & \text{if } |AD_a| = bitrate, \\ AD_a||10^* & \text{if } |AD_a| < bitrate, \end{cases}$$

where 1 represents the byte `0x01`, and 0 represents the byte `0x00`.

In Version 1, the padding rule for the last block of the message $M$ is the following:

$$M_m \leftarrow \begin{cases} M_m & \text{if } |M_m| = bitrate, \\ M_m||10^* & \text{if } |M_m| < bitrate, \end{cases}$$

where 1 represents the byte `0x01`, and 0 represents the byte `0x00`.

In order to solve the issue pointed out in [1] we modify the padding rule as following: "Append 1 in any case, and fill the rest of the block with 0s". Thus, the changes will be:

The padding rule for the associated data $AD$ is the following:

$$AD = AD_1||AD_2||\dots||AD_a||10^*$$

where 1 represents the byte `0x01`, and 0 represents the byte `0x00`.

The padding rule for the message $M$ is the following:

$$M = M_1||M_2||\dots||M_m||10^*$$

where 1 represents the byte `0x01`, and 0 represents the byte `0x00`.

Note that if the associated data $AD$ (the message $M$) has length that is a multiple of the *bitrate*, then the number of processed blocks of $AD$ ($M$) is increased by one, and thus $a \leftarrow a + 1$ ($m \leftarrow m + 1$).

# Update in Chapter 4. Features

In this chapter we give clarification about the feature *Tag second preimage resistance* - resistance against finding second preimage for an authentication tag when the key is known (insider attack) for short messages.

# Acknowledgment

We would like to thank Gaëtan Leurent and Thomas Fuhr for their detailed observation on the $\pi$-Cipher, pointing out the problem with the padding function in v1, and giving us a note for removing a bug in the reference C code.

# References

[1] Gaëtan Leurent and Thomas Fuhr. Observation on picipher. Message on the crypto-competitions mailing list, November, 2014.