

Padding Chunk and Parameter for the Stream Control Transmission Protocol (SCTP)

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This document defines a padding chunk and a padding parameter and describes the required receiver side procedures. The padding chunk is used to pad a Stream Control Transmission Protocol (SCTP) packet to an arbitrary size. The padding parameter is used to pad an SCTP INIT chunk to an arbitrary size.

Table of Contents

1.	Introduction	2
2.	Conventions	2
3.	Padding Chunk (PAD)	2
4.	Padding Parameter (PAD)	3
5.	IANA Considerations	4
5.1.	A New Chunk Type	4
5.2.	A New Parameter Type	4
6.	Security Considerations	4
7.	Acknowledgments	4
8.	References	5
8.1.	Normative References	5
8.2.	Informative References	5

1. Introduction

This document defines a padding chunk and a padding parameter and describes the required receiver side procedures. The padding chunk is used to pad an SCTP packet to an arbitrary size. The padding parameter is used to pad an SCTP INIT chunk to an arbitrary size. The usage of the PAD chunk for path MTU discovery is described in PMTU [4]. The inappropriate usage of the PAD parameter or PAD chunk can result in wasted bandwidth.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL", when they appear in this document, are to be interpreted as described in RFC 2119 [1].

3. Padding Chunk (PAD)

This chunk is used to pad an SCTP packet. A PAD chunk can be used to enlarge the packet by 4 to 65536 bytes in steps of 4 bytes. An SCTP packet MAY contain multiple PAD chunks.

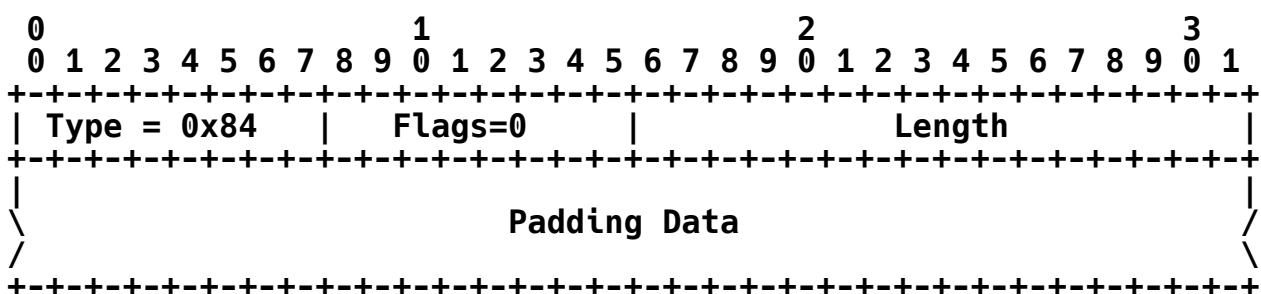


Figure 1

Type: 1 byte (unsigned integer)

This value MUST be set to 0x84 for all PAD chunks.

Flags: 1 byte (unsigned integer)

This value SHOULD be set to zero on transmit and MUST be ignored on receipt.

Length: 2 bytes (unsigned integer)

This value holds the length of the Padding Data plus 4.

Padding Data: n bytes (unsigned integer)

This holds the Padding Data. The Padding Data **MUST** be ignored by the receiver.

The receiver of the PAD chunk **MUST** discard this chunk and continue processing the rest of the chunks in the packet. Please note that this is also the required processing behavior for any unknown chunk having the same highest-order two bits of the type as the PAD chunk.

4. Padding Parameter (PAD)

This parameter is used to pad an INIT chunk. A PAD parameter can be used to enlarge the INIT chunk by 4 bytes as the minimum to the maximum size of the INIT chunk in steps of 4 bytes. An INIT chunk **MAY** contain multiple PAD parameters.

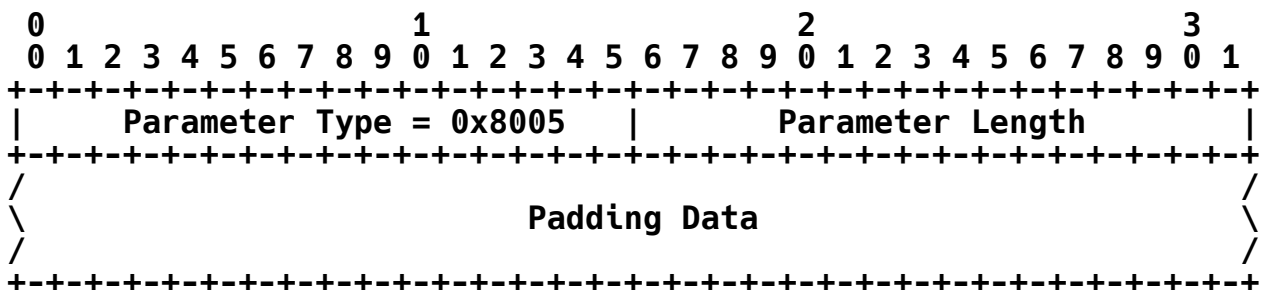


Figure 2

Parameter Type: 2 bytes (unsigned integer)

This value **MUST** be set to 0x8005.

Parameter Length: 2 bytes (unsigned integer)

This value holds the length of the Padding Data plus 4.

The PAD parameter **MAY** be included only in the INIT chunk. It **MUST NOT** be included in any other chunk. The receiver of the PAD parameter **MUST** silently discard this parameter and continue processing the rest of the INIT chunk. This means that the size of the generated COOKIE parameter in the INIT-ACK **MUST NOT** depend on the existence of the PAD parameter in the INIT chunk. A receiver of a PAD parameter **MUST NOT** include the PAD parameter within any State Cookie parameter it generates.

5. IANA Considerations

This document is the reference for all registrations described in this section. All registrations have been listed in the document available at sctp-parameters [3]. The changes are described below.

5.1. A New Chunk Type

A chunk type for the PAD chunk has been assigned by IANA. The value has been assigned as described in Figure 1. The following has been added to the "CHUNK TYPES" table of sctp-parameters [3]:

CHUNK TYPES

ID Value	Chunk Type	Reference
-----	-----	-----
132(0x84)	Padding Chunk (PAD)	[RFC4820]

5.2. A New Parameter Type

A parameter type has been assigned for the PAD parameter by IANA. The value has been assigned as described in Figure 2. The following has been added to the "CHUNK PARAMETER TYPES" table in sctp-parameters [3]:

INIT Chunk Parameter Types

Chunk Parameter Type	Value
-----	-----
Padding	32773(0x8005)

6. Security Considerations

This document does not add any additional security considerations to the ones given in RFC 2960 [2].

7. Acknowledgments

The authors wish to thank Matthew J. Zekauskas and Lars Eggert for their invaluable comments.

8. References

8.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., and V. Paxson, "Stream Control Transmission Protocol", RFC 2960, October 2000.

8.2. Informative References

- [3] "IANA registry",
<<http://www.iana.org/assignments/sctp-parameters>>.
- [4] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

Authors' Addresses

Michael Tuexen
Muenster Univ. of Applied Sciences
Stegerwaldstr. 39
48565 Steinfurt
Germany

EMail: tuexen@fh-muenster.de

Randall R. Stewart
Cisco Systems, Inc.
4875 Forest Drive
Suite 200
Columbia, SC 29206
USA

EMail: rrs@cisco.com

Peter Lei
Cisco Systems, Inc.
955 Happfield Dr.
Arlington Heights, IL 60004
US

Phone: +1 773 695-8201
E-Mail: peterlei@cisco.com

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.