(12) PATENT **(11)** Application No. **AU 199870282 B2**
(19) AUSTRALIAN PATENT OFFICE **(10)** Patent No. **723002**

(54) Title
Method and system for producing and checking a hash total for digital data grouped in several data segments

(51)[7] International Patent Classification(s)
H04L 009/32

(21) Application No: **199870282** (22) Application Date: **1998.02.25**

(87) WIPO No: **WO98/47264**

(30) Priority Data

(31) Number	(32) Date	(33) Country
19715486	**1997.04.14**	**DE**

(43) Publication Date : **1998.11.11**
(43) Publication Journal Date : **1998.12.24**
(44) Accepted Journal Date : **2000.08.17**

(71) Applicant(s)
Siemens Aktiengesellschaft

(72) Inventor(s)
Martina Hanck; Gerhard Hoffmann; Klaus Lukas

(74) Agent/Attorney
SPRUSON and FERGUSON,GPO Box 3898,SYDNEY NSW 2001

(56) Related Art
JP 06-315027
US 5673318

OPI DATE 11/11/98 APPLN. ID 70282/98
AOJP DATE 24/12/98 PCT NUMBER PCT/DE98/00563

AU9870282

(51) **Internationale Patentklassifikation 6 :** **H04L 9/32** **A1**

(11) **Internationale Veröffentlichungsnummer:** **WO 98/47264**

(43) **Internationales Veröffentlichungsdatum:** 22. Oktober 1998 (22.10.98)

(21) **Internationales Aktenzeichen:** PCT/DE98/00563

(22) **Internationales Anmeldedatum:** 25. Februar 1998 (25.02.98)

(30) **Prioritätsdaten:**
197 15 486.7 14. April 1997 (14.04.97) DE

(71) **Anmelder** *(für alle Bestimmungsstaaten ausser US)*: SIEMENS AKTIENGESELLSCHAFT [DE/DE]; Wittelsbacherplatz 2, D–80333 München (DE).

(72) **Erfinder; und**
(75) **Erfinder/Anmelder** *(nur für US)*: HANCK, Martina [DE/DE]; Am Grenzweg 2, D–85635 Höhenkirchen (DE). HOFFMANN, Gerhard [DE/DE]; Gozbertstrasse 8/II, D–81547 München (DE). LUKAS, Klaus [DE/DE]; Niemöllerallee 6, D–81793 München (DE).

(81) **Bestimmungsstaaten:** AU, ID, JP, US, europäisches Patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

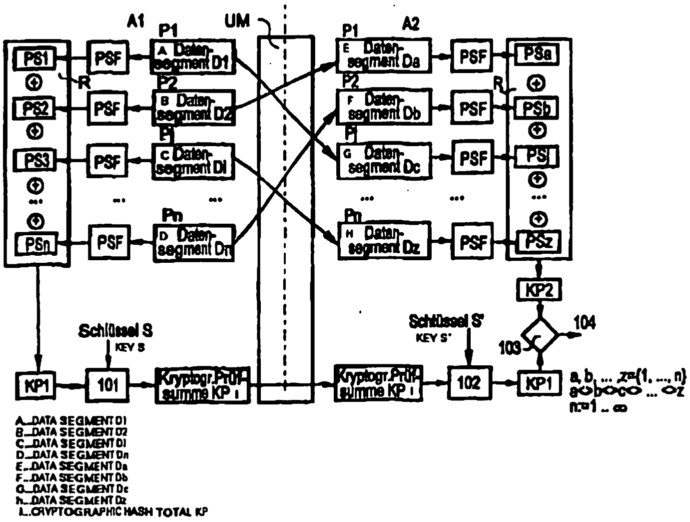
Veröffentlicht
Mit internationalem Recherchenbericht.
Vor Ablauf der für Änderungen der Ansprüche zugelassenen Frist; Veröffentlichung wird wiederholt falls Änderungen eintreffen.

(54) **Title:** METHOD AND SYSTEM FOR PRODUCING AND CHECKING A HASH TOTAL FOR DIGITAL DATA GROUPED IN SEVERAL DATA SEGMENTS

(54) **Bezeichnung:** VERFAHREN UND ANORDNUNG ZUR BILDUNG UND ÜBERPRÜFUNG EINER PRÜFSUMME FÜR DIGITALE DATEN, DIE IN MEHRERE DATENSEGMENTE GRUPPIERT SIND

(57) Abstract

The invention relates to methods and systems for producing a hash total and checking a hash total for digital data, said data being grouped into data segments. According to this method, a hash total is produced for each data segment. The individual hash totals are combined to form a first commutative hash total using a commutative link. In order to check the first commutative hash total, another hash total is produced for each data segment and these hash totals are combined to form a second commutative hash total using a commutative link. The first commutative hash total and the second commutative hash total are then checked to make sure that they coincide.



(57) Zusammenfassung

Es werden Verfahren und Anordnungen zur Bildung einer Prüfsumme und zur Überprüfung einer Prüfsumme für digitale Daten, die in mehrere Datensegmente gruppiert sind, angegeben. Bei dem Verfahren wird für jedes Datensegment eine Prüfsumme gebildet. Die einzelnen Prüfsummen werden unter Verwendung einer kommutativen Verknüpfung zu einer ersten kommutativen Prüfsumme verknüpft. Zur Überprüfung der ersten kommutativen Prüfsumme wird für jedes Datensegment wiederum eine Prüfsumme gebildet und die Prüfsumme wiederum unter Verfahren einer kommutativen Verknüpfung zu einer zweiten kommutativen Prüfsumme verknüpft. Die erste kommutative Prüfsumme und die zweite kommutative Prüfsumme werden auf Übereinstimmung überprüft.

Abstract

Method and arrangement for forming and checking a checksum for digital data which are grouped into a number of data segments

Methods and arrangements for forming a checksum and for checking a checksum for digital data which are grouped into a number of data segments are specified. In the method, a checksum is formed for each data segment. The individual checksums are combined to form a first commutative checksum by using a commutative operation. To check the first commutative checksum, a checksum is again formed for each data segment and the checksum is again combined to form a second commutative checksum under the method of a commutative operation. The first commutative checksum and the second commutative checksum are checked for a match.

Description

Method and arrangement for forming and checking a checksum for digital data which are grouped into a number of data segments

In digital communications, i.e. during the exchange of digital data, it is frequently desirable to protect the transmission of the electronic data with respect to the most varied aspects.

A very significant aspect is the protection of the digital data to be transmitted against unauthorized modification, the so-called protection of the integrity of the data.

As a protection against unauthorized modification of digital data, the so-called cryptographic checksum, for example the digital signature, is known from [1]. The method described in [1] is based on forming a hashing value from the digital user data and the subsequent cryptographic processing of the hashing value by means of a cryptographic key. The result is a cryptographic checksum. To check the integrity, a corresponding cryptographic key is used for performing the inverse cryptographic operation on the checksum formed and the result is compared with the hashing value again calculated from the user data. The integrity of the user data is ensured when the hashing values are matched.

This previously customary procedure necessitates that the complete user data must be present on the receiver side in the identical order in which they were present when the hashing value was formed since otherwise the formation of the hashing value leads to an errored value. In digital communications, however, it is frequently customary to subdivide and to transmit the user data to be transmitted in relatively small data segments which are also called data packets, due to protocol boundary conditions.

The data segments are frequently not tied to a defined order or it is not possible to guarantee a defined sequential arrival of the data segments. In the method from [1], it is therefore required for the complete user data to be reassembled again on the receiver side, that is to say after the transmission of the data segments, in the order in which they were originally sent. The data to be transmitted can only be verified in this order. However, this frequently means considerable additional expenditure for the flow control of the data segments inasmuch as this is possible at all within the framework of the protocol used.

From [2], commutative operations are known. In [2], a general definition for commutative operations is also specified. Illustratively, a commutative operation can be understood to be an operation in which the order of individual operations is unimportant and each order of individual operation always leads to the same total operation. A commutative operation can be, for example, an EXOR operation, an additive operation or also a multiplicative operation.

From [3], a method and a device for generating check code segments for the occurrence of source data and for determining errors in the source data are known.

The invention is thus based on the object of specifying methods and arrangements for forming and checking a first commutative checksum for digital data which are grouped into a number of data segments, in which a flow control for the individual data segments is no longer required.

According to one aspect of the present invention there is provided a method of forming a first commutative checksum for digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

a segment checksum is formed for each data segment;

the first commutative checksum is formed by a commutative operation on the segment checksums; and

the first commutative checksum is cryptographically protected by using at least one cryptographic operation.

According to another aspect of the present invention there is provided a method of checking a predetermined cryptographic commutative checksum which is allocated to digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

the cryptographic commutative checksum is subjected to an inverse cryptographic operation to form a first cryptographic checksum;

a second segment checksum is formed for each data segment;

a second commutative checksum is formed by a commutative operation on the second segment checksums; and

the second commutative checksum is checked for a match with the first commutative checksum.

According to still another aspect of the present invention there is provided a method of forming and checking a first commutative checksum for digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

a segment checksum is formed for each data segment;

the first commutative checksum is formed by a commutative operation on the segment checksums;

the first commutative checksum is cryptographically protected by using at least one cryptographic operation, a cryptographic commutative checksum being formed;

the cryptographic commutative checksum is subjected to an inverse cryptographic operation to form a first reconstructed cryptographic checksum;

a second segment checksum is formed for each data segment of the digital data to which the first commutative checksum is allocated;

a second commutative checksum is formed by a commutative operation on the second segment checksums; and

the second commutative checksum is checked for a match with the first reconstructed commutative checksum.

According to still another aspect of the present invention there is provided an apparatus for forming a first commutative checksum for digital data which is grouped into a number of data segments, said apparatus comprising:

means for forming a segment checksum for each data segment;

means for forming the first commutative checksum by a commutative operation on the segment checksums; and

means for cryptographically protecting the first commutative checksum by using at least one cryptographic operation.

According to still another aspect of the present invention there is provided an apparatus for checking a predetermined first commutative checksum which is allocated to digital data which is grouped into a number of data segments, said apparatus comprising:

means for subjecting the cryptographic commutative checksum to an inverse cryptographic operation to form a first cryptographic checksum;

means for forming a second segment checksum for each data segment;

means for forming a second commutative checksum by a commutative operation on the second segment checksum; and

means for checking the second commutative checksum for a match with the first commutative checksum.

According to still another aspect of the present invention there is provided an apparatus for forming and checking a first commutative checksum for digital data which is grouped into a number of data segments, said apparatus comprising:

means for forming a segment checksum for each data segment;

means for forming the first commutative checksum by a commutative operation on the segment checksums;

means for cryptographically protecting the first commutative checksum by using at least one cryptographic operation, by forming a cryptographic commutative checksum;

means for subjecting the cryptographic commutative checksum to an inverse cryptographic operation to form a first reconstructed cryptographic checksum;

means for forming a second segment checksum for each data segment of the digital data to which the first commutative checksum is allocated;

means for forming a second commutative checksum by a commutative operation on the second segment checksums; and

means for checking the second commutative checksum for a match with the first reconstructed commutative checksum.

A considerable advantage of the methods and of the arrangements can be seen in the fact that, by using a commutative operation for individual checksums of the data segments, a flow control for the order of the individual data segments is no longer required.

Furthermore, it is no longer required to reassemble the complete user data in the original order in which the first commutative checksums were formed. The order of the individual data segments is no longer of significance in the formation of the commutative checksum.

If the digital data are transmitted between two arrangements, a further advantage of the methods can be seen in the fact that the checking of the integrity can already be begun before all data segments have been received since it is no longer required to maintain the original order in forming the first checksum. This leads to a timesaving in the checking of the integrity of the data.

Illustratively, the invention can be seen in the fact that a checksum is formed in the case of a number of data segments which, together, form the data to be protected, and the individual checksums of the data segments are commutatively combined with one another.

Advantageous further developments of the invention are obtained from the dependent claims.

It is advantageous to protect the first commutative checksum cryptographically by using at least one cryptographic operation.

The result of this further development is that the cryptographic security of the data is considerably increased. A cryptographic operation in this sense is, for example, the encrypting of the first commutative checksum with a symmetric or also with an assymetric encryption method which forms a cryptographic checksum. On the receiver side, the inverse cryptographic method to the cryptographic method is performed in order to ensure cryptographic security.

To form a checksum within the context of the document, various possibilities are known:

- a checksum can be formed by forming hashing values for the individual data segments;

- the checksums can also be formed by so-called cyclic codes (Cyclic Redundancy Check, CRC);
- a cryptographic one-way function can also be used for forming the checksums for the data segments.

The methods can be advantageously used in various application scenarios.

The methods can be used both in the transmission of digital data for protection against manipulation of the data, and in the archiving of digital data in a computer in which the first commutative checksum is formed and stored together with the data to be archived. The first commutative checksum can be checked when the digital data are loaded from the archive memory in order to detect any manipulation of the archived data.

The method can be advantageously used for protecting digital data, the data segments of which are not tied to an order. Examples of such data segments are packet-oriented communication protocols, for example network management protocols such as the Simple Network Management Protocol (SNMP) or the Common Management Information Protocol (CMIP).

In the text which follows, an illustrative embodiment of the invention will be explained in greater detail with reference to a Figure. Even if the illustrative embodiment is explained with reference to the Simple Network Management Protocol (SNMP) in the text which follows, this does not represent any restriction on the applicability of the method. The method can be used whenever it is of importance to ensure integrity protection for digital data which are grouped into a number of data segments.

The Figure shows two arrangements, data segments being transmitted from the first arrangement to the second arrangement.

In the Figure, a first computer arrangement A1, in which data segments (Di, i = 1 .. n) are stored, is shown symbolically. The data segments Di together form the digital data which are also designated as user data, for which it is of importance to ensure their integrity.

Both the first computer arrangement A1 and a second computer arrangement A2 described in the text which follows in each case contain an arithmetic and logic unit R which is arranged in such a manner that the method steps described in the text which follows are performed.

In the first arrangement A1, the data segments Di are arranged at positions Pi within the total data stream. For each data segment Di, a first segment checksum PSi is [lacuna] by using a checksum function PSF. The individual first segment checksum PSi are combined to form a first commutative checksum KP1 by a commutative operation as defined and described in [2]. The commutative operation on the individual checksums PSi are shown symbolically by an EXOR symbol ⊕ in the Figure.

The first commutative checksum KP1 is subjected to a cryptographic method, a symmetric or asymmetric method, by using a first cryptographic key S (step 101). The result of the cryptographic operation is a cryptographic checksum KP.

Both the data segments Di and the cryptographic checksum KP are transmitted by a transmission medium, preferably a line or also a logical connection which is symbolically shown by a communication link UM in the Figure, to a second arrangement A2 where they are received.

The crossing arrows of the data segments Di in the Figure indicate that, due to the transmission of the data segments Di, these are received in positions Pj (j = a .. z) which are displaced compared with the order in the first arrangement A1.

Thus, a data segment D2 at the first position P1 is received as data segment Da in the second arrangement A2. Data segment D1 is received as data segment Dc in the second arrangement. Data segment Dn is received as received data segment Db at the second position P2 in the second arrangement A2.

In accordance with the method used, either the first cryptographic key S is used for performing the inverse cryptographic operation on the cryptographic checksum KP if a symmetric encryption method is used, or a second cryptographic key S′ is used if an asymmetric cryptographic method is used.

The result of the inverse cryptographic operation (step 102) is again the first commutative checksum KP1 with correct encryption and decryption.

This checksum is stored in the second arrangement A2. For the comparison of the data segments Dj, which are now received in permutated order compared with the original order during the formation of the first commutative checksum KP1, second segment checksums Psj are formed for the received data segments Dj, again using the same checksum methods PSF.

The resultant second checksums PSj are again commutatively combined with one another to form a second commutative checksum KP2.

In a further step 103, a check is made whether the first commutative checksum KP1 matches the second commutative checksum KP2.

If this is so, the integrity of the data segments Di, and thus the integrity of all the digital data, is ensured (step 104) if the cryptographic methods used or, respectively, the methods used for forming checksums ensure the corresponding cryptographic security.

If the first cryptographic checksum KP1 does not match the second cryptographic checksum KP2, the integrity of the data segments Di would be violated and a manipulation of the data is found and preferably reported to a user of the system.

The protocol data units (PDU) in SNMP are structured in such a manner that the user information (so-called variable bindings) can contain a list of objects (object indicators, OID/value pairs). The order of the objects within a PDU is not specified so that it is possible for a permutation of the objects to occur during the transmission of the PDUs between the first arrangement A1 and the second arrangement A2. The invention now makes it possible to form a single cryptographic checksum over all objects of an SNMP PDU without having to take into consideration the order of the objects or of the PDUs.

In the text which follows, alternatives to the illustrative embodiment described above will be explained.

The method for forming the checksum PSF can be, for example, a method for forming hashing values. However, methods for forming cyclic codes (Cyclic Redundancy Check, CRC) using feedback-type shift registers can also be used. In addition, cryptographic one-way functions can be used for forming the checksums PSi and, respectively, Psj.

Furthermore, the commutative operation can have the additional property of associativity.

Both the method for forming the checksum and the method for checking a checksum can be performed independently of one another. However, the method for forming the checksum and the method for checking the checksum can also be performed jointly.

Furthermore, it is provided not to transmit digital data but to archive the digital data, that is to say to store them in the first arrangement A1, together with the first commutative checksum KP1. When the archived data are reused, that is to say when the data segments Di are loaded from the memory of the first arrangement A1, the method for checking the first commutative checksum KP1 as described above will then be performed. The first arrangement A1 and the second arrangement A2 can thus be identical.

Illustratively, the invention can be seen in that in the case of a number of data segments which, together, represent the data to be protected, a checksum is formed for each data segment and the individual checksums of the data segments are commutatively combined with one another. This makes it possible to form and to check a checksum without having to take into consideration the order of the data segments.

In this document, the following publications have been quoted:

[1] W. Stallings, Sicherheit in Netzwerk und Internet (Security in Network and Internet), Prentice Hall, ISBN 3-930436-29-9, pp. 203-223, 1995

[2] K.-H. Kiyek and F. Schwarz, Mathmatik für Informatiker (Mathematics for Computer Scientists), Teubner Verlag, ISBN 3-519-03277-X, pp. 11-13, 1989

[3] DE-A 2 048 365

The claims defining the invention are as follows:

1. A method of forming a first commutative checksum for digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

a segment checksum is formed for each data segment;

the first commutative checksum is formed by a commutative operation on the segment checksums; and

the first commutative checksum is cryptographically protected by using at least one cryptographic operation.

2. A method of checking a predetermined cryptographic commutative checksum which is allocated to digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

the cryptographic commutative checksum is subjected to an inverse cryptographic operation to form a first cryptographic checksum;

a second segment checksum is formed for each data segment;

a second commutative checksum is formed by a commutative operation on the second segment checksums; and

the second commutative checksum is checked for a match with the first commutative checksum.

3. A method of forming and checking a first commutative checksum for digital data which is grouped into a number of data segments, by a computer, said method comprising the steps of:

a segment checksum is formed for each data segment;

the first commutative checksum is formed by a commutative operation on the segment checksums;

the first commutative checksum is cryptographically protected by using at least one cryptographic operation, a cryptographic commutative checksum being formed;

the cryptographic commutative checksum is subjected to an inverse cryptographic operation to form a first reconstructed cryptographic checksum;

a second segment checksum is formed for each data segment of the digital data to which the first commutative checksum is allocated;

a second commutative checksum is formed by a commutative operation on the second segment checksums; and

the second commutative checksum is checked for a match with the first reconstructed commutative checksum.

4. The method according to any one of claims 1 to 3, wherein the segment checksums are formed in accordance with at least one of the following types:

- forming a hashing value,
- forming CRC codes,
- using at least one cryptographic one-way function.

5. The method according to any one of claims 1 to 4, wherein the cryptographic operation is a symmetric cryptographic method.

6. The method according to any one of claims 1 to 4, wherein the cryptographic operation is an asymmetric cryptographic method.

7. The method according to any one of claims 1 to 6, wherein the commutative operation exhibits the property of associativity.

8. The method according to any one of claims 1 to 7, wherein digital data are protected, the data segments of which are not tied to an order.

9. The method according to any one of claims 1 to 7, wherein said digital data is processed in accordance with a network management protocol.

10. An apparatus for forming a first commutative checksum for digital data which is grouped into a number of data segments, said apparatus comprising:

means for forming a segment checksum for each data segment;

means for forming the first commutative checksum by a commutative operation on the segment checksums; and

means for cryptographically protecting the first commutative checksum by using at least one cryptographic operation.

11. An apparatus for checking a predetermined first commutative checksum which is allocated to digital data which is grouped into a number of data segments, said apparatus comprising:

means for subjecting the cryptographic commutative checksum to an inverse cryptographic operation to form a first cryptographic checksum;

means for forming a second segment checksum for each data segment;

means for forming a second commutative checksum by a commutative operation on the second segment checksum; and

means for checking the second commutative checksum for a match with the first commutative checksum.

12. An apparatus for forming and checking a first commutative checksum for digital data which is grouped into a number of data segments, said apparatus comprising:

means for forming a segment checksum for each data segment;

means for forming the first commutative checksum by a commutative operation on the segment checksums;

means for cryptographically protecting the first commutative checksum by using at least one cryptographic operation, by forming a cryptographic commutative checksum;

means for subjecting the cryptographic commutative checksum to an inverse cryptographic operation to form a first reconstructed cryptographic checksum;

means for forming a second segment checksum for each data segment of the digital data to which the first commutative checksum is allocated;

means for forming a second commutative checksum by a commutative operation on the second segment checksums; and

means for checking the second commutative checksum for a match with the first reconstructed commutative checksum.

13. The apparatus according to any one of claims 10 to 12, wherein the segment checksums are formed in accordance with at least one of the following types:

- forming a hashing value,
- forming CRC codes,
- using at least one cryptographic one-way function.

14. The apparatus according to any one of claims 10 to 13, wherein the cryptographic operation is a symmetric cryptographic method.

15. The apparatus according to any one of claims10 to 13, wherein the cryptographic operation is an asymmetric cryptographic method.

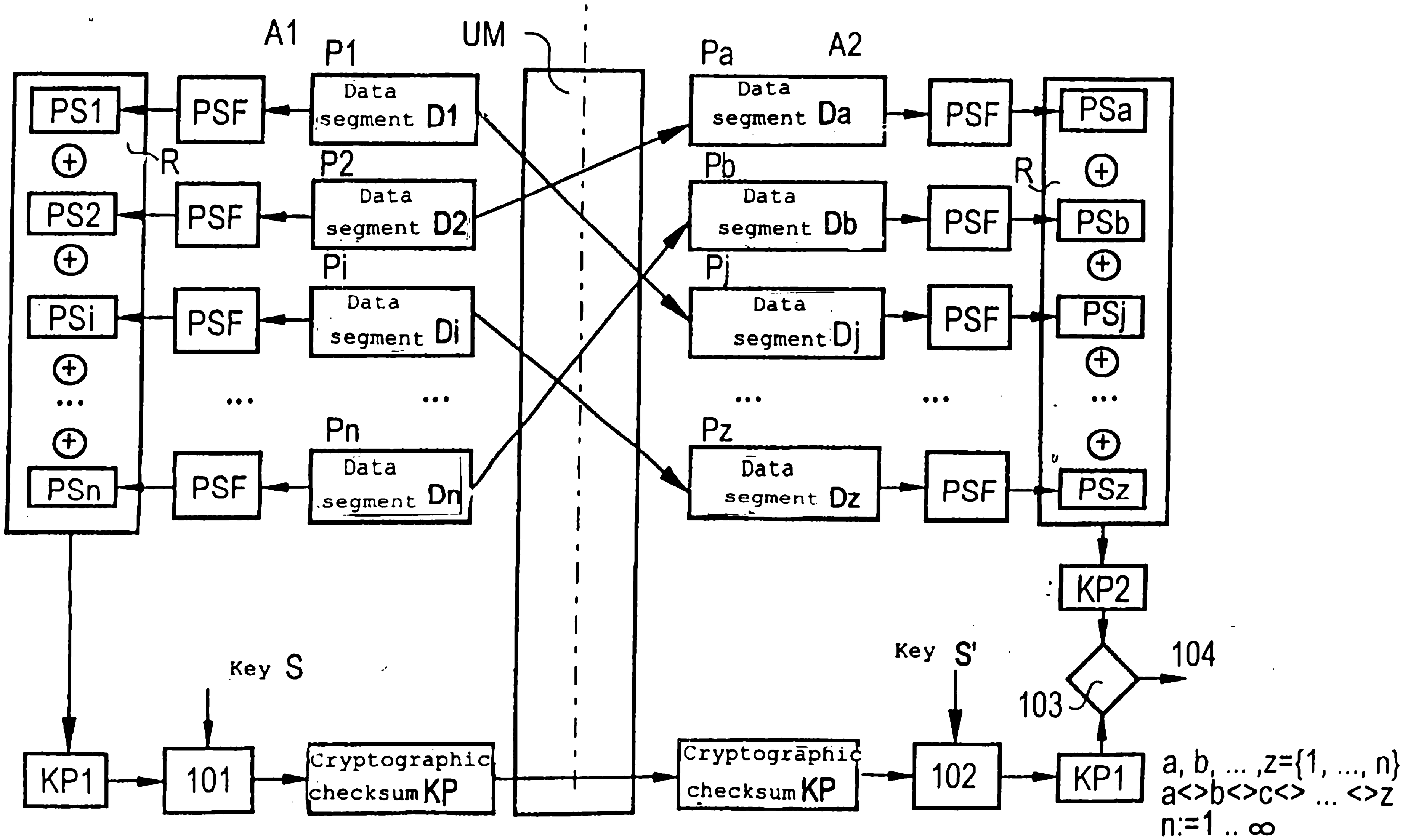
16. The apparatus according to any one of claims 10 to 15, wherein the commutative operation exhibits the property of associativity.

17. The apparatus according to any one of claims 10 to 16, wherein the digital data are protected such that the data segments are not tied to an order.

18. The apparatus according to any one of claims 10 to 16, wherein the digital data is processed in accordance with a network management protocol.

19. A method for forming a first commutative checksum for digital data which is grouped into a number of digital data segments, said method being substantially as hereinbefore described with reference to the accompanying drawing.

DATED this thirteenth Day of June 2000
Siemens Aktiengesellschaft
Patent Attorneys for the Applicant
SPRUSON & FERGUSON


A1
P1
UM
Pa
A2
PS1
PSF
Data segment D1
Data segment Da
PSF
PSa
R
P2
Pb
R
PS2
PSF
Data segment D2
Data segment Db
PSF
PSb
Pi
Pj
PSi
PSF
Data segment Di
Data segment Dj
PSF
PSj
...
Pn
Pz
PSn
PSF
Data segment Dn
Data segment Dz
PSF
PSz
KP2
104
Key S
Key S'
103
KP1
101
Cryptographic checksum KP
Cryptographic checksum KP
102
KP1
a, b, ... ,z={1, ..., n}
a<>b<>c<> ... <>z
n:=1 .. ∞