Nordea

Technical Specification

Nordea File protection with HMAC SHA256-128

Version 1.2

1 General

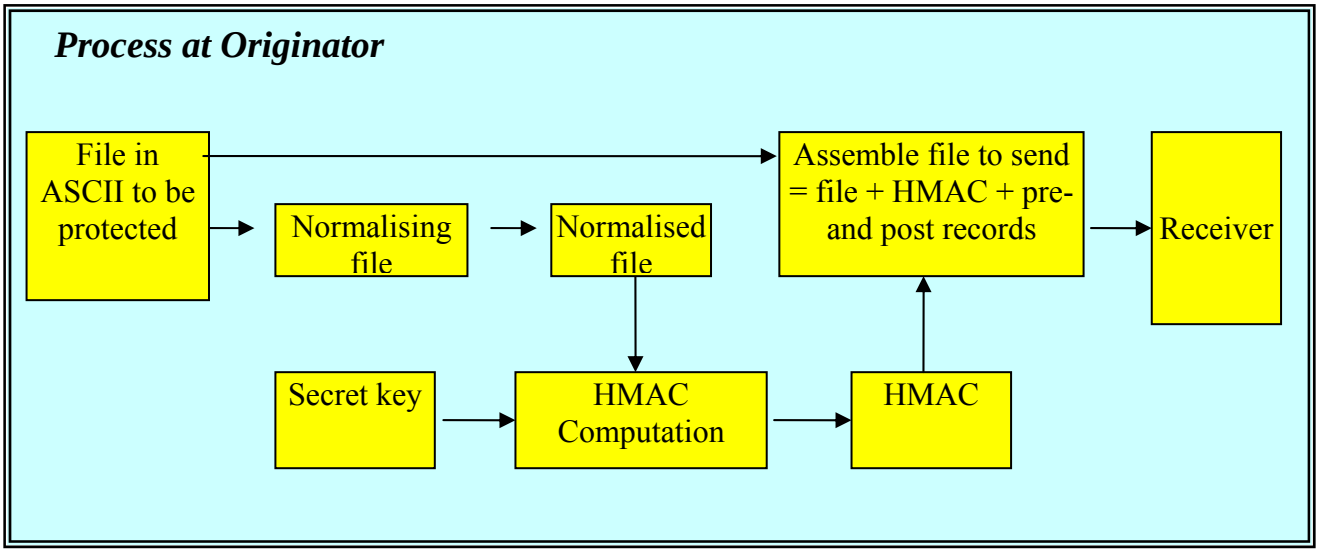
This document describes the steps involved to protect a file from being intentionally or unintentionally modified. To be more precise — it makes it possible to detect such modification but **cannot stop modifications** from happening.
The process involves two steps, one step at the originator of the file and one step at the receiver of the file.

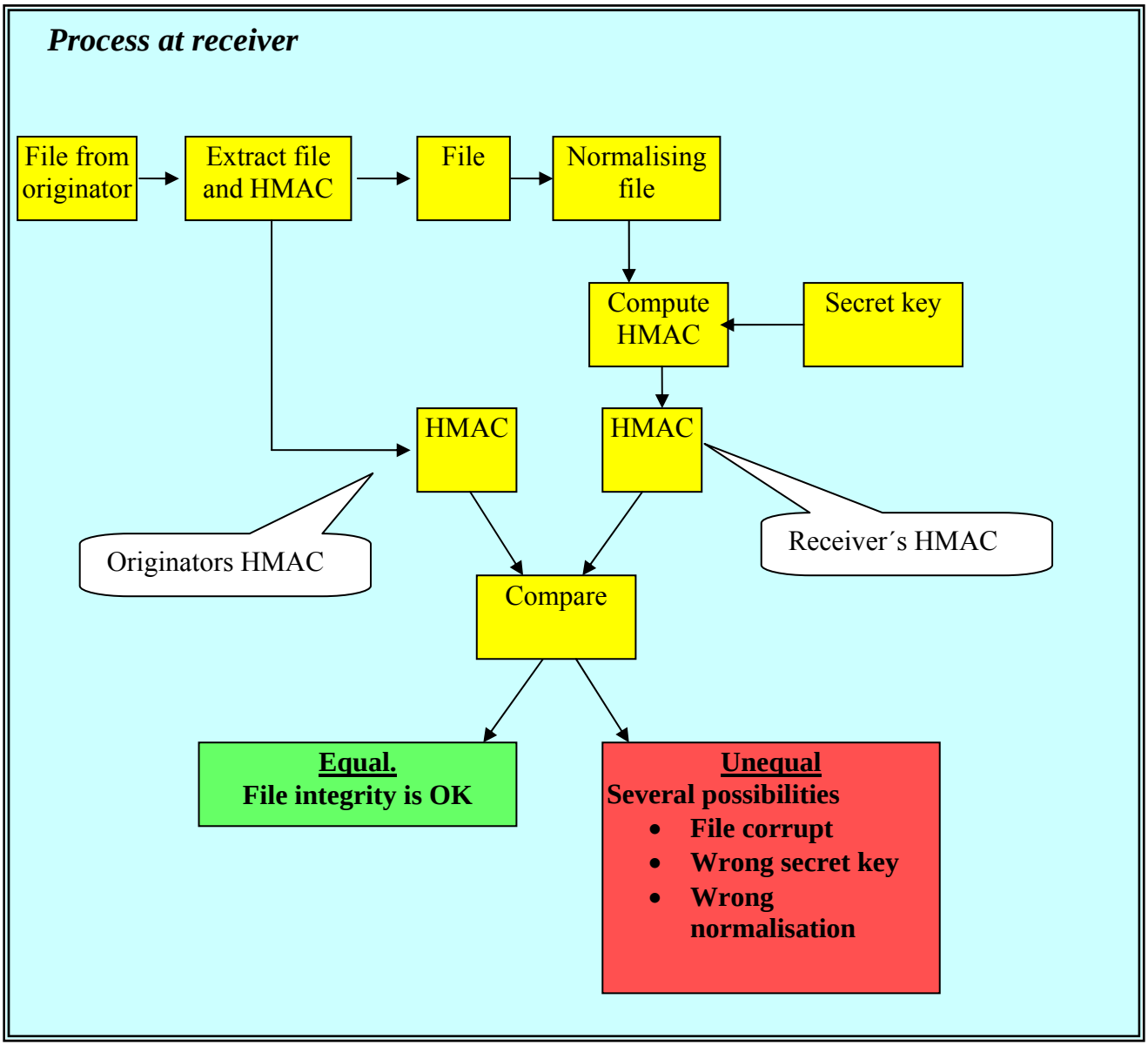
The protection is independent of the content of the file.

The originator of the file, which is often the sender of the file, computes a cryptographic checksum of the file called the HMAC. The receiver of the file also computes a HMAC and compares the one received from the sender with the one the receiver has computed. If the two are the same the file has not been modified and its integrity has been verified.

The cryptographic computation of the HMAC is done on the file using hex values after it has been normalised. The file itself though is not changed due to the normalisation process.
This means that the unmodified file plus HMAC computed on the normalised file is sent to the receiver.
It is the responsibility of the receiver to verify the integrity and thus do the second HMAC computation and comparison.

The graphs below show the process at both ends.





Several different methods exist for calculation of a cryptographic checksum. Nordea accept files protected with HMAC-SHA256-128 method.

The method is public and the security relies on keeping some of the input to the seal calculation secret, i.e. the key. Keys used are 128 bits long and are changed at certain intervals. The keys are generated by Nordea and sent to Nordea's customers well in advance of the time for replacement.

To be able to send files to Nordea the sender must have a seal agreement with Nordea and have received a customer id and a secret HMAC key to be able to sign the agreement. Additionally the rules set forth in this document must be followed.

In principle, Nordea's method of tamper protection with HMAC-SHA256-128 means that a file to be sent to Nordea is run through an algorithm (256-bit secure hash algorithm, SHA256) that calculates a cryptographic checksum on the file, a hash-based method authentication code (HMAC).

The standard uses a secret key that is 128 bits long to generate a MAC that is also 128 bits long. Note that the MAC should be truncated to 128 bits since SHA256 normally generates a 256-bit value. It is the first 128 bits that should be used as the HMAC.

1 References

The calculation of HMAC-SHA256-128 is described in a three international documents.

Information Technology Laboratory at US National Institute of Standards, NIST has published 2 Federal Information Processing Standards, FIPS, which are relevant.

1. FIPS-198-1 describes the Secure Hash Algorithm; SHA using a 256 bit key.
2. FIPS-180-3 specifies Hashed Message Authentication Code, HMAC calculation

http://www.itl.nist.gov/fipspubs/index.htm

The Internet Engineering Task Force, IETF, has published a Request For Comments, RFC, document that is also applicable.

3. RFC 4868 specifies HMAC-SHA256-128 and how the calculated value is truncated.

http://www.rfc-editor.org/rfc/rfc4868.txt

2 Definitions

HASH algorithm	A function used to compute a signature of a file, a hash code, such that it is difficult: For a given hashcode The HASH algorithm itself does not use any keys.
Hash code	The output of a hash algorithm
HMAC	A seal computed using a hash algorithm on a file which has added secret non-transmitted data. The non-transmitted data being part of the computation serves as the secret key.
KVV	Key Verification Value. Functions as a seal on the key and is used to verify that the correct key is used.
MAC	Message Authentication Code, often used as synonym for seal. Seal, MAC and HMAC may be used interchangeably meaning the same thing in this specification.

Normalisation	Rules for ensuring that HMAC calculations are done in a consistent way.
SEAL	Equivalent with a cryptographic checksum, SEAL is a generic term and several different algorithms and methods can be used to calculate a seal, E.G. DES, MAA, MD5. and HMAC.

2 Tamper protection with HMAC-SHA256-128

When a data file with information is created, the seal is calculated with HMAC-SHA256-128 as per the following:

1. Use the current key provided by Nordea.

2. Initiate the algorithm with the current key value.

3. Read the file and normalise the contents.

4. Send the normalised data to the algorithm in order to calculate the HMAC.

5. When the file is read, carry out the final operation, which results in a calculated HMAC.

6. Calculate a key verification value, KVV.

7. Enter the HMAC and KVV in record %22 and the production date in record %20.

8. Send the files to Nordea for further processing.

When calculating the MAC, the entire file is read and all characters in the file are processed in the algorithm. Note that the % posts not shall be a part of the calculation.

When calculating the MAC, if end-of-line characters are used (CRLF, Carriage Return Line Feed), these characters should not be included in the calculation.

2.1 Data format

Because the algorithm is binary, great weight must be placed on the data format the file should have. To prevent problems with potential character conversions, bit patterns to be used in the calculation of the MAC are defined for each character (see the table at the end of this document).

2.1.1 Requirement

Every character included in the file must be included in the calculation as per above. Also, the file must have the format and the character set that the applicable applications presuppose.

2.2 Basic file structure

The seal is always placed in the %-22 record. Files that today have a seal sum within the file must change the sealing process. We don’t support the HMAC standard for these type of files. For example if the Invoice payment service (FS) is used with a S post at the end of the file it is mandatory to execute a HMAC calculation and remove the post and place the seal in the communication record %-22.

2.2.1 Seal Information

The seal information important to tamper protection is placed in the %020 and %022 records. Creation date is placed in the record "%020". MAC and KVV in %022. The %- records must always be 80 characters long regardless of the length of following records and must be structured as indicated in the tables below. More about the File transmission header (%001), Contents (%020, %022) and Trailer (%002) is described in Appendix 1. For detailed descriptions about communication methods see the document "Girolink Internet and other Communication methods".

Positions	Field description
1-4	"%020"
5–14	Destination node
15–24	Source node.
25–31	**External reference 1. Production date must always be stated**
32–38	Number of items. (Stated where possible for allocation of space.)
39–48	External reference 2. Free text field, e.g. where customer number is stated.
49–80	Reserve/Blank.

Positions	Field description
1-4	"%022"
5-11	Number of records in file (Fantom records excluded), right justified "0" padded.
12-43	**KVV for key used, 128 bits, presented as 32 hexadecimal digits.**
44-75	**MAC for the file, 128 bits, presented as 32 hexadecimal digits.**
76-80	Blank/reserve

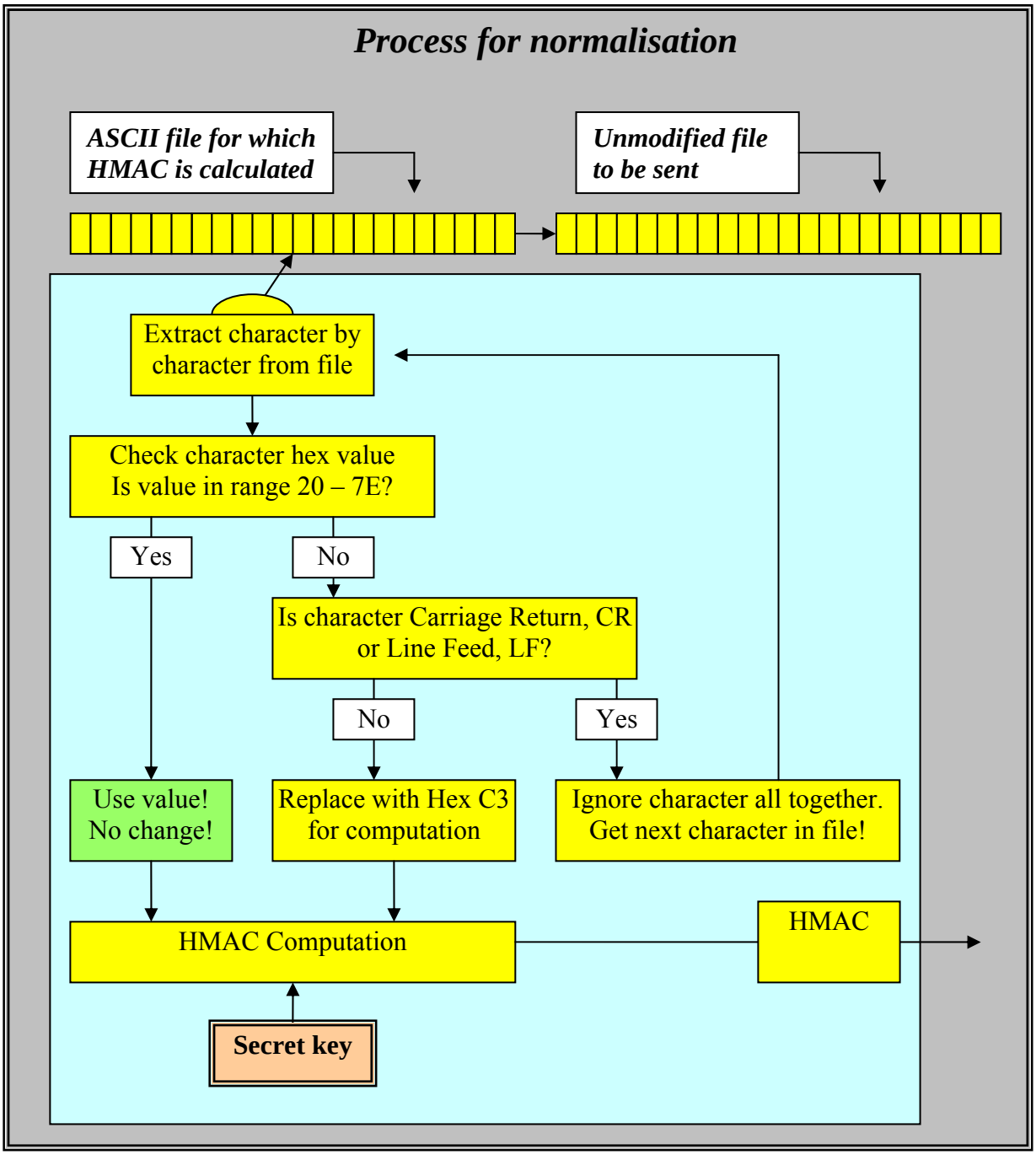
2.3 Character translation

2.3.1 General

Keyboard characters are not represented in the same way in different environments. For instance, if Å, Ä or Ö is written in DOS, only ■ is visible when viewing the contents of a file with these characters from Windows. If the file looks differently at Nordea than it did at the sender, the calculated MAC will also differ. Consequently, the file's contents may need to be converted to a type of normalised standard character set so that the file always looks the same at the time of calculation regardless of where it was created. This means that a potential character conversion must take place right before the MAC is calculated and the file is sent to Nordea.

2.3.2 Normalised character set

For the method pertaining to this document (HMAC-SHA256-128), all characters in the 7-bit ASCII table (see the table at the end of this document), excluding the control character, are included, i.e. all characters from hex 20 (space) to hex 7E (~), inclusive. All characters that are not listed in the table are replaced by hex C3 (dec 195) before seal calculation.



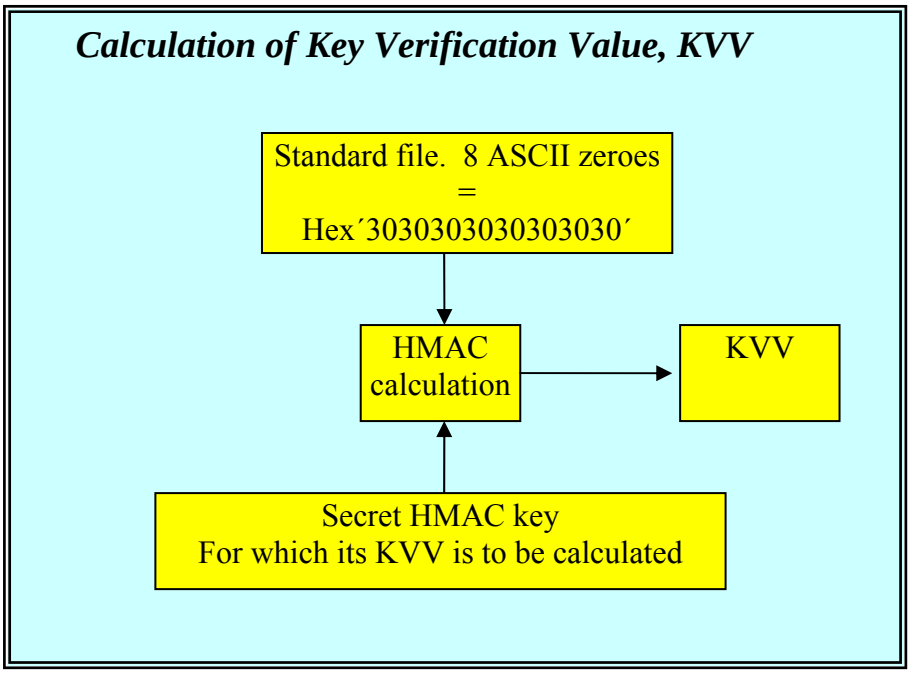
3 Management of keys

3.1 Principle

HMAC-SHA256-128 requires a key that is 128 bits long for the calculation of the HMAC on the data. This is normally represented as 32 hexadecimal digits in manual processing. A key is valid, until a new key is requested, which is then delivered in a sealed key envelope. The key is secret to all outsiders and only those assigned, and consequently entrusted, to handle the key shall have knowledge of it.

3.2 Responsibility for keys

Within Nordea, the keys are handled with a high level of secrecy. The objective of key management is to achieve a situation in which the key cannot be revealed or abused in Nordea's environment. Therefore, great responsibility rests with the customer to manage keys and introduce program support for this so as to achieve a high level of security.



3.3 Storage

The basis of key management is that the keys are stored in a secure manner when not in use. What constitutes a secure manner naturally depends on how the current environment is structured at each customer.

The following fundamental requirements must be set for secure key management:

- Keys are locked away when not in use.
- Keys are retrieved and entered in manually at each use.
- Processing should take place in an environment such that it can be reasonably ensured that the key values are not disclosed to unauthorised parties.

3.4 Use of keys

When the key is to be used for calculation of a MAC, it must be entered into the algorithm. If no encryption is available in the environment in which tamper protection is used, the key should be entered manually via a keyboard at each use. To check the entry, the KVV belonging to each key can be used. The KVV can be stored in the system for comparison with the KVV obtained for the entered key value.

The key in HMAC-SHA256-128 is represented as a series of 32 hexadecimal digits in processing. The digits represent 4-bit groups and the key value can therefore be said to comprise 128 bits.

3.5 Key verification value

There is a key verification value (KVV) for each key. This is calculated as a MAC for a "standard file" with the current key. This KVV need not be secret in the same way as a key and can be used to verify that the key value is correct without disclosing the key value.

A KVV shall be calculated each time a key is used to protect a file. This value shall be placed in the indicated location stated above and shall be included in the check of the file done at Nordea.
To check if an erroneous key value is entered, the customer's system can verify the KVV before calculation of the MAC is begun. The “Standard file” for the calculation of KVV has the contents: “00000000”, i.e. 8 zeros in ASCII-format, hex 30.

3.6 Requirement

In the description above, it is stated that keys must be handled in such a way so that they cannot be disclosed to or used by unauthorised parties. The customer is charged with great responsibility for its keys and their handling. The key verification value (KVV) shall be used to reduce the possibility that erroneous key values are used.

4 Definition of characters used

This table shows the characters to be used in the calculation of a MAC in accordance with Nordea's method HMAC-SHA256-128. The characters that are not in the table are replaced, as previously mentioned, with hex C3 (dec 195) before seal calculation.

Hexadecimal	**Binary**	**Char. as per 7-bit ASCII ISO/IEC 646**
20	0010 0000	SP
21	0010 0001	!
22	0010 0010	″ (double quote)
23	0010 0011	#
24	0010 0100	$
25	0010 0101	%
26	0010 0110	&
27	0010 0111	' (single quote)
28	0010 1000	(
29	0010 1001	)
2A	0010 1010	*
2B	0010 1011	+
2C	0010 1100	,
2D	0010 1101	- (minus sign)
2E	0010 1110	.
2F	0010 1111	/
30	0011 0000	0
31	0011 0001	1
32	0011 0010	2
33	0011 0011	3
34	0011 0100	4
35	0011 0101	5
36	0011 0110	6
37	0011 0111	7
38	0011 1000	8
39	0011 1001	9
3A	0011 1010	:
3B	0011 1011	;
3C	0011 1100	<
3D	0011 1101	=
3E	0011 1110	>
3F	0011 1111	?
40	0100 0000	@ , É
41	0100 0001	A
42	0100 0010	B
43	0100 0011	C
44	0100 0100	D

45	0100 0101	E
46	0100 0110	F
47	0100 0111	G
48	0100 1000	H
49	0100 1001	I
4A	0100 1010	J
4B	0100 1011	K
4C	0100 1100	L
4D	0100 1101	M
4E	0100 1110	N
4F	0100 1111	O
50	0101 0000	P
51	0101 0001	Q
52	0101 0010	R
53	0101 0011	S
54	0101 0100	T
55	0101 0101	U
56	0101 0110	V
57	0101 0111	W
58	0101 1000	X
59	0101 1001	Y
5A	0101 1010	Z
5B	0101 1011	[, Ä
5C	0101 1100	\ , Ö
5D	0101 1101	] , Å]
5E	0101 1110	^ , Ü
5F	0101 1111	_ (underscore)
60	0110 0000	` , é
61	0110 0000	a
62	0110 0001	b
63	0110 0010	c
64	0110 0100	d
65	0110 0101	e
66	0110 0110	f
67	0110 0111	g
68	0110 1000	h
69	0110 1001	i
6A	0110 1010	j
6B	0110 1011	k
6C	0110 1100	l
6D	0110 1101	m
6E	0110 1110	n
6F	0110 1111	o

70	0111 0000	p
71	0111 0001	q
72	0111 0010	r
73	0111 0011	s
74	0111 0100	t
75	0111 0101	u
76	0111 0110	v
77	0111 0111	w
78	0111 1000	x
79	0111 1001	y
7A	0111 1010	z
7B	0111 1011	{ , ä
7C	0111 1100	| , ö
7D	0111 1101	} , å
7E	0111 1110	~ , ü

4.1 EBCDIC – ASCII Conversion Table

When calculating a MAC, ASCII shall always be used. Consequently, if the customer's platform uses EBCDIC, a character conversion must be applied before the MAC calculation.
The following conversion table shall be used in these cases.
The outer frame represents the hexadecimal EBCDIC value. To translate SPACE (x'40') in EBCDIC to ASCII, the following is done:

1. Find 4x in the left-hand column.
2. On the 4x row, find the value below the x0 column.
3. The value found is 20, which corresponds to SPACE (x'40') in EBCDIC.

	x0	x1	x2	X3	X4	X5	X6	X7	X8	X9	xA	xB	xC	xD	xE	xF
0x	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3
1x	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3
2x	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3
3x	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3
4x	20	C3	C3	7B	C3	C3	C3	7D	C3	C3	C3	2E	3C	28	2B	21
5x	26	60	C3	C3	C3	C3	C3	C3	C3	C3	C3	5D	2A	29	3B	5E
6x	2D	2F	C3	23	C3	C3	C3	24	C3	C3	7C	2C	25	5F	3E	3F
7x	C3	5C	C3	C3	C3	C3	C3	C3	C3	60	3A	5B	5C	27	3D	22
8x	C3	61	62	63	64	65	66	67	68	69	C3	C3	C3	C3	C3	C3
9x	C3	6A	6B	6C	6D	6E	6F	70	71	72	C3	C3	C3	C3	C3	5D
Ax	C3	7E	73	74	75	76	77	78	79	7A	C3	C3	C3	C3	C3	C3
Bx	C3	C3	C3	C3	C3	5B	C3	C3	C3	C3	C3	7C	C3	C3	C3	C3
Cx	7B	41	42	43	44	45	46	47	48	49	C3	C3	C3	C3	C3	C3
Dx	7D	4A	4B	4C	4D	4E	4F	50	51	52	C3	C3	7E	C3	C3	C3
Ex	40	C3	53	54	55	56	57	58	59	5A	C3	C3	40	C3	C3	C3
Fx	30	31	32	33	34	35	36	37	38	39	C3	C3	5E	C3	C3	C3

3 Sending format

Sending format S2 facilitates standardised file identification. This is essential to a uniform standard for media management.

Sending format S2 also makes it possible to run several sub-files of the same type in a single sending. Each sub-file is to be surrounded by file items (%020 and %022) and the entire sending is to be surrounded by sending items (%001 and %002).

In other respects all per cent items have the same length as the application items. Letters must always be upper-case. All items are alfa-numerical and written left-aligned and blanks filled in (does not concern the number field in the File Trailer %022, which must be numerical and written right aligned and zero filled in. In connection with the Mac-code the File Trailer also contains the Seal, see below).

Compulsory information is in extra bold type. Other information is not essential but can if specified improve the handling related to file transmission.

A data forwarding agent must always specify the origin in the address fields (%020).

Transmission header

Positions	**Field description**
1–4	**%001**
5–14	**Delivering node**. (Node ID.)
15–20	**Password.** (Is NOT specified for SNI or TCP/IP transfer.)
21	**0 (zero)**. Indicates delivery.
22–24	File type.
25–30	**External reference**. Dates must be specified as yymmdd.
31	Free field for e.g. sending number during the day.
32	**0** (zero).
33–80	Reserve/Blank.

File header
An address post for each application file.

Positions	**Field description**
1-4	"%020"
5–14	Destination node
15–24	Source node.
25–31	**External reference 1. Production date must always be stated**
32–38	Number of items. (Stated where possible for allocation of space.)
39–48	External reference 2. Free text field, e.g. where customer number is stated.
49–80	Reserve/Blank.

File Trailer

Positions	Field description
1-4	"%022"
5-11	Number of records in file (Fantom records excluded), right justified "0" padded.
12-43	**KVV for key used, 128 bits, presented as 32 hexadecimal digits.**
44-75	**MAC for the file, 128 bits, presented as 32 hexadecimal digits.**
76-80	Blank/reserve

Transmission Trailer

Positions	Field description
1–4	**%002**
5–80	Reserve/Blank.

4 Possible errors

If the HMAC verification fails one or more of the following might be the cause of the problem:

- The HMAC is correctly calculated by the originator of the file but gets corrupted during transport.
- The file itself gets corrupted during transport
- The originator and the receiver do not use the same key for HMAC computation
- An error occurred during normalisation. When a character outside the normal range (20-7F) was detected the calculation did not use the hex ´C3 value for calculation but the value itself instead.
  E.G the value hex FF was not detected in file by either the originator or the receiver so instead of hex C3 hex FF was used in the HMAC calculation.
- One or more Carriage Return and/or Line feed was included in the calculation.

5 Example of test file to Invoice payment service with HMAC

```
%001TCP1234567      001P0082700
%020FS-T      0123456789090105        A9999
0A99990310231
2A9999  99999999  TESTBOLAGET AB            TESTVÄGEN 1  TEL
08-123456 SEKSEK
```

```
53 SEK      11111FAKTURA1                     00000010000031023VÅR
REFERENS 12345
53 SEK     222222FAKTURA2                     00000020000031023VÅR
REFERENS 23456
53 SEK    3333333FAKTURA3                     00000030000031023VÅR
REFERENS 33333
63 SEK    3333333KREDITFAKT3
000000500000310230410 23VÅR REFERENS
43         444444MEDDELANDERAD1
MEDDELANDERAD2
43         444444MEDDELANDERAD3
MEDDELANDERAD4
53 SEK     444444FAKTURA4                     00000040000031027VÅR
REFERENS 45678
34        1234567     TESTMOTTAGARE 1
5555555
54 SEK    1234567FAKTURA5                     00000050000031024VÅR
REFERENS 56478
34        2345678     TESTMOTTAGARE 2                 9960
2287240
54 SEK    2345678FAKTURA6                     00000060000031024VÅR
REFERENS 67890
7A9999  99999999  000000160000+
SEKSEK
%022000001401234567890123456789012345678900123456789012345678901234567890123456789012
%002
```

(NB! The source node and delivery node are unique for each customer/Service office and is administered by Plusgirot.