ETSI/SAGE **Version: 1.1**
Technical report **Date: 6th September 2006**

Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Document 5: Design and Evaluation Report

Document History		
V1.0	**6th February 2006**	**Publication**
V1.1	**6th September 2006**	**No change to the technical content at all, just removal of an unwanted page header**

Reference

Keywords
3GPP, security, SAGE, algorithm

ETSI Secretariat

Postal address
F-06921 Sophia Antipolis Cedex - FRANCE

Office address
650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16
Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° 7803/88

X.400
c= fr; a=atlas; p=etsi; s=secretariat

Internet
secretariat@etsi.fr
http://www.etsi.fr

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for ETSI members and non-members, and can be found in SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/ipr).

At the time of writing of this report, no information of any Intellectual Property Rights (IPRs) has been drawn to the attention neither to the Task Force nor to ETSI. See Section 9.8 of this report for further information about IPR investigations conducted by the Task Force.

No guarantee can be given as to the existence of other IPRs not referenced in SR 000 314 (or the updates on the ETSI Web server (http://www.etsi.org/ipr) which are, or may be, or may become, essential to the present document.

Foreword

This Report has been produced by ETSI SAGE Task Force 278 on Design of the second UMTS encryption and integrity protection algorithms UEA2 and UIA2 (SAGE TF 3GPP).

The work described in this report was undertaken in response to a request made by 3GPP.

Version 1 of this report was submitted to the 3GPP SA3 group in January 2006.

1 Scope

This public report contains a detailed summary of the design and evaluation work performed during the development of the 3GPP Confidentiality and Integrity Algorithms UEA2 and UIA2. The report also includes summaries of evaluations that were conducted by independent external evaluators, and reflects modifications that were done to the design based on this feedback.

2 References

For the purposes of this report, the following references apply:

[1] 3G TS 33. 102 V 6.3.0 (2004-12) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Security Architecture.

[2] 3G TS 33. 105 V 6.0.0 (2004-06) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Cryptographic Algorithm Requirements.

[3] 3G TR 33. 901 V 1.0.0 (1999-06) 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Criteria for cryptographic algorithm design process.

[4] 3G TS 25.321 V3.0.0: 3rd Generation Partnership Project; Technical Specification Group (TSG) RAN; Working Group 2 (WG2); MAC protocol specification.

[5] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Document 1: UEA2 and UIA2 Specification; Version: 1.0; Date: 10th January 2006.

[6] ETSI/SAGE Specification. Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2. Document 2: SNOW 3G Specification; Version: 1.0; Date: 10th January 2006.

[7] ETSI/SAGE Specification. Specification of the MILENAGE-2G Algorithms: an Example Algorithm Set for the GSM Authentication and Key Generation Functions A3 and A8. Version 1.0. May 2002.

Additional references to external documents are provided in Annex A.

3 Abbreviations

For the purposes of the present report, the following abbreviations apply:

AES	Advanced Encryption Standard
AuC	Authentication Centre
CK	Cipher Key
ETSI	European Telecommunications Standards Institute
GF(q)	The finite field of q elements
GSMA	GSM Association

3GPP	3rd Generation Partnership Project
FSM	Finite State Machine
f8	UMTS confidentiality (encryption) algorithm
f9	UMTS integrity algorithm
HW	Hardware
IBS	Input Bit Stream
IK	Integrity Key
IPR	Intellectual Property Rights
IV	Initialization Vector
LFSR	Linear Feedback Shift Register
MAC	Message Authentication Code
OBS	Output Bit Stream
OTP	One Time Pad
PDU	Protocol Data Unit
RLC	Radio Link Control
RNC	Radio Network Controller
SA3	3GPP Systems and Architecture Group
SAGE	Security Algorithms Group of Experts
SAGE TF 3GPP	SAGE Task Force for the design of the standard 3GPP Confidentiality and Integrity Algorithms
SDU	Signalling Data Unit
SW	Software
UE	User Equipment
UEA2	UMTS Encryption Algorithm Suite 2
UIA2	UMTS Integrity Algorithm Suite 2
UMTS	Universal Mobile Telecommunications System
USIM	User Services Identity Module
XL	Extended Linearization
XSL	Extended Sparse Linearization

4 Structure of this report

The material presented in this report is organised in the subsequent clauses, as follows:

- Clause 5 provides background information on the second 3GPP Confidentiality and Integrity Algorithms UEA2 and UIA2;
- Clause 6 provides a summary of the algorithm requirements;
- Clause 7 consists of a brief presentation of the actual designs;
- Clause 8 provides information on the design criteria and the design work;
- Clause 9 gives an overview of the evaluation work carried out by SAGE TF 3GPP and other parties and the conclusions of the evaluations;
- Annex A includes a list of external references that are related to the results in this report.

5 Background to the design and evaluation work

The development of the second issue of standardised 3GPP confidentiality and integrity algorithms was undertaken in response to an initiative from 3GPP SA3 and GSMA Security Group. Due to the fact that deployment of new algorithms in networks and handsets takes a very long time, it was decided to develop a second suite of confidentiality and integrity algorithms for 3GPP in case the transition to new algorithms was needed. Even without any indications of weaknesses in the KASUMI-based versions of f8 and f9, the new algorithms should be fundamentally different from the previous, so that an attack on one algorithm is very unlikely to translate into an attack on the other.

The new version of f8 and f9 should meet all relevant security goals with a high degree of confidence, but the available resources for design and evaluation were rather limited in terms of time and funding. The ETSI SAGE group therefore decided on the following strategies for the work:

- Reuse of people and results from the previous 3GPP project.
- Investigate published and well-studied algorithms and schemes for the design.
- Concentrate new designs to establish cryptographic resistance against algebraic attacks.
- Initiate additional evaluation by inviting leading researchers for independent review of the final design.

Based upon the 3GPP requirements and the work conducted by the task force, a modified version of SNOW 2.0 [20] was developed and named SNOW 3G. SNOW 3G is the cryptographic engine of the second suite of 3GPP encryption and integrity algorithms f8 and f9 (UEA2 and UIA2).

6 Summary of algorithm requirements

The general security architecture for 3GPP is specified in ref. [1]. The complete set of security services needed is realised using a set of cryptographic functions identified in ref. [2]. Out of the full algorithm set there is a need for two algorithms fully standardised. These are:

- f8 – Confidentiality algorithm
- f9 – Integrity algorithm

The requirements for the f8 and f9 algorithms were specified in ref. [2]. For the second suite of 3GPP cryptographic algorithms, the same functional and security requirements apply. In addition to cryptographic separation from the KASUMI-based f8/f9 algorithms, the only modification was to support a throughput of 10 Mbit/s. For the completeness of this report we include some of the main requirements:

6.1 f8 – Confidentiality algorithm

The function f8 shall only be used to protect the confidentiality of user data and signalling data sent over the radio access link between UE and RNC.

The algorithm should be designed to accommodate a range of implementation options including hardware and software implementations.

For hardware implementations, it should be possible to implement one instance of the algorithm using less than 10,000 gates (working assumption).

It must be possible to implement the algorithm to achieve an encryption speed in the order of 10 Mbit/s on the downlink and on the uplink. This throughput should be available at a minimal clock speed of 20 MHz.

The f8 algorithm will be used to encrypt frames of variable length up to approximately 20000 bits.

The function f8 should be a symmetric synchronous stream cipher.

The length of the cipher key CK is 128 bits. In case the effective key length should need to be made smaller than 128 bits, the most significant bits of CK shall carry the effective key information, whereas the remaining, least significant bits shall be set zero.

Additional input parameters: COUNT, BEARER, DIRECTION and LENGTH.

This plaintext block consists of the payload of the particular RLC PDUs / MAC SDUs to be encrypted in a single 10ms physical layer frame for a given bearer and transmission direction. It may consist of user traffic or signalling data. The structure of the plaintext block cannot be specified at present.

6.2 f9 – Integrity algorithm

The MAC function f9 shall be used to authenticate the data integrity and data origin of signalling data transmitted between UE and RNC.

The algorithm should be designed to accommodate a range of implementation options including hardware and software implementations.

The function f9 shall be a MAC function.

The length of the integrity key IK is 128 bits. In case the effective key length should need to be made smaller than 128 bits, the most significant bits of IK shall carry the effective key information, whereas the remaining, least significant bits shall be set zero.

Additional input parameters: COUNT, FRESH and LENGTH.

The algorithm shall output a 32-bit MAC.

6.3 Generic requirements for 3GPP cryptographic functions and algorithms

The functions should be designed with a view to their continued use for a period of at least 20 years. Successful attacks with a workload significantly less than exhaustive key search through the effective key space should be impossible.

The designers of above functions should design algorithms to a strength that reflects the above qualitative requirements.

The algorithm will be openly published for public scrutiny. A number of independent and qualified parties shall, prior to publication, evaluate the strength of the algorithm.

Legal restrictions on the use or export of equipment containing cryptographic functions may prevent the use of such equipment in certain countries.

It is the intention that UE and USIMs that embody such algorithms should be free from restrictions on export or use, in order to allow the free circulation of 3G terminals. Network equipment, including RNC and AuC, may be expected to come under more stringent restrictions. It is the intention that RNC and AuC that embody such algorithms should be exportable under the conditions of the Wassenaar Arrangement [30].

7 3GPP confidentiality and integrity algorithms

The detailed specifications of the UEA2 and UIA2 algorithms are found in ref. [5] and ref. [6]. For this report we include a general overview of the designs. The basic building block is the stream cipher SNOW 3G, which is a two component stream cipher with an internal state of 608 bits initialized by a 128-bit cipher key CK and a 128-bit initialization vector IV.

7.1 SNOW 3G

The structure of SNOW 3G is depicted in the following diagram:

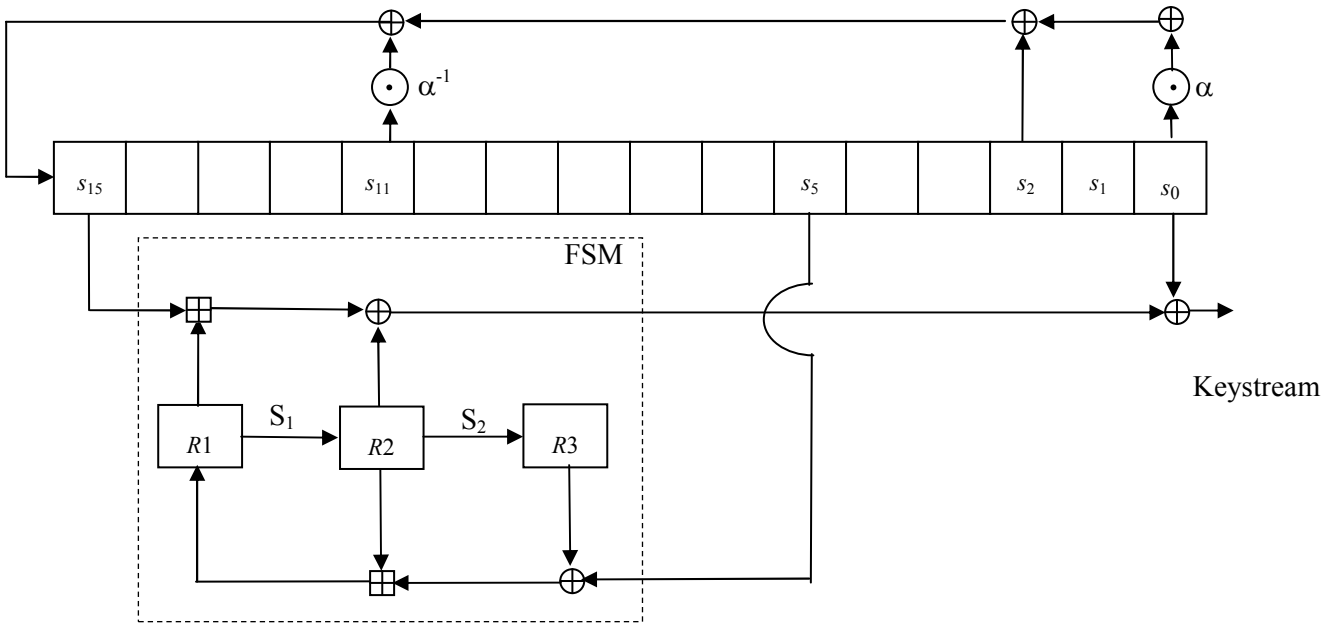


Figure 1: Schematic drawing of SNOW 3G

SNOW 3G consists of a Linear Feedback Shift Register (LFSR) and a Finite State Machine (FSM). The LFSR is constructed from 16 stages, each holding 32 bits and the feedback is defined by a primitive polynomial over the finite field GF(2^{32}). The FSM is based upon three 32-bit registers R1, R2, and R3. The operation of the FSM involves input from the LFSR and uses two substitution box ensembles S1 and S2. The mixing operations are exclusive OR and addition modulo 2^{32}. See ref. [6] for details on the specification of S-boxes, the S-box ensembles, the loading of key variable and IVs and the generation of the keystream.

7.2 Confidentiality function UEA2

The new confidentiality algorithm for f8 (UEA2) now encrypts and decrypts frames using SNOW 3G as a standard synchronous stream cipher. Ref. [5] defines how the system parameters COUNT, BEARER and DIRECTION are used together with the Confidentiality Key (CK) to initialise the keystream generator.

The output from SNOW 3G consists of 32 bit words that are xored to corresponding Input Bit Stream (IBS).

The main stream cipher principles of UEA2 are shown in the following diagram. The produced cipher text block is denoted as Output Bit Stream (OBS).

For decryption the same scheme is used to recover the plain text block (IBS) from the received cipher text (OBS). Sender and receiver will synchronize for each frame using the frame counter COUNT-C.

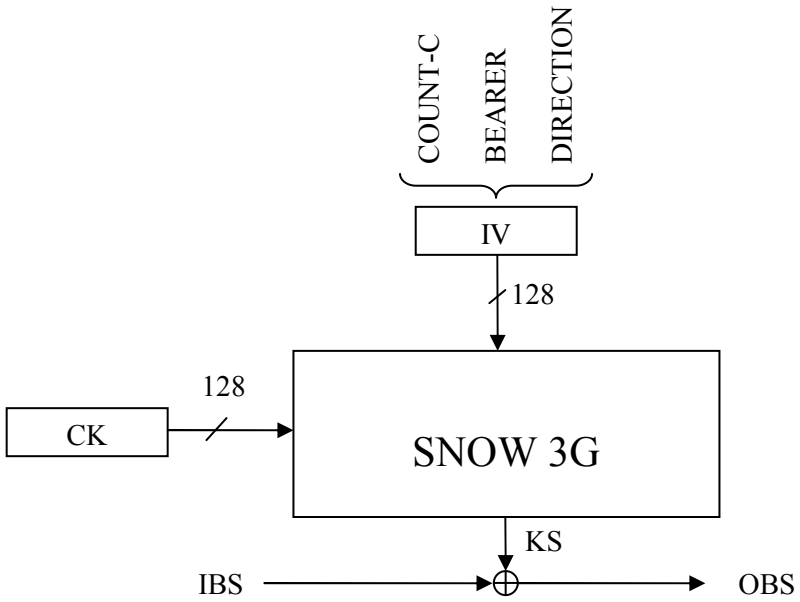


Figure 2: Principles of the UEA2 encryption operation

7.3 Integrity function UIA2

The new integrity algorithm f9 (UIA2) computes a 32-bit Message Authentication Code (MAC) on an input message under an integrity key IK. The message may be between 1 and 20000 bits in length. The UIA algorithm is based on universal hashing and the GMAC-scheme for generation of the MAC.

Prior to processing the message, the SNOW 3G generator is initialized with the integrity key IK and the initialization vector IV, and three random values are generated. P and Q of length 64 bits and the one-time-pad value OTP of length 32 bits.

The message is padded with a 64-bit length field and divided into a list of 64-bit blocks. This list represents a polynomial $f(x)$ of degree greater than or equal to 1 over the finite field $GF(2^{64})$. This polynomial is then evaluated at the secret point P using standard finite field arithmetic modulo the irreducible polynomial $x^{64} + x^4 + x^3 + x + 1$. The 64-bit result from this operation is then multiplied by the secret 64-bit value Q, truncated to 32 bits and xored to the secret 32-bit value OTP.

The structure of f9 is depicted in the following diagram:

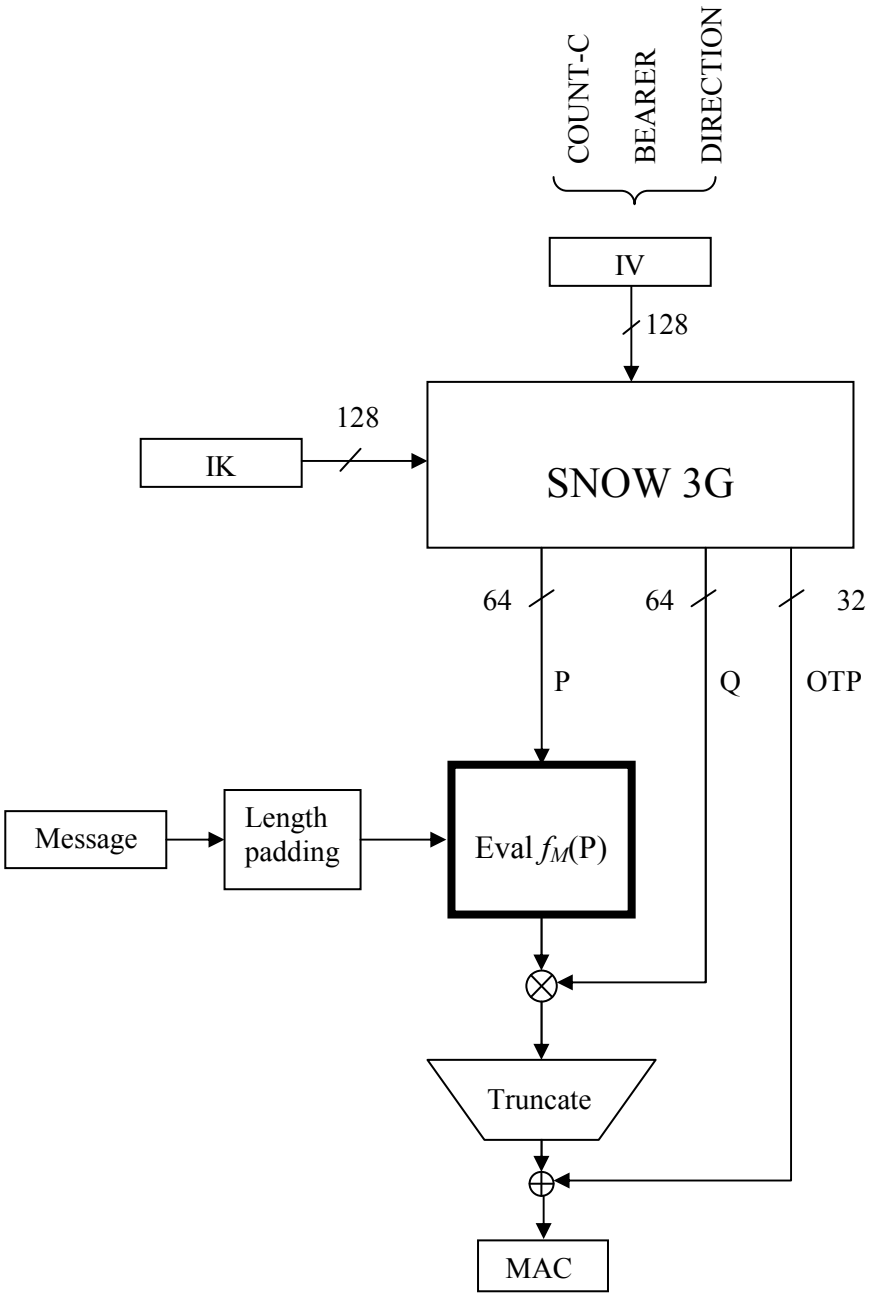


Figure 3: Principles of the UIA2 MAC computation

8 Rationale for the chosen design

8.1 General comments

The essential design goals for the second suite of 3GPP confidentiality and integrity algorithms were that the algorithms should:

- provide a high level of security within the 3GPP context;
- meet their implementation requirements — in particular, allow a low power, low gate count implementation in hardware;
- be founded on cryptographic principles different from the KASUMI-based constructions – so that progress in cryptanalysis towards one of the algorithms should not directly lead to attacks against the other.

The designers have added new components to the published algorithms SNOW 2.0, ref. [20], but have tried to keep the algorithm as simple as possible. They wanted the algorithms to be secure against all practical attacks, and carefully decided not to over-complicate them just to provide a very high security margin against unrealistic theoretical attacks. Great care was taken to ensure that any modification to SNOW 2.0 should not weaken the cryptographic strength of the algorithm.

The following types of attack against the stream cipher SNOW 3G were particularly considered:

- algebraic attacks;
- distinguishing attacks.

Attacks against the UEA2 and UIA2 constructions were also considered.

8.1.1 Use of AES in UMTS

A good choice for a new 3GPP encryption algorithm could have been the NIST approved Advanced Encryption Standard (AES), ref. [24]. This algorithm has been reviewed and applied in the context of 3GPP before. At the time of designing the first 3GPP cryptographic algorithms, the AES project was still in an early phase and no winner had been selected. However, the candidate algorithm Rijndael was selected by the design team of the 3GPP authentication and key management framework as the cryptographic engine of the MILENAGE specification, ref. [7]. Rijndael was later to become the AES algorithm and has by now achieved widespread use and high acceptance.

The main argument for not taking AES as the cryptographic core of UEA2 and UIA2 was that the AES uses the same principles for its non-linear components as KASUMI. In fact much of the recent research in algebraic attacks against block ciphers was triggered by the inherent algebraic structure of the AES S-box. See ref. [15] for an early paper discussing the possibility of weaknesses in block ciphers using S-boxes that can be expressed by simple algebraic equations. Even if it is not known if the XL/XSL-algorithms may be used to break AES, at was agreed that the new UEA2/UIA2 algorithms should include special protection against algebraic attacks. Instead of just changing the S-box of a well-established algorithm like AES, the project team looked for other strong algorithms that could safely be enhanced with protection mechanisms against this new family of attacks.

8.1.2 Selection of SNOW 2.0

The decision to use SNOW 2.0 as the basis for UEA2 and UIA2 was based on the fact that the algorithm had been available for some years, the algorithm was based on sound cryptographic principles, a lot of public analysis has been undertaken, and no major flaws have been identified. The algorithm allows for efficient implementation both in hardware and software, and SNOW 2.0 is one out of two stream ciphers chosen for the forthcoming international standard ISO/IEC IS 18033-4.

8.2 Design Policy of SNOW 2.0

The 3GPP algorithm SNOW 3G is based on the stream cipher SNOW 2.0, ref. [20], which again was an improved version of SNOW 1.0, ref. [19].

The first version of SNOW was submitted to the NESSIE project. The main design goal was to design a cryptographic primitive that would provide an excellent trade-off between security and performance. High throughput is achieved using a word-oriented stream cipher that outputs a sequence of keystream words of length 32 bit.

A few weaknesses were reported on the original SNOW construction. First Hawkes and Rose described a guess-and-determine attack on SNOW 1.0 in 2002, ref. [23]. This is a key recovery attack which applies to the 256-bit key version of the algorithm. The attack has a data complexity of 2^{95} words and process complexity of 2^{224} operations.

A second attack on SNOW 1.0 was described by Coppersmith, Halevi, and Jutla, ref. [14]. Their attack was using the technique of linear cryptanalysis to distinguish the output of SNOW 1.0 from a truly random bit sequence. This attack needs about 2^{95} words of output and the computational complexity is about 2^{100} operations.

The main changes from SNOW 1.0 to SNOW 2.0 were to modify the feedback polynomial and to ensure that the FSM takes two inputs from the shift register. In SNOW 2.0 the finite field GF(2^{32}) is seen as an extension field of degree four over the finite field GF(2^8). The feedback loop involves two multiplications that can be implemented as a byte shift together with an unconditional XOR with one of 256 possible patterns. This results in a better spreading of the bits in the feedback loop and improves the resistance against correlation attacks. The use of two constants in the feedback loops also improves resistance against bitwise linear approximation attacks. The introduction of two inputs to the FSM part makes a guess-and-determine attack more difficult. The modified S-box in SNOW 2.0 also provides stronger diffusion since each output bit now depends on each input bit. The S-box is now defined by the AES S-box followed by the AES *MixColumn* operation.

Watanabe et. al, ref. [31], present a distinguishing attack against SNOW 2.0 using the linear masking method, ref. [14]. Their attack requires 2^{225} words of output (2^{230} bits) and 2^{225} steps of analysis to distinguish the output of SNOW 2.0 from a truly random bit sequence. Their conclusion is that SNOW 2.0 generates sufficiently random sequences as a 128-bit keystream cipher, but is slightly weak as a 256-bit keystream cipher. Recent research, ref. [26], has refined this result and it has been shown that there is a linear distinguisher on SNOW 2.0 which requires 2^{177} bits of keystream and 2^{172} operations. These new results will also apply to SNOW 3G, but do not constitute any attack since the key size of this algorithm is 128 bits.

8.3 Changes from SNOW 2.0 to SNOW 3G

The resistance of SNOW 2.0 to algebraic attacks was studied by Billet and Gilbert, ref. [11]. They show that a modified version of SNOW 2.0, where the two additions modulo 2^{32} are replaced by standard XOR, is highly vulnerable to an algebraic attack with time complexity about 2^{50} and requires no more than 1000 words of output. They further show that SNOW 2.0 can be representing as a system of overdetermined quadratic equations by just collecting 17 known keystream bits. The feasibility of solving such systems is an open area of research, but this paper shows that the intractability of attacking SNOW 2.0 using algebraic attacks will be similar to the AES and KASUMI.

Based in these results, the main goal for the design team was to increase the resistance of SNOW 2.0 against algebraic attacks. This goal was achieved by the introduction of the 32-bit register R3 and the second ensemble of S-boxes S2 in the FSM-component of SNOW 3G.

8.4 Choice of integrity mechanism

Three possible principles for the design of UIA2 were considered. One option was to re-use the KASUMI-based scheme for f9 developed for the first suite of 3GPP algorithms with a different block cipher. This integrity algorithm is a strengthened variant of the widely used CBC-MAC. However, this scheme would not work with a stream cipher like SNOW 3G, and introducing another block cipher into the design would have an unacceptable impact on HW resources.

The second alternative was to use a MAC-construction based on a collision-free hash function like HMAC using SHA-256 as the basic hash function. HMAC is quite efficient and provides good provable security under fairly standard security assumptions. Furthermore some implementors are considering the inclusion of SHA-256 in their handsets. HMAC is generally seen as a strong and efficient message authentication algorithm and is much used in applications like IPSEC, but this approach would also require implementation of the full SHA-256 algorithm.

The MAC-algorithm that was chosen in the end is based on universal hash functions and is quite similar to GMAC, ref. [22], and is also known as a Carter-Wegman type of MAC. MAC-constructions like UIA2 have recently gained a lot of attention since they provide efficient MAC-schemes with strong provable security. The final design of UIA2 combines the standard GMAC computation with a final processing step to obtain maximal security for a 32-bit shortened (truncated) MAC even if the hashing operations take place in GF(2^{64}).

9 Algorithm evaluation

9.1 Evaluation criteria

9.1.1 General principles

Evaluation will be done by members of the project team as described in ref. [3].

The evaluation must confirm that operational, security and formal requirements are fulfilled.

Analysis and findings from the design phase shall be available to all members of the project team.

All members of the project team shall approve the final deliverables.

Evaluation work should take maximal advantage of published results on known components and constructions.

9.1.2 Implementation aspects

Evaluation must ensure that the complexity requirements from ref. [2] are met.

Evaluation must ensure that the performance requirements from ref. [2] are met. Note: The target throughput shall be 10 Mbit/s at clock speed 20 MHz.

Correctness of specification and test data shall be confirmed by two independent implementations.

9.1.3 Mathematical evaluation

Criteria for mathematical evaluation shall be tailored towards the actual design.

Basic components such as S-boxes and combining functions shall be analysed with respect to algebraic and statistical properties.

Core primitives such as block and stream ciphers shall be evaluated according to established cryptographic principles and resistance against known attacks.

Final constructions of UEA2 and UIA2 shall be evaluated according to standard security notions. There should be no efficient test enabling an adversary to distinguish the UEA2 construction from a truly random generator. The integrity algorithm should resist existential forgery by an adaptive adversary.

The operational context of UEA2 and UIA2 will be as much as possible taken into account in the analysis.

9.1.4 Statistical evaluation

If needed, statistical test shall be conducted on core components and the f8/f9 functions to ensure that pseudorandom properties are present.

9.1.5 IPR investigations

Evaluation must ensure that no IPR regulations will limit the use of UEA2 and UIA2 within its intended scope of 3G.

9.2 Principles of algorithm evaluation

The evaluation work performed by the extended task force was divided into three different parts. The main investigations in each part are described in this section. Due to the fact that SNOW 3G is based on a well-known and analysed algorithm for which no statistical weaknesses was known, it was agreed to focus the available resources on the mathematical evaluation. In-depth statistical testing of the algorithm was not done since any weakness or bias found would most likely be traced to corresponding problems in SNOW 2.0.

9.2.1 Analysis of various components of SNOW 3G

SNOW 3G consists of two interacting modules, the LFSR and the FSM. The evaluation must analyse the cryptographic, statistical and algebraic properties of the following components:

- The shift register
- The FSM as a whole
- The two S-boxes of the FSM
- The combining functions

9.2.2 Analysis of SNOW 3G as a stream cipher

SNOW 3G is designed as a modern synchronous stream cipher, and evaluation should ensure that the cipher achieves the necessary properties with respect to pseudorandom properties of the generated keystream and resistance against known attacks. Special attention was taken towards the following attacks:

Algebraic attacks: Is it possible to express (or approximate) the non-linear component (or a multiple of the function) of the cipher as a multivariate polynomial of low algebraic degree?

Guess and determine attacks: Is it possible to guess some internal values of the cipher state, and then to determine other internal values using the mathematical relations between internal values and generated keystream?

Linear distinguishing attacks: Is it possible to distinguish the output from the cipher from a truly random sequence of the same length using a mask based on linear approximation of the non-linear component?

9.2.3 Analysis of the encryption and integrity modes

This part of the analysis consists of investigating the strength of constructions used for deriving the UEA2 and UIA2 algorithms from the SNOW 3G stream cipher – in order to make sure that the UEA2 and UIA2 construction do not substantially deviate from the following ideal requirements:

- UEA2 : There should be no efficient test enabling an adversary to distinguish the UEA2 algorithm (as seen as a pseudorandom function generator associating a key with a mapping from the IV set to the output sequences set) from a truly random function generator.

- UIA2 : The integrity algorithm should resist existential forgery by an adaptive adversary, i.e. it should be computationally infeasible for an adversary to infer any additional MAC value of an N+1th message from a set of N adaptively obtained MAC values corresponding to N messages.

The operational context of use of the UEA2 and UIA2 algorithms (repetition of IV values, redundancy of the plaintext, etc.) should be taken into account in the analysis.

9.3 Security principles of UEA2 and UIA2

9.3.1 Supporting arguments for the UEA2 construction

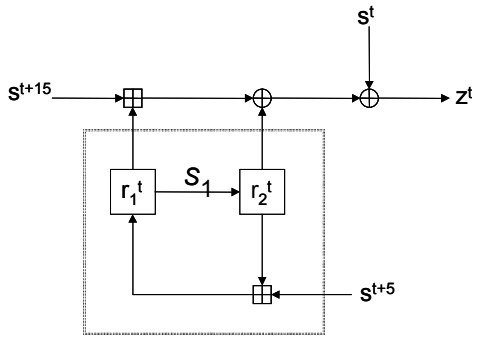
The goals in designing SNOW 3G were in essence:

- to improve on the resistance of SNOW 2.0 to algebraic attacks,
- to retain all the good security properties of SNOW 2.0, and
- to make only a very slight increase in implementation complexity.

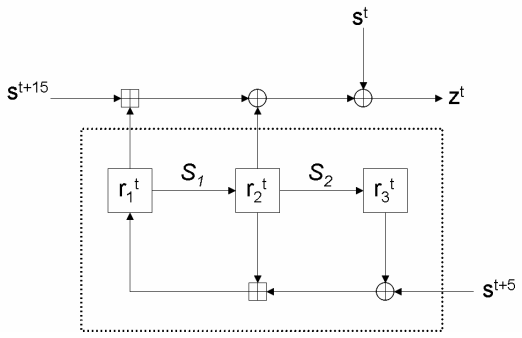
We therefore chose a minimal change to the SNOW 2.0 architecture, but sufficient to strengthen the algorithm against the concerns highlighted by Billet and Gilbert, ref. [11].

9.3.1.1 The architectural change from SNOW 2.0 to SNOW 3G

In SNOW 3G we introduce a third memory element into the SNOW 2.0 nonlinear combiner.



If we simplify SNOW 2.0 by replacing mod 2^{32} addition by XOR, we have at each time clock *two* linear equations involving the *two* nonlinear register words, the (assumed known) keystream and the (fixed length) initial linear register state. Hence (ref. [11]) we can eliminate all intermediate values of r_1 and r_2, and express r_2^t as a linear combination of initial linear register state words, known keystream words, and the initial value r_2^0.



Replacing mod 2^{32} addition by XOR, we have at each time clock *two* linear equations involving the *three* nonlinear register words, the (assumed known) keystream and the (fixed length) initial linear register state:

$$z^t = s^t \oplus s^{t+15} \oplus r_1^t \oplus r_2^t$$

$$r_1^{t+1} = r_2^t \oplus r_3^t \oplus s^{t+5}$$

The elimination of intermediate values is no longer possible.

9.3.1.2 Design criteria for the new S-box ensemble *S2*

If S is an S-box, where $(y_0, \ldots, y_{n-1}) = S(x_0, \ldots, x_{m-1})$, we say that S has *algebraic immunity r* if there is no degree r equation in the variables $\{x_0, \ldots, x_{m-1}, y_0, \ldots, y_{n-1}\}$ that holds for all values of the input vector $(x_0, \ldots, x_{m-1})$. Algebraic attacks tend to be possible when the algebraic immunity of a cipher's non-linear components is low. For instance, the AES S-box has algebraic immunity only 1 — there are quadratic relations involving the input and output bits.

Henceforth we write "AI" for "algebraic immunity".

In SNOW 3G, the S-box ensemble $S1$ is identical to that used in SNOW 2.0. It was carefully selected to provide strong security against, for instance, differential and linear attacks. However, its AI is only 1 since it is built from the AES S-box and so the resistance it provides against algebraic attacks is not as high as it might be.

The S-box ensemble $S2$ was built from four copies of a new 8-bit to 8-bit S-box S_2. Since the only function of the third register element is to provide resistance to algebraic attacks the new S-box does not necessarily need to provide resistance against other forms of attack. It doesn't need to be "strong" in all the senses usually associated with an S-box and we don't necessarily mind if it has relatively strong linear or differential characteristics. Even rather strong correlation between input bits and output bits, or linear combinations of input bits and output bits are unlikely to be critical factors. There are few criteria for the selection of S_2:

- it should have a higher AI than $S1$;
- it should not lead to a design that was weaker than SNOW 2.0, and
- it should be relatively "cheap" to implement.

An initial version of the S-box S_2 concentrated on trying to achieve a high algebraic immunity. However this particular choice of S_2 had a very negative effect on SW performance and it was decided to modify the construction. Some analysis by the task force showed that the computation for the $S2$ ensemble was taking as much as 93% of the CPU load. A new version of S_2 was therefore designed with all four instances of the S-box S_2 being combined with the same MixColumn operation that was used to construct the S-box ensemble $S1$ out of the 8-bit to 8-bit S-box S_1.

9.3.2 Rationale behind the Construction of UIA2

The integrity function UIA is based on the principles of *universal* and *strongly universal* hash functions introduced by Wegman and Carter in their seminal papers [13] and [32]. They show that there is a direct correspondence between strongly universal hash functions and unconditionally secure authentication codes. The use of universal hash functions as a tool for building a secure Message Authentication Code (MAC) has recently received much attention within the cryptographic community due to its nice security properties and possibilities for efficient implementation, see ref. [22], [25], and [28].

For many reasons, Galois MAC construction was considered an attractive solution for the new message integrity function UIA. The only problem was that the forgery probability grows as the messages get longer. The message length in the UMTS system is currently upper-bounded by 20 000 bits, but longer messages may need to be allowed in the future. Using the upper bound of 2^{16} for the message length, computing in GF(2^{64}) and truncating to 32-bit MAC, the forgery probability becomes 2^{-22}. This means an unacceptable degradation in security compared to other more traditional computationally secure MAC constructions. Note that the forgery probability is only doubled, if the MAC is computed in the Galois field GF(2^{32}). With the current construction, the higher security offered by the larger field is lost in truncation.

This avoid this drawback UIA is using a two-stage MAC construction. The idea is to concatenate two universal hash families as described by Stinson [29], Bierbrauer et al [10] and Nevelsteen and Preneel [25] and recently referred to as "secure truncation" by D. Bernstein, ref. [9].

9.3.2.1 Theoretical background

Let us recall the definitions of ε-AXU and ε-AU hash families.

Definition: [25] Let $H = \{h_k : A \rightarrow B, k \in K\}$ be a family of hash functions mapping elements of set A to set B. The family H is ε-Almost-Xor-Universal **(ε-AXU)** if,

$$\forall\, x, x' \in A, \quad x \neq x' \quad \Rightarrow \quad \forall \delta \in B, \quad \mathbf{Pr}_k \{h_k(x) \oplus h_k(x') = \delta\} \leq \varepsilon.$$

If the condition holds only for $\delta = 0$, that is,

$$\forall\, x, x' \in A, \quad x \neq x' \quad \Rightarrow \quad \mathbf{Pr}_k \{h_k(x) \oplus h_k(x') = 0\} \leq \varepsilon,$$

the family H is called ε-Almost-Universal **(ε-AU).**

To compute a 32-bit MAC UIA adopts a two-stage construction. In the first phase we work over GF(2^{64}) and we use an ε_1-AU hash function family with longer hash codes. In stage two we use an ε_2-AXU hash function family, with shorter hash codes.

There is a composition theorem due to Stinson, ref. [10] and [29], which allows different universal hash function families to be combined. The most useful result for us is the following:

> *If there exists an ε_1-AU family H_1 of hash functions from A to B and an ε_2-AXU family H_2 of hash functions from B to C, then there exists an ε-AXU family H of hash functions from A to C where $H = H_1 \times H_2$, and $\varepsilon = \varepsilon_1 + \varepsilon_2 - \varepsilon_1\varepsilon_2 \leq \varepsilon_1 + \varepsilon_2$.*

9.3.2.2 The practical instantiation in UIA2

Many practical instantiations of a two-stage MAC can now be constructed based on the various constructions of ε-AXU and ε-AU hash families known from literature. The construction in UIA2 authentication function makes its forgery probabilities close to ideal. For the UIA MAC construction, we use an $L \cdot 2^{-64}$-AU hash function family in the first stage, where L is the length of the message in 64-bit blocks (after padding and length appending). In the second stage we have a 2^{-32}-AXU hash function family and so our two-stage construction provides an $(L \cdot 2^{-64} + 2^{-32})$-AXU hash function family, that is, an 2^{-31}-AXU hash function family.

Suppose that the message to be authenticated (after appropriate formatting) consists of L 64-bit blocks $M_{L-1}, \ldots, M_1, M_0$. Given a 64-bit quantity k, a 64-bit intermediate tag T_{in} is computed using a $L \cdot 2^{-64}$-AU hash family as

$$T_{in} = M_{L-1} k^{L-1} + \ldots + M_1 k + M_0 \text{ over GF}(2^{64}).$$

Given a second 64-bit quantity λ the 32-bit message authentication tag is computed by computing first the product $\lambda \cdot T_{in}$ over GF(2^{64}), and truncating the result to the least significant 32 bits. This is an instantiation of a 2^{-32}-AXU hash family (secure truncation), see Lemma 10 of [10]. The authentication tag is obtained by xor-ing this result with a 32-bit one time pad. For this construction the worst case forgery probability is bounded by $L \cdot 2^{-64} + 2^{-32} \approx 2^{-32}$ for messages with length up to 2^{38} bits. The message length in the UMTS system is currently upper-bounded by 20 000 bits, which is well below this limit.

9.4 Mathematical analysis of SNOW 3G

During the design of SNOW 3G an important aim was to remain close to the original SNOW 2.0. In this way it is hoped that much of the trust already invested in SNOW 2.0 can be directly inherited within SNOW 3G. Consequently SNOW 3G shares many features with SNOW 2.0 with the only significant difference between the two being the inclusion of an additional register R3 within the Finite State Machine (FSM).

The inclusion of the set of S-boxes S_2 and the register R3 is not expected to diminish the security of the SNOW 2.0. Such a negative interaction would be highly remarkable. Instead the new register is primarily intended to provide additional resistance to attacks that might strive to exploit the algebraic foundations of the cipher [11].

9.4.1 Properties of components

SNOW 3G consists of two major building blocks, the LFSR and the FSM. The generated keystream is a result of the combined operation of these two components, and the security of the cipher relies on careful design of both parts.

9.4.1.1 The Linear Feedback Shift Register (LFSR)

The LFSR consists of 16 stages of 32 bits each giving a total of 512 bits. The feedback polynomial is given by

$$f(x) = \alpha x^{16} + x^{14} + \alpha^{-1} x^5 + 1$$

defined over GF(2^{32}). α is again a root of $x^4 + \beta^{23}x^3 + \beta^{245}x^2 + \beta^{48}x + \beta^{239}$ in GF(2^8) where β is a root of the binary polynomial $x^8 + x^7 + x^5 + x^3 + 1$. Since $f(x)$ is a primitive polynomial the LFSR will generate a maximal sequence with period $2^{512} - 1$. The polynomial has been carefully selected to offer a good trade-off between performance and security. Considered bitwise, the feedback polynomial has a weight of 251, and the evaluation has not been able to find any multiples of low weight.

9.4.1.2 S-box S_1

The substitution box $S1$ is based on the bytewise substitution of Rijndael, ref. [24], combined with the MixColumn operation of Rijndael. The properties of the AES S-box are well-known. The best linear approximation has a bias of 2^{-4} and the best differential characteristic has a probability of 2^{-6}. Combined with the good diffusion properties of the MixColumn operations, the $S1$ construction offers good protection against many known attacks.

Due to its algebraic structure, it is also well known that there are 39 algebraic equations in the eight input and eight output bits. This fact means that the algebraic immunity of the S-box is limited to 1, resulting in the possibility of algebraic attacks on this component.

9.4.1.3 S-box S_2

The new S-box $S2$ consists of four bytewise substitutions followed by the MixColumn operation of Rijndael. The starting point for the 8-to-8-bit S-box is a specific permutation polynomial over GF(2^8), defined by the equation

$$g_{49}(x) = x + x^9 + x^{13} + x^{15} + x^{33} + x^{41} + x^{45} + x^{47} + x^{49}.$$

This is an example of a Dickson polynomial, ref. [17], which offer AI = 2, nonlinearity 96, that is, bias 2^{-3}, and the maximum differential probability is 2^{-5}. The use of this function was proposed to the design team by Enes Pasalic [27]. In order to avoid short cycles, the complete definition of the new S-box is based on the permutation polynomial $g_{49}(x)$ to which is added the constant term 0x25 using $x^8 + x^6 + x^5 + x^3 + 1$ (= 0) as the irreducible polynomial for the GF(2^8) representations. It has been verified that this results in a permutation with shortest cycle length 10. Other tests on S_2 have established that there are no annihilators of degree less than or equal to two.

9.4.1.4 Combining functions

The non-linear combing function of SNOW 3G is addition modulo 2^{32}. This operator has been used in many encryption algorithms, and the binary representation of the component functions have algebraic degrees that increases with 1 for every component. However, the algebraic immunity will never be higher than 2, and the resistance against algebraic attacks will not grow with the word size. It can also be shown that there are many linear approximations to the bitwise component functions that have high bias.

9.4.2 Resistance against attacks

The resistance of SNOW 2.0 to standard stream cipher cryptanalysis is well established. During the design of SNOW 3G particular focus was paid to the distinguishing attacks of Watanabe *et al,* ref. [31], and the algebraic attacks of Billet and Gilbert, ref. [11].

9.4.2.1 Distinguishing attacks.

In ref. [26], the linear distinguishing attack on SNOW 2.0 is improved by showing that there is linear approximation relation over the terms of the keystream of SNOW 2.0 with bias $2^{-86.89}$. This is significantly larger than the bias $2^{-107.26}$ of the best linear approximation found by Watanabe *et al* whose estimate of the bias actually was $2^{-112.25}$.

Both external evaluation teams considered the linear distinguishing attack by Watanabe *et al* as one of the known attacks that might be applied to SNOW 3G. The evaluators from Leuven considered the attack infeasible, while recognising that they did not have possibility to perform thorough analysis of this cryptanalytic method. The ENS team derived linear approximate relation of SNOW 3G by constructing an equation system and eliminating the intermediate variables.

It can be observed, as shown in Figure 4, that there is a linear mask of the FSM of SNOW 3G, which involves outputs from three subsequent rounds only.

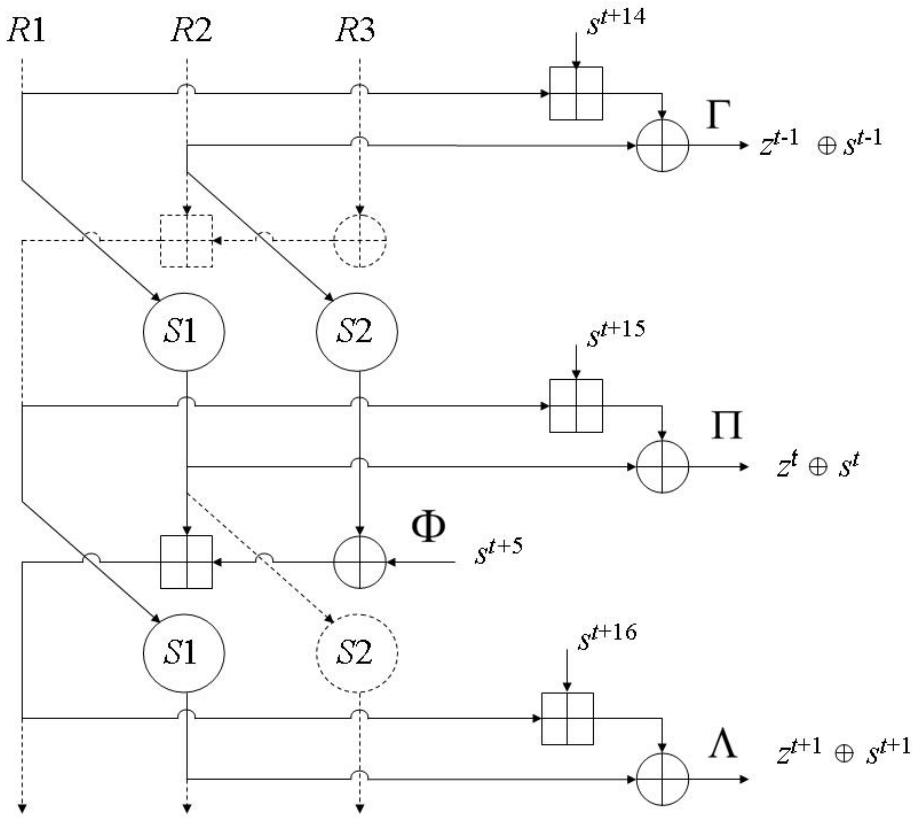


Figure 4: Linear masking of SNOW 3G over three rounds

Then, by making use of the linear recursion of the LFSR sequence the following linear approximate relation can be derived for the terms of the keystream sequence:

$$\begin{aligned} &\Gamma \cdot (z^{t+15} \oplus z^{t+1}) \oplus [\Gamma\alpha] \cdot z^{t-1} \oplus [\Gamma\alpha^{-1}] \cdot z^{t+10} \\ &\oplus \Pi \cdot (z^{t+16} \oplus z^{t+2}) \oplus [\Pi\alpha] \cdot z^{t} \oplus [\Pi\alpha^{-1}] \cdot z^{t+11} \\ &\oplus \Lambda \cdot (z^{t+17} \oplus z^{t+3}) \oplus [\Lambda\alpha] \cdot z^{t+1} \oplus [\Lambda\alpha^{-1}] \cdot z^{t+12} \end{aligned} \qquad (1)$$

Here we denote by $[\Gamma\alpha]$ the mask for which the following holds:

$$[\Gamma\alpha] \cdot x = \Gamma \cdot \alpha x, \text{ for all } x \in GF(2^{32}),$$

where the product αx, for $\alpha, x \in GF(2^{32})$, is computed in $GF(2^{32})$.

The bias of the linear approximate relation (1) is computed as follows. First the bias of the linear masking system of the FSM depicted in Figure 4 is computed with the mask triplet Γ, Π, Λ. This bias value is denoted by b_1. Then the mask triplet is replaced by $[\Gamma\alpha]$, $[\Pi\alpha]$, $[\Lambda\alpha]$, and the resulting bias is denoted by b_α. Similarly, a third mask triplet $[\Gamma\alpha^{-1}]$, $[\Pi\alpha^{-1}]$, $[\Lambda\alpha^{-1}]$ is used to compute a bias value $b_{\alpha'}$. Finally, the total bias value b is estimated using the *piling-up* lemma as $b = 8\, b_1^2\, b_\alpha\, b_{\alpha'}$.

In the three-round case the output mask of the second instance of S1 must take all three values: Λ (twice), $\Lambda\alpha$ and $\Lambda\alpha^{-1}$**.** Due to the diffusion properties of the MixColumn transformation one can show that the minimum number of S-boxes is at least seven. The S-boxes of S1 are the AES S-boxes with bias at most 2^{-4}. The new S2 function has the same structure as S1 except that the AES S-boxes are replaced by nonlinear S-boxes with bias at

most 2^{-3}. We also observe that the output mask must also take three values: Φ (twice), $\Phi\alpha$ and $\Phi\alpha^{-1}$, as the same mask shall be applied to the LFSR input s^{t+5}. Finally, whatever input and output masks for the second instance of $S1$ are used, at least four active S-boxes are need. Hence there are always at least 11 active AES S-boxes and seven active S_2 S-boxes, giving an upper bound of

$$b \leq 2^{16}(2^{-4})^{11}(2^{-3})^{7} = 2^{-49}$$

to the bias b of the three-round linear approximation of SNOW 3G. It is unclear how tight this bound might be. Let us just mention that the corresponding upper bound to the two round linear approximation of SNOW 2.0 would be $2^{6}(2^{-4})^{7} = 2^{-22}$ while the best known bias of a linear approximate relation for SNOW 2.0 has bias 2^{-87}.

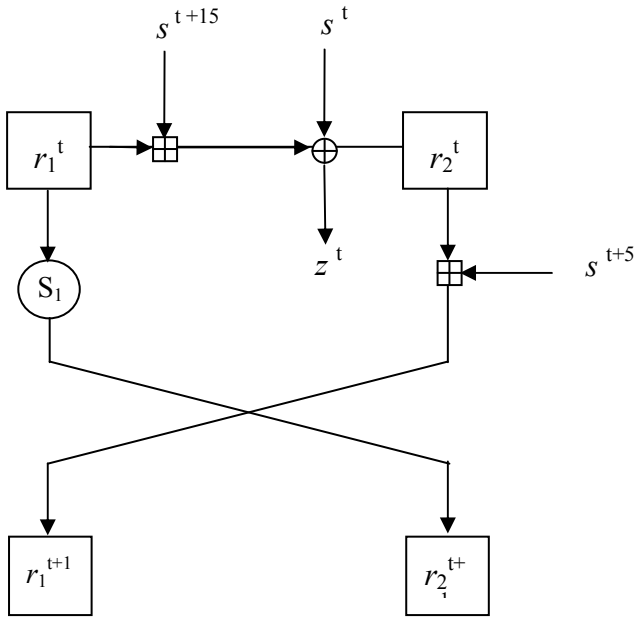
9.4.2.2 Algebraic attacks.

It is shown, ref. [11], that the simplified version of SNOW 2.0, where additions modulo 2^{32} are replaced by xor (named SNOW$^{\oplus}$ 2.0 in the sequel), is vulnerable to a simple algebraic attack of complexity about 2^{50} operations. The actual SNOW 2.0 can be captured by an overdefined system of quadratic equations. However no efficient method to solve this system faster than exhaustive search is known to exist.

Here we give some evidence that the simplified version of SNOW 3G named SNOW$^{\oplus}$ 3G, where additions mod 2^{32} are replaced by an exclusive or, is significantly more resistant to algebraic attacks than SNOW$^{\oplus}$ 2.0. No realistic algebraic attack SNOW$^{\oplus}$ 3G appears to be feasible except for a few bad choices for S-box S_2 which are easy to avoid. Since it is natural to conjecture that the complexity of the best algebraic attacks against SNOW 3G is strictly higher than the complexity of the best algebraic attacks against SNOW$^{\oplus}$ 3G, there is strong evidence that SNOW 3G is further out of reach of algebraic cryptanalysis methods than in the case of SNOW 2.0.

Our claim is supported by the two following informal arguments. In the first argument, we assume that the algebraic immunity of additional S-box S_2 used in SNOW 3G is larger or equal to 1, i.e. larger or equal to the algebraic immunity of S-box S_1. In the second argument, we assume that the algebraic immunity of S-box S_2 is larger or equal to 2.

1: If the algebraic immunity of S-box S_2 is at least 1, then, except for a few bad choices of S_2, knowledge of the keystream does not allow us to express all the SNOW$^{\oplus}$ 3G memory bits as known linear combinations of the initial state variables s^0 to s^{15} and/or the initial FSM memory state, as for SNOW$^{\oplus}$ 2.0. This difference comes from the fact that the ratio between the amount of keystream output at each time clock and the FSM memory size is smaller in the case of SNOW$^{\oplus}$ 3G, namely one third instead of one half. The SNOW$^{\oplus}$ 2.0 case is described in Figure 5. At each time clock, two linear equations (i) and (ii) relating the memory words, the initial state variables and the observed keystream words z^t can be derived.



SNOW$^{\oplus}$ 2.0 is governed by the following linear equations:

$$z^t = r_1{}^t \oplus s^{t+15} \oplus s^t \oplus r_2{}^t \qquad \text{(i)}$$

$$r_1{}^{t+1} = r_2{}^t \oplus s^{t+5} \qquad \text{(ii)}$$

and by the following non-linear equations:

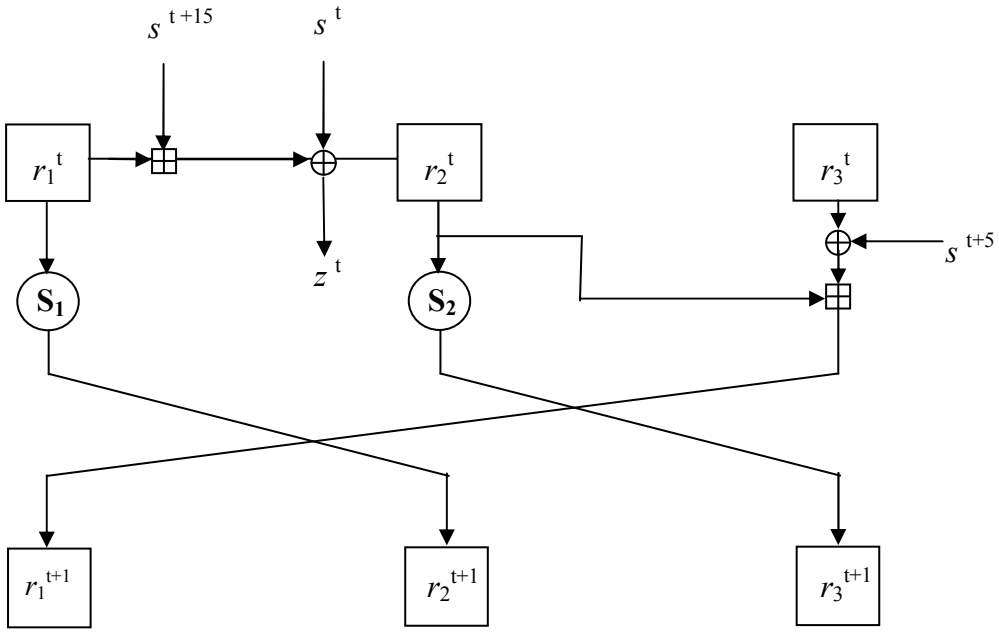
$r_2{}^{t+1} = S_1(r_1{}^t)$ and the resulting quadratic equations (iii)

Figure 5: SNOW 2.0 and the SNOW $^{\oplus}$ 2.0 equations

This allows us to eliminate two intermediate memory words per time clock in the equations relating the initial LFSR and FSM state and the current memory state. The SNOW$^{\oplus}$ 3G case is described in Figure 6. At each time clock, two linear equations (1) and (2) that relate the memory words, the initial state variables, and the observed keystream words z^t can be derived. But this is not sufficient to eliminate all the three intermediate memory words introduced at each time clock. Moreover, since S-boxes S_1 and S_2 have algebraic immunity at least 1, none of the equations associated with S_1 and S_2 is linear.

In order to show that a few (somewhat artificial) bad choices for S_2 that provide a third linear equation do nevertheless exist, let us assume that there exists a non zero triplet of linear mappings L_1, L_2 and L such that $L_1\, S_1^{-1} \oplus L_2\, S_2 \oplus L = 0$.

Due to relations (3) and (4), which imply $r_1{}^{t-1} = S_1^{-1}(r_2{}^t)$ and $r_3{}^{t+1} = S_2(r_2{}^t)$ for all t, the additional linear equation $L_1\, r_1{}^{t-1} \oplus L_2\, r_3{}^t \oplus L\, r_2{}^t = 0$ now holds, thus potentially leading to the same kind of linearization attacks as for SNOW$^{\oplus}$ 2.0. The situation where such linear mappings exist can however be easily avoided, and one can even show that if S_2 has algebraic immunity at least 2, then there exists no (L, L_1, L_2) triplet of linear mappings satisfying $L_1\, S_1^{-1} \oplus L_2\, S_2 \oplus L = 0$ and the additional condition that L_1 and L_2 be invertible.



SNOW$^{\oplus}$ 3G is governed by the following linear equations:

$$z^t = r_1{}^t \oplus s^{t+15} \oplus s^t \oplus r_2{}^t \quad (1)$$

$$r_1{}^{t+1} = r_2{}^t \oplus r_3{}^t \oplus s^{t+5} \quad (2)$$

and by the following non-linear equations:

$$r_2{}^{t+1} = S_1(r_1{}^t) \quad \text{and the resulting quadratic equations} \quad (3)$$

$$r_3{}^{t+1} = S_2(r_2{}^t) \quad \text{and the resulting equations of degree } \mathrm{AI}(S_2)+1 \quad (4)$$

Figure 6: SNOW 3G and the SNOW $^{\oplus}$ 3G equations

In summary, it seems that outside a few exceptional cases such as the one described above, there is, an "accumulation effect" in the SNOW$^{\oplus}$ 3G FSM equations, i.e. the degree of the equations relating the FSM state at time t, the initial LFSR and FSM state and the keystream bits increases as t grows, and does not stay equal to 1 as in the case of SNOW$^{\oplus}$ 2.0.

2: If the algebraic immunity of S-box S_2 is at least 2, then the system represented by equations (1) to (4) in the initial 32 (16+3) = 608 initial LFSR and FSM state bits and some extra FSM state variables has the following property (that the system of equations associated with SNOW $^{\oplus}$ 2.0 does not have): if one only considers those of the equations which degree is at most two, one does not get an overdefined system of equations. Thus in order to get an overdefined of equations, one must incorporate equations of degree strictly larger than 2. This property is due to the following reasons:

- Since S_2 has algebraic immunity at least two[1], none of the equations resulting from (4) has degree at most 2, so the only available equations of degree at most 2 are equations (1), (2), and (3).

- However, the quadratic system associated with equations (1), (2), and (3) is highly underdefined. Given any keystream sequence (z^t) it easy to extend any arbitrary initial LFSR state value and any arbitrary initial memory value r_2^0 to a solution of the quadratic system associated with equations (1), (2) and (3): as a matter of fact, (1) provides r_1^0 and it suffices then to successively apply , for consecutive values of t, equation (3) in order to derive r_2^{t+1} from r_1^t, equation (1) in order to derive r_1^{t+1} from r_2^{t+1}, and finally equation (2) in order to derive r_3^t from r_2^t and r_1^{t+1}.

This clearly indicates that the system of available equations of degree at most 2 does not allow us to recover the initial and FSM states of SNOW$^{\oplus}$ 3G. Therefore, equations of degree at least 3, resulting form (4) have to be taken into account. We thus get a system of degree at least 3 in the initial 32 (16+3) = 608 initial LFSR and FSM state bits and some extra state variables. Even if one applies efficient computational algebra methods to solve this system, one can expect the resulting complexity to be greater or equal to the complexity of solving a full rank linear system in all the $\binom{608}{3} \approx 2^{25}$ monomials of degree 3 in the initial LFSR and FSM state bits. Therefore, one can expect a complexity of at least 2^{57} (and probably a much higher complexity) for any algebraic attack against SNOW$^{\oplus}$ 3G. This gives some additional confirmation that SNOW$^{\oplus}$ 3G offers much better resistance to algebraic attacks than SNOW$^{\oplus}$ 2.0 and that SNOW 3G appears to be out of reach of realistic algebraic attacks.

9.5 Implementation attacks

Among all attacks against cryptographic algorithms, we focus on so-called side-channel attacks, or 'implementation attacks', i.e. attacks which make use of an additional (physical) information channel through which secrets might leak. These additional channels include, but are not limited to, timing measurements, power consumption, electromagnetic radiation, etc.

9.5.1 Evaluation of SNOW 3G

SNOW 3G has been designed for use as the base algorithm for the second set of 3GPP confidentiality and integrity algorithms. As most symmetric key algorithms, it uses simple linear operations such as byte or word shifts and exclusive or operations, modular addition as well as substitutions through non linear 32 by 32 bit S-boxes. It also heavily uses polynomial evaluation over GF(2^{32}) during the linear feedback shift register computations.

SNOW 3G is vulnerable to power analysis and timing attacks. Some of these attacks may be thwarted by using adequate hardware protections on the mobile equipment. Others seem far less relevant in the 3GPP context of use and are therefore discarded.

[1] An even more demanding condition, which would strengthen the argument given here, would consist of requiring that the single equations of degree at most 2 relating the variables of the triplet (x, y = S1(x), z=S2(y)) be those quadratic equations relating the variables of the pair (x, y = S1(x)).

9.5.1.1 Timing Attacks

In the case of SNOW 3G and the Carter-Wegman MAC computation approach, the most vulnerable time-dependent operations are the polynomial evaluations over $GF(2^{32})$ and $GF(2^{64})$. In the 3GPP context, these operations are implemented in hardware on the mobile equipment, thus timing measurements seized by a remote attacker may not be all that meaningful in practice.

Nevertheless, if necessary, care should be taken to ensure that these operations are implemented in constant time using operations that are the same, whatever the intermediate result. In particular, no conditional instructions depending on internal values should be used. The same operations need to be performed independently of the intermediate results of the inner computation.

In this way, efficient protection against timing attacks can be achieved.

9.5.1.2 Power Analysis Attacks

Every cryptographic algorithm is potentially vulnerable to Simple or Differential Power Analysis. In a very simple approach, an attacker monitors the power consumption curves of the device while executing cryptographic operations on different input data. In Simple Power Analysis, the overall form of the curve reveals which actual data are being manipulated, thus leaking information on the secret key. For Differential Power Analysis, the attacker defines a selection function, which consists for example in one output bit of a non-linear S-box and which depends on some known input data and a portion of the secret key which he is trying to guess. The correct guess is confirmed if the partition of power curves obtained by sorting them according to the value of the output bit reveals a common pattern in all curves in a given set. Otherwise, the curves are randomly sorted and no information can be retrieved.

In the case of SNOW 3G, the model for such attacks is that the mobile equipment and the USIM are not directly available to the attacker, therefore power measurements seem out of scope in this context. If the attacker has physical access to the mobile device, the card will just as simply reveal the secret session keys to him since it reveals them to the mobile equipment, which does not provide tamper resistance per se. Therefore, there are simpler ways to extract the keys when access to the device is made possible.

Similarly to Power Attacks, Electromagnetic radiation attacks also require the attacker to have direct access to the handset, therefore making them unrealistic in the 3GPP context of use.

9.5.2 Conclusion on implementation attacks

For UEA2 and UIA2 algorithms, the context does not allow for advanced implementation attacks. The only attack which may seem relevant here is when the attacker does not need to have physical access to the mobile equipment. Since the algorithms are implemented in hardware, it seems rather unlikely that such attacks may be executed amidst the noise surrounding a typical communication. Nevertheless, if timing attacks are of concern, simple constant operation computations will provide for efficient protection against such attacks.

9.6 Results from complexity evaluation

Throughout the development of the UEA2 and UIA2 algorithms, the task force has investigated the performance of the design proposal with respect to performance and complexity requirements. See section 6 in this document for more details on the requirements.

A high level requirement would be that the new algorithm suite should be similar to the old KASUMI-based algorithms with respect to complexity and performance.

9.6.1 KASUMI HW performance

For different optimizations of Kasumi, the HW complexity in a Mitsubishi 0.18 micron CMOS standard cell technology has been reported by the following table:

	KASUMI (1)	KASUMI (2)	KASUMI (3)
Size	9916 Gates	6107 Gates	3698 Gates
Maximal Clock	25.8 MHz	36.2 MHz	88.5 MHz
Maximal Speed	412 Mbps	290 Mbps	101 Mbps
Latency	64bits / 4 cycles	64 bits / 8 cycles	64 bits / 56 cycles
Key Setup Time	0 cycles	8 cycles	8 cycles

9.6.2 SNOW HW performance

For SNOW-based algorithms similar implementation efforts give the following results:

SNOW 2.0 and variant

	Size	Delay	Speed
SNOW 2.0	7128 Gates	19.31 nsec	1.66 Gbps
SNOW 2.0 with random S_1	8502 Gates	19.48 nsec	1.64 Gbps

SNOW 3G with different S2-boxes

	Size	Delay	Speed
SNOW 3G with $S_2 = S_1$	9498 Gates	18.57 nsec	1.72 Gbps
SNOW 3G with random S_2	10918 Gates	18.57 nsec	1.72 Gbps
SNOW 3G with final S_2	10908 Gates	18.57 nsec	1.72 Gbps

These tables show that the HW-complexity of SNOW 3G is comparable to that of KASUMI, and the throughput speed is significantly better. We also note that the actual choice of S2 has little impact to the speed. In fact, the critical (speed dominant) path of SNOW 2.0/SNOW 3G and variants is the route S_{t+15} -> ADD -> XOR -> XOR -> Z_t, which does not pass $S1$ or $S2$. This is the reason why all variants above run in almost the same speed.

During the design and analysis phase, a great many different S-box ensembles were tested with the final $S2$ offering the best compromise of design simplicity, security, and performance.

9.6.3 UIA2 complexity

Using standard techniques for Galois field multiplication, it has been estimated that a single GF(2^{64}) is about 8% of the cost of the AES-128 pipeline. From this we can conclude that performance of the GMAC-based UIA2 computation should be superior to the KASUMI-based UIA1.

9.6.4 SW performance

Even if hardware implementation is the preferred technology for realisation of the UMTS encryption/integrity algorithms, some manufacturers are using SW implementations, in the terminals and/or in the RNC units. Initial implementations suggest that SNOW 3G can be expected to provide a raw encryption speed that is at least 80% that provided by SNOW 2.0. This estimate has been confirmed by the task force compiling the example code using two different compilers, ggc (optimization –O3) and icc.

Ref. [20] includes a section discussing implementation aspects of the SNOW 2.0 algorithm. The report shows that SNOW 2.0 can be implemented with a key setup of 937 clock cycles using 18 cycles per word for keystream generation. This results in a typical encryption speed more than 3Gbits/sec on a Intel Pentium 4 running at 1.8GHz.

Based on these findings it is clear that UEA2 is well suited for software implementation, is faster than the current UEA1, and meets the implementation requirements given in ref. [2].

The software performance for Galois-based MACs is discussed in ref. [22] where a software implementation of GMAC is reported to process 1500-byte packets using 10.2 cycles per byte. The performance of GMAC in software is typically several times faster than other standard MAC constructions.

9.7 Results from independent evaluation

After the final design decisions had been made within the SAGE group, contracts were made with two external evaluators. Their task was to conduct an independent evaluation of the design proposals, with respect to security and performance issues. The external evaluations were undertaken over a three months period from September to November 2005, and the results of this work are provided in ref. [12] and [21]. In this section we include the main conclusions from the work as well as the task force responses to the issues raised in the reports.

9.7.1 Evaluator 1

The report from evaluator 1 gives a detail background of the SNOW stream cipher family, discussing design strategies and published attacks on SNOW 1.0, SNOW 2.0, and

Sosemanuk. The latter is a new stream cipher submitted to eStream, providing performance very similar to SNOW 2.0.

9.7.1.1 Extracts from evaluation report

The evaluators describes their detail analysis of SNOW 3G. They confirmed the algebraic immunity of the then S_2 as well as noting several structural features of that S-box. The team also made software implementations of SNOW 3G confirming that the new S-box resulted in a performance that was 15 to 70 times slower than SNOW 2.0. This S-box has since been replaced.

The security assessment of the structure of SNOW 3G involved analysis of the cipher against the following class of attacks:

- Algebraic attacks,
- Guess-and-determine attacks,
- Distinguishing attacks based on linear approximations, and
- Initialization attacks based on differential cryptanalysis and collision attacks.

For each type of attacks the evaluator tried to use state-of-the art cryptanalysis to mount an attack against SNOW 3G. Within the time spend on this analysis it was not possible to find attacks offering better performance than an exhaustive search for the 128 bit key.

For attacks against the modes of operations UEA2 and UIA2, the external evaluation confirms that any attack against UEA2 that could give one bit of plaintext information can be translated to a distinguishing attack on SNOW 3G. Similar arguments are given for UIA2.

9.7.1.2 Evaluation summary

The conclusions of the report were as follows:

- "we did not find any plausible attack against SNOW3G and we believe that SNOW 3G is secure and adequately addresses all potential imperfections found in the analysis of SNOW 2.0,
- the addition of a third register in the FSM is a very successful design idea, that plays a key role in the strengthening of the security,
- the S2 function involves a rather high cost in terms of performance; furthermore, it is unclear that raising the algebraic immunity of S2 at a higher level than 2 results in a significant improvement of the security. Also, if S2 ever gets changed, we would favor the choice of a permutation, that increases the resistance of the key mixing phase."

9.7.1.3 Task force response

The concern raised by Evaluator 1 on the choice of S2 was taken into account by the task force resulting in the redesign of S_2 reported in section 9.3.1.2.

9.7.2 Evaluator 2

Evaluation team 2 conducted a similar task investigating the security and performance issues of UEA2 and UIA2. Their resulting report is also based on theoretical derivations and practical experimentation.

9.7.2.1 Extracts from evaluation report

The report from the second evaluation team includes a detail analysis of the components of 3G. Starting with the LFSR from SNOW 2.0, they confirmed that the period of the generated sequence is of maximal length 2^{512}-1 and that bit-wise consideration gave a feedback polynomial of high weight.

The report then discussed the non-linear properties of the addition operation modulo 2^{32}, and concluded that this operation can be represented by a system of linear and quadratic equations. This operation can also be approximated by linear relations with a high probability. By itself the modular addition does not give sufficient resistance against algebraic attacks.

The team also confirmed the algebraic properties of the two S-boxes and provided a careful analysis of the new S-box S_2, which has since been changed.

Cryptanalytic attacks on SNOW 3G were described along the following strategies:

1. Structural attacks. Trying several well known attacks based on the structural architecture of SNOW 3G gave complexities that were well above the key size of the algorithm.

2. Linear attacks. The report described general strategies for distinguishing attacks based on linear approximations, and argued that they are most likely unsuccessful in building a distinguisher for SNOW 3G.

3. Algebraic attacks. The team discussed possible strategies for mounting algebraic attacks against SNOW 3G, but concluded that the introduction of R3 succeeded very well against the problems identified by Gilbert and Billet, ref. [11].

   Resynchronization attacks. The team investigated the resistance of SNOW 3G against chosen IV attacks. Their best approach made use of the fact that the (then) S-box S_2 was not a permutation and they argued that the diffusion rate of SNOW 3G with 32 clocks for the initialization of 19 registers does not have a huge security margin against resynchronization attacks.

This evaluation group also considered the modes of use of SNOW 3G in UEA2 and UIA2. For UEA2 they focused on generic time-memory-data trade-off attacks. Discussing both Babbage-Golic trade-off and Biryukov-Shamir trade-off, they concluded that the best counter-measure against such attacks would be to increase the entropy of the IV. In the current specifications this is limited to the five bits in the Bearer identity. For the integrity algorithm they confirm the forgery probabilities given in section 9.3.2.2 of this report. An interesting observation was made to the rare occasion with $P = 0$ or $Q = 0$. In this case the MAC is completely insensitive to the message.

The report pointed to the yearly report from ECRYPT on algorithms and key sizes, ref. [18]. That report concludes that a 128-bit key provides long-term protection against all types of adversaries. For the MAC it is stated that a 32-bit MAC could be a problem in case an opponent can broadcast a message to many users. Finally the report stated four different

issues that might be used in an attack, and discusses possible strategies to overcome the identified problems. These were:

1. choice of S2 with regards to unbalance and inefficiency in SW,
2. lack of entropy in the IV,
3. insertion of the IV, and
4. the number of clocks during resynchronization.

9.7.2.2 Evaluation summary

The Executive summary from the second team reads:

"We have performed a security evaluation of the 3GPP Confidentiality and Integrity Algorithms UEA2 and UIA2, with an effort of approximately 20 person-days. We have applied state-of-the-art cryptanalytic techniques on the underlying stream cipher SNOW 3G, and found no practical attacks. We believe that SNOW 3G offers a sufficient security margin against current cryptanalytic techniques. In view of the very fast evolution in the field of cryptography and particularly stream ciphers, we recommend to update this evaluation three years from now.

The 3GPP Confidentiality Algorithm UEA2 is based on SNOW 3G and provides adequate confidentiality protection. The 3GPP Integrity Algorithm UIA2 provides message authentication with a sufficient security level. The key size of 128 bits for both UEA2 and UIA2 should provide security against exhaustive search for the next 30 years, and the MAC size of 32 bits is deemed sufficient for the intended application.

Some minor improvements are proposed that can strengthen the primitives against tradeoff and resynchronization attacks and improve the software performance, while at the same time keeping the good resistance to the other attacks described.

Finally we observe that our report is the result of only a limited time of review. Others with more time and dedicated resources may well find attacks that we have not been able to identify during this time."

9.7.2.3 Task force response

As reported in section 9.3.1.2, the task force redesigned S_2 into a fast permutation. This resolved the first potential problems identified by Evaluator 2.

Adding more entropy to the IV was not seen as a necessary modification to the design. Sufficient entropy is provided by the key, and the IV is defined by the external parameters given by the UMTS system.

The task force agreed that modifying the IV insertion could lead to faster dissemination of IV data in the system. The way in which the SNOW 3G IV is constructed was therefore changed, so that each bit of contributing data (COUNT-C, BEARER and DIRECTION for UEA2, and COUNT-I, FRESH and DIRECTION for UIA2) affects two bits of the SNOW 3G IV, instead of just one. This means that the contributing data are indeed diffused faster, without changing the way in which the SNOW 3G IV itself is used by the SNOW 3G algorithm.

Increasing the number of clocks during resynchronization was also considered. This addresses essentially the same concern as the IV diffusion point, and so was rendered less beneficial by

the change to the IV construction. The task force therefore agreed to stick to the 32 clocks used by SNOW 2.0. For efficiency reasons is it important to keep the initialization phase of the algorithm as short as possible.

For UIA2 the rare occasion of generating $P = 0$ and $Q = 0$ has a low probability $p = 2^{-63}$, and this situation is covered by the security proof for the scheme. The task force could not find sufficient argument for complicating the specifications with special handling of this case.

9.8 Results from IPR investigations

The evaluation must ensure that no IPR regulations will limit the use of UEA2 and UIA2 within its intended scope of 3G. The intention is that the algorithm specifications shall be fully standardized and be published as a 3GPP specification and that all copyright on the algorithm and test data specifications shall be owned jointly by the 3GPP partner organisations. See 33.105 [2]

(This does not preclude that users of the algorithm and the algorithm specification, shall be required to sign a licence agreement with any of the 3GPP partner organisations.)

SAGE has conducted some limited investigations relating to IPR, and has not been informed of existence of any Intellectual Property Right (IPR) which could be, or could become essential to these specifications.

Specifically the original designers of SNOW, have stated to SAGE that they have no claims on IP rights. SNOW2.0 is the basis for SNOW3G which makes up UEA2 and the modifications from SNOW2.0 to SNOW 3G have been designed completely inside the ETSI SAGE design team.

SNOW 3G is also an essential part for UIA2. Other parts of UIA2 build upon the Galois/Counter Mode invented by John Viega and David McGrew. They have issued an IPR statement on the NIST webpage:

http://csrc.nist.gov/CryptoToolkit/modes/proposedmodes/gcm/gcm-nist-ipr.pdf

It contains the following statement: “The authors are unaware of any intellectual property rights that pertain to the Galois/Counter Mode of operation (GCM) [22], nor do they claim any such rights.”

The other parts of UIA 2 have been designed completely inside the ETSI SAGE design team.

In conclusion we have strong reason to believe that no IPR is connected to the specified algorithms.

However, pursuant to the ETSI IPR Policy, no full investigation, including IPR searches, has been carried out. No guarantee can be given as to the existence of any IPRs which are, or may be, or may become, essential to the algorithm specifications.

9.9 Conclusion of evaluation

The 3GPP confidentiality and integrity algorithms have been subject to an extensive mathematical and cryptanalytic review in order to reveal any weakness in the design. This work has been conducted by the task force itself, by additional manufacturers with competence in the field, and by two independent parties. The work has involved some of the leading experts in the field. *The general conclusion is that the algorithms are based on sound*

design principles, and no practical attacks were found. The algorithms are well fitted for their intended use.

The algorithms have specifically been designed for use within the 3GPP context. It has not been the intention to increase the security margins in order to develop general-purpose algorithms for multiple unknown applications. The design is a carefully trade-off providing full strength algorithms and efficient implementation and use in the next generation mobile systems.

The 3GPP algorithms have been designed to resist a suite of well-known cryptanalytic attacks. However, one can never prove that a cryptographic algorithm will resist new attacks in the future. Due to this fact and the very limited time span that was available for the work, the task force will propose that the results from this report are reviewed on a regular basis. A basic review of the offered security and usability of the 3GPP confidentiality and integrity algorithms should be conducted every five years.

10 Annex A - External references

[8] F. Armknecht. On the Existence of low-degree Equations for Algebraic Attacks. Cryptology ePrint Archive, Report 2004/185, 2004. http://eprint.iacr.org/2004/185.

[9] D. Bernstein. Stronger security bounds for Wegman-Carter-Shoup authenticators. Proceedings of Eurocrypt'05. Available via http://cr.yp.to/antiforgery/securitywcs-20050227.pdf.

[10] J. Bierbrauer, T. Johansson, G. Kabatianskii, and B. Smeets. On families of hash functions via geometric codes and concatenation. Proceedings of CRYPTO '93, LNCS 773, 331-342, Springer-Verlag, 1993.

[11] O. Billet and H. Gilbert. Resistance of SNOW 2.0 Against Algebraic Attacks. In Alfred Menezes editor, Topics in Cryptology -- CT-RSA~2005, Lecture Notes in Computer Science, vol. 3376, Springer Verlag, 2005.

[12] A. Braeken, L. R. Knudsen, J. Lano, B. Preneel and H. Wu. Security Evaluation of the 3GPP Encryption and Integrity Algorithms UEA2 and UIA2.

[13] J. L. Carter and M, N, Wegman. Universal Classes of Hash Functions. J. Computer and System Sciences 18 (1979), 143-154.

[14] D. Coppersmith, S. Halevi, and C. S. Jutla. Cryptanalysis of stream ciphers with linear masking. In M. Yung, editor, Advances in Cryptology -- CRYPTO 2002, Lecture Notes in Computer Science, pages 515--532. Springer -Verlag, 2002.

[15] N. T. Courtois. Algebraic attacks on combiners with memory and several outputs. Cryptology ePrint Archive, Report 2003/125, 2003. http://eprint.iacr.org/.

[16] N. T. Courtois and J. Pieprzyk. Cryptanalysis of Block Ciphers with Overdefined Systems of Equations. Proceedings of ASIACRYPT 2002, LNCS 2501, pp. 267-287, 2002.

[17] L. E. Dickson. Linear Groups with an Exposition of the Galois Field theory, Teubner, Leipzig, 1901; Dover, New York, 1958.

[18] ECRYPT Network of Exellence. ECRYPT Yearly Report on Algorithms and Keysizes, 2004. http://www.ecrypt.eu.org/documents/D.SPA.10-1.1.pdf.

[19] P. Ekdahl and T. Johansson. SNOW --- a new stream cipher. Submission can be downloaded at http://www.cryptonessie.org, 2000.

[20] P. Ekdahl and T. Johansson. A new version of the stream cipher SNOW. In K. Nyberg and H. M. Heys, editors, Selected Areas in Cryptography -- SAC 2002, Lecture Notes in Computer Science, pages 47--61. Springer -Verlag, 2002.

[21] L. Granboulan and J. Stern. Evaluation of UEA2 and UIA2.

[22] D. McGrew and J. Viega. The security and performance of the Galois/Counter mode of operation. available via the IACR eprint server at http://eprint.iacr.org/2004/193.pdf.

[23] P. Hawkes and G. G. Rose. Guess-and-determine attacks on SNOW. In K. Nyberg and H. M. Heys, editors, Selected Areas in Cryptography -- SAC 2002, Lecture Notes in Computer Science, pages 37--46. Springer -Verlag, 2002.

[24] National Institute of Standards and Technology. Advanced Encryption Standard. FIPS 197. 26 November 2001.

[25] W. Nevelsteen and B. Preneel. Software performance of universal hash functions. Proceedings of EUROCRYPT '99, LNCS 1592, 24-41, Springer-Verlag, 1999.

[26] K. Nyberg, Improved linear distinguishers for SNOW 2.0. Draft November 25, 2005, submitted.

[27] E. Pasalic. Private communication. May 2005.

[28] V. Shoup. On fast and provably secure message authentication based on universal hashing. Proceedings of CRYPTO '96, LNCS 1109, 313-328, Springer-Verlag, 1996.

[29] D. Stinson. Universal hashing and authentication codes. Proceedings of CRYPTO '91, LNCS 576, 74-85, Springer-Verlag, 1992.

[30] Wassenaar Arrangement, December 1998.

[31] D. Watanabe, A. Biryukov and C. de Canni'ere. A distinguishing attack on SNOW 2.0 with linear masking method. In M. Matsui and R. Zuccherato, ed-itors, Selected Areas in Cryptography -- SAC 2003, Lecture Notes in Computer Science, pages 222--233. Springer -Verlag, 2003.

[32] M. N. Wegman and J. L. Carter. New Hash Functions and their Use in Authentication and Set Equality. J. Computer and System Sciences 22 (1981), 265-279.