

**YD**

# 中华人民共和国通信行业标准

YD/T 1444-2006

---

## 流控制传送协议(SCTP)测试方法

Test methods for stream control transmission protocol(SCTP)

2006-05-31 发布

2006-10-01 实施

---

中华人民共和国信息产业部 发布

## 目 次

前 言	II
1 范围	1
2 规范性的引用文件	1
3 测试结构	1
4 测试项目	1
4.1 测试内容中的消息和参数	1
4.2 关于测试项目中地址格式的说明	1
4.3 关于测试项目中原语要求的说明	1
4.4 测试判定原则	1
4.5 测试项目	2
4.5.1 偶联建立	2
4.5.2 偶联中止	30
4.5.3 无效消息处理	46
4.5.4 重复消息的处理	60
4.5.5 故障处理	71
4.5.6 差错消息处理	85
4.5.7 数据块的捆绑和复用	91
4.5.8 DATA 消息的传送	99
4.5.9 数据块的证实	114
4.5.10 异常情况的测试	117
4.5.11 重传定时器	120

## 前 言

本标准非等效采用 IETF RFC 2960《流控制传送协议》(2000年10月)。与 RFC 2960“流控制传送协议(2000年10月)”的主要区别如下:

1. 增加了关于测试项中原语的说明。测试项中的原语只作为说明用,在实际测试中不予要求。同时本文件中还增加了对本测试规范的评判标准。

2. 对使用 IPv6 地址进行测试的情况进行了说明;同时对于地址格式的测试,建议应依据现有网络使用的 IP 地址的状况;建议不使用主机名地址格式。

3. 在偶联建立的测试中,明确了 INIT 消息中出局流数量不可以为 0。

4. 对于由 HB-Request 原语产生的 HeartBeat 消息,在收到的 HeartBeat ACK 不正确时, SCTP 不必重发 HeartBeat 消息。

5. 如果用于偶联建立的 INIT 消息中支持的地址类型不为接收方所支持时,被测系统应当发送 ABORT 消息。

本标准是 YD/T 1194-2002《流控制传送协议(SCTP)》的配套标准。

本标准由中国通信标准化协会提出并归口。

本标准起草单位:信息产业部电信研究院

中兴通讯股份有限公司

华为技术有限公司

上海贝尔阿尔卡特股份有限公司

本标准主要起草人:吕 军 续合元 张 宜 高 峰 连 超 林 铭 吕 严

## 流控制传送协议 (SCTP) 测试方法

### 1 范围

本标准规定了对流控制传送协议 (SCTP) 的测试程序和测试方法, 并对需要在测试过程中注意的问题和在实际使用本标准时的判定原则进行了规定。

本标准适用于采用 SCTP 协议的软交换设备、信令网关和 IP 网内的相关设备。

### 2 规范性的引用文件

下列文件中的条款通过在本标准中引用而成为本标准的条款, 凡是注日期的引用文件, 其随后所有的修改单 (不包括勘误的内容) 或修订版均不适用于本标准, 然而鼓励根据本标准达成协议的各方研究是否可以适用这些文件的最新版本。凡是不注日期的引用文件, 其最新版本适用于本标准。

YD/T 1194-2002	流控制传送协议 (SCTP)
RFC 2960	流控制传送协议 (2000 年 10 月)

### 3 测试结构

测试结构如图 1 所示。

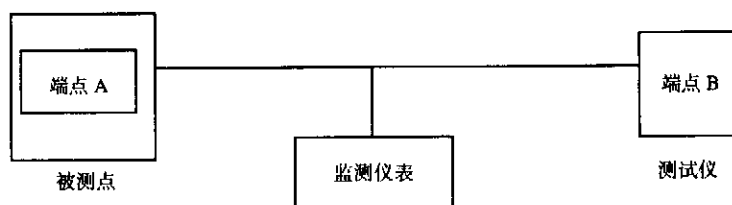


图 1 SCTP 测试结构

### 4 测试项目

#### 4.1 测试内容中的消息和参数

对测试项目中涉及到的消息、参数和原语的格式说明, 可参见 YD/T 1194 - 2002。

#### 4.2 关于测试项目中地址格式的说明

在与地址格式相关的测试中, 地址格式的应用应根据网络的应用情况进行测试, 即当前网络主要应用 IPv4 地址时, 本标准只对 IPv4 地址做要求。只有当网络中使用 IPv6 地址时, 则本标准应要求的地址格式包括 IPv4 和 IPv6 两种格式。

当网络可以支持 DNS 服务时, 方可考虑使用主机名地址。为了防备对 DNS 的攻击, 建议在 SCTP 层应用时不使用主机名地址。

#### 4.3 关于测试项目中原语要求的说明

测试项目中描述的高层应用同 SCTP 层之间原语, 主要用来说明不同层之间的交互关系, 在实际测试中可以不进行要求。

#### 4.4 测试判定原则

测试项目编号上标记为任选的项目, 在进行入网测试时可以根据实际情况不做要求, 其余项目是人网测试的必备项目。当被测设备的所有必备项目合格后, 该设备方满足设备进网要求。

4.5 测试项目

4.5.1 偶联建立

测试编号: 1.1	
参考: RFC 2960 第5.1节	
项目: 偶联建立	
分项目: 正常的偶联建立	
目的: 检验正常的偶联建立程序	
测试预置条件: 端点A和端点B之间的偶联未建立, 设置数据由A点的高层发送建立Associate原语, 建立到端点B偶联	
结构: 1	测试类型: VAT、CPT
<p>预期的信号单元顺序:</p>	
测试说明:	
1.	由端点A的高层协议 ( ULP ) 发送Associate原语, 启动偶联建立, 并用监视仪记录消息流程。
2.	检查A: 偶联是否建立?
3.	检查B: 偶联建立后发送的第一个数据块 ( Data Chunk ) 是否由SACK证实?
4.	检查C: 在已建立的偶联上接收和发送不同长度的数据块。
5.	检查D: 所有的DATA消息都被正确接收, 无消息丢失。
6.	对于有效性测试, 在反方向上重复测试。

测试编号: 1.2	
参考: RFC 2960 第4节和5.1.6节	
项目: 正常的偶联建立	
分项目: T1 - Init 定时器	
目的: 验证T1 - Init 定时器超时后, 并重发INIT消息	
测试预置条件: 端点A和端点B之间的偶联未建立, 设置数据由A点的高层发送建立Associate原语, 建立到端点B偶联, 在端点B设置数据让B不响应INIT-ACK	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
<pre> sequenceDiagram     participant ULP as 高层应用(ULP)     participant A as 端点 A     participant B as 端点 B      ULP-&gt;&gt;A: Associate     A-&gt;&gt;B: INIT     A: 启动定时器T1 - Init     A: 定时器T1 - Init超时     A-&gt;&gt;B: INIT   </pre>	
测试说明:	
1.	由端点A发送INIT消息, 尝试建立到端点B的偶联。用监视仪记录消息流程。
2.	检查A: 在定时器T1 - Init超时后, 端点A重新发送INIT消息。
3.	检查B: 重发的INIT消息的Init-Tag的值与前一个INIT消息的Init-Tag值相同。
4.	检查C: 消息顺序是否如上所示?

测试编号: 1.3																												
参考: RFC 2960 第4节和5.1.6节																												
项目: 正常的偶联建立																												
分项目: COOKIE-ECHO 使用的T1 - Cookie定时器																												
目的: 验证T1 - Cookie定时器超时后, 并重发INIT消息																												
测试预置条件: 端点A和端点B之间的偶联未建立, 设置数据由A点的高层发送建立Associate原语, 建立到端点B偶联, 在端点B设置数据让B不发送COOKIE - ACK响应COOKIE-ECHO																												
结构: 1	测试类型: VAT																											
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 40%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">INIT</td> <td></td> </tr> <tr> <td style="text-align: center;">INIT ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">COOKIE_ECHO</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">启动T1 - Cookie定时器</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">T1 - Cookie定时器超时</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">COOKIE_ECHO</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">重新启动T1 - Cookie定时器</td> <td></td> </tr> </table>		端点 B	端点 A	高层应用(ULP)			<===== Associate		INIT		INIT ACK	----->			COOKIE_ECHO			启动T1 - Cookie定时器			T1 - Cookie定时器超时			COOKIE_ECHO			重新启动T1 - Cookie定时器	
端点 B	端点 A	高层应用(ULP)																										
		<===== Associate																										
	INIT																											
INIT ACK	----->																											
	COOKIE_ECHO																											
	启动T1 - Cookie定时器																											
	T1 - Cookie定时器超时																											
	COOKIE_ECHO																											
	重新启动T1 - Cookie定时器																											
测试说明:																												
1.	由端点A发送INIT消息, 尝试建立到端点B的偶联, 用监视仪记录消息流程。																											
2.	检查A: 在定时器T1 - Cookie超时后, 端点A重新发送COOKIE_ECHO消息。																											
3.	检查B: 消息顺序是否如上所示?																											

测试编号: 1.4	
参考: RFC 2960 第4节	
项目: 正常的偶联建立	
分项目: 超过了MAX.INIT.RETRANS, 未能收到INIT-ACK	
目的: 验证如果重发的INIT次数超过了MAX.INIT.RETRANS, 则应中止偶联建立	
测试预置条件: 端点A和B之间未建立偶联, 在端点B设置数据, 让B不响应INIT-ACK	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A                      高层应用(ULP)
	<===== Associate
<-----	INIT
	启动定时器T1 - Init
	定时器T1 - Init超时
<-----	INIT
	启动定时器T1 - Init
	定时器T1 - Init超时
<-----	INIT
:	:
重发INIT的次数超过了 MAX.INIT.RETRANS	Communications Lost    =====>
	<===== Associate
<-----	INIT
INIT ACK    ----->	
<-----	COOKIE_ECHO
COOK_ACK    ----->	Communication up    =====>
DATA        ----->	
<-----	SACK
测试说明:	
1.	通过由端点A发送INIT尝试建立到端点B的偶联, 用监视仪记录消息流程。
2.	检查A: 如果INIT消息重复发送了MAX.INIT.RETRANS次, 而未能收到INIT-ACK, 则应中止偶联建立并向高层报告。
3.	检查B: 验证端点A可以重新尝试建立到端点B的一个新偶联。
4.	检查C: 消息顺序是否如上所示?



测试编号： 1.5	
参考： RFC 2960 第4节和5.1.6节	
项目： 正常的偶联建立	
分项目： 超过了MAX.INIT.RETRANS， 未能收到COOKIE - ACK	
目的： 验证如果重发的COOKIE - ECHO次数超过了MAX.INIT.RETRANS， 则应中止偶联建立	
测试预置条件： 端点A和B之间未建立偶联， 在端点B设置数据， 使B不响应COOKIE - ACK	
结构： 1	测试类型： VAT
预期的信号单元顺序：	
端点 B	端点 A <span style="float: right;">高层应用(ULP)</span>
	<==== Associate
INIT-ACK	INIT ←-----
←-----	----->
←-----	COOKIE-ECHO 启动定时器T1 - Cookie 定时器T1 - Cookie超时
←-----	COOKIE-ECHO 启动定时器T1 - Cookie 定时器T1 - Cookie超时
←-----	COOKIE-ECHO
:	:
重发INIT的次数超过了 MAX.INIT.RETRANS	Communications Lost =====>
	<==== Associate
INIT ACK	INIT ←-----
←-----	----->
←-----	COOKIE_ECHO Communication up =====>
COOK_ACK DATA	←-----
←-----	SACK
测试说明：	
1.	通过由端点A发送INIT尝试建立到端点B的偶联，用监视仪记录消息流程。
2.	检查A： 如果COOKIE-ECHO消息重复发送了MAX.INIT.RETRANS次，而未能收到COOKIE-ACK，则应中止偶联建立并向高层报告。
3.	检查B： 验证端点A可以重新尝试建立到端点B的一个新偶联。
4.	检查C： 消息顺序是否如上所示？

测试编号：1.6	
参考：RFC 2960 第5.1节	
项目：正常的偶联建立	
分项目：未能收到COOKIE-ECHO消息	
目的：如果没有收到COOKIE-ECHO消息，检验端点处于关闭状态，且未对该偶联分配资源	
测试预置条件：端点A和B之间未建立偶联，在端点B设置数据，让B不响应COOKIE-ECHO	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
INIT	----->
	<-----
	INIT-ACK(带有Cookie)
	<===== Associate
	<-----
INIT ACK	----->
	<-----
	INIT
COOKIE_ACK	----->
	<-----
	COOKIE_ECHO
	Communication up =====>
DATA	----->
	<-----
	SACK
测试说明：	
1.	从端点B启动到端点A的偶联建立，在收到INIT-ACK后，端点B不响应COOKIE-ECHO，用监视仪记录消息流程。
2.	检查A：端点B处于关闭状态。
3.	检查B：消息顺序是否如上所示。
4.	检查C：端点A能否尝试建立一个新的偶联。

测试编号: 1.7																																																																							
参考: RFC 2960 第5.3.1节																																																																							
项目: 正常的偶联建立																																																																							
分项目: 发起偶联建立的端点在建立不同的偶联应使用不同的启动标签值																																																																							
目的: 验证当重新建立到对端的偶联时, 需要使用不同的Init-Tag值, 即与先前的偶联的Init-Tag值不同																																																																							
测试预置条件: 端点间没有建立的偶联, 在端点B设置数据, 偶联可以在端点A和B之间建立并终止																																																																							
结构: 1	测试类型: VAT																																																																						
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%;"></td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">INIT ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">COOKIE_ECHO</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">Communication up</td> <td style="text-align: center;">=====</td> <td style="text-align: center;">Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">DATA</td> <td style="text-align: center;">&lt;=====</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">DATA Arrive</td> <td style="text-align: center;">=====</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">Communication Lost</td> <td style="text-align: center;">=====</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT (不同的Init-Tag)</td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">Associate</td> </tr> <tr> <td style="text-align: center;">INIT ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)				<=====	Associate		<-----	INIT			INIT ACK	----->					<-----	COOKIE_ECHO			COOKIE_ACK	----->	Communication up	=====	Send		<-----	DATA	<=====		SACK	----->				DATA	----->	DATA Arrive	=====			<-----	SACK			SHUTDOWN	----->	SHUTDOWN ACK				<-----	Communication Lost	=====			<-----	INIT (不同的Init-Tag)	<=====	Associate	INIT ACK	----->			
端点 B		端点 A		高层应用(ULP)																																																																			
			<=====	Associate																																																																			
	<-----	INIT																																																																					
INIT ACK	----->																																																																						
	<-----	COOKIE_ECHO																																																																					
COOKIE_ACK	----->	Communication up	=====	Send																																																																			
	<-----	DATA	<=====																																																																				
SACK	----->																																																																						
DATA	----->	DATA Arrive	=====																																																																				
	<-----	SACK																																																																					
SHUTDOWN	----->	SHUTDOWN ACK																																																																					
	<-----	Communication Lost	=====																																																																				
	<-----	INIT (不同的Init-Tag)	<=====	Associate																																																																			
INIT ACK	----->																																																																						
测试说明:																																																																							
1.	在端点A和B启动偶联建立, 该INIT中包含一个启动标签 (Init-Tag), 用SHUTDOWN消息终止该偶联, 并重新启动偶联建立。																																																																						
2.	检查A: 第二个偶联建立的INIT消息中包含的启动标签应与上一个启动标签值不同。																																																																						

测试编号: 1.8																																																	
参考: RFC 2960 第5.3.1节																																																	
项目: 正常的偶联建立																																																	
分项目: 接收偶联启动命令的端点在建立不同偶联时应使用不同的启动标签值																																																	
目的: 验证当收到对端的重新建立偶联的命令时, 需要使用不同的Init-Tag值, 即与先前的偶联的Init-Tag值不同																																																	
测试预置条件: 端点间没有建立的偶联, 在端点B设置数据, 偶联可以在端点A和B之间建立并终止																																																	
结构: 1	测试类型: VAT																																																
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <thead> <tr> <th style="text-align: left;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT ACK ( Init-Tag )</td> </tr> <tr> <td>COOKIE_ECHO</td> <td style="text-align: center;">&lt;-----</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">-----&gt;</td> <td>COOKIE_ACK</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Communication UP =====&gt;</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== SEND</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA</td> </tr> <tr> <td>SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td>DATA</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: right;">DATA Arrive =====&gt;</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK</td> </tr> <tr> <td>SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Communication Lost =====&gt;</td> </tr> <tr> <td>INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT ACK( 不同的Init-Tag )</td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	INIT	----->			<-----	INIT ACK ( Init-Tag )	COOKIE_ECHO	<-----			----->	COOKIE_ACK			Communication UP =====>			<===== SEND		<-----	DATA	SACK	----->		DATA	----->	DATA Arrive =====>		<-----	SACK	SHUTDOWN	----->			<-----	SHUTDOWN ACK			Communication Lost =====>	INIT	----->			<-----	INIT ACK( 不同的Init-Tag )
端点 B	端点 A	高层应用(ULP)																																															
INIT	----->																																																
	<-----	INIT ACK ( Init-Tag )																																															
COOKIE_ECHO	<-----																																																
	----->	COOKIE_ACK																																															
		Communication UP =====>																																															
		<===== SEND																																															
	<-----	DATA																																															
SACK	----->																																																
DATA	----->	DATA Arrive =====>																																															
	<-----	SACK																																															
SHUTDOWN	----->																																																
	<-----	SHUTDOWN ACK																																															
		Communication Lost =====>																																															
INIT	----->																																																
	<-----	INIT ACK( 不同的Init-Tag )																																															
测试说明:																																																	
1.	由端点B启动端点A和B之间的偶联建立, 该端点A发送的INIT-ACK中包含一个启动标签 ( Init-Tag ), 由端点B用SHUTDOWN消息终止该偶联, 并重新启动偶联建立。																																																
2.	检查A: 第二个偶联建立的INIT-ACK消息中包含的启动标签应与上一个启动标签值不同。																																																

测试编号: 1.9																																														
参考: RFC 2960 第3.3.2节																																														
项目: 正常的偶联建立																																														
分项目: INIT消息中的任选参数																																														
目的: 验证端点A在收到的INIT消息中包含一个或多个任选参数时的处理																																														
测试预置条件: 端点A和B之间的偶联未建立, 在B点设置数据, 在B点发送的INIT消息中包含所有的任选参数, 端点A也应支持IPv6的地址																																														
结构: 1	测试类型: VAT																																													
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 40%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: left; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ECHO</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">COOKIE_ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">Communication UP</td> <td style="text-align: center;">=====&gt;</td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">DATA Arrive</td> <td style="text-align: center;">=====&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	INIT	----->					<-----	INIT ACK			COOKIE_ECHO	----->					<-----	COOKIE_ACK					Communication UP	=====>		DATA	----->						DATA Arrive	=====>			<-----	SACK		
端点 B		端点 A		高层应用(ULP)																																										
INIT	----->																																													
	<-----	INIT ACK																																												
COOKIE_ECHO	----->																																													
	<-----	COOKIE_ACK																																												
		Communication UP	=====>																																											
DATA	----->																																													
		DATA Arrive	=====>																																											
	<-----	SACK																																												
测试说明:																																														
1.	由端点B启动端点A和B之间的偶联建立, 端点B发送的INIT中应包含所有的任选参数 (IPv4地址、IPv6地址、COOKIE-ECHO的有效期、支持的地址类型), 用监视仪记录消息流程。																																													
2.	检查A: INIT消息被接收, 并响应INIT-ACK。																																													
3.	检查B: 端点A和B间的偶联成功建立。																																													

测试编号: 1.10																																														
参考: RFC 2960 第3.3.3节																																														
项目: 正常的偶联建立																																														
分项目: INIT ACK消息中的任选参数																																														
目的: 验证端点A在收到的INIT ACK消息中包含一个或多个任选参数时的处理																																														
测试预置条件: 端点A和B之间的偶联未建立, 在B点设置数据, 在B点发送的INIT ACK消息中包含所有的任选参数, 端点A也应支持IPv6的地址																																														
结构: 1	测试类型: VAT																																													
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 20%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="width: 20%;"></th> <th style="text-align: left; width: 15%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">←=====</td> <td>Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>INIT</td> <td></td> <td></td> </tr> <tr> <td>INIT ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>COOKIE_ECHO</td> <td></td> <td></td> </tr> <tr> <td>COOKIE_ACK</td> <td style="text-align: center;">-----&gt;</td> <td>Communication up</td> <td style="text-align: center;">=====</td> <td>Send</td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>DATA</td> <td style="text-align: center;">←=====</td> <td>Send</td> </tr> <tr> <td>DATA</td> <td style="text-align: center;">-----&gt;</td> <td>DATA Arrive</td> <td style="text-align: center;">=====</td> <td>Send</td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>SACK</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)				←=====	Associate		←-----	INIT			INIT ACK	----->					←-----	COOKIE_ECHO			COOKIE_ACK	----->	Communication up	=====	Send		←-----	DATA	←=====	Send	DATA	----->	DATA Arrive	=====	Send		←-----	SACK		
端点 B		端点 A		高层应用(ULP)																																										
			←=====	Associate																																										
	←-----	INIT																																												
INIT ACK	----->																																													
	←-----	COOKIE_ECHO																																												
COOKIE_ACK	----->	Communication up	=====	Send																																										
	←-----	DATA	←=====	Send																																										
DATA	----->	DATA Arrive	=====	Send																																										
	←-----	SACK																																												
测试说明:																																														
1.	由端点A启动端点A和B之间的偶联建立, 端点B发送的INIT ACK中应包含所有的任选参数 (IPv4地址、IPv6地址、不识别的参数等), 用监视仪记录消息流程。																																													
2.	检查A: INIT ACK消息被接收, 并响应COOKIE-ECHO。																																													
3.	检查B: 端点A和B间的偶联被成功建立。																																													
4.	重复测试: 如果COOKIE不是INIT ACK必备参数后的第一个参数, 而是在任选参数后, 或在任选参数中, 则接收INIT-ACK。																																													

测试编号: 1.11																																	
参考: RFC 2960 第5.1.1节																																	
项目: 流参数不匹配																																	
分项目: INIT ACK中的出局流和入局流参数与INIT中的参数不匹配																																	
目的: 检验INIT ACK中的出局流和入局流参数与INIT中的参数不匹配时系统的处理																																	
测试预置条件: 端点A和B之间的偶联未建立, 且INIT消息中的出局流数量不为0																																	
结构: 1	测试类型: VAT																																
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 40%; text-align: center;">端点 B</td> <td style="width: 20%;"></td> <td style="width: 20%; text-align: center;">端点 A</td> <td style="width: 20%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">INIT (OS=X MIS=Y)</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">端点A的OS=Z, 且Z&gt;Y</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK (OS=Y MIS=X)</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ECHO</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">COOKIE_ACK</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Communication up</td> <td style="text-align: right;">=====&gt;</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;=====</td> <td style="text-align: right;">Send (流ID=Y)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Send Failure</td> <td style="text-align: right;">=====&gt;</td> </tr> </table> <p style="text-align: center;">OS = 出局流数量 MIS = 最大入局流数量</p>		端点 B		端点 A	高层应用(ULP)	INIT (OS=X MIS=Y)	----->	端点A的OS=Z, 且Z>Y			<-----	INIT ACK (OS=Y MIS=X)		COOKIE_ECHO	----->	COOKIE_ACK			<-----					Communication up	=====>			<=====	Send (流ID=Y)			Send Failure	=====>
端点 B		端点 A	高层应用(ULP)																														
INIT (OS=X MIS=Y)	----->	端点A的OS=Z, 且Z>Y																															
	<-----	INIT ACK (OS=Y MIS=X)																															
COOKIE_ECHO	----->	COOKIE_ACK																															
	<-----																																
		Communication up	=====>																														
		<=====	Send (流ID=Y)																														
		Send Failure	=====>																														
测试说明:																																	
1.	由端点B启动偶联的建立, 在发送的INIT消息中MIS为Y, 端点A的MIS小于端点B的出局流数量, 用监视仪记录消息流程。																																
2.	检查A: 端点A发送的INIT-ACK的出局流数量应等于收到的INIT中的MIS。																																
3.	检验B: 端点A的出局流ID和MIS流ID的范围应为1~(Y-1)。																																
4.	检查C: 消息顺序是否如上所示?																																
5.	注: 端点A也可以中止偶联。																																

测试编号：1.12	
参考：RFC 2960 第3.3.2节	
项目：流参数不匹配	
分项目：INIT中的出局流和入局流参数设置为0	
目的：如果收到的INIT消息中的OS或MIS字段为0，则端点A发送ABORT消息，端点处于关闭状态	
测试预置条件：端点A和B间的偶联未建立，在端点B设置数据，使得发送的INIT消息中的OS或MIS参数为0	
结构：1'	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre>       端点 B                               端点 A           高层应用(ULP)       INIT(OS=0, MIS=y)  -----&gt;                                      &lt;----- ABORT </pre>	
测试说明：	
1.	由端点B启动到端点A的偶联建立，发送的INIT消息中的OS为0，MIS可以为0以外的任意值。用监视仪记录消息流程。
2.	检查A：端点B是否响应了ABORT消息，其原因值应为“无效的必备参数”
3.	检查B：消息顺序是否如上所示？
4.	重复测试，MIS为0，OS可以为0以外的任意值。
5.	重复测试，OS和MIS均为0。



测试编号: 1.13	
参考: RFC 2960 第5.1.1节	
项目: 流参数不匹配	
分项目: INIT-ACK中的出局流参数与入局流参数不匹配	
目的: 如果INIT和INIT-ACK消息中的入局流及出局流参数不匹配, 则使用这两个参数中最小的值建立偶联	
测试预置条件: 链路不工作	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)      ULP-&gt;&gt;A: Associate     A-&gt;&gt;B: INIT (OS=Z MIS=Y)     B-&gt;&gt;A: INIT ACK (OS = Y MIS = X)     A-&gt;&gt;ULP: COOKIE_ECHO     ULP-&gt;&gt;A: Communication up     A-&gt;&gt;B: COOKIE_ACK     A-&gt;&gt;ULP: Send     ULP-&gt;&gt;A: Send(流ID=X)     A-&gt;&gt;B: DATA (流ID=X-1)     A-&gt;&gt;ULP: Send Failure     B-&gt;&gt;A: SACK     </pre>	
测试说明:	
1.	由端点A启动偶联建立, 端点B发送的INIT-ACK消息中MIS参数小于端点A发送的OS, 用监视仪记录消息流程。
2.	检查A: 端点A响应COOKIE-ECHO。
3.	检查B: 端点A的OS的流ID的范围为0~(X-1)。
4.	检查C: 消息顺序是否如上所示? 注: 端点A也可以中止偶联。

测试编号: 1.14	
参考: RFC 2960 第3.3.3节	
项目: 流参数不匹配	
分项目: INIT-ACK消息中的入局流和出局流参数为0	
目的: 如果收到的INIT-ACK消息中的OS或MIS参数为0, 则端点A需要发送ABORT消息, 且端点B处于关闭状态	
测试预置条件: 端点A和B间的偶联未建立, 且端点A的OS为Z, 在端点B设置数据, 让B响应的INIT-ACK消息中的OS或MIS参数为0	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
<p>端点 B</p> <p>INIT ACK (OS = 0 MIS = Z)</p>	<p>端点 A</p> <p>INIT (OS=Z MIS=Y)</p> <p>ABORT</p> <p>Communication Lost</p>
	<p>高层应用(ULP)</p> <p>Associate</p> <p>=====&gt;</p> <p>=====&gt;</p> <p>=====&gt;</p> <p>=====&gt;</p>
测试说明:	
1.	由端点A启动到端点B的偶联建立, 收到的INIT-ACK消息中的OS为0, MIS为INIT中的OS, 用监视仪记录消息流程。
2.	检查A: 端点A是否发送了ABORT消息?
3.	重复测试, 收到的INIT-ACK消息中的MIS为0, OS为INIT中的MIS。
4.	检查B: 端点A是否发送了ABORT消息?
5.	重复测试, 收到的INIT-ACK消息中的MIS和OS都为0。
6.	检查C: 端点A是否发送了ABORT消息?

测试编号：1.15																																	
参考：RFC 2960 第3.3.3.1节																																	
项目：正常的偶联建立																																	
分项目：INIT-ACK中有“不识别的参数”																																	
目的：如果收到的INIT消息中有不识别的TLV参数，则这些不识别的参数应当在INIT-ACK中送回																																	
测试预置条件：端点A和端点B间的偶联未建立，在端点B设置数据，使发送的INIT中包括不识别的参数类型，且参数类型的最高两个比特设置为11																																	
结构：1	测试类型：VAT																																
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: right; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td>消息中包含未定义的任 选参数</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT ACK (该消息中包 括“不识别的参数”字段)</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ECHO</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>COOKIE_ACK</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">Communication up</td> <td style="text-align: right;">=====&gt;</td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	INIT	----->	消息中包含未定义的任 选参数			<-----	INIT ACK (该消息中包 括“不识别的参数”字段)		COOKIE_ECHO	----->				<-----	COOKIE_ACK				Communication up	=====>	DATA	----->				<-----	SACK	
端点 B		端点 A	高层应用(ULP)																														
INIT	----->	消息中包含未定义的任 选参数																															
	<-----	INIT ACK (该消息中包 括“不识别的参数”字段)																															
COOKIE_ECHO	----->																																
	<-----	COOKIE_ACK																															
		Communication up	=====>																														
DATA	----->																																
	<-----	SACK																															
测试说明：																																	
1.	由端点B启动到端点A的偶联，B发送的INIT消息中，应当包括一些未定义的参数类型，这些未定义的参数类型的最高两个比特设置为11，用监视仪记录消息流程。																																
2.	检查A：端点A接收该INIT消息并响应INIT-ACK消息。																																
3.	检查B：端点B发送的INIT-ACK消息中的不识别参数字段应当包括从INIT消息中收到的未定义的参数。																																
4.	检查C：端点A和B的偶联正常建立。																																

测试编号：1.16	
参考：RFC 2960 1.4节	
项目：正常的偶联建立	
分项目：传送地址在多个偶联中	
目的：如果用于偶联建立的INIT消息中包含的一个传送地址已经在偶联中，则端点用ABORT响应该INIT消息	
测试预置条件：端点A和B间未建立偶联，在端点B设置数据，由B点发送INIT消息建立到A的偶联	
结构：1	测试类型：VAT
预期的信号单元顺序：	
<p>端点 B</p> <p>INIT (IP地址为X和Y, 端口为a)</p> <p>COOKIE_ECHO</p> <p>INIT (IP地址为X和Z, 端口为a)</p> <p>DATA</p> <p>SACK</p>	<p>端点 A</p> <p>INIT ACK</p> <p>COOKIE_ACK</p> <p>Communication up</p> <p>ABORT</p> <p>SACK</p> <p>DATA</p> <p>Send</p>
测试说明：	
1.	首先由B点启动到端点A的偶联建立，发送的INIT中包含IP地址X和Y，用监视仪记录消息流程。
2.	再由B点启动到端点A的偶联建立，发送的INIT消息中包含的IP地址为X和Z。
3.	检查A：第二次的偶联建立不成功，端点A发送ABORT消息。
4.	检查B：端点A和B之间已有的偶联不受影响。

测试编号: 1.17																																																																																		
参考: RFC 2960 第5.1.2节A																																																																																		
项目: 正常的偶联建立																																																																																		
分项目: INIT消息中无传送地址																																																																																		
目的: 在收到的INIT消息中未包含IP地址参数时, 检查端点的动作																																																																																		
测试预置条件: 端点A和端点B间的偶联未建立, 设置数据由B点发送的INIT消息中不包括IP地址参数																																																																																		
结构: 1	测试类型: VAT																																																																																	
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"></td> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%;"></td> <td style="width: 10%;"></td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 30%;"></td> <td style="width: 10%;"></td> <td style="width: 30%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">&lt;-----</td> <td></td> <td style="text-align: center;">INIT ACK</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">COOKIE_ECHO</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">&lt;-----</td> <td></td> <td style="text-align: center;">COOKIE_ACK</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">Communication up</td> <td style="text-align: center;">=====&gt;</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">Send</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">&lt;-----</td> <td></td> <td style="text-align: center;">DATA</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </table>			端点 B			端点 A				高层应用(ULP)		INIT	----->									<-----		INIT ACK						COOKIE_ECHO	----->									<-----		COOKIE_ACK											Communication up	=====>								<=====	Send				<-----		DATA						SACK	----->						
	端点 B			端点 A				高层应用(ULP)																																																																										
	INIT	----->																																																																																
		<-----		INIT ACK																																																																														
	COOKIE_ECHO	----->																																																																																
		<-----		COOKIE_ACK																																																																														
						Communication up	=====>																																																																											
						<=====	Send																																																																											
		<-----		DATA																																																																														
	SACK	----->																																																																																
测试说明:																																																																																		
1.	由B点启动到A点的偶联建立, 在B点发送的INIT消息中没包含IP地址, 用监视仪记录消息流程。																																																																																	
2.	检查A: INIT-ACK消息被发送到传送INIT消息的IP包的起源IP地址。																																																																																	
3.	检查B: 端点A发送的数据都使用传送INIT消息的IP包的起源IP地址。																																																																																	
4.	检查C: 端点A和端点B的偶联是否建立?																																																																																	

测试编号: 1.18 ( 任选 )																																					
参考: RFC 2960 第5.1.2节																																					
项目: 正常的偶联建立																																					
分项目: INIT消息中包含了主机名的地址																																					
目的: 当收到的INIT消息中没有IP地址而只有主机名地址时端点采取的动作																																					
测试前置条件: 端点A和B之间未建立偶联, 在B点设置数据, 使发送的INIT消息中只包含有主机名地址, 而没有IP地址。同时端点B发送的主机名地址应当可以由端点A解析																																					
结构: 1	测试类型: VAT																																				
预期的信号单元顺序:																																					
<table border="0"> <tr> <td>端点 B</td> <td></td> <td>端点 A</td> <td>高层应用(ULP)</td> </tr> <tr> <td>INIT( 包括主机名)</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>INIT ACK</td> <td></td> </tr> <tr> <td>COOKIE_ECHO</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>COOKIE_ACK</td> <td></td> </tr> <tr> <td></td> <td></td> <td>Communication up</td> <td>=====&gt;</td> </tr> <tr> <td></td> <td></td> <td></td> <td>&lt;===== Send</td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>DATA</td> <td></td> </tr> <tr> <td>SACK</td> <td>-----&gt;</td> <td></td> <td></td> </tr> </table>	端点 B		端点 A	高层应用(ULP)	INIT( 包括主机名)	----->				<-----	INIT ACK		COOKIE_ECHO	----->				<-----	COOKIE_ACK				Communication up	=====>				<===== Send		<-----	DATA		SACK	----->			
端点 B		端点 A	高层应用(ULP)																																		
INIT( 包括主机名)	----->																																				
	<-----	INIT ACK																																			
COOKIE_ECHO	----->																																				
	<-----	COOKIE_ACK																																			
		Communication up	=====>																																		
			<===== Send																																		
	<-----	DATA																																			
SACK	----->																																				
测试说明:																																					
1.	由B点启动到A点的偶联建立, B点发送的INIT消息只包含有主机地址, 无其他IP地址参数, 用监视仪记录消息流程。																																				
2.	检查A: INIT-ACK消息被发送到的IP地址, 是根据INIT消息中主机地址解析出的IP地址。																																				
3.	检查B: 端点A发送的数据都使用解析出的IP地址。																																				
4.	检查C: 端点A和端点B的偶联是否建立?																																				
5.	重复测试, 如果INIT消息中既包含IP地址 ( IPv4或IPv6 ), 又包括主机名地址参数, 则端点A应忽略IP地址参数, 而应使用主机名解析出的IP地址建立偶联。																																				

测试编号: 1.19																												
参考: RFC 2960 第5.1.2节																												
项目: 正常的偶联建立																												
分项目: INIT-ACK消息中无传送地址																												
目的: 当INIT-ACK消息中没有包含IP地址参数时, 端点的采取的动作																												
测试预置条件: 端点A和B之间未建立偶联, 在B点设置数据, 使发送的INIT-ACK消息中没有IP地址																												
结构: 1	测试类型: VAT																											
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 40%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;----- INIT</td> <td></td> </tr> <tr> <td style="text-align: center;">INIT ACK (未包括IP地址参数)</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;----- COOKIE_ECHO</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: right;">Communication up =====&gt;</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;----- DATA</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> </table>		端点 B	端点 A	高层应用(ULP)			<===== Associate		<----- INIT		INIT ACK (未包括IP地址参数)	----->			<----- COOKIE_ECHO		COOKIE_ACK	----->	Communication up =====>			<===== Send		<----- DATA		SACK	----->	
端点 B	端点 A	高层应用(ULP)																										
		<===== Associate																										
	<----- INIT																											
INIT ACK (未包括IP地址参数)	----->																											
	<----- COOKIE_ECHO																											
COOKIE_ACK	----->	Communication up =====>																										
		<===== Send																										
	<----- DATA																											
SACK	----->																											
测试说明:																												
1.	由A点启动到B点的偶联建立, B点发送的INIT-ACK消息未包括IP地址参数, 用监视仪记录消息流程。																											
2.	检查A: COOKIE-ECHO消息被发送携带INIT-ACK消息的IP数据包的起源IP地址。																											
3.	检查B: 端点A发送的数据都使用携带INIT-ACK消息的IP数据包的起源IP地址。																											
4.	检查C: 端点A和端点B的偶联是否建立?																											

测试编号：1.20（任选）	
参考：RFC 2960 第5.1.2节	
项目：正常的偶联建立	
分项日：INIT-ACK消息中只包括主机名，而无其他IP地址	
目的：当INIT-ACK消息中只有主机名地址而没有包含IP地址时，端点的采取的动作	
测试预置条件：端点A和B之间未建立偶联，在B点设置数据，使发送的INIT-ACK消息中只包含有主机名地址，而没有IP地址。同时端点B发送的主机名地址应当可以由端点A解析	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Associate     B-&gt;&gt;A: INIT     A-&gt;&gt;B: INIT ACK (只有主机名地址参数)     A-&gt;&gt;B: COOKIE_ECHO     B-&gt;&gt;A: COOKIE_ACK     B-&gt;&gt;ULP: Communication up     ULP-&gt;&gt;A: Send     A-&gt;&gt;B: DATA     B-&gt;&gt;A: SACK </pre>	
测试说明：	
1.	由A点启动到B点的偶联建立，在B点发送的INIT-ACK消息只包含有主机地址，无其他IP地址参数，用监视仪记录消息流程。
2.	检查A：COOKIE-ECHO消息被发送到的IP地址，是根据INIT-ACK消息中主机地址解析出的IP地址。
3.	检查B：端点A发送的数据都使用解析出的IP地址。
4.	检查C：端点A和端点B的偶联是否建立？
5.	重复测试，如果INIT-ACK消息中既包含IP地址（IPv4或IPv6），又包括主机名地址参数，则端点A应忽略IP地址参数，而应使用主机名解析出的IP地址建立偶联。



<b>测试编号:</b> 1.21	
<b>参考:</b> RFC 2960 第5.1.2节	
<b>项目:</b> 正常的偶联建立	
<b>分项目:</b> INIT消息中有一个或多个传送地址	
<b>目的:</b> 如果收到的INIT消息中有一个或多个IP地址参数, 则端点应当使用这些IP地址以及传送INIT消息的起源IP地址和起源端口号作为目的地传送地址	
<b>测试预置条件:</b> 端点A和B之间未建立偶联, 在B点设置数据, 使发送的INIT消息中包含一个或多个IP地址, 可以是IPv4或IPv6的地址	
<b>结构:</b> 1	<b>测试类型:</b> VAT, CPT
<b>预期的信号单元顺序:</b>	
端点 B	端点 A
INIT ( IP地址 = X,Y,Z端口 = a )	----->
	<-----
COOKIE_ECHO	INIT ACK
	----->
	<-----
	COOKIE_ACK
	Communication up =====>
DATA ( IP=X, Port=a )	----->
	<-----
DATA ( IP=Y, Port=a )	SACK
	----->
	<-----
DATA ( IP=Z, Port=a )	SACK
	----->
	<-----
	SACK
<b>测试说明:</b>	
1.	由B点启动到A点的偶联建立, 发送的INIT消息中包含一个或多个同类IP地址 ( IPv4或IPv6地址 ), 用监视仪记录消息流程。
2.	检查A: INIT消息被接收, 且响应的INIT-ACK消息应发送到收到INIT消息的传送地址。
3.	检查B: 端点A发送的除INIT ACK之外的其他消息可以发送到从INIT消息中收到的任何传送地址。
4.	检查C: 当端点A收到来自包含在INIT消息中的传送地址的消息时, 应能正确接收并用SACK响应。
5.	使用IPv6或IPv4地址的组合重复测试。

测试编号: 1.22	
参考: RFC 2960 第5.1.2节	
项目: 正常的偶联建立	
分项日: INIT-ACK消息中包含一个或多个传送地址	
目的: 如果收到INIT-ACK消息中有一个或多个IP地址, 则端点应当使用这些IP地址以及传送INIT-ACK消息的起源IP地址和起源端口号作为目的地传送地址	
测试预置条件: 端点A和B之间未建立偶联, 在B点设置数据, 使响应的INIT-ACK消息中包含一个或多个IP地址, 可以是IPv4或IPv6的地址	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
<p>端点 B</p> <p>INIT ACK (传送地址=X, Y, Z 端口=a)</p> <p>COOKIE_ACK</p> <p>DATA ( IP=X, Port=a )</p> <p>DATA ( IP=Y, Port=a )</p> <p>DATA ( IP=Z, Port=a )</p>	<p>端点 A</p> <p>INIT</p> <p>COOKIE_ECHO</p> <p>Communication up</p> <p>SACK</p> <p>SACK</p> <p>SACK</p>
	<p>高层应用(ULP)</p> <p>Associate</p>
测试说明:	
1.	由端点A启动到B的偶联建立, B响应的INIT-ACK消息中可以包含一个或多个IP地址, 用监视仪记录消息流程。
2.	检查A: 端点A接收该INIT-ACK消息, 且响应的COOKIE-ECHO消息应发送到包含在INIT-ACK消息中的传送地址或携带INIT-ACK消息的IP数据包的起源IP地址。
3.	检查B: 端点A发送的其他消息都应发送到包含在INIT-ACK消息中的传送地址或携带INIT-ACK消息的IP数据包的起源IP地址。
4.	检查C: 当端点A收到来自包含在INIT-ACK消息中的传送地址的消息时, 应能正确接收并用SACK响应。
5.	使用IPv6或IPv4地址的组合重复测试。

测试编号: 1.23 ( 任选 )	
参考: RFC 2960 第5.1.2节	
项目: 正常的偶联建立	
分项目: INIT消息中的主机地址不可解析	
目的: 当收到的INIT消息中的主机名地址无法解析时系统采取的动作	
测试预置条件: 端点A和端点B之间的偶联未建立, 在端点B设置数据, 使其发送的INIT消息中只包含主机名, 且该主机名地址在端点A无法解析	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre>       端点 B                                端点 A                                高层应用(ULP)           INIT -----&gt;           (只包含主机名)           &lt;----- ABORT                                      (差错原因为不能解析                                      的地址) </pre>	
测试说明:	
1.	由端点B启动到端点A的偶联建立, B发送的INIT消息中只包含了一个无法解析的主机名地址。用监视仪记录消息流程。
2.	检查A: 端点A针对该INIT消息发送ABORT消息, 且该差错原因为无法解析的地址。
3.	重复测试, 如果该INIT消息中除了包含无法解析的主机名地址外, 还包含IP地址时, 端点A也应响应ABORT消息, 且该差错原因为无法解析的地址。

测试编号：1.24（任选）	
参考：RFC 2960 第5.1.2节	
项目：正常的偶联建立	
分项目：INIT-ACK消息中包含不能解析的主机名地址	
目的：当收到的INIT-ACK消息中只包含有不能解析的主机名地址时，验证系统采取的动作	
测试预置条件：端点A和端点B之间的偶联未建立，在端点B设置数据，使其发送的INIT-ACK消息中只包含主机名，且该主机名地址在端点A无法解析	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Associate     B-&gt;&gt;A: INIT-ACK (包含不能解析的主机地址)     A-&gt;&gt;B: ABORT (差错原因为不能解析的地址)   </pre>	
测试说明：	
1.	由端点A启动到端点B的偶联建立，B发送的INIT-ACK消息中只包含了一个无法解析的主机名地址。用监视仪记录消息流程。
2.	检查A：端点A针对该INIT-ACK消息发送ABORT消息，且该差错原因为无法解析的地址。
3.	重复测试，如果该INIT-ACK消息中除了包含无法解析的主机名地址外，还包含IP地址时，端点A也应响应ABORT消息，且该差错原因为无法解析的地址。
注：端点A在发送了ABORT后，可以重新发送INIT消息重新建立偶联。	

测试编号: 1.25																																									
参考: RFC 2960 第5.1.2节																																									
项目: 正常的偶联建立																																									
分项目: INIT消息中支持的地址参数字段																																									
目的: 当收到的INIT消息中包括支持的地址参数字段, 验证系统采取的动作																																									
测试预置条件: 端点A和B之间的偶联未建立, 在B设置数据, 在INIT消息中发送支持的地址参数字段。端点A应能使用支持的地址字段中提到的地址类型																																									
结构: 1	测试类型: VAT, CPT																																								
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: right; width: 30%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: right; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT (带有支持的地址字段)</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE_ECHO</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">COOKIE_ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">Communication up</td> <td style="text-align: center;">=====&gt;</td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	INIT (带有支持的地址字段)	----->					<-----	INIT ACK			COOKIE_ECHO	----->					<-----	COOKIE_ACK						Communication up	=====>	DATA	----->					<-----	SACK		
端点 B		端点 A		高层应用(ULP)																																					
INIT (带有支持的地址字段)	----->																																								
	<-----	INIT ACK																																							
COOKIE_ECHO	----->																																								
	<-----	COOKIE_ACK																																							
			Communication up	=====>																																					
DATA	----->																																								
	<-----	SACK																																							
测试说明:																																									
1.	端点B启动到A的偶联建立, 在发送的INIT消息中应当包括支持的地址字段。用监视仪记录消息流程。																																								
2.	检查A: 发送的INIT-ACK消息的地址类型必须与收到的INIT消息中支持的地址字段一致。																																								
3.	检查B: 偶联成功地建立。																																								
4.	重复测试: 支持的地址字段中包括IPv6和IPv4的地址类型的组合。																																								

测试编号：1.26																
参考：RFC 2960 第5.1.2节																
项目：正常的偶联建立																
分项目：INIT消息中支持的地址类型都不为接收方支持																
目的：当接收方收到的INIT消息中支持的地址类型都不为接收方支持时，系统采取的动作																
测试预置条件：端点A和端点B之间的偶联未建立。在B点设置数据，使发送的INIT消息中包含端点A不能支持的地址类型字段																
结构：1	测试类型：VAT															
<p>预期的信号单元顺序：</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="width: 20%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td style="text-align: center;">INIT (带有支持的地址类型参数)</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">ABORT (原因)</td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)		INIT (带有支持的地址类型参数)	----->					<-----	ABORT (原因)	
端点 B		端点 A		高层应用(ULP)												
	INIT (带有支持的地址类型参数)	----->														
		<-----	ABORT (原因)													
测试说明：																
1.	端点B启动到A的偶联建立，在发送的INIT消息中支持的地址字段中的所有地址类型A点都不支持。用监视仪记录消息流程。															
2.	检查A：端点A 用ABORT中止偶联建立。															
3.	检查B：端点A发送的ABORT分组的的差错原因为“不可解析的地址”。															

测试编号: 1.27																
参考: RFC 2960 第3.3.2节																
项目: 正常的偶联建立																
分项目: INIT消息的启动标签为0																
目的: 检验当收到的INIT消息的启动标签为0时系统采取的动作																
测试预置条件: 端点A和B之间的偶联未建立, 在B点设置数据, 使端点B发送INIT的启动标签为0																
结构: 1	测试类型: VAT															
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 20%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td style="text-align: center;">INIT (启动标签为0)</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">ABORT</td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)		INIT (启动标签为0)	----->					<-----	ABORT	
端点 B		端点 A		高层应用(ULP)												
	INIT (启动标签为0)	----->														
		<-----	ABORT													
测试说明:																
1.	端点B启动到A的偶联建立, 在发送的INIT消息中启动标签为0。															
2.	检查A: 端点A响应ABORT消息, 其中错误原因为无效的必备参数。															
3.	重复测试: INIT消息中的启动标签不为0, 但a_rwnd参数为0。验证端点A响应ABORT消息, 其中错误原因为无效的必备参数。															
4.	重复测试: INIT消息中的其他参数都有效, 但验证标签不为0, 验证端点A响应ABORT消息, 其中错误原因为无效的必备参数。															

测试编号: 1.28	
参考: RFC 2960 第3.3.3节	
项目: 正常的偶联建立	
分项目: 收到的INIT-ACK消息的启动标签为0	
目的: 检验当收到响应INIT消息的INIT-ACK消息的启动标签等于0时系统采取的动作	
测试预置条件: 端点A和B之间的偶联未建立, 在B点设置数据, 使端点B响应INIT-ACK消息的启动标签为0	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A                      高层应用(ULP)
	←===== Associate
	INIT
INIT ACK (启动标签为0)	----->
	←----- ABORT (注)
注: 端点A是否发送ABORT消息取决于实施。	
测试说明:	
1.	从端点A启动到B的偶联建立, B响应的INIT-ACK消息中的启动标签为0。
2.	检查A: 端点A舍弃该INIT-ACK消息, 端点A是否发送ABORT消息取决于实施。
3.	重复测试: 如果INIT-ACK消息的启动标签不为0, 但a_rwnd参数为0, 则端点A舍弃该INIT-ACK消息, 端点A是否发送ABORT消息取决于实施。



4.5.2 偶联中止

测试编号: 2.1	
参考: RFC 2960 第9.1节	
项目: 偶联中止	
分项目: ABORT的产生	
目的: 当高层协议发送ABORT原语时, 则系统发送ABORT原语中止偶联	
测试预置条件: 端点A和端点B的偶联已建立, 在端点A设置数据, 让A的高层协议发送ABORT原语	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     Note over A: 高层应用(ULP) Abort (包含特定 差错原因)     Note over A: ABORT (包含特定差错原因)     A--&gt;&gt;B:      Note over B: 偶联被中止     </pre>	
测试说明:	
1.	由端点A的高层协议发送ABORT原语, 该原语中包含特定的差错原因
2.	检查A: 端点A发送的ABORT消息与高层原语携带的差错原因是否一致?
3.	检查B: 偶联是否被中止。
4.	重复测试, ABORT原语中未包括差错原因。

测试编号：2.2							
参考：RFC 2960 第9.1节							
项目：偶联中止							
分项目：收到没有差错原因的ABORT消息时，端点应中止偶联							
目的：验证收到没有差错原因的ABORT消息时应当可以中止偶联							
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，发送到端点A的ABORT消息中没有差错原因							
结构：1	测试类型：VAT						
<p>预期的信号单元顺序：</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 33%; text-align: center;">端点 B</td> <td style="width: 33%; text-align: center;">端点 A</td> <td style="width: 33%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">           ABORT            (无差错原因)         </td> <td style="text-align: center;">           -----&gt;         </td> <td style="text-align: center;">           Communication Lost =====&gt;         </td> </tr> </table>		端点 B	端点 A	高层应用(ULP)	ABORT (无差错原因)	----->	Communication Lost =====>
端点 B	端点 A	高层应用(ULP)					
ABORT (无差错原因)	----->	Communication Lost =====>					
测试说明：							
1.	由端点B发送没有差错原因的ABORT消息来中止端点A和端点B之间的偶联。						
2.	应设置ABORT消息中Peer's V-tag 和Local V-tag的T比特。						
3.	检查A：对该ABORT消息不需要发送证实，同时该偶联被清除。						
4.	检查B：SCTP向ULP发送Communication Lost原语通知偶联关闭。						
5.	重复测试：验证标签值等于发送的启动标签。						
6.	重复测试：ABORT消息中带有一个或多个差错原因。						

测试编号: 2.3																																																																	
参考: RFC 2960 第9.2节																																																																	
项目: 偶联中止																																																																	
分项目: 从高层协议收到终止原语后终止偶联																																																																	
目的: 验证端点收到终止原语后, 在所有未证实的数据被对端证实后端点发送SHUTDOWN消息																																																																	
测试预置条件: 端点A和B之间的偶联已建立, 由A点的高层协议发送终止原语																																																																	
结构: 1	测试类型: VAT																																																																
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="text-align: center; width: 10%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: right; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td style="text-align: center;">未被证实</td> <td style="text-align: center;">&lt;-----</td> <td>DATA</td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Terminate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA (重发的)</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA (重发的)</td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">所有的DATA都被证实</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN</td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN COMPLETE</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Communication Lost</td> <td style="text-align: right;">=====&gt;</td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				<===== Send		<-----	DATA					<===== Send	未被证实	<-----	DATA					<===== Terminate		<-----	DATA (重发的)			<-----	DATA (重发的)		DATA	----->				<-----	SACK		SACK	----->			所有的DATA都被证实					<-----	SHUTDOWN		SHUTDOWN ACK	----->				<-----	SHUTDOWN COMPLETE				Communication Lost	=====>
端点 B		端点 A	高层应用(ULP)																																																														
			<===== Send																																																														
	<-----	DATA																																																															
			<===== Send																																																														
未被证实	<-----	DATA																																																															
			<===== Terminate																																																														
	<-----	DATA (重发的)																																																															
	<-----	DATA (重发的)																																																															
DATA	----->																																																																
	<-----	SACK																																																															
SACK	----->																																																																
所有的DATA都被证实																																																																	
	<-----	SHUTDOWN																																																															
SHUTDOWN ACK	----->																																																																
	<-----	SHUTDOWN COMPLETE																																																															
		Communication Lost	=====>																																																														
测试说明:																																																																	
1.	先由端点A发送一些DATA分组, B点暂不予以证实。由A点的高层发送终止原语。																																																																
2.	检查A: 端点A在发送SHUTDOWN消息前应保证所有的DATA分组全部被证实, 端点A可以重发未证实的DATA分组。																																																																
3.	检查B: 端点A在所有数据被证实后发送SHUTDOWN消息。																																																																
4.	B点在发送SACK前, 还应发送一个DATA分组, 并用SACK证实。																																																																
5.	检查C: SHUTDOWN消息中累积证实TSN值应当与端点B的DATA分组的TSN一致。																																																																

测试编号：2.4	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：发送了SHUTDOWN消息后T2-Shutdown定时器超时	
目的：验证在发送SHUTDOWN消息后T2-Shutdown定时器启动，在该定时器超时后端点A应重发SHUTDOWN消息	
测试预置条件：端点A和端点B之间的偶联已建立，在B设置数据不响应SHUTDOWN-ACK	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     B-&gt;&gt;A: SHUTDOWN     Note over A: T2-Shutdown定时器     A--&gt;&gt;B: Terminate     B-&gt;&gt;A: SHUTDOWN     Note over A: T2-Shutdown定时器     A--&gt;&gt;B: Terminate </pre>	
测试说明：	
1.	由端点A终止到端点B的偶联，端点A发送了SHUTDOWN消息后，端点B不响应SHUTDOWN-ACK。
2.	检查A：在T2-Shutdown定时器超时后，端点A重发SHUTDOWN消息。
3.	重复测试：如果端点B是一个多归属的端点，该SHUTDOWN消息将重发到其他的传送地址。

测试编号：2.5	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：SHUTDOWN消息发送的次数超过了偶联的重传次数（ASSOCIATION.MAX.RETRANS）	
目的：验证当SHUTDOWN消息发送的次数超过了偶联的重传次数时，则清除该偶联	
测试预置条件：端点A和端点B间的偶联已建立，在B设置数据不响应SHUTDOWN-ACK时，设置偶联的最大重传次数（ASSOCIATION.MAX.RETRANS）为X。	
结构：1	测试类型：VAT
预期的信号单元顺序：	
<pre> sequenceDiagram     participant A as 端点A     participant B as 端点B     Note over A: SHUTDOWN     Note over B: T2-Shutdown定时器     Note over A: SHUTDOWN     Note over B: T2-Shutdown定时器     Note over A: SHUTDOWN     Note over A: SHUTDOWN消息的重发次数到X次     Note over B: Communication Lost     Note over B: Terminate     </pre>	
测试说明：	
1.	端点A终止到端点B的偶联，端点A发送SHUTDOWN消息，端点B不响应SHUTDOWN-ACK消息。
2.	检查A：端点A在重发了X次SHUTDOWN消息后，端点A的SCTP向ULP发送Communication Lost原语。
	注：T2-Shutdown定时器的值可以在每次重传后增加。

测试编号：2.6	
参考：RFC 2960 第8.5.1节	
项目：偶联中止	
分项目：收到响应SHUTDOWN消息的SHUTDOWN-ACK消息	
目的：在 SHUTDOWN 发送状态下，系统接收 SHUTDOWN-ACK 消息后应终止偶联并发送 SHUTDOWN-COMPLETE 消息	
测试预置条件：端点A和端点B的偶联已建立，端点B应能响应端点A发送的SHUTDOWN消息。	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     B-&gt;&gt;A: SHUTDOWN ACK     A-&gt;&gt;B: SHUTDOWN     A-&gt;&gt;B: SHUTDOWN COMPLETE     A--&gt;&gt;A: Communication Lost     ULP--&gt;&gt;A: Terminate </pre>	
测试说明：	
1.	由A点的高层协议发送TERMINATE原语，终止端点A和B之间的偶联，端点A发送SHUTDOWN消息，端点B响应SHUTDOWN-ACK消息。
2.	检查A：端点A在收到SHUTDOWN-ACK消息后应响应SHUTDOWN-COMPLETE消息。
3.	检查B：偶联被清除。

测试编号：2.7	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：在SHUTDOWN发送状态下，收到高层的DATA发送请求	
目的：验证在SHUTDOWN发送状态下，系统在收到高层的DATA发送请求时应予以拒绝	
测试预置条件：端点A和端点B之间的偶联已建立，在端点A的ULP设置数据，在SHUTDOWN发送状态下请求发送DATA数据	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Terminate     A-&gt;&gt;B: SHUTDOWN     ULP-&gt;&gt;A: Send     A-&gt;&gt;ULP: Send Failure     </pre>	
测试说明：	
1.	由A点的高层协议发送TERMINATE原语，终止端点A和B之间的偶联，端点A发送SHUTDOWN消息，在SHUTDOWN发送状态下，A点的ULP请求发送DATA数据。
2.	检查A：数据发送请求被拒绝。
3.	检查B：端点A仍处于SHUTDOWN发送状态。

测试编号：2.8																																																			
参考：RFC 2960 第9.2节																																																			
项目：偶联中止																																																			
分项目：在SHUTDOWN接收状态下收到从高层的数据发送请求																																																			
目的：验证系统在收到SHUTDOWN状态下收到数据发送请求时的动作																																																			
测试预置条件：端点A和端点B的偶联已建立，在A点设置数据，在收到SHUTDOWN状态下，由ULP发送数据																																																			
结构：1	测试类型：VAT																																																		
<p>预期的信号单元顺序：</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%;"></td> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 10%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: right;">SEND</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">DATA</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">未被B点证实 SHUTDOWN -----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">DATA (重发的)</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: right;">Send</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">Send Failure =====&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">SACK -----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SHUTDOWN-ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">SHUTDOWN-COMPLETE -----&gt;</td> <td></td> <td></td> <td></td> </tr> </table>			端点 B	端点 A		高层应用(ULP)				<=====	SEND		<-----	DATA				未被B点证实 SHUTDOWN ----->					<-----	DATA (重发的)						<=====	Send				Send Failure =====>			SACK ----->					<-----	SHUTDOWN-ACK				SHUTDOWN-COMPLETE ----->			
	端点 B	端点 A		高层应用(ULP)																																															
			<=====	SEND																																															
	<-----	DATA																																																	
	未被B点证实 SHUTDOWN ----->																																																		
	<-----	DATA (重发的)																																																	
			<=====	Send																																															
			Send Failure =====>																																																
	SACK ----->																																																		
	<-----	SHUTDOWN-ACK																																																	
	SHUTDOWN-COMPLETE ----->																																																		
测试说明：																																																			
1.	端点A收到端点B发送SHUTDOWN消息后，端点A的ULP请求发送数据。																																																		
2.	检查A：数据发送应当被拒绝。																																																		
3.	检查B：端点A仍处于收到SHUTDOWN状态。																																																		
4.	检查C：在端点A收到SACK后，应当发送SHUTDOWN-ACK消息。																																																		
5.	在收到SHUTDOWN状态下，端点A的ULP发送TERMINATE原语后不会触发SHUTDOWN消息。																																																		



测试编号：2.9	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：在SHUTDOWN未决状态下收到高层的数据发送请求	
目的：验证系统在SHUTDOWN未决状态下收到数据发送请求时的动作	
测试预置条件：端点A和端点B的偶联已建立，在A点设置数据，在SHUTDOWN未决状态下，由ULP发送数据	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre>           端点 B                                端点 A                                高层应用(ULP)           未被B点证实                            DATA                                &lt;===== SEND           &lt;-----           SACK -----&gt;           SHUTDOWN-ACK -----&gt;           &lt;----- SHUTDOWN           &lt;----- SHUTDOWN-COMPLETE            SHUTDOWN消息尚未发送           &lt;===== TERMINATE           &lt;===== Send           Send Failure =====&gt;                 </pre>	
测试说明：	
1.	端点A有未证实的DATA，端点A发送TERMINATE原语，但SHUTDOWN消息尚未发送，端点A的ULP请求发送数据。
2.	检查A：数据发送应当被拒绝。
3.	检查B：端点A仍处于SHUTDOWN未决状态。
4.	检查C：在端点A收到SACK后，应当发送SHUTDOWN消息。

测试编号：2.10	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：在SHUTDOWN-ACK发送状态下收到高层的数据发送请求	
目的：验证系统在SHUTDOWN-ACK发送状态下收到数据发送请求时的动作	
测试预置条件：端点A和端点B的偶联已建立，在A点设置数据，在发送SHUTDOWN-ACK后，由ULP发送数据	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A
SHUTDOWN ----->	SHUTDOWN-ACK <-----
	<===== Send
	Send Failure =====>
SHUTDOWN-COMPLETE ----->	Communication Lost =====>
测试说明：	
1.	端点A收到端点B发送的SHUTDOWN消息后，端点A响应SHUTDOWN-ACK消息，端点的ULP请求发送数据。
2.	检查A：数据发送应当被拒绝。
3.	检查B：端点A仍处于发送SHUTDOWN-ACK状态。

测试编号: 2.11	
参考: RFC 2960 第9.2节	
项目: 偶联中止	
分项目: 在SHUTDOWN发送状态下从对端收到DATA分组	
目的: 验证在SHUTDOWN发送状态下收到对端的DATA时系统的动作	
测试预置条件: 端点A和端点B的偶联已建立, 在A点设置数据, 在发送SHUTDOWN后, 由端点B发送DATA分组	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     B-&gt;&gt;A: DATA     A-&gt;&gt;B: SHUTDOWN     Note over A: 启动T2-shutdown定时器     A-&gt;&gt;B: SACK + SHUTDOWN     Note over A: 重新启动T2-shutdown定时器     B-&gt;&gt;A: SHUTDOWN ACK     Note over A: 清除偶联     A-&gt;&gt;B: SHUTDOWN COMPLETE     Note over A: Communication Lost     ULP-&gt;&gt;A: Terminate     </pre>	
测试说明:	
<ol style="list-style-type: none"> <li>1. 端点A发送SHUTDOWN消息后, 端点B响应DATA消息。</li> <li>2. 检查A: 端点A对DATA数据的证实可以与SHUTDOWN消息同时发送。</li> <li>3. 检查B: 端点A仍处于发送SHUTDOWN状态。</li> <li>4. 检查C: 端点A收到SHUTDOWN ACK消息后清除偶联。</li> </ol>	

测试编号：2.12	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：在收到SHUTDOWN状态下从对端收到DATA分组	
目的：验证在收到SHUTDOWN状态下收到对端的DATA时系统的动作	
测试预置条件：端点A和端点B的偶联已建立，在B点设置数据，在发送SHUTDOWN后，由端点B发送DATA分组，此时A点的重发缓冲区中应包含未证实的数据	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: SEND     B-&gt;&gt;A: SHUTDOWN     B-&gt;&gt;A: DATA     B-&gt;&gt;A: SACK     A-&gt;&gt;B: SHUTDOWN ACK     Note over A: Communication Lost   </pre>	
测试说明：	
1.	端点A收到端点B发送的SHUTDOWN消息后，由于端点A有尚未证实的数据，因此暂不向B响应SHUTDOWN-ACK消息，此时端点B向端点A发送DATA。
2.	检查A：此时收到的DATA分组予以忽略。
3.	检查B：端点A仍处于收到SHUTDOWN状态。
4.	检查C：端点A收到端点B的SACK消息后，在无未证实的消息时发送SHUTDOWN-ACK消息。

测试编号: 2.13	
参考: RFC 2960 第9.2节	
项目: 偶联中止	
分项目: 在收到SHUTDOWN状态下又从对端收到SHUTDOWN消息	
目的: 验证在收到SHUTDOWN状态下收到对端的SHUTDOWN消息时系统的动作	
测试预置条件: 端点A和端点B的偶联已建立, 在B点设置数据, 在发送SHUTDOWN后, 再由端点B发送SHUTDOWN消息, 此时A点的重发缓冲区中应包含未证实的数据	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: SEND     B-&gt;&gt;A: SHUTDOWN     A-&gt;&gt;B: DATA     B-&gt;&gt;A: SHUTDOWN     A-&gt;&gt;B: DATA (重发的) (不予理睬)     B-&gt;&gt;A: SACK     A-&gt;&gt;B: SHUTDOWN ACK     Note over A,B: Communication Lost     </pre>	
测试说明:	
1.	端点A收到端点B发送SHUTDOWN消息后, 由于端点A有尚未证实的数据, 因此暂不向B响应SHUTDOWN-ACK消息, 此时端点B向端点A再次发送SHUTDOWN消息。
2.	检查A: 此时收到的SHUTDOWN消息予以忽略。
3.	检查B: 端点A仍处于收到SHUTDOWN状态。
4.	检查C: 端点A收到端点B的SACK消息后, 在无未证实的消息时发送SHUTDOWN-ACK消息。

测试编号：2.14	
参考：RFC 2960 第9.2节	
项目：偶联中止	
分项目：SHUTDOWN-ACK消息发送后T2-shutdown定时器超时	
目的：验证SHUTDOWN-ACK消息发送后T2-shutdown定时器超时，系统应重发SHUTDOWN-ACK消息	
测试预置条件：端点A和端点B的偶联已建立，在B点设置数据，在收到SHUTDOWN-ACK后，不响应SHUTDOWN-COMPLETE消息	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
SHUTDOWN	----->
	<-----
	SHUTDOWN ACK
	T2-shutdown定时器
	<-----
	SHUTDOWN ACK
测试说明：	
1.	端点A收到端点B发送SHUTDOWN消息后，端点A响应SHUTDOWN-ACK消息，此时端点B不回应SHUTDOWN消息。
2.	检查A：T2-shutdown定时器超时后端点A将重发SHUTDOWN-ACK消息。
3.	检查B：如果端点B是一个多归属的端点，则重发的SHUTDOWN-ACK消息将发送到另外的传送地址（与第一个SHUTDOWN-ACK消息发送的传送地址不同）。

测试编号: 2.15																																																								
参考: RFC 2960 第9.2节																																																								
项目: 偶联中止																																																								
分项目: SHUTDOWN-ACK消息重发次数超过了偶联最大重传次数ASSOCIATION.MAX.RETRANS																																																								
目的: 验证当SHUTDOWN-ACK消息发次数超过了偶联最大重传次数ASSOCIATION.MAX.RETRANS后, 该偶联将被清除																																																								
测试预置条件: 端点A和端点B的偶联已建立, 在B点设置数据, 在收到SHUTDOWN-ACK后, 不响应SHUTDOWN-COMPLETE消息, 设置ASSOCIATION.MAX.RETRANS为X																																																								
结构: 1	测试类型: VAT																																																							
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: right; width: 30%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: right; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">T2-shutdown定时器</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">:</td> <td style="text-align: center;">:</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">T2-shutdown定时器</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">达到最大重传次数X</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">Communication Lost</td> <td style="text-align: center;">=====</td> <td style="text-align: right;">&gt;</td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	SHUTDOWN	----->					<-----	SHUTDOWN ACK					T2-shutdown定时器				<-----	SHUTDOWN ACK				:	:				<-----	SHUTDOWN ACK					T2-shutdown定时器				<-----	SHUTDOWN ACK					达到最大重传次数X					Communication Lost	=====	>
端点 B		端点 A		高层应用(ULP)																																																				
SHUTDOWN	----->																																																							
	<-----	SHUTDOWN ACK																																																						
		T2-shutdown定时器																																																						
	<-----	SHUTDOWN ACK																																																						
	:	:																																																						
	<-----	SHUTDOWN ACK																																																						
		T2-shutdown定时器																																																						
	<-----	SHUTDOWN ACK																																																						
		达到最大重传次数X																																																						
		Communication Lost	=====	>																																																				
测试说明:																																																								
1.	端点A收到端点B发送的SHUTDOWN消息后, 端点A响应SHUTDOWN-ACK消息, 此时端点B不回送SHUTDOWN-COMPLETE消息。																																																							
2.	检查A: T2-shutdown定时器超时后端点A将重发SHUTDOWN-ACK消息。																																																							
3.	检查B: 如果重发SHUTDOWN-ACK消息的次数超过最大重传次数X, 偶联应当被中止。																																																							
注: SHUTDOWN-ACK消息重发后T2-shutdown定时器时长可以增加。																																																								

测试编号: 2.16	
参考: RFC 2960 第9.2节	
项目: 偶联中止	
分项目: 收到了响应SHUTDOWN-ACK消息的SHUTDOWN-COMPLETE消息	
目的: 验证在收到了响应SHUTDOWN-ACK消息的SHUTDOWN-COMPLETE消息后系统的动作	
测试预置条件: 端点A和端点B的偶联已建立, 在B点设置数据, 在收到SHUTDOWN-ACK后, 立即响应SHUTDOWN-COMPLETE消息	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre>       端点 B                                端点 A                                高层应用(ULP)       SHUTDOWN -----&gt;       &lt;----- SHUTDOWN ACK       SHUTDOWN-COMPLETE -----&gt;                                      Communication Lost =====&gt; </pre>	
测试说明:	
1.	端点A收到端点B发送的SHUTDOWN消息后, 端点A响应SHUTDOWN-ACK消息, 此时端点B立即回送SHUTDOWN-COMPLETE消息。
2.	检查A: 端点A收到了SHUTDOWN-COMPLETE消息。
3.	检查B: 偶联被清除。
4.	重复测试: SHUTDOWN-COMPLETE消息的T比特设置为1, 且验证标签等于对端的验证标签。



4.5.3 无效消息处理

测试编号: 3.1																																														
参考: RFC 2960 第5.1节																																														
项目: 无效消息的处理																																														
分项目: 无效的INIT消息——消息长度小于包含所有必备参数的长度、OS=0、MIS=0或a_rwnd=0																																														
目的: 验证系统收到无效的INIT消息后的响应																																														
测试预置条件: 端点A和端点B之间的偶联未建立, 在端点B设置数据发送的INIT消息长度小于包含所有必备参数后INIT消息的长度、OS=0、MIS=0或a_rwnd=0																																														
结构: 1	测试类型: VAT																																													
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 15%;"></th> <th style="width: 35%; text-align: center;">端点 B</th> <th style="width: 10%;"></th> <th style="width: 20%; text-align: center;">端点 A</th> <th style="width: 20%; text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">1)</td> <td style="vertical-align: top;">INIT INIT消息长度小于包含所有必备参数的长度</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center; vertical-align: middle;">&lt;-----</td> <td style="vertical-align: middle;">ABORT</td> <td></td> </tr> <tr> <td style="vertical-align: top;">2)</td> <td style="vertical-align: top;">INIT(OS=0)</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center; vertical-align: middle;">&lt;-----</td> <td style="vertical-align: middle;">ABORT</td> <td></td> </tr> <tr> <td style="vertical-align: top;">3)</td> <td style="vertical-align: top;">INIT(MIS=0)</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center; vertical-align: middle;">&lt;-----</td> <td style="vertical-align: middle;">ABORT</td> <td></td> </tr> <tr> <td style="vertical-align: top;">4)</td> <td style="vertical-align: top;">INIT(a_rwnd=0)</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center; vertical-align: middle;">&lt;-----</td> <td style="vertical-align: middle;">ABORT</td> <td></td> </tr> </tbody> </table>			端点 B		端点 A	高层应用(ULP)	1)	INIT INIT消息长度小于包含所有必备参数的长度	----->					<-----	ABORT		2)	INIT(OS=0)	----->					<-----	ABORT		3)	INIT(MIS=0)	----->					<-----	ABORT		4)	INIT(a_rwnd=0)	----->					<-----	ABORT	
	端点 B		端点 A	高层应用(ULP)																																										
1)	INIT INIT消息长度小于包含所有必备参数的长度	----->																																												
		<-----	ABORT																																											
2)	INIT(OS=0)	----->																																												
		<-----	ABORT																																											
3)	INIT(MIS=0)	----->																																												
		<-----	ABORT																																											
4)	INIT(a_rwnd=0)	----->																																												
		<-----	ABORT																																											
测试说明:																																														
1.	由端点B发送无效INIT消息, 该INIT消息的长度小于包含所有必备参数后INIT消息的长度、OS=0、MIS=0或a_rwnd=0。																																													
2.	检查A: 端点A是否响应了ABORT消息。																																													
3.	检查B: ABORT消息中的验证标签应当等于INIT消息的启动标签。																																													
4.	消息序列是否如上所示。																																													

测试编号: 3.2	
参考: RFC 2960 第5.1节	
项目: 无效消息的处理	
分项目: INIT-ACK消息中的必备参数丢失	
目的: 验证系统在收到无效的INIT-ACK消息后的响应	
测试前置条件: 端点A和端点B间的偶联未建立, 在B设置数据使发送的INIT-ACK消息的长度小于包含必备参数的长度、OS=0、MIS=0或a_rwnd=0	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A
高层应用(ULP)	
1)	<===== Associate
	INIT
INIT-ACK(长度小于包含 必备参数的长度)	----->
	<----- ABORT (注)
2)	<===== Associate
	INIT
INIT-ACK (OS=0)	----->
	<----- ABORT (注)
3)	<===== Associate
	INIT
INIT-ACK (MIS=0)	----->
	<----- ABORT (注)
4)	<===== Associate
	INIT
INIT-ACK (a_rwnd=0)	----->
	<----- ABORT (注)
注: ABORT消息是否发送取决于实施	
测试说明:	
1.	由端点A启动偶联建立, B点发送的INIT-ACK消息的长度小于包含必备参数的长度、OS=0、MIS=0或a_rwnd=0。
2.	检查A: 端点A响应ABORT消息。
3.	如果发送ABORT消息, 则ABORT消息中的验证标签应当等于INIT-ACK消息的启动标签。

测试编号：3.3																																																							
参考：RFC 2960 第8.5节																																																							
项目：无效消息的处理																																																							
分项目：消息中包含无效的验证标签																																																							
目的：验证系统在收到包含无效验证标签的消息后的响应																																																							
测试预置条件：端点A和端点B间的偶联未建立，在端点B设置数据，使B点发送的消息中包含的验证标签与接收的验证标签不一致																																																							
结构：1	测试类型：VAT																																																						
预期的信号单元顺序：																																																							
<table border="0" style="width: 100%;"> <tr> <td style="width: 35%;"></td> <td style="width: 35%; text-align: center;">端点 A</td> <td style="width: 30%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td>1) 端点 B</td> <td></td> <td></td> </tr> <tr> <td>    INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT-ACK</td> </tr> <tr> <td>    COOKIE-ECHO</td> <td style="text-align: center;">-----&gt;</td> <td>舍弃该消息</td> </tr> <tr> <td>该消息的验证标签与 INIT-ACK消息的验证标签 不符</td> <td></td> <td></td> </tr> <tr> <td>2)</td> <td></td> <td style="text-align: right;">&lt;===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT</td> </tr> <tr> <td>    INIT-ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td>该消息的验证标签与INIT 消息的启动标签不符</td> <td></td> <td style="text-align: right;">  T1-init定时器超时</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT</td> </tr> <tr> <td>3)</td> <td></td> <td style="text-align: right;">&lt;===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>INIT</td> </tr> <tr> <td>    INIT-ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>COOKIE-ECHO</td> </tr> <tr> <td>    COOKIE-ACK</td> <td style="text-align: center;">-----&gt;</td> <td>舍弃该消息</td> </tr> <tr> <td>该消息的验证标签与 INIT-ACK消息的验证标签 不符</td> <td></td> <td style="text-align: right;">  T1-init定时器超时</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>COOKIE-ECHO</td> </tr> </table>		端点 A	高层应用(ULP)	1) 端点 B			INIT	----->			<-----	INIT-ACK	COOKIE-ECHO	----->	舍弃该消息	该消息的验证标签与 INIT-ACK消息的验证标签 不符			2)		<===== Associate		<-----	INIT	INIT-ACK	----->		该消息的验证标签与INIT 消息的启动标签不符		T1-init定时器超时		<-----	INIT	3)		<===== Associate		<-----	INIT	INIT-ACK	----->			<-----	COOKIE-ECHO	COOKIE-ACK	----->	舍弃该消息	该消息的验证标签与 INIT-ACK消息的验证标签 不符		T1-init定时器超时		<-----	COOKIE-ECHO	
	端点 A	高层应用(ULP)																																																					
1) 端点 B																																																							
INIT	----->																																																						
	<-----	INIT-ACK																																																					
COOKIE-ECHO	----->	舍弃该消息																																																					
该消息的验证标签与 INIT-ACK消息的验证标签 不符																																																							
2)		<===== Associate																																																					
	<-----	INIT																																																					
INIT-ACK	----->																																																						
该消息的验证标签与INIT 消息的启动标签不符		T1-init定时器超时																																																					
	<-----	INIT																																																					
3)		<===== Associate																																																					
	<-----	INIT																																																					
INIT-ACK	----->																																																						
	<-----	COOKIE-ECHO																																																					
COOKIE-ACK	----->	舍弃该消息																																																					
该消息的验证标签与 INIT-ACK消息的验证标签 不符		T1-init定时器超时																																																					
	<-----	COOKIE-ECHO																																																					
测试说明：																																																							
1.	由端点A或端点B启动偶联建立，在端点B响应的消息中其验证标签不等于收到的INIT-ACK消息的验证标签。																																																						
2.	检查A：验证标签不正确的消息被舍弃。																																																						
3.	检查B：相应的定时器超时后，端点A重发对应的消息。																																																						

测试编号: 3.4	
参考: RFC 2960 第8.5节	
项目: 无效消息的处理	
分项目: 消息中包含无效的验证标签	
目的: 验证系统在收到包含无效验证标签的消息后的响应	
测试预置条件: 端点A和端点B间的偶联已建立, 在端点B设置数据, 使B点发送的消息中包含的验证标签无效	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A                      高层应用(ULP)
1)                      DATA                      ----->	端点A不响应SACK, 并舍弃该消息
该消息的验证标签与INIT-ACK消息的验证标签不符	
2)	<===== SEND
	<----- DATA
SACK                      ----->	T3-rtx定时器超时
该消息的验证标签与INIT-ACK消息的验证标签不符	
	<----- DATA
3)	
HEARTBEAT                      ----->	舍弃该消息, 不发送HEARTBEAT-ACK消息
验证标签与INIT-ACK消息的验证标签不符	
4)	<===== Request
	<----- HEARTBEAT
HEARTBEAT-ACK                      ----->	舍弃该消息
验证标签与INIT-ACK消息的验证标签不符	不需要重发HEARTBEAT
测试说明:	
1.	偶联已建立, 当B点发送的消息中验证标签与INIT-ACK消息的验证标签不符。
2.	检查A: 验证标签不正确的消息被舍弃。
3.	检查B: 如果有相应的定时器在运行, 在相应的定时器超时后, 端点A重发对应的消息。

测试编号: 3.5 (1)																																																																	
参考: RFC 2960 第6.8节																																																																	
项目: 无效消息的处理																																																																	
分项目: SCTP分组的CRC-32校验码不正确																																																																	
目的: 检查系统在收到无效消息时的动作																																																																	
测试预置条件: 端点A和端点B之间的偶联未建立, 在端点B设置数据, 使其发送的偶联控制消息中包含无效的CRC-32校验码																																																																	
结构: 1	测试类型: VAT																																																																
预期的信号单元顺序:																																																																	
1)	<table border="0"> <tr> <td>端点 B</td> <td></td> <td>端点 A</td> <td>高层应用(ULP)</td> </tr> <tr> <td>INIT</td> <td>-----&gt;</td> <td>端点A舍弃该消息</td> <td></td> </tr> <tr> <td>无效CRC-32校验码</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2)</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>&lt;===== Associate</td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>INIT (启动T1-Init定时器)</td> <td></td> </tr> <tr> <td>INIT-ACK</td> <td>-----&gt;</td> <td>端点A舍弃该消息</td> <td></td> </tr> <tr> <td>无效CRC-32校验码</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>T1-Init定时器超时</td> <td></td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>INIT (启动T1-Init定时器)</td> <td></td> </tr> <tr> <td>3)</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>INIT</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td>&lt;-----</td> <td>INIT-ACK</td> <td></td> </tr> <tr> <td>COOKIE ECHO</td> <td>-----&gt;</td> <td>端点A舍弃该消息</td> <td></td> </tr> <tr> <td>无效CRC-32校验码</td> <td></td> <td></td> <td></td> </tr> </table>	端点 B		端点 A	高层应用(ULP)	INIT	----->	端点A舍弃该消息		无效CRC-32校验码				2)							<===== Associate		<-----	INIT (启动T1-Init定时器)		INIT-ACK	----->	端点A舍弃该消息		无效CRC-32校验码						T1-Init定时器超时			<-----	INIT (启动T1-Init定时器)		3)								INIT	----->				<-----	INIT-ACK		COOKIE ECHO	----->	端点A舍弃该消息		无效CRC-32校验码			
端点 B		端点 A	高层应用(ULP)																																																														
INIT	----->	端点A舍弃该消息																																																															
无效CRC-32校验码																																																																	
2)																																																																	
			<===== Associate																																																														
	<-----	INIT (启动T1-Init定时器)																																																															
INIT-ACK	----->	端点A舍弃该消息																																																															
无效CRC-32校验码																																																																	
		T1-Init定时器超时																																																															
	<-----	INIT (启动T1-Init定时器)																																																															
3)																																																																	
INIT	----->																																																																
	<-----	INIT-ACK																																																															
COOKIE ECHO	----->	端点A舍弃该消息																																																															
无效CRC-32校验码																																																																	
测试说明:																																																																	
1.	当系统收到的消息的CRC-32校验码不正确时, 检验端点A舍弃该控制消息。																																																																

测试编号: 3.5 (2)																						
参考: RFC 2960 第6.8节																						
项目: 无效消息的处理																						
分项目: SCTP分组的CRC-32校验码不正确																						
目的: 检查系统在收到无效消息时的动作																						
测试预置条件: 端点A和端点B之间的偶联未建立, 在端点B设置数据, 使其发送的控制消息中包含无效的CRC-32校验码																						
结构: 1	测试类型: VAT																					
预期的信号单元顺序:																						
<p>4)</p> <table border="0"> <thead> <tr> <th style="text-align: center;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: center;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td style="text-align: right;">←===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">-----&gt; INIT</td> <td></td> </tr> <tr> <td style="text-align: center;">INIT-ACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: left;">端点A舍弃该消息</td> </tr> <tr> <td></td> <td style="text-align: center;">-----&gt; COOKIE-ECHO(启动T1-Init定时器)</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE-ACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: left;">T1-Init定时器超时</td> </tr> <tr> <td style="text-align: center;">(无效CRC-32校验码)</td> <td style="text-align: center;">-----&gt; COOKIE-ECHO</td> <td></td> </tr> </tbody> </table>	端点 B	端点 A	高层应用(ULP)			←===== Associate		-----> INIT		INIT-ACK	----->	端点A舍弃该消息		-----> COOKIE-ECHO(启动T1-Init定时器)		COOKIE-ACK	----->	T1-Init定时器超时	(无效CRC-32校验码)	-----> COOKIE-ECHO		
端点 B	端点 A	高层应用(ULP)																				
		←===== Associate																				
	-----> INIT																					
INIT-ACK	----->	端点A舍弃该消息																				
	-----> COOKIE-ECHO(启动T1-Init定时器)																					
COOKIE-ACK	----->	T1-Init定时器超时																				
(无效CRC-32校验码)	-----> COOKIE-ECHO																					
测试说明:																						
1.	当系统收到的消息的CRC-32校验码不正确时, 检验端点A舍弃该控制消息。																					

测试编号: 3.6 (1)		
参考: RFC 2960 第6.8节		
项目: 无效消息的处理		
分项目: SCTP分组的CRC-32校验码不正确		
目的: 检查系统在收到无效消息时的动作		
测试预置条件: 端点A和端点B之间的偶联已建立, 在端点B设置数据, 使其发送的控制消息中包含无效的CRC-32校验码		
结构: 1	测试类型: VAT	
预期的信号单元顺序:		
端点 B	端点 A	高层应用(ULP)
1)		
DATA (包含无效CRC-32校验码)	----->	舍弃该分组, 不发送SACK
2)		
	<-----	SEND
SACK (包含无效CRC-32校验码)	----->	舍弃该分组, 重发定时器超 时后重发DATA
3)		
HEARTBEAT (包含无效CRC-32校验码)	----->	舍弃该分组, 不响应 HEARTBEAT ACK
4)		
	<-----	HB Request
HEARTBEAT ACK (包含无效CRC-32校验码)	----->	舍弃该分组, 超时而重发 HEARTBEAT
测试说明:		
1.	当系统收到的消息的CRC-32校验码不正确时, 检验端点A舍弃该消息。	

测试编号：3.6（2）	
参考：RFC 2960 第6.8节	
项目：无效消息的处理	
分项目：SCTP分组的CRC-32校验码不正确	
目的：检查系统在收到无效消息时的动作	
测试预置条件：端点A和端点B之间的偶联已建立，在端点B设置数据，使其发送的控制消息中包含无效的CRC-32校验码	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
5)	
SHUTDOWN (包含无效CRC-32校验码)	舍弃该分组，不发送SACK
	<===== SEND
	<----- DATA
6)	
SHUTDOWN ACK (包含无效CRC-32校验码)	舍弃该分组，T2-shutdown 定时器超时后重发
	<===== Terminate
	<----- SHUTDOWN
	<----- SHUTDOWN
7)	
ABORT (包含无效CRC-32校验码)	舍弃该分组
	<===== SEND
	<----- DATA
测试说明：	
1.	当系统收到的消息的CRC-32校验码不正确时，检验端点A舍弃该消息。



测试编号：3.7（1）																	
参考：RFC 2960 第5.1.5节																	
项目：无效消息的处理																	
分项目：无效的COOKIE-ECHO消息																	
目的：验证系统在收到无效的COOKIE-ECHO消息时的响应																	
测试预置条件：端点A和端点B的偶联未建立，在B点设置数据使其发送的COOKIE-ECHO消息的MD5编码与从INIT中收到的MD5编码不同																	
结构：1	测试类型：VAT																
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 40%;"></th> <th style="text-align: right; width: 30%;">端点 A</th> <th style="text-align: right; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td style="text-align: center;">INIT -----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: right;">INIT-ACK (包含COOKIE)</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE ECHO (错误的MD5编码)</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: right;">舍弃该消息，且不响应 COOKIE ACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)		INIT ----->				<-----	INIT-ACK (包含COOKIE)		COOKIE ECHO (错误的MD5编码)	----->	舍弃该消息，且不响应 COOKIE ACK	
端点 B		端点 A	高层应用(ULP)														
	INIT ----->																
	<-----	INIT-ACK (包含COOKIE)															
COOKIE ECHO (错误的MD5编码)	----->	舍弃该消息，且不响应 COOKIE ACK															
测试说明：																	
1.	由端点B启动到端点A的偶联建立，由B发送的COOKIE ECHO消息中的MD5编码与收到的INIT ACK消息的MD5编码不同，用监视仪记录消息流程。																
2.	检查A：端点B发送的COOKIE ECHO消息被舍弃。																
3.	检查B：端点A不响应COOKIE ACK消息。																

测试编号：3.7(2)	
参考：RFC 2960 第5.1.5节	
项目：无效消息的处理	
分项目：无效的COOKIE-ECHO消息	
目的：验证系统在收到无效的COOKIE-ECHO消息时的响应	
测试预置条件：端点A和端点B的偶联未建立，在B点设置数据使其在INIT ACK消息中规定的COOKIE存活时间后，发送正确的COOKIE-ECHO消息	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
INIT	----->
	<-----
	INIT-ACK ( COOKIE的存活时间为X秒)
COOKIE ECHO	----->
在X秒之后发送该消息	舍弃该消息，且不响应
	COOKIE ACK
	<-----
	ERROR (原因值为过期的COOKIE)
测试说明：	
1.	由端点B启动到端点A的偶联建立，端点B在INIT ACK消息中规定的COOKIE存活时间之后，发送正确的COOKIE-ECHO消息，用监视仪记录消息流程。
2.	检查A：B发送的COOKIE ECHO消息被舍弃。
3.	检查B：端点A发送ERROR消息，原因值为过期的COOKIE，其过期的测量值可以为0或COOKIE超时后经过的时间。

测试编号：3.8	
参考：RFC 2960 第8.5.1节	
项目：无效消息的处理	
分项目：无效的ABORT消息	
目的：检验系统在收到无效的ABORT消息后的响应	
测试预置条件：端点A和B之间的偶联已建立，在端点B设置数据，B点发送的ABORT消息中包含无效的验证标签	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p>	
测试说明：	
1.	由B点在建立的偶联上发送ABORT消息，其中ABORT消息的验证标签值应当不同于INIT-Tag和对端的验证标签，用监视仪记录消息流程。
2.	检查A：端点A舍弃该消息，且不采取任何动作，偶联仍处于工作状态。

测试编号：3.9	
参考：RFC 2960	
项目：无效消息的处理	
分项目：数据块长度大于分组长度	
目的：验证当收到的分组长度小于定义的数据块长度，则舍弃该分组	
测试预置条件：端点A和端点B的偶联未建立，在端点B设置数据，使端点B发送的数据块的长度值大于消息长度	
结构：I	测试类型：VAT
预期的信号单元顺序： <pre>sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     Note over B: INIT (数据块长度大于分组长度)     B--&gt;&gt;A: INIT     A--&gt;&gt;ULP: 舍弃该消息</pre>	
测试说明：	
1.	由端点B启动偶联建立，发送的INIT消息中的数据块长度的值大于分组长度，用监视仪记录消息流程。
2.	检查A：INIT消息被舍弃。
3.	检查B：用INIT-ACK, SACK, COOKIE-ECHO, COOKIE-ACK, SHUTDOWN, SHUTDOWN-ACK, HEARTBEAT, HEARTBEAT-ACK, ABORT, DATA和ERROR消息重复测试。

测试编号：3.10	
参考：RFC 2960 第8.5.1节	
项目：无效消息的处理	
分项目：无效的SHUTDOWN-ACK消息	
目的：验证系统在收到无效的SHUTDOWN - ACK消息后的响应	
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，使端点B发送的SHUTDOWN-ACK消息的验证标签无效，系统仍应处于已发送SHUTDOWN状态	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Terminate     B-&gt;&gt;A: SHUTDOWN-ACK (无效的验证标签)     B-&gt;&gt;A: DATA     A-&gt;&gt;A: SHUTDOWN 忽略该消息     A-&gt;&gt;A: SACK     </pre>	
测试说明：	
1.	端点A发送SHUTDOWN终止偶联，端点B响应的SHUTDOWN-ACK分组的验证标签不等于自己的或对端的验证标签。
2.	检查A：端点A忽略收到的这个SHUTDOWN-ACK消息。
3.	检查B：端点B仍处于已发送SHUTDOWN状态。

测试编号：3.11																															
参考：RFC 2960 第8.5.1节																															
项目：无效消息的处理																															
分项目：无效的SHUTDOWN-COMPLETE消息																															
目的：验证系统在收到无效的SHUTDOWN - COMPLETE消息后的响应																															
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，使端点B发送的SHUTDOWN-COMPLETE消息的验证标签无效，系统仍应处于已发送SHUTDOWN ACK状态																															
结构：1	测试类型：VAT																														
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: left; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td>SHUTDOWN-COMPLETE (无效的验证标签)</td> <td style="text-align: center;">-----&gt;</td> <td>忽略该消息</td> <td></td> <td></td> </tr> <tr> <td>SHUTDOWN-COMPLETE</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">Communication Lost</td> <td style="text-align: center;">=====&gt;</td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	SHUTDOWN	----->					<-----	SHUTDOWN ACK			SHUTDOWN-COMPLETE (无效的验证标签)	----->	忽略该消息			SHUTDOWN-COMPLETE	----->							Communication Lost	=====>
端点 B		端点 A		高层应用(ULP)																											
SHUTDOWN	----->																														
	<-----	SHUTDOWN ACK																													
SHUTDOWN-COMPLETE (无效的验证标签)	----->	忽略该消息																													
SHUTDOWN-COMPLETE	----->																														
			Communication Lost	=====>																											
测试说明：																															
1.	端点B发送SHUTDOWN终止偶联，端点A响应的SHUTDOWN-ACK，此时端点B响应的SHUTDOWN-COMPLETE分组的验证标签不等于自己的或对端的验证标签。																														
2.	检查A：端点A忽略收到的这个SHUTDOWN-COMPLETE消息。																														
3.	检查B：端点B当前的状态不受影响。																														
4.	重复测试：如果SHUTDOWN-COMPLETE消息的验证标签与对端的验证标签匹配，但T比特未设置为1。																														

4.5.4 重复消息的处理

测试编号: 4.1	
参考: RFC 2960 § 8                      状态转移图: 图10;	
项目: 重复的消息	
分项目: INIT的碰撞	
目的: 验证在INIT出现碰撞的情况下系统的响应	
测试预置条件: 端点A和端点B的偶联未建立, 在端点A和端点B设置数据, 保证两端同时发送INIT	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     B-&gt;&gt;A: INIT     A--&gt;A: T1-Init定时器     A--&gt;B: INIT ACK     B-&gt;&gt;A: INIT     ULP--&gt;&gt;: Associate     </pre>	
测试说明:	
1.	由两端同时启动偶联建立, 端点A和端点B同时发送INIT消息。
2.	检查A: 端点A用INIT-ACK消息响应端点B发送的INIT。
3.	检查B: 端点A发送的INIT-ACK消息的验证标签的值应当设置为端点B发送的INIT消息的标签值, 启动标签则设置为端点A的标签值(应当与端点A发送的INIT消息的启动标签值相同)。
4.	检查C: 在发送了INIT-ACK消息后, 端点A的T1-Init定时器仍旧运行, 且端点A保持在COOKIE_Wait状态。
5.	重复测试: 端点B在收到COOKIE_ECHO后, 发送INIT消息, 此时INIT-ACK消息COOKIE参数中的Tie_Tag标签的值应当设置为端点B发送的INIT消息的标签值。

测试编号: 4.2																										
参考: RFC 2960 第5.2.2节																										
项目: 重复的消息																										
分项目: 在已建立节点收到重复的INIT消息																										
目的: 检验系统在偶联已建立状态下收到INIT消息后的处理																										
测试预置条件: 端点A和端点B的偶联已建立, 在端点B设置数据, 由B点发送一个INIT消息, 起源IP地址和目的地IP地址与已建立的偶联的地址相同																										
结构: 1	测试类型: VAT																									
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%; text-align: center;">端点 B</th> <th style="width: 40%;"></th> <th style="width: 30%; text-align: center;">端点 A</th> <th style="width: 10%;"></th> <th style="width: 10%; text-align: center;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	INIT	----->					<-----	INIT ACK			DATA	----->					<-----	SACK		
端点 B		端点 A		高层应用(ULP)																						
INIT	----->																									
	<-----	INIT ACK																								
DATA	----->																									
	<-----	SACK																								
测试说明:																										
1.	由端点B在已建立的偶联上发送INIT。																									
2.	检查A: 端点A发送INIT-ACK消息来响应INIT消息。																									
3.	检查B: INIT-ACK消息中, 验证标签字段可以设置为从INIT中收到的对端的新的标签值。																									
4.	检查C: 在INIT-ACK消息中, Init-Tag不等于当前偶联的Init-tag, 该值应当是随机生成的。																									
5.	检查D: INIT-ACK消息发送的COOKIE应当是新产生的一个COOKIE, 这个COOKIE使用当前偶联的验证标签和对端验证标签作为本地和对端的Tie-Tag。																									
6.	检查E: INIT-ACK中的参数均使用当前偶联的参数(例如OS的数量, a_rwnd)。																									
7.	检查F: 在发送INIT-ACK消息后, 端点A不再采取任何动作, 现存的偶联不受影响。																									
8.	重复测试, INIT消息在接收到A点SHUTDOWN消息后发送。																									
9.	重复测试, INIT消息是在发送SHUTDOWN或接收SHUTDOWN状态下发送到端点A。																									



测试编号：4.3																													
参考：RFC 2960 第9.2节																													
项目：重复的消息																													
分项目：在SHUTDOWN ACK发送状态下收到的INIT消息																													
目的：检验系统在SHUTDOWN ACK发送状态下收到INIT消息后的处理																													
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，在收到了SHUTDOWN-ACK消息后，由B点发送一个INIT消息																													
结构：1	测试类型：VAT																												
<p>预期的信号单元顺序：</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: left; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> </tr> <tr> <td></td> <td></td> <td>启动T2-Shutdown定时器</td> <td></td> </tr> <tr> <td>INIT</td> <td style="text-align: center;">-----&gt;</td> <td>舍弃该INIT消息</td> <td></td> </tr> <tr> <td></td> <td></td> <td>T2-Shutdown定时器超时</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	SHUTDOWN	----->				<-----	SHUTDOWN ACK				启动T2-Shutdown定时器		INIT	----->	舍弃该INIT消息				T2-Shutdown定时器超时			<-----	SHUTDOWN ACK	
端点 B		端点 A	高层应用(ULP)																										
SHUTDOWN	----->																												
	<-----	SHUTDOWN ACK																											
		启动T2-Shutdown定时器																											
INIT	----->	舍弃该INIT消息																											
		T2-Shutdown定时器超时																											
	<-----	SHUTDOWN ACK																											
测试说明：																													
1.	由B点发送SHUTDOWN消息来终止A和B之间的偶联，在接收到端点A响应的SHUTDOWN-ACK消息后，端点B向已建立偶联的端点A的传送地址发送INIT消息。																												
2.	检查A：端点A舍弃INIT消息，在T2-Shutdown定时器超时后重发SHUTDOWN-ACK消息。																												

测试编号：4.4	
参考：RFC 2960 第5.2.3节	
项目：重复的消息	
分项目：在COOKIE_SEND状态下收到INIT-ACK消息	
目的：检验系统在COOKIE_SEND状态下收到INIT-ACK消息后的处理	
测试预置条件：端点A和B间的偶联未建立，在端点B设置数据，在收到COOKIE-ECHO状态下发送INIT-ACK消息	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
	<===== Associate
INIT-ACK	<-----> INIT(启动T1-Init定时器)
INIT-ACK	<-----> COOKIE-ECHO
COOKIE-ACK	<-----> 舍弃该INIT-ACK消息
DATA	<-----> Communication Up =====>
	<-----> SACK
测试说明：	
1.	由端点A启动到端点B的偶联建立，端点B在收到COOKIE-ECHO消息后发送INIT-ACK消息。
2.	检查A：在COOKIE_Sent状态下舍弃收到的INIT-ACK消息。
3.	检查B：端点A的状态不改变，且端点A和端点B的偶联可以成功建立。

测试编号: 4.5																	
参考: RFC 2960 第5.2.5节																	
项目: 重复的消息																	
分项目: 在偶联建立状态下收到COOKIE ACK消息																	
目的: 检验系统在偶联建立状态下收到COOKIE ACK消息后的处理																	
测试预置条件: 端点A和B间的偶联已建立, 在端点B设置数据, 发送COOKIE-ACK消息																	
结构: 1	测试类型: VAT																
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="text-align: center;">端点 A</td> <td style="text-align: right;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">COOKIE-ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td style="text-align: right;">舍弃该COOKIE-ACK消息</td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> </tr> </table>		端点 B		端点 A	高层应用(ULP)	COOKIE-ACK	----->		舍弃该COOKIE-ACK消息	DATA	----->				<-----	SACK	
端点 B		端点 A	高层应用(ULP)														
COOKIE-ACK	----->		舍弃该COOKIE-ACK消息														
DATA	----->																
	<-----	SACK															
测试说明:																	
1.	由端点A和端点B的偶联已建立。																
2.	检查A: 在偶联建立状态下舍弃收到的COOKIE-ACK消息, 偶联的状态不受影响。																

测试编号: 4.6																																				
参考: RFC 2960 第9.2节																																				
项目: 重复的消息																																				
分项目: SHUTDOWN消息的碰撞																																				
目的: 检验系统对SHUTDOWN消息的碰撞情况的处理																																				
测试预置条件: 端点A和B间的偶联已建立, 在端点A和B设置数据, 同时发送SHUTDOWN消息																																				
结构: 1	测试类型: VAT																																			
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 40%;"></th> <th style="text-align: right; width: 30%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;=====</td> <td>Terminate</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>启动T2-Shutdown定时器</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td>重新启动T2-Shutdown定时器</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)				<=====	Terminate		<-----	SHUTDOWN					启动T2-Shutdown定时器			SHUTDOWN	----->					<-----	SHUTDOWN ACK					重新启动T2-Shutdown定时器		
端点 B		端点 A		高层应用(ULP)																																
			<=====	Terminate																																
	<-----	SHUTDOWN																																		
		启动T2-Shutdown定时器																																		
SHUTDOWN	----->																																			
	<-----	SHUTDOWN ACK																																		
		重新启动T2-Shutdown定时器																																		
测试说明:																																				
1.	由端点的两端同时终止偶联并同时发送SHUTDOWN消息。																																			
2.	检查A: 端点A用SHUTDOWN-ACK响应收到的SHUTDOWN消息。																																			
3.	检查B: T2-Shutdown定时器重新启动。																																			

测试编号: 4.7																													
参考: RFC 2960 第9.2节																													
项目: 重复的消息																													
分项目: 在COOKIE_WAIT状态下收到SHUTDOWN消息																													
目的: 检验系统在COOKIE_WAIT状态下收到SHUTDOWN消息后的处理																													
测试预置条件: 端点A和B间的偶联未建立, 在端点B设置数据, 在收到INIT消息后发送SHUTDOWN消息																													
结构: 1	测试类型: VAT																												
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: center; width: 30%;">端点 A</th> <th style="text-align: right; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">←===== Associate</td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>INIT</td> <td></td> </tr> <tr> <td></td> <td></td> <td>启动T1-Init定时器</td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td>舍弃该消息</td> <td></td> </tr> <tr> <td></td> <td></td> <td>T1-Init定时器超时</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">←-----</td> <td>INIT</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				←===== Associate		←-----	INIT				启动T1-Init定时器		SHUTDOWN	----->	舍弃该消息				T1-Init定时器超时			←-----	INIT	
端点 B		端点 A	高层应用(ULP)																										
			←===== Associate																										
	←-----	INIT																											
		启动T1-Init定时器																											
SHUTDOWN	----->	舍弃该消息																											
		T1-Init定时器超时																											
	←-----	INIT																											
测试说明:																													
1.	由A点启动端点A和B之间的偶联, 端点A发送INIT消息, 端点B用SHUTDOWN消息予以响应。																												
2.	检查A: 端点A舍弃该SHUTDOWN消息。																												
3.	检查B: T1-Init定时器超时后A点重发INIT消息。																												
4.	重复测试: 在COOKIE ECHO状态下收到SHUTDOWN消息, 系统处理方式相同。																												

测试编号：4.8													
参考：RFC 2960 第9.2节													
项目：重复的消息													
分项目：在关闭状态下收到SHUTDOWN消息													
目的：检验系统在关闭状态下收到SHUTDOWN消息后的处理													
测试预置条件：在端点B设置数据，在A点处于关闭状态下发送SHUTDOWN消息													
结构：1	测试类型：VAT												
<p>预期的信号单元顺序：</p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="text-align: center;">端点 A</td> <td style="text-align: right;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">ABORT</td> <td></td> </tr> </table>		端点 B		端点 A	高层应用(ULP)	SHUTDOWN	----->				<-----	ABORT	
端点 B		端点 A	高层应用(ULP)										
SHUTDOWN	----->												
	<-----	ABORT											
测试说明：													
1.	从端点B发送SHUTDOWN消息，此时端点A处于关闭状态。												
2.	检查A：端点A用ABORT消息响应收到的SHUTDOWN消息。												
3.	检查B：ABORT消息的验证标签应当设置为SHUTDOWN消息的验证标签。												

测试编号: 4.9	
参考: RFC 2960 第9.2节	
项目: 重复的消息	
分项目: 在收到SHUTDOWN状态下收到SHUTDOWN消息	
目的: 检验系统在收到SHUTDOWN状态下又收到SHUTDOWN消息后的处理	
测试预置条件: 在端点B设置数据, 在已建立的偶联上发送SHUTDOWN消息	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Send     B-&gt;&gt;A: SHUTDOWN     A-&gt;&gt;B: DATA     B-&gt;&gt;A: SHUTDOWN     A-&gt;&gt;B: DATA(重发的消息)     B-&gt;&gt;A: SACK     A-&gt;&gt;B: SHUTDOWN-ACK     Note over A: Communication Lost     B-&gt;&gt;A: SHUTDOWN     </pre>	
测试说明:	
1.	从端点B发送SHUTDOWN消息终止到端点A的偶联, 此时端点A处于收到SHUTDOWN状态, 再由端点B发送SHUTDOWN消息。
2.	检查A: 端点A忽略该收到的SHUTDOWN消息。
3.	检查B: 端点A的状态不受影响。
4.	检查C: 在收到端点B对数据的证实后, 端点A发送SHUTDOWN-ACK消息。

测试编号：4.10	
参考：RFC 2960 第9.2节	
项目：重复的消息	
分项目：在SHUTDOWN ACK发送状态下收到有效的COOKIE-ECHO消息，其中本地和对端的Tag与当前TCB不匹配，但本地与对端的Tie-Tag与当前的TCB相匹配	
目的：检验系统在SHUTDOWN ACK发送状态下收到有效的COOKIE-ECHO消息后的处理	
测试预置条件：端点A和B的偶联已建立，在端点B设置数据，发送的有效的COOKIE-ECHO消息中，本地和对端的Tag与当前TCB不匹配，但本地与对端的Tie-Tag与当前的TCB相匹配	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
SHUTDOWN	----->
	<----- SHUTDOWN-ACK
COOKIE-ECHO	----->
	<----- SHUTDOWN-ACK
	<----- ERROR
测试说明：	
1.	当端点A处于SHUTDOWN-ACK发送状态时从端点B发送有效的COOKIE-ECHO消息，其中本地和对端的Tag与当前TCB不匹配，但本地与对端的Tie-Tag与当前的TCB相匹配。
2.	检查A：端点A忽略收到COOKIE-ECHO消息，并重发SHUTDOWN-ACK消息。
3.	检查B：端点A发送ERROR消息，其中差错原因为“在关闭状态下收到COOKIE”。
4.	检查C：端点A的状态不改变。



测试编号: 4.11																					
参考: RFC 2960 第9.2节																					
项目: 重复的消息																					
分项目: 在SHUTDOWN ACK发送状态下收到DATA消息																					
目的: 检验系统在SHUTDOWN ACK发送状态下收到DATA消息后的处理																					
测试预置条件: 端点A和B的偶联已建立, 在端点B设置数据, 在收到端点A发送的SHUTDOWN-ACK消息后发送DATA消息																					
结构: 1	测试类型: VAT																				
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 20%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SHUTDOWN-ACK</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">舍弃该数据</td> <td></td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)	SHUTDOWN	----->					<-----	SHUTDOWN-ACK			DATA	----->	舍弃该数据		
端点 B		端点 A		高层应用(ULP)																	
SHUTDOWN	----->																				
	<-----	SHUTDOWN-ACK																			
DATA	----->	舍弃该数据																			
测试说明:																					
1.	当端点A处于SHUTDOWN-ACK发送状态时从端点B发送DATA消息。																				
2.	检查A: 端点A忽略收到DATA消息。																				
3.	检查B: 端点A的状态不改变。																				
4.	检查C: 重复测试, 处于收到Shutdown, Cookie Wait或Cookie Echoed状态下, 系统收到DATA。																				
5.	检查D: 重复测试, 用SACK消息替代DATA消息重复测试。																				

## 4.5.5 故障处理

测试编号: 5.1																					
参考: RFC 2960 第8.1节																					
项目: 故障处理																					
分项目: 连续重发的次数大于偶联的最大重传次数																					
目的: 验证当连续重发的次数大于偶联的最大重传次数后, 目的地传送地址应当被标记为不可用, 且偶联被关闭																					
测试预置条件: 端点A和B的偶联已建立, 设置B点的传送地址为X, Y和Z, 且不响应端点A发送到端点B的DATA消息																					
结构: 1	测试类型: VAT																				
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;">端点 B</th> <th style="width: 20%;"></th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">不响应SACK</td> <td style="text-align: center;">←-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (IP=X) T3-Rtx定时器超时</td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td style="text-align: center;">不响应SACK</td> <td style="text-align: center;">←-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (IP=Y) T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td style="text-align: center;">不响应SACK</td> <td style="text-align: center;">←-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (IP=Z) T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td colspan="4" style="text-align: center;">           端点B启动重传计数器            :            :            到端点B的重传计数器均大于偶联的最大重传次数时            Communication Lost =====&gt;         </td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	不响应SACK	←-----	DATA (IP=X) T3-Rtx定时器超时	<===== Send	不响应SACK	←-----	DATA (IP=Y) T3-Rtx定时器超时		不响应SACK	←-----	DATA (IP=Z) T3-Rtx定时器超时		端点B启动重传计数器 : : 到端点B的重传计数器均大于偶联的最大重传次数时 Communication Lost =====>			
端点 B		端点 A	高层应用(ULP)																		
不响应SACK	←-----	DATA (IP=X) T3-Rtx定时器超时	<===== Send																		
不响应SACK	←-----	DATA (IP=Y) T3-Rtx定时器超时																			
不响应SACK	←-----	DATA (IP=Z) T3-Rtx定时器超时																			
端点B启动重传计数器 : : 到端点B的重传计数器均大于偶联的最大重传次数时 Communication Lost =====>																					
测试说明:																					
1.	端点A和B的偶联已建立, 由端点A发送DATA消息, 端点B不发送SACK。																				
2.	检查A: 当到端点B的重传计数器的次数都超过了最大重传次数后, 则标记该偶联为不可及。																				
3.	检查B: SCTP应当向ULP通知偶联故障。																				

测试编号: 5.2	
参考: RFC 2960 第8.1节	
项目: 故障处理	
分项目: 当收到SACK后, 到对应端点的重传计数器应当重新清0	
目的: 验证系统在收到SACK后, 到对应端点的重传计数器应当全部重新清0	
测试预置条件: 链路工作	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A
	<===== Send
不响应SACK <-----	DATA (IP=X)   T3-Rtx定时器超时
不响应SACK <-----	DATA (IP=Y)   T3-Rtx定时器超时
不响应SACK <-----	DATA (IP=Z)   T3-Rtx定时器超时
	到端点B的重传计数器小于偶联的最大重传次数
SACK ----->	DATA (IP=X)
	重传计数器清0
	<===== SEND
不响应SACK <-----	DATA (IP=X) T3-Rtx定时器超时
不响应SACK <-----	DATA (IP=Y) T3-Rtx定时器超时
不响应SACK <-----	DATA (IP=Z) T3-Rtx定时器超时
	:
	:
	当端点B的重传计数器均大于偶联的最大重传次数时
	Communication Lost =====>
测试说明:	
1.	端点A和B的偶联已建立, 由端点A发送DATA消息, 在端点B设置数据, 当重传次数小于端点A的偶联最大重传次数时, 端点B响应SACK。
2.	检查A: 端点A的重传计数器清0。
3.	由端点A再次发送DATA, 端点B不响应SACK。
4.	检查B: 当端点A的重传次数大于偶联的最大重传次数时, 端点A向高层报告偶联故障。

测试编号: 5.3																																																	
参考: RFC 2960 § 4.1	状态转移图: 图11;																																																
项目: 故障处理																																																	
分项目: 重传计数器大于通路的最大重传次数																																																	
目的: 验证当重传计数器的值大于通路的最大重传次数时, 标记该目的地传送地址标记为未激活																																																	
测试预置条件: 端点A和B的偶联已建立, 在B点设置数据, B点是一个多归属的点, 不响应由A点发送到某个地址的DATA消息																																																	
结构: 1	测试类型: VAT																																																
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 10%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: right; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td>不响应SACK</td> <td style="text-align: center;">&lt;-----</td> <td>DATA (IP=X)</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td>不响应SACK</td> <td style="text-align: center;">&lt;-----</td> <td>DATA (IP=X)</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td>不响应SACK</td> <td style="text-align: center;">&lt;-----</td> <td>DATA (IP=X)</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="border-left: 1px solid black; padding-left: 5px;">T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td colspan="4" style="text-align: center;">到端点B的重传计数器大于通路的最大重传次数</td> </tr> <tr> <td colspan="4" style="text-align: center;">NETWORK STATUS CHANGE =====&gt;</td> </tr> <tr> <td colspan="4" style="text-align: center;">IP地址X未激活</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA (IP=Y)</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				<===== Send	不响应SACK	<-----	DATA (IP=X)				T3-Rtx定时器超时		不响应SACK	<-----	DATA (IP=X)				T3-Rtx定时器超时		不响应SACK	<-----	DATA (IP=X)				T3-Rtx定时器超时		到端点B的重传计数器大于通路的最大重传次数				NETWORK STATUS CHANGE =====>				IP地址X未激活					<-----	DATA (IP=Y)	
端点 B		端点 A	高层应用(ULP)																																														
			<===== Send																																														
不响应SACK	<-----	DATA (IP=X)																																															
		T3-Rtx定时器超时																																															
不响应SACK	<-----	DATA (IP=X)																																															
		T3-Rtx定时器超时																																															
不响应SACK	<-----	DATA (IP=X)																																															
		T3-Rtx定时器超时																																															
到端点B的重传计数器大于通路的最大重传次数																																																	
NETWORK STATUS CHANGE =====>																																																	
IP地址X未激活																																																	
	<-----	DATA (IP=Y)																																															
测试说明:																																																	
1.	端点A和B的偶联已建立, 由端点A发送DATA, 端点B不响应SACK, 端点A将重发该DATA消息。																																																
2.	检查A: 当端点A重发的次数大于通路最大重传次数时, 则端点A标记该地址为未激活。																																																
3.	检查B: 如果还有其他地址可用, 则把DATA消息发送到另外的地址。																																																
<p>注: 在实际测试中, 可能出现向地址X、Y分别重传DATA的情况。进行本项测试时, 当到地址X的重传次数超过了通路最大重传次数时, 但到地址Y的通路最大重传次数尚未超过。</p>																																																	

测试编号: 5.4	
参考: RFC 2960 第8.2节	
项目: 故障处理	
分项目: 在收到对端的HEARTBEAT-ACK或SACK消息后重发计数器应当清0	
目的: 验证系统在收到HEARTBEAT-ACK或用于证实某些未证实的TSN的SACK消息后, 则对应该目的地址的重传计数器应当清0	
测试预置条件: 端点A和B之间的偶联已建立, 在端点B设置数据, B为一个多归属的点, 端点B不响应端点A发送的DATA	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p>	
测试说明:	
1.	端点A向端点B发送DATA消息, B点收到的重发的DATA数目小于最大通路重发次数时响应SACK。
2.	检查A: 端点A的重传计数器应当清0。
3.	检查B: 消息顺序是否如上所示。
4.	重复测试, 用HEARTBEAT-ACK消息代替SACK进行测试。
注: 在这个测试中, 重发的DATA必须发送到相同的IP地址。	

测试编号: 5.5																																					
参考: RFC 2960 第8.2节																																					
项目: 故障处理																																					
分项目: 如果收到了一个未完成的TSN的SACK, 而该TSN的DATA是发送到另外的传送地址时, 该地址的重传计数器不清0。																																					
目的: 验证收到的SACK是来自其他地址, 而证实的TSN又是先前发送到另外传送地址的情况下, 到另外的传送地址的重传计数器不清0																																					
测试预置条件: 端点A和B的偶联已建立, 端点B是一个多归属的点, 在端点B设置数据, 使端点B在收到到某个传送地址的DATA后, 不发送到SACK																																					
结构: 1	测试类型: VAT																																				
预期的信号单元顺序:																																					
<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 40%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td style="text-align: center;">不响应SACK</td> <td style="text-align: center;">&lt;----- DATA (IP=X)</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">  T3-Rtx定时器超时</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">:</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">:</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">到端点B的重传计数器大于通路的最大重传次数</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">NETWORK STATUS CHANGE</td> <td style="text-align: right;">=====&gt;</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;----- DATA (IP=Y)(重发的)</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">重传计数器不清0</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send (IP=X)</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;----- DATA (IP=Y)</td> <td></td> </tr> </table>		端点 B	端点 A	高层应用(ULP)			<===== Send	不响应SACK	<----- DATA (IP=X)			T3-Rtx定时器超时			:			:			到端点B的重传计数器大于通路的最大重传次数			NETWORK STATUS CHANGE	=====>		<----- DATA (IP=Y)(重发的)		SACK	----->	重传计数器不清0			<===== Send (IP=X)		<----- DATA (IP=Y)	
端点 B	端点 A	高层应用(ULP)																																			
		<===== Send																																			
不响应SACK	<----- DATA (IP=X)																																				
	T3-Rtx定时器超时																																				
	:																																				
	:																																				
	到端点B的重传计数器大于通路的最大重传次数																																				
	NETWORK STATUS CHANGE	=====>																																			
	<----- DATA (IP=Y)(重发的)																																				
SACK	----->	重传计数器不清0																																			
		<===== Send (IP=X)																																			
	<----- DATA (IP=Y)																																				
测试说明:																																					
1.	端点A到端点B的偶联已建立, 从端点A向端点B的某个传送地址发送DATA, 在端点B不发送SACK响应到该地址的DATA。当重发计数器超过两个通路的最大重传次数后, 该地址变为不激活, 此后的重传将发送到另外的传送地址, 此时端点B响应到该传送地址的DATA消息。																																				
2.	检查A: 在收到这个SACK后, 重传计数器并未清0。																																				
3.	检查B: 此后的数据均发送到另外的传送地址。																																				

测试编号：5.6																																																								
参考：RFC 2960 第8.3节																																																								
项目：故障处理																																																								
分项目：HEART-BEAT消息的周期发送																																																								
目的：验证当一个端点处于空闲状态时，当相当长时间无数据传递时，系统应当周期地向该端点发送HEART-BEAT消息																																																								
测试预置条件：端点A和B的偶联已建立，端点B是一个多归属的点，端点A和B长时间不传送数据																																																								
结构：1	测试类型：VAT																																																							
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%; text-align: center;">端点 B</th> <th style="width: 40%;"></th> <th style="width: 30%; text-align: center;">端点 A</th> <th style="width: 10%;"></th> <th style="width: 10%; text-align: center;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT (IP=X)</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">HEARTBEAT-ACK(来自X)</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">经过HEARTBEAT周期后</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT (IP=Y)</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">:</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT(IP=Y)(重传)</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重发n次，大于通路最大重传次数（Y变为未激活）</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">NETWORK STATUS CHANGE</td> <td style="text-align: center;">=====&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT(IP=Y)(重传)</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重传计数器不再增加</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)		<-----	HEARTBEAT (IP=X)			HEARTBEAT-ACK(来自X)	----->						经过HEARTBEAT周期后				<-----	HEARTBEAT (IP=Y)					:				<-----	HEARTBEAT(IP=Y)(重传)					重发n次，大于通路最大重传次数（Y变为未激活）					NETWORK STATUS CHANGE	=====>			<-----	HEARTBEAT(IP=Y)(重传)					重传计数器不再增加		
端点 B		端点 A		高层应用(ULP)																																																				
	<-----	HEARTBEAT (IP=X)																																																						
HEARTBEAT-ACK(来自X)	----->																																																							
		经过HEARTBEAT周期后																																																						
	<-----	HEARTBEAT (IP=Y)																																																						
		:																																																						
	<-----	HEARTBEAT(IP=Y)(重传)																																																						
		重发n次，大于通路最大重传次数（Y变为未激活）																																																						
		NETWORK STATUS CHANGE	=====>																																																					
	<-----	HEARTBEAT(IP=Y)(重传)																																																						
		重传计数器不再增加																																																						
测试说明：																																																								
1.	端点A到端点B的偶联已建立，在端点A和端点B之间不发送任何数据，在设置的HEARTBEAT周期后，系统将自动发送HEARTBEAT。																																																							
2.	检查B：如果到某个传送地址重发的HEARTBEAT消息次数超过通路的重传次数后，该传送地址应变为不激活。																																																							
3.	检查C：发送HEARTBEAT的周期应当等于用户设置的周期加上到该目的地地址的RTO。																																																							
4.	当端点B所有的传送地址被标记为未激活时，重复该测试。应当向某个地址发送HB消息。																																																							

测试编号：5.7													
参考：RFC 2960 第8.3节													
项目：故障处理													
分项目：HEART-BEAT请求原语													
目的：验证当高层发送HEART-BEAT请求原语时，系统应当向该端点发送HEART-BEAT消息													
测试预置条件：端点A和B的偶联已建立，在端点A高层设置数据，由端点A的ULP发送HEART-BEAT请求原语													
结构：1	测试类型：VAT												
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="text-align: center; width: 30%;">端点 A</th> <th style="text-align: right; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td style="text-align: right;">← HeartBeat</td> </tr> <tr> <td></td> <td style="text-align: center;">← HEARTBEAT</td> <td></td> </tr> <tr> <td style="text-align: left;">HEARTBEAT -ACK(来自X)</td> <td style="text-align: center;">→</td> <td></td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)			← HeartBeat		← HEARTBEAT		HEARTBEAT -ACK(来自X)	→	
端点 B	端点 A	高层应用(ULP)											
		← HeartBeat											
	← HEARTBEAT												
HEARTBEAT -ACK(来自X)	→												
测试说明：													
1.	端点A到端点B的偶联已建立，由端点A的ULP发送HEART-BEAT请求原语。												
2.	检查A：端点A是否向B发送了HEARTBEAT消息。												
3.	重复测试：端点B的所有传送地址标记为未激活。												



测试编号: 5.8																	
参考: RFC 2960 第8.3节																	
项目: 故障处理																	
分项目: 用HEARTBEAT-ACK消息响应收到的HEART-BEAT消息																	
目的: 验证当收到远端发送HEART-BEAT消息时, 系统应当用HEARTBEAT-ACK消息予以响应																	
测试预置条件: 端点A和B的偶联已建立, 在端点B设置数据, 由端点B发送HEART-BEAT消息																	
结构: 1	测试类型: VAT																
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: center; width: 30%;">端点 B</td> <td style="width: 40%;"></td> <td style="text-align: center; width: 30%;">端点 A</td> <td style="text-align: right; width: 10%;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">HEARTBEAT</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT-ACK</td> <td></td> </tr> <tr> <td></td> <td></td> <td colspan="2">消息中的信息应当是从 HEARTBEAT消息中复制的</td> </tr> </table>		端点 B		端点 A	高层应用(ULP)	HEARTBEAT	----->				<-----	HEARTBEAT-ACK				消息中的信息应当是从 HEARTBEAT消息中复制的	
端点 B		端点 A	高层应用(ULP)														
HEARTBEAT	----->																
	<-----	HEARTBEAT-ACK															
		消息中的信息应当是从 HEARTBEAT消息中复制的															
测试说明:																	
1.	端点A到端点B的偶联已建立, 由端点B向端点A发送HEART-BEAT消息。																
2.	检查A: 端点A是否向B响应了HEARTBEAT-ACK消息。																
3.	检查B: HEARTBEAT-ACK消息中的信息应当是从HEARTBEAT消息中复制得到的。																

测试编号: 5.9																																																																		
参考: RFC 2960 第8.3节																																																																		
项目: 故障处理																																																																		
分项目: 来自于未激活地址的HEARTBEAT-ACK的消息																																																																		
目的: 验证当收到来自未激活地址的HEARTBEAT-ACK消息时, 系统应当把该传送地址标记为激活, 且把该目的地传送地址的重传计数器清0																																																																		
测试预置条件: 端点A和B的偶联已建立, 在端点B设置数据, 端点B不响应收到的DATA消息																																																																		
结构: 1	测试类型: VAT																																																																	
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 40%;"></td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 10%;"></td> <td style="width: 10%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">Send</td> </tr> <tr> <td style="text-align: center;">不响应SACK</td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">DATA (IP=X)</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">  T3-Rtx定时器超时</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">:</td> <td style="text-align: center;">:</td> <td></td> <td></td> </tr> <tr> <td></td> <td colspan="3" style="text-align: center;">到端点B的重传计数器大于通路的最大重传次数</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">NETWORK STATUS CHANGE (未激活)</td> <td style="text-align: center;">=====</td> <td style="text-align: center;">&gt;</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">HeartBeat</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">HEARTBEAT</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">HEARTBEAT-ACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">NETWORK STATUS CHANGE (激活)</td> <td style="text-align: center;">=====</td> <td style="text-align: center;">&gt;</td> </tr> <tr> <td></td> <td></td> <td></td> <td style="text-align: center;">&lt;=====</td> <td style="text-align: center;">Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">DATA (IP=X)</td> <td></td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)				<=====	Send	不响应SACK	<-----	DATA (IP=X)					T3-Rtx定时器超时				:	:				到端点B的重传计数器大于通路的最大重传次数						NETWORK STATUS CHANGE (未激活)	=====	>				<=====	HeartBeat		<-----	HEARTBEAT			HEARTBEAT-ACK	----->						NETWORK STATUS CHANGE (激活)	=====	>				<=====	Send		<-----	DATA (IP=X)		
端点 B		端点 A		高层应用(ULP)																																																														
			<=====	Send																																																														
不响应SACK	<-----	DATA (IP=X)																																																																
		T3-Rtx定时器超时																																																																
	:	:																																																																
	到端点B的重传计数器大于通路的最大重传次数																																																																	
		NETWORK STATUS CHANGE (未激活)	=====	>																																																														
			<=====	HeartBeat																																																														
	<-----	HEARTBEAT																																																																
HEARTBEAT-ACK	----->																																																																	
		NETWORK STATUS CHANGE (激活)	=====	>																																																														
			<=====	Send																																																														
	<-----	DATA (IP=X)																																																																
测试说明:																																																																		
1.	端点A到端点B的偶联已建立, 端点B不响应端点A发送DATA消息, 使端点A标记端点B为未激活。由端点A的高层发送HEARTBAET请求原语。																																																																	
2.	检查A: 在收到端点B响应的HEARTBAET-ACK消息后, 端点B的状态是否变为激活状态。																																																																	
3.	端点A又可以向端点B发送DATA消息。																																																																	

测试编号: 5.10																					
参考: RFC 2960 第8.4节																					
项目: 故障处理																					
分项目: 收到了一个有效的DATA分组, 而端点与发送该分组的地址未建立偶联(OOTB)																					
目的: 验证系统在收到OOTB的数据时应当响应ABORT消息																					
测试预置条件: 端点A和和端点B之间无偶联, 在端点B设置数据向端点A发送一个有效的DATA消息																					
结构: 1	测试类型: VAT																				
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 30%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: left; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td>通过了校验码测试</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>ABORT</td> <td></td> </tr> <tr> <td></td> <td></td> <td>舍弃该收到的消息</td> <td></td> </tr> <tr> <td></td> <td></td> <td>端点A的状态不改变</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	DATA	----->	通过了校验码测试			<-----	ABORT				舍弃该收到的消息				端点A的状态不改变	
端点 B		端点 A	高层应用(ULP)																		
DATA	----->	通过了校验码测试																			
	<-----	ABORT																			
		舍弃该收到的消息																			
		端点A的状态不改变																			
测试说明:																					
1.	端点A到端点B的偶联未建立, 由端点B发送一个有效的DATA给端点A。																				
2.	检查A: 端点A是否响应了ABORT消息。																				
3.	检查B: ABORT消息中的验证标签是否等于收到的DATA消息的验证标签, ABORT数据块的T比特是否设置为1。																				
4.	端点A舍弃DATA消息, 且A的状态不改变。																				
5.	用SHUTDOWN, INIT-ACK 和SACK重复测试。																				

测试编号：5.11		
参考：RFC 2960 第8.4节		
项目：故障处理		
分项目：收到了一个有效的ABORT分组，而端点与发送该分组的地址未建立偶联（OOTB）		
目的：验证系统在收到OOTB的ABORT时不应再响应ABORT消息，直接舍弃该消息		
测试预置条件：端点A和端点B之间无偶联，在端点B设置数据向端点A发送一个有效的ABORT消息		
结构：1	测试类型：VAT	
预期的信号单元顺序：		
端点 B	端点 A	高层应用(ULP)
ABORT	----->	通过了校验码测试 舍弃该收到的消息 端点A的状态不改变
测试说明：		
1.	端点A到端点B的偶联未建立，由端点B发送一个有效的ABORT给端点A。	
2.	检查A：端点A是否直接舍弃了该ABORT消息。	
3.	端点A的状态不改变。	
4.	重复测试：设置ABORT数据块的T比特为0和1的情况。	

测试编号: 5.12										
参考: RFC 2960 第8.4节										
项目: 故障处理										
分项目: 收到了一个有效的SHUTDOWN-ACK分组, 而端点与发送该分组的地址未建立偶联(OOTB)										
目的: 验证系统在收到OOTB的SHUTDOWN-ACK分组时再响应SHUTDOWN-COMPLETE消息, 并舍弃收到的SHUTDOWN-ACK消息										
测试预置条件: 端点A和和端点B之间无偶联, 在端点B设置数据向端点A发送一个有效的SHUTDOWN-ACK消息										
结构: I	测试类型: VAT									
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">SHUTDOWN-ACK</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td style="vertical-align: top;">通过了校验码测试, 舍弃该收到的消息</td> </tr> <tr> <td></td> <td style="text-align: center; vertical-align: middle;">&lt;-----</td> <td style="vertical-align: top;">SHUTDOWN-COMPLETE 端点A的状态不改变</td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	SHUTDOWN-ACK	----->	通过了校验码测试, 舍弃该收到的消息		<-----	SHUTDOWN-COMPLETE 端点A的状态不改变
端点 B	端点 A	高层应用(ULP)								
SHUTDOWN-ACK	----->	通过了校验码测试, 舍弃该收到的消息								
	<-----	SHUTDOWN-COMPLETE 端点A的状态不改变								
测试说明:										
1.	端点A到端点B的偶联未建立, 由端点B发送一个有效的SHUTDOWN-ACK给端点A。									
2.	检查A: 端点A是否响应了SHUTDOWN-COMPLETE消息, SHUTDOWN-COMPLETE消息的验证标签应当等于收到的SHUTDOWN-ACK消息的验证标签, SHUTDOWN--COMPLETE数据块的T比特应设置为1。									
3.	检查B: 端点A是否舍弃了该SHUTDOWN-ACK消息。									
4.	端点A的状态不改变。									

测试编号: 5.13							
参考: RFC 2960 第8.4节							
项目: 故障处理							
分项目: 收到了一个有效的SHUTDOWN-COMPLETE分组, 而端点与发送该分组的地址未建立偶联(OOTB)							
目的: 验证系统在收到OOTB的SHUTDOWN-COMPLETE消息, 直接舍弃收到的SHUTDOWN-ACK消息							
测试预置条件: 端点A和和端点B之间无偶联, 在端点B设置数据向端点A发送一个有效的SHUTDOWN-COMPLETE消息							
结构: 1	测试类型: VAT						
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">SHUTDOWN-COMPLETE</td> <td style="text-align: center; vertical-align: middle;">-----&gt;</td> <td style="vertical-align: top;">通过了校验码测试 舍弃该收到的消息 端点A的状态不改变</td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	SHUTDOWN-COMPLETE	----->	通过了校验码测试 舍弃该收到的消息 端点A的状态不改变
端点 B	端点 A	高层应用(ULP)					
SHUTDOWN-COMPLETE	----->	通过了校验码测试 舍弃该收到的消息 端点A的状态不改变					
测试说明:							
1.	端点A到端点B的偶联未建立, 由端点B发送一个有效的SHUTDOWN-COMPLETE给端点A。						
2.	检查A: 端点A是否直接舍弃了SHUTDOWN-COMPLETE消息。						
3.	端点A的状态不改变。						
4.	重复测试: 设置SHUTDOWN-COMPLETE数据块的T比特为0和1的情况。						
5.	用差错原语为“过期的COOKIE”的ERROR 消息重复测试。						
6.	用COOKIE-ACK重复测试。						

测试编号: 5.14							
参考: RFC 2960 第8.4节							
项目: 故障处理							
分项目: 收到了一个有效的分组, 且地址是非单播地址, 此时端点与发送该分组的地址未建立偶联(OOTB)							
目的: 验证系统在收到的消息地址为非单播地址时, 直接舍弃收到的消息							
测试预置条件: 端点A和端点B之间无偶联, 在端点B设置数据向端点A发送一个有效的消息(例如COOKIE-ECHO消息), 该消息中起源地址是非单播地址							
结构: 1	测试类型: VAT						
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 33%; text-align: center;">端点 B</td> <td style="width: 33%; text-align: center;">端点 A</td> <td style="width: 33%; text-align: center;">高层应用(ULP)</td> </tr> <tr> <td style="text-align: center;">COOKIE-ECHO</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">舍弃该收到的消息 端点A的状态不改变</td> </tr> </table>		端点 B	端点 A	高层应用(ULP)	COOKIE-ECHO	----->	舍弃该收到的消息 端点A的状态不改变
端点 B	端点 A	高层应用(ULP)					
COOKIE-ECHO	----->	舍弃该收到的消息 端点A的状态不改变					
测试说明:							
1.	端点A到端点B的偶联未建立, 由端点B发送一个有效的COOKIE-ECHO给端点A, 其起源地址为非单播地址。						
2.	检查A: 端点A是否直接舍弃了该消息。						
3.	端点A的状态不改变。						
4.	用其他类型消息且起源地址为非单播地址重复测试。						
5.	重复测试: 把目的地地址设置为非单播地址。						

## 4.5.6 差错消息处理

测试编号: 6.1	
参考: RFC 2960 第5.2.6节	
项目: ERROR消息	
分项目: 在收到Cookie_Echoed状态下收到差错原因为“过期的COOKIE”的ERROR消息	
目的: 验证在收到Cookie_Echoed状态下, 能够正确处理收到差错原因为“过期的COOKIE”的ERROR消息	
测试预置条件: 端点A和端点B的偶联未建立, 在端点B设置数据, 在收到COOKIE ECHO消息后, 响应的ERROR消息中包括的差错原因为“过期的COOKIE”	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
<p>端点 B</p> <p>INIT ACK</p> <p>ERROR (过期的COOKIE)</p>	<p>端点 A</p> <p>=====&gt;</p> <p>INIT(启动T1-Init定时器)</p> <p>COOKIE ECHO</p>
	<p>高层应用(ULP)</p> <p>Associate</p>
测试说明:	
1.	由端点A启动到B的偶联建立。在端点A响应了COOKIE-ECHO消息后, 端点B回送一个ERROR消息, 差错原因为“过期的Cookie”。
2.	端点在收到差错原因为“过期的Cookie”的ERROR消息后, 可以采取以下动作之一: <ul style="list-style-type: none"> <li>a) 舍弃先前的TCB并向高层报告不能建立偶联;</li> <li>b) 向远端点发送一个新的INIT数据块, 使远端点生成一个新的状态COOKIE, 尝试重新建立过程;</li> <li>c) 向远端点发送一个新的INIT数据块, 并且通过增加Cookie Preservative 参数请求对状态COOKIE的存活时间进行延长。(推荐使用该方法)</li> </ul>



测试编号：6.2																																																									
参考：RFC 2960 第5.2.6节																																																									
项目：ERROR消息																																																									
分项目：在其他状态下（除了Cookie_Echoed状态）下收到差错原因为“过期的COOKIE”的ERROR消息																																																									
目的：验证系统在其他状态下（除了Cookie_Echoed状态）下，能对收到差错原因为“过期的COOKIE”的ERROR消息进行正确处理																																																									
测试前置条件：端点A和端点B之间的偶联已建立，在端点B设置数据，由B发送差错原因为“过期的COOKIE”的ERROR消息																																																									
结构：1	测试类型：VAT																																																								
<p>预期的信号单元顺序：</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 10%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: left; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td>←===== Associate</td> </tr> <tr> <td></td> <td>←-----</td> <td>INIT(启动T1-Init定时器)</td> <td></td> </tr> <tr> <td>INIT ACK</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td>←-----</td> <td>COOKIE ECHO</td> <td></td> </tr> <tr> <td>COOKIE ACK</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td>Communication Up =====&gt;</td> </tr> <tr> <td></td> <td></td> <td></td> <td>←===== Send</td> </tr> <tr> <td></td> <td>←-----</td> <td>DATA</td> <td></td> </tr> <tr> <td>SACK</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td>ERROR</td> <td>-----&gt;</td> <td>直接舍弃该ERROR消息</td> <td></td> </tr> <tr> <td>(过期的COOKIE)</td> <td></td> <td></td> <td></td> </tr> <tr> <td>DATA</td> <td>-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td>←-----</td> <td>SACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				←===== Associate		←-----	INIT(启动T1-Init定时器)		INIT ACK	----->				←-----	COOKIE ECHO		COOKIE ACK	----->						Communication Up =====>				←===== Send		←-----	DATA		SACK	----->			ERROR	----->	直接舍弃该ERROR消息		(过期的COOKIE)				DATA	----->				←-----	SACK	
端点 B		端点 A	高层应用(ULP)																																																						
			←===== Associate																																																						
	←-----	INIT(启动T1-Init定时器)																																																							
INIT ACK	----->																																																								
	←-----	COOKIE ECHO																																																							
COOKIE ACK	----->																																																								
			Communication Up =====>																																																						
			←===== Send																																																						
	←-----	DATA																																																							
SACK	----->																																																								
ERROR	----->	直接舍弃该ERROR消息																																																							
(过期的COOKIE)																																																									
DATA	----->																																																								
	←-----	SACK																																																							
测试说明：																																																									
1.	由端点A启动到B的偶联建立，在端点A处于非Cookie_Echoed状态，由B发送ERROR消息。																																																								
2.	检查A：端点A是否舍弃了该ERROR消息。																																																								
3.	检查B：偶联的状态不受影响。																																																								

测试编号：6.3	
参考：RFC 2960 第3.3.10.1节	
项目：ERROR消息	
分项目：其他的差错原因——无效的流标识符	
目的：验证系统收到包含无效的流标识符的DATA时，系统响应的ERROR消息的差错原因为“无效的流标识符”	
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，B发送的DATA中包含的流序号无效	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
DATA (流标识符=X)	X大于端点A的MIS
	<----- ERROR (差错原因为无效的流标识符，流标识符为X)
测试说明：	
1.	端点A和B之间的偶联已建立，从端点B发送一个DATA消息，其流标识符对于该偶联无效。
2.	检查A：端点A响应的ERROR消息中的差错原因为无效的流标识符，此外该消息中还应包含无效的流标识符X。

测试编号: 6.4	
参考: RFC 2960 第3.3.10.2节	
项目: ABORT消息	
分项目: 其他的差错原因——丢失必备参数	
目的: 验证系统收到的INIT ACK消息中没有Cookie时, 系统响应的ABORT消息的差错原因为“丢失必备参数”	
测试预置条件: 端点A和端点B的偶联未建立, 在端点B设置数据, B点响应的INIT ACK消息中未包含Cookie	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     B-&gt;&gt;A: INIT ACK 未包含Cookie     A-&gt;&gt;ULP: Associate     A-&gt;&gt;B: ABORT (差错原因丢失 必备参数)     </pre>	
测试说明:	
1.	由端点A启动到端点B的偶联建立, 端点B响应的INIT ACK消息中未包含Cookie。
2.	检查A: 端点A响应的ABORT消息中的差错原因为必备参数丢失, 并指示出丢失的参数。

测试编号: 6.5	
参考: RFC 2960 第3.3.10.8节	
项目: ERROR消息	
分项: 其他的差错原因——不识别的参数(参数类型的高两比特为11)	
目的: 验证系统收到的INIT ACK消息中包含的TLV参数在接收方未定义, 系统响应的ERROR消息的差错原因为“不识别的参数”	
测试预置条件: 端点A和端点B的偶联未建立, 在端点B设置数据, 由B发送的INIT ACK消息中包含一个未知的参数(例如: 参数类型编码为OFFF), 但编码规则符合规范要求	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Associate     B--&gt;&gt;A: INIT ACK (未知的参数)     A--&gt;&gt;B: ERROR (不识别的参数) </pre>	
测试说明:	
1.	由端点A启动到端点B的偶联建立, 端点B响应的INIT ACK消息中包含一个未知的参数。
2.	检查A: 端点A响应的ERROR消息中的差错原因为“不识别的参数”, 并携带该不识别的参数。
3.	用备用的参数类型重复测试, 其编码规则符合TLV的编码规则。

测试编号：6.6																									
参考：RFC 2960 第3.3.10.9节																									
项目：ERROR消息																									
分项目：在收到的COOKIE ECHO消息中包含了不识别参数的差错原因																									
目的：验证系统在收到的COOKIE ECHO消息中包含了不识别参数的差错原因，系统应能完成偶联建立过程																									
测试预置条件：端点A和端点B之间的偶联未建立																									
结构：1	测试类型：VAT																								
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 30%; text-align: center;">端点 B</th> <th style="width: 30%; text-align: center;">端点 A</th> <th style="width: 40%; text-align: center;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK</td> </tr> <tr> <td style="text-align: center;">COOKIE ECHO (包含的差错原因为“不识别的参数”)</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">COOKIE ACK</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">Communication Up =====&gt;</td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	INIT	----->			<-----	INIT ACK	COOKIE ECHO (包含的差错原因为“不识别的参数”)	----->			<-----	COOKIE ACK			Communication Up =====>	DATA	----->			<-----	SACK
端点 B	端点 A	高层应用(ULP)																							
INIT	----->																								
	<-----	INIT ACK																							
COOKIE ECHO (包含的差错原因为“不识别的参数”)	----->																								
	<-----	COOKIE ACK																							
		Communication Up =====>																							
DATA	----->																								
	<-----	SACK																							
测试说明：																									
1.	由端点B启动到端点A的偶联建立，在端点B响应的COOKIE ECHO消息中包含的差错原因应为“不识别参数”。																								
2.	检查A：端点A执行正常的偶联建立程序，响应COOKIE ACK消息。																								
3.	检查B：端点A和B之间的偶联是否建立。																								

## 4.5.7 数据块的捆绑和复用

测试编号: 7.1																
参考: RFC 2960 第6.10节																
项目: 数据块的捆绑和复用																
分项目: 数据块与INIT消息的捆绑																
目的: 验证如果在INIT消息中捆绑了DATA消息, 系统应当舍弃该INIT消息																
测试预置条件: 端点A和B之间的偶联未建立, 在B点设置数据, 发送的INIT消息中捆绑了DATA数据块																
结构: 1	测试类型: VAT															
<p>预期的信号单元顺序:</p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: center; width: 33%;">端点 B</td> <td style="width: 33%;"></td> <td style="text-align: center; width: 33%;">端点 A</td> <td style="width: 33%;"></td> <td style="text-align: center;">高层应用(ULP)</td> </tr> <tr> <td></td> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">直接舍弃该分组</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">其中捆绑了DATA数据块</td> <td></td> <td></td> <td></td> </tr> </table>		端点 B		端点 A		高层应用(ULP)		INIT	----->	直接舍弃该分组			其中捆绑了DATA数据块			
端点 B		端点 A		高层应用(ULP)												
	INIT	----->	直接舍弃该分组													
	其中捆绑了DATA数据块															
测试说明:																
1.	由端点B启动到端点A的偶联建立, 端点B发送的INIT消息中捆绑了DATA数据块。INIT数据块是作为该SCTP分组中的第一个数据块。															
2.	检查A: 端点A应当舍弃整个分组, 并且不采取任何动作。															
3.	重复测试: 在SCTP分组中的DATA数据块包含在INIT数据块之前。															

测试编号：7.2	
参考：RFC 2960 第6.10节	
项目：数据块的捆绑和复用——控制数据块和DATA数据块的捆绑	
分项目：INIT ACK数据块和DATA数据块的捆绑	
目的：验证系统应舍弃收到的INIT ACK数据块和DATA数据块捆绑的SCTP分组	
测试预置条件：端点A和端点B之间的偶联未建立，在端点B设置数据，使其响应的INIT ACK和DATA数据块捆绑	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
	<===== Associate
	INIT(启动T1-Init定时器)
INIT ACK (捆绑了DATA数据块)	舍弃该INIT ACK
	T1-Init定时器超时
	INIT (重发的)
INIT ACK	
	COOKIE ECHO
COOKIE ACK	
	Communication Up =====>
DATA	
	SACK
测试说明：	
1.	由端点A启动到端点B的偶联建立，B点响应的INIT-ACK消息与DATA数据块捆绑，但INIT-ACK是SCTP分组中的第一个数据块。
2.	检查A：端点A舍弃该INIT ACK消息。
3.	检查B：端点A的状态不改变。当Init-T1定时器超时后，端点A重发INIT消息。
4.	重复测试：INIT-ACK不是SCTP分组中的第一个数据块。

测试编号：7.3																																	
参考：RFC 2960 第6.10节																																	
项目：数据块的捆绑和复用——控制数据块和DATA数据块的捆绑																																	
分项目：DATA数据块与SHUTDOWN COMPLETE数据块的捆绑																																	
目的：验证所有与SHUTDOWN COMPLETE数据块捆绑的DATA数据块都将被舍弃																																	
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，端点B响应的SHUTDOWN COMPLETE消息中捆绑了DATA数据块																																	
结构：1	测试类型：VAT																																
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 20%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: right; width: 20%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN COMPLETE</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">捆绑了数据块</td> <td style="text-align: center;">&lt;-----</td> <td>ABORT</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				<===== Send		<-----	DATA		SACK	----->			SHUTDOWN	----->				<-----	SHUTDOWN ACK		SHUTDOWN COMPLETE	----->			捆绑了数据块	<-----	ABORT	
端点 B		端点 A	高层应用(ULP)																														
			<===== Send																														
	<-----	DATA																															
SACK	----->																																
SHUTDOWN	----->																																
	<-----	SHUTDOWN ACK																															
SHUTDOWN COMPLETE	----->																																
捆绑了数据块	<-----	ABORT																															
测试说明：																																	
1.	由端点B终止偶联，端点B发送SHUTDOWN消息，端点A响应SHUTDOWN ACK消息，端点B响应的SHUTDOWN COMPLETE消息中应捆绑DATA数据块。																																
2.	检查A：端点A应当舍弃该SHUTDOWN COMPLETE消息，并仍处于Shutdown_Ack发送状态。																																
3.	检查B：端点A响应ABORT消息。																																



测试编号：7.4																																					
参考：RFC 2960 第5.1.5节和6.10节																																					
项目：数据块的捆绑和复用——控制数据块和DATA数据块的捆绑																																					
分项目：DATA数据块与COOKIE ECHO数据块捆绑，且COOKIE ECHO数据块是该分组中的第一个数据块																																					
目的：验证所有与COOKIE ECHO数据块捆绑的DATA数据块都将被接收，并用COOKIE ACK响应该COOKIE ECHO数据块																																					
测试预置条件：端点A和端点B的偶联未建立，在端点B设置数据，端点B响应的COOKIE ECHO消息中捆绑了DATA数据块																																					
结构：1	测试类型：VAT																																				
<p>预期的信号单元顺序：</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; width: 30%;">端点 B</th> <th style="width: 40%;"></th> <th style="text-align: center; width: 30%;">端点 A</th> <th style="text-align: center; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">INIT</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">INIT ACK</td> <td></td> </tr> <tr> <td style="text-align: center;">COOKIE ECHO (捆绑DATA数据块)</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">COOKIE ACK</td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">Communication Up</td> <td style="text-align: center;">=====&gt;</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	INIT	----->				<-----	INIT ACK		COOKIE ECHO (捆绑DATA数据块)	----->				<-----	COOKIE ACK				Communication Up	=====>		<-----	SACK		DATA	----->				<-----	SACK	
端点 B		端点 A	高层应用(ULP)																																		
INIT	----->																																				
	<-----	INIT ACK																																			
COOKIE ECHO (捆绑DATA数据块)	----->																																				
	<-----	COOKIE ACK																																			
		Communication Up	=====>																																		
	<-----	SACK																																			
DATA	----->																																				
	<-----	SACK																																			
测试说明：																																					
1.	由端点B启动到端点A的偶联建立，由端点B响应的COOKIE ECHO消息中捆绑DATA数据块。																																				
2.	检查A：端点A接收该COOKIE ECHO消息，并响应COOKIE ACK消息。																																				
3.	检查B：端点A发送的SACK应与COOKIE ACK消息捆绑，用来证实与COOKIE ECHO数据块捆绑的DATA数据块。																																				
4.	重复测试：如果与DATA捆绑的COOKIE ECHO数据块不是该SCTP分组的第一个数据块，则端点A应舍弃该SCTP分组。																																				
5.	重复测试：如果与该COOKIE ECHO数据块捆绑的是一个控制数据块，则应当舍弃该分组。																																				

测试编号：7.5	
参考：RFC 2960 第5.1.5节和6.10节	
项目：数据块的捆绑和复用——控制数据块和DATA数据块的捆绑	
分项目：DATA数据块与COOKIE ACK数据块捆绑，且COOKIE ACK数据块是该分组中的第一个数据块	
目的：验证所有与COOKIE ACK数据块捆绑的DATA数据块都将被接收，且偶联正常建立	
测试预置条件：端点A和端点B的偶联未建立，在端点B设置数据，端点B响应的COOKIE ACK消息中捆绑了DATA数据块	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
	←===== Associate
	←----- INIT
INIT ACK	----->
	←----- COOKIE ECHO
COOKIE ACK (捆绑DATA数据块)	-----> 接收该COOKIE ACK
	Communication Up =====>
	←----- SACK
DATA	----->
	←----- SACK
测试说明：	
1.	由端点A启动到端点B的偶联建立，由端点B响应的COOKIE ACK消息中捆绑DATA数据块。
2.	检查A：端点A接收该COOKIE ACK消息。
3.	检查B：端点A发送SACK证实与COOKIE ACK数据块捆绑的DATA数据块。
4.	重复测试：与COOKIE ACK捆绑的是SACK数据块，消息顺序同上。
5.	重复测试：如果与DATA捆绑的COOKIE ACK数据块不是该SCTP分组的第一个数据块，则端点A应舍弃该SCTP分组。
6.	重复测试：如果与该COOKIE ACK数据块捆绑的是一个控制数据块，则应当舍弃该分组。

测试编号: 7.6																					
参考: RFC 2960 第5.1.5节和6.10节																					
项目: 数据块的捆绑和复用																					
分项目: SHUTDOWN数据块与SACK数据块的捆绑																					
目的: 验证系统会接收与SACK捆绑的SHUTDOWN数据块																					
测试预置条件: 端点A和端点B的偶联已建立, 在端点B设置数据, SACK数据块与SHUTDOWN数据块捆绑																					
结构: 1	测试类型: VAT																				
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 10%;"></th> <th style="text-align: left; width: 30%;">端点 A</th> <th style="text-align: right; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>DATA</td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN 与SACK捆绑</td> <td style="text-align: center;">-----&gt;</td> <td>该SHUTDOWN被接收, 并 进入收到Shutdown状态</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SHUTDOWN ACK</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)				<===== Send		<-----	DATA		SHUTDOWN 与SACK捆绑	----->	该SHUTDOWN被接收, 并 进入收到Shutdown状态			<-----	SHUTDOWN ACK	
端点 B		端点 A	高层应用(ULP)																		
			<===== Send																		
	<-----	DATA																			
SHUTDOWN 与SACK捆绑	----->	该SHUTDOWN被接收, 并 进入收到Shutdown状态																			
	<-----	SHUTDOWN ACK																			
测试说明:																					
1.	由端点B启动终止偶联, 发送的SHUTDOWN消息与SACK数据块捆绑, 其中SACK是对最后一个DATA数据块的证实。																				
2.	检查A: 端点A接收该SHUTDOWN消息, 并进入收到Shutdown状态。																				
3.	重复测试, 当与SHUTDOWN数据块捆绑的SACK不是最后一个数据块时, 程序同上。																				
4.	重复测试, 如果与SHUTDOWN数据块捆绑的是DATA, 则端点A舍弃该分组。																				

测试编号: 7.7																			
参考: RFC 2960 第9.2节																			
项目: 数据块的捆绑和复用																			
分项目: DATA数据块与SACK数据块的捆绑																			
目的: 验证系统会接收与DATA捆绑的SACK数据块																			
测试前置条件: 端点A和端点B的偶联已建立, 在端点B设置数据, SACK数据块与DATA数据块捆绑																			
结构: 1	测试类型: VAT																		
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 40%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td></td> <td style="text-align: center;">DATA</td> <td></td> </tr> <tr> <td style="text-align: center;">SACK</td> <td style="text-align: center;">-----&gt;</td> <td style="text-align: center;">该SACK被接收,</td> </tr> <tr> <td style="text-align: center;">与DATA数据块捆绑</td> <td></td> <td style="text-align: center;">DATA Arrive =====&gt;</td> </tr> <tr> <td></td> <td style="text-align: center;">SACK</td> <td></td> </tr> </table>		端点 B	端点 A	高层应用(ULP)			<===== Send		DATA		SACK	----->	该SACK被接收,	与DATA数据块捆绑		DATA Arrive =====>		SACK	
端点 B	端点 A	高层应用(ULP)																	
		<===== Send																	
	DATA																		
SACK	----->	该SACK被接收,																	
与DATA数据块捆绑		DATA Arrive =====>																	
	SACK																		
测试说明:																			
<ol style="list-style-type: none"> <li>1. 端点A与端点B间的偶联已建立, 端点B发送的SACK应与DATA数据块捆绑。</li> <li>2. 检查A: 端点A接收该SACK和DATA数据块。</li> <li>3. 重复测试: 如果SACK是分组中的最后一个数据块, 则舍弃该SACK数据块。</li> </ol>																			

测试编号: 7.8																
参考: RFC 2960 第9.2节																
项目: 数据块的捆绑和复用																
分项目: DATA数据块与SHUTDOWN ACK数据块的捆绑																
目的: 验证系统应舍弃与DATA捆绑的SHUTDOWN ACK数据块																
测试预置条件: 端点A和端点B的偶联已建立, 在端点B设置数据, 端点B发送的SHUTDOWN ACK数据块与DATA数据块捆绑																
结构: 1	测试类型: VAT															
<p>预期的信号单元顺序:</p> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 30%; text-align: center;">端点 B</td> <td style="width: 30%; text-align: center;">端点 A</td> <td style="width: 40%; text-align: right;">高层应用(ULP)</td> </tr> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Terminate</td> </tr> <tr> <td></td> <td style="text-align: center;">SHUTDOWN</td> <td></td> </tr> <tr> <td style="text-align: center;">SHUTDOWN ACK 与DATA数据块捆绑</td> <td style="text-align: center;">-----&gt;</td> <td style="border-left: 1px solid black; padding-left: 10px;">该SHUTDOWN ACK被舍弃</td> </tr> <tr> <td></td> <td style="text-align: center;">SHUTDOWN</td> <td></td> </tr> </table>		端点 B	端点 A	高层应用(ULP)			<===== Terminate		SHUTDOWN		SHUTDOWN ACK 与DATA数据块捆绑	----->	该SHUTDOWN ACK被舍弃		SHUTDOWN	
端点 B	端点 A	高层应用(ULP)														
		<===== Terminate														
	SHUTDOWN															
SHUTDOWN ACK 与DATA数据块捆绑	----->	该SHUTDOWN ACK被舍弃														
	SHUTDOWN															
测试说明:																
1.	由端点A启动终止偶联, B点发送的SHUTDOWN ACK消息与DATA数据块捆绑, DATA是最后一个数据块。															
2.	检查A: 端点A舍弃该分组。															
3.	检查B: 端点A的状态不受影响。															
4.	重复测试: 与SHUTDOWN ACK消息捆绑的DATA数据块是该分组的第一个数据块, 程序同上。															
5.	重复测试, 如果与SHUTDOWN ACK消息捆绑的是控制数据块, 则该数据块对于端点A是有效的数据块。															

## 4.5.8 DATA 消息的传送

测试编号：8.1	
参考：RFC 2960 第3.3节和6.9节	
项目：DATA消息	
分项目：未分段的DATA消息	
目的：当DATA消息的长度 $\leq$ 偶联的通路MTU，端点A发送数据时无需分段	
测试预置条件：端点A和端点B的偶联已建立，在端点A设置数据，使其发送的用户数据 $\leq$ MTU的长度	
结构：1	测试类型：VAT、CPT
<p>预期的信号单元顺序：</p> <p style="text-align: center;"> <span style="margin-right: 100px;">端点 B</span> <span style="margin-right: 100px;">端点 A</span> <span style="margin-right: 100px;">高层应用(ULP)</span> </p> <p style="text-align: center;"> <span style="margin-right: 100px;">B和E比特设置为11</span> <span style="margin-right: 100px;">DATA</span> <span style="margin-right: 100px;">Send</span> </p> <p style="text-align: center;"> <span style="margin-right: 100px;">←-----</span> <span style="margin-right: 100px;">←=====</span> <span style="margin-right: 100px;">≦</span> </p> <p style="text-align: center;"> <span style="margin-right: 100px;"></span> <span style="margin-right: 100px;">[TSN=1, Strm=0, Seq=1]</span> <span style="margin-right: 100px;"></span> </p>	
测试说明：	
1.	检查端点A发送的DATA消息的B比特和E比特设置为11。

测试编号：8.2	
参考：RFC 2960 第3.3节和6.9节	
项目：DATA消息	
分项目：分段DATA消息的发送	
目的：当DATA消息的长度大于偶联的通路MTU，端点A发送数据时应进行分段	
测试前置条件：端点A和端点B的偶联已建立，在端点A设置数据，使其发送的用户数据大于MTU的长度，本测试项将用户数据分4段发送	
结构：1	测试类型：VAT、CPT
预期的信号单元顺序：	
端点 B	端点 A
	高层应用(ULP)
	<===== Send
B和E比特设置为10	<----- DATA [TSN=1, Strm=0, Seq=1]
SACK ( TSN=1 )	----->
B和E比特设置为00	<----- DATA [TSN=2, Strm=0, Seq=1]
B和E比特设置为00	<----- DATA [TSN=3, Strm=0, Seq=1]
B和E比特设置为01	<----- DATA [TSN=4, Strm=0, Seq=1]
测试说明：	
1.	检查端点A发送的第一个DATA消息的B比特和E比特设置为10。
2.	检查端点A发送的第二个DATA消息的B比特和E比特设置为00。
3.	检查端点A发送的第三个DATA消息的B比特和E比特设置为00。
4.	检查端点A发送的第四个DATA消息的B比特和E比特设置为01。

测试编号：8.3																																									
参考：RFC 2960 第3.3节和6.9节																																									
项目：DATA消息																																									
分项目：接收分段的DATA消息																																									
目的：验证系统可以正确的接收分段的数据																																									
测试前置条件：端点A和端点B的偶联已建立，在端点B设置数据，发送3个分段的DATA消息																																									
结构：1	测试类型：VAT、CPT																																								
<p>预期的信号单元顺序：</p> <table style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; width: 30%;">端点 B</th> <th style="width: 40%;"></th> <th style="text-align: center; width: 30%;">端点 A</th> <th style="text-align: center; width: 10%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">( TSN=1, Strm=0, Seq =1 )</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK (TSN ACK=1)</td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">( TSN=2, Strm=0, Seq =1 )</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK (TSN ACK=2)</td> <td></td> </tr> <tr> <td style="text-align: center;">DATA</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> </tr> <tr> <td style="text-align: center;">( TSN=3, Strm=0, Seq =1 )</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td style="text-align: center;">SACK (TSN ACK=3)</td> <td></td> </tr> </tbody> </table>		端点 B		端点 A	高层应用(ULP)	DATA	----->			( TSN=1, Strm=0, Seq =1 )					<-----	SACK (TSN ACK=1)		DATA	----->			( TSN=2, Strm=0, Seq =1 )					<-----	SACK (TSN ACK=2)		DATA	----->			( TSN=3, Strm=0, Seq =1 )					<-----	SACK (TSN ACK=3)	
端点 B		端点 A	高层应用(ULP)																																						
DATA	----->																																								
( TSN=1, Strm=0, Seq =1 )																																									
	<-----	SACK (TSN ACK=1)																																							
DATA	----->																																								
( TSN=2, Strm=0, Seq =1 )																																									
	<-----	SACK (TSN ACK=2)																																							
DATA	----->																																								
( TSN=3, Strm=0, Seq =1 )																																									
	<-----	SACK (TSN ACK=3)																																							
测试说明：																																									
1.	端点A的高层收到了一个完整的用户数据。																																								



测试编号: 8.4	
参考: RFC 2960 第6.1节和6.3.2节	
项目: DATA消息	
分项目: 停止T3-rtx定时器	
目的: 验证系统在收到SACK数据块后, 应停止T3-rtx定时器	
测试预置条件: 在端点B设置数据, 由端点B用SACK响应DATA数据块	
结构: 1	测试类型: VAT
<p>预期的信号单元顺序:</p> <pre> sequenceDiagram     participant B as 端点 B     participant A as 端点 A     participant ULP as 高层应用(ULP)     ULP-&gt;&gt;A: Send     A-&gt;&gt;B: DATA (TSN=1)     Note over A: 小于T3-rtx的设置值     B-&gt;&gt;A: SACK (TSN=1)     A-&gt;&gt;B: DATA (TSN=2)     Note over A: 等于T3-rtx的设置值     A-&gt;&gt;B: DATA (TSN=2) 重发的     </pre>	
测试说明:	
1.	检查A: 端点A发送了DATA消息后, 应启动T3-rtx定时器。
2.	检查B: 端点A在收到了SACK后, T3-rtx定时器应停止。

测试编号：8.5																
参考：RFC 2960 第6.1节、第6.3.3节和6.3.2节																
项目：DATA消息																
分项目：T3-rtx定时器超时																
目的：验证系统在T3-rtx定时器超时后，系统应重发DATA数据块																
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，不由端点B响应SACK																
结构：1	测试类型：VAT															
<p>预期的信号单元顺序：</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center; width: 30%;">端点 B</th> <th style="text-align: center; width: 30%;">端点 A</th> <th style="text-align: right; width: 40%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td style="text-align: right;">&lt;===== Send</td> </tr> <tr> <td style="text-align: center;">&lt;-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (TSN=1) 等于T3-rtx的设置值</td> <td></td> </tr> <tr> <td style="text-align: center;">&lt;-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (TSN=1) 重发的 等于T3-rtx的设置值</td> <td></td> </tr> <tr> <td style="text-align: center;">&lt;-----</td> <td style="border-left: 1px solid black; padding-left: 5px;">DATA (TSN=1) 重发的</td> <td></td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)			<===== Send	<-----	DATA (TSN=1) 等于T3-rtx的设置值		<-----	DATA (TSN=1) 重发的 等于T3-rtx的设置值		<-----	DATA (TSN=1) 重发的	
端点 B	端点 A	高层应用(ULP)														
		<===== Send														
<-----	DATA (TSN=1) 等于T3-rtx的设置值															
<-----	DATA (TSN=1) 重发的 等于T3-rtx的设置值															
<-----	DATA (TSN=1) 重发的															
测试说明：																
1.	检查A：端点A发送了DATA消息后，应启动T3-rtx定时器。															
2.	检查B：在T3-rtx定时器超时后，端点A应重发DATA消息。															
3.	检查C：在重发了DATA消息后，T3-rtx定时器重新启动。															

测试编号: 8.6																																																																		
参考: RFC 2960 第6.2节																																																																		
项目: DATA消息																																																																		
分项目: 重复的DATA																																																																		
目的: 验证端点在收到重复的DATA数据块后, 应当在响应的SACK中予以报告																																																																		
测试预置条件: 端点A和端点B的偶联已建立, 在端点B设置数据, 由端点B发送重复的DATA																																																																		
结构: 1	测试类型: VAT																																																																	
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; width: 30%;">端点 B</th> <th style="width: 10%;"></th> <th style="text-align: center; width: 20%;">端点 A</th> <th style="width: 10%;"></th> <th style="text-align: right; width: 30%;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>DATA ( TSN=X )</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK</td> <td>DATA Arrive</td> <td style="text-align: center;">=====&gt;</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">( TSN ACK=X )</td> <td></td> <td></td> </tr> <tr> <td>DATA ( TSN=X )</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td>DATA ( TSN=X )</td> <td style="text-align: center;">-----&gt;</td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK ( TSN ACK=X )</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重复的TSN个数为2</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重复的TSN值为X</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重复的TSN值为X</td> <td></td> <td></td> </tr> <tr> <td>DATA ( TSN=X )</td> <td style="text-align: center;">-----&gt;</td> <td>SACK ( TSN ACK=X )</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重复的TSN个数为1</td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">重复的TSN值为X</td> <td></td> <td></td> </tr> </tbody> </table>		端点 B		端点 A		高层应用(ULP)	DATA ( TSN=X )	----->					<-----	SACK	DATA Arrive	=====>			( TSN ACK=X )			DATA ( TSN=X )	----->				DATA ( TSN=X )	----->					<-----	SACK ( TSN ACK=X )					重复的TSN个数为2					重复的TSN值为X					重复的TSN值为X			DATA ( TSN=X )	----->	SACK ( TSN ACK=X )					重复的TSN个数为1					重复的TSN值为X		
端点 B		端点 A		高层应用(ULP)																																																														
DATA ( TSN=X )	----->																																																																	
	<-----	SACK	DATA Arrive	=====>																																																														
		( TSN ACK=X )																																																																
DATA ( TSN=X )	----->																																																																	
DATA ( TSN=X )	----->																																																																	
	<-----	SACK ( TSN ACK=X )																																																																
		重复的TSN个数为2																																																																
		重复的TSN值为X																																																																
		重复的TSN值为X																																																																
DATA ( TSN=X )	----->	SACK ( TSN ACK=X )																																																																
		重复的TSN个数为1																																																																
		重复的TSN值为X																																																																
测试说明:																																																																		
1.	由端点B发送一个TSN为X的DATA, 端点A用SACK进行响应。端点B再发两个重复的DATA ( TSN=X )。																																																																	
2.	检查A: 端点A响应的SACK中重复的TSN数量字段应为2, 并重复报告两个重复的TSN=X。																																																																	
3.	端点B再发一个重复的DATA ( TSN=X )。																																																																	
4.	检查B: 端点A响应的SACK中重复的TSN数量字段应为1, 并报告重复的TSN=X。																																																																	

测试编号：8.7		
参考：RFC 2960 第6.2.1节		
项目：DATA消息		
分项目：缓冲区容量		
目的：验证系统在收到对端的rwnd指示对端已无可用缓冲区空间时，系统不发送新的DATA		
测试预置条件：端点A和端点B的偶联已建立，在端点B设置数据，由端点B发送的SACK中的rwnd=0		
结构：1		测试类型：VAT
预期的信号单元顺序：		
端点 B	端点 A	高层应用(ULP)
		<===== Send
	DATA ( TSN=1 )	
SACK ( TSN ACK=1 a_rwnd=0 )	----->	<===== Send
	DATA ( TSN=2 ) (已发送)	
		<===== Send
	不发送DATA	
SACK ( TSN ACK=2 a_rwnd=5 )	----->	
	DATA ( TSN=3 )	
		<===== Send
	DATA ( TSN=4 )	
SACK ( TSN ACK=4 )	----->	
测试说明：		
1.	端点A发送一个DATA，端点B响应SACK，其中a_rwnd=0。	
2.	检查A：如果端点A收到的SACK中的a_rwnd=0，则端点A不应发送新的DATA。	
3.	检查B：已发送DATA消息应能正确接收。	

测试编号：8.8		
参考：RFC 2960 第6.1节和7.2节		
项目：DATA消息		
分项目：出现拥塞		
目的：验证如果cwnd等于未证实的数据量，则发送方不应发送新的数据		
测试预置条件：在端点B设置，当端点A处于拥塞状态后时，则发送SACK证实所有未证实的数据		
结构：1		测试类型：VAT
预期的信号单元顺序：		
端点 B	端点 A	高层应用(ULP)
		<===== Send
	<----- DATA ( TSN=1 )	
		<===== Send
	<----- DATA ( TSN=2 )	
		<===== Send
	<----- DATA ( TSN=3 )	
	未证实的数据量等于cwnd	
		<===== Send
SACK ( TSN ACK=3 )	----->	
	<----- DATA ( TSN=4 )	
SACK ( TSN ACK=4 )	----->	
		<===== Send
	<----- DATA ( TSN=5 )	
SACK ( TSN ACK=5 )	----->	
测试说明：		
1.	端点B对端点A发送的DATA不响应SACK。	
2.	检查A：如果端点A未证实的数量已经等于cwnd，则此时端点不应发送新的数据。	
3.	如果cwnd大于0，则到某个目的地地址发送方可以有cwnd未证实的字节。	

测试编号：8.9		
参考：RFC 2960 第6.1节和6.2节		
项目：DATA消息		
分项目：重发功能		
目的：验证如果端点需要发送新的数据，则需要重发的那些未证实的DATA数据块应当优先发送		
测试预置条件：在端点B设置，在B发送的SACK中存在GAP		
结构：1	测试类型：VAT	
预期的信号单元顺序：		
端点 B	端点 A	高层应用(ULP)
		<===== Send
	<----- DATA ( TSN=1 )	
		<===== Send
	<----- DATA ( TSN=2 )	
		<===== Send
	<----- DATA ( TSN=3 )	
		<===== Send
	<----- DATA ( TSN=4 )	
		<===== Send
SACK ( TSN ACK=2, Frag=1, start=2, End=2 )	----->	
	需要重发TSN=3	<===== Send
	<----- DATA ( TSN=3 )	
	<----- DATA ( TSN=5 )	
SACK ( TSN ACK=5 )	----->	
测试说明：		
1.	端点A和B的偶联已建立，端点B发送的SACK中应包含GAP。	
2.	检查A：如果端点A收到的SACK中包含有GAP，则端点A只从发送队列中清除TSN ACK之前的数据。	
3.	检查B：当TSN=3的DATA被标记为重发时，且收到了发送新数据的请求，则端点A应当先发送重发的DATA。	

测试编号: 8.10																									
参考: RFC 2960 第6.6节																									
项目: DATA消息																									
分项目: 顺序递交功能																									
目的: 验证系统使用顺序递交功能时, 在向高层递交用户数据时应当按照流顺序号码进行递交																									
测试前置条件: 在端点B设置数据, 端点B发送的数据的流顺序号码出现失序																									
结构: 1	测试类型: VAT																								
<p>预期的信号单元顺序:</p> <table border="0"> <thead> <tr> <th style="text-align: left;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td>DATA ( TSN =1, Seq=3 Ubit=0 )</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK ( TSN ACK=1 )      无数据递交</td> </tr> <tr> <td>DATA ( TSN =2, Seq=2 Ubit=0 )</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK ( TSN ACK=2 )      无数据递交</td> </tr> <tr> <td>DATA ( TSN =3, Seq=1 Ubit=0 )</td> <td style="text-align: center;">-----&gt;</td> <td></td> </tr> <tr> <td></td> <td style="text-align: center;">&lt;-----</td> <td>SACK ( TSN ACK=3 )</td> </tr> <tr> <td></td> <td style="text-align: center;">DATA Arrive (Seq=1, Seq=2, Seq=3) =====&gt;</td> <td></td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	DATA ( TSN =1, Seq=3 Ubit=0 )	----->			<-----	SACK ( TSN ACK=1 )      无数据递交	DATA ( TSN =2, Seq=2 Ubit=0 )	----->			<-----	SACK ( TSN ACK=2 )      无数据递交	DATA ( TSN =3, Seq=1 Ubit=0 )	----->			<-----	SACK ( TSN ACK=3 )		DATA Arrive (Seq=1, Seq=2, Seq=3) =====>	
端点 B	端点 A	高层应用(ULP)																							
DATA ( TSN =1, Seq=3 Ubit=0 )	----->																								
	<-----	SACK ( TSN ACK=1 )      无数据递交																							
DATA ( TSN =2, Seq=2 Ubit=0 )	----->																								
	<-----	SACK ( TSN ACK=2 )      无数据递交																							
DATA ( TSN =3, Seq=1 Ubit=0 )	----->																								
	<-----	SACK ( TSN ACK=3 )																							
	DATA Arrive (Seq=1, Seq=2, Seq=3) =====>																								
测试说明:																									
1.	端点A和B的偶联已建立, 端点B发送的DATA的流顺序号码是失序的。																								
2.	端点A用SACK对每个收到的DATA进行证实。																								
3.	检查A: 当端点A收到的DATA消息的流顺序正确后, 才向高层递交数据。																								

测试编号: 8.11																						
参考: RFC 2960 第6.6节																						
项目: DATA消息																						
分项目: 无序递交功能																						
目的: 验证系统使用无序递交功能时, 可以根据收到的消息(未分段)顺序直接向高层递交用户数据																						
测试预置条件: 在端点B设置数据, 端点B发送的数据的流顺序号码出现失序																						
结构: 1	测试类型: VAT																					
<p>预期的信号单元顺序:</p> <table border="0" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">端点 B</th> <th style="text-align: center;">端点 A</th> <th style="text-align: right;">高层应用(ULP)</th> </tr> </thead> <tbody> <tr> <td style="vertical-align: top;">           DATA ( TSN =1, Seq=3 Ubit=1 )         </td> <td style="text-align: center; vertical-align: middle;">           -----&gt;         </td> <td></td> </tr> <tr> <td></td> <td style="text-align: center; vertical-align: middle;">           &lt;-----         </td> <td style="vertical-align: top;">           SACK ( TSN ACK=1 )            DATA Arrive (Seq=3) =====&gt;         </td> </tr> <tr> <td style="vertical-align: top;">           DATA ( TSN =2, Seq=2 Ubit=1 )         </td> <td style="text-align: center; vertical-align: middle;">           -----&gt;         </td> <td></td> </tr> <tr> <td></td> <td style="text-align: center; vertical-align: middle;">           &lt;-----         </td> <td style="vertical-align: top;">           SACK ( TSN ACK=2 )            DATA Arrive (Seq=2) =====&gt;         </td> </tr> <tr> <td style="vertical-align: top;">           DATA ( TSN =3, Seq=1 Ubit=0 )         </td> <td style="text-align: center; vertical-align: middle;">           -----&gt;         </td> <td></td> </tr> <tr> <td></td> <td style="text-align: center; vertical-align: middle;">           &lt;-----         </td> <td style="vertical-align: top;">           SACK ( TSN ACK=3 )            DATA Arrive (Seq=1) =====&gt;         </td> </tr> </tbody> </table>		端点 B	端点 A	高层应用(ULP)	DATA ( TSN =1, Seq=3 Ubit=1 )	----->			<-----	SACK ( TSN ACK=1 ) DATA Arrive (Seq=3) =====>	DATA ( TSN =2, Seq=2 Ubit=1 )	----->			<-----	SACK ( TSN ACK=2 ) DATA Arrive (Seq=2) =====>	DATA ( TSN =3, Seq=1 Ubit=0 )	----->			<-----	SACK ( TSN ACK=3 ) DATA Arrive (Seq=1) =====>
端点 B	端点 A	高层应用(ULP)																				
DATA ( TSN =1, Seq=3 Ubit=1 )	----->																					
	<-----	SACK ( TSN ACK=1 ) DATA Arrive (Seq=3) =====>																				
DATA ( TSN =2, Seq=2 Ubit=1 )	----->																					
	<-----	SACK ( TSN ACK=2 ) DATA Arrive (Seq=2) =====>																				
DATA ( TSN =3, Seq=1 Ubit=0 )	----->																					
	<-----	SACK ( TSN ACK=3 ) DATA Arrive (Seq=1) =====>																				
测试说明:																						
1.	端点A和B的偶联已建立, 端点B发送的DATA的流顺序号码是失序的。																					
2.	端点A用SACK对每个收到的DATA进行证实。																					
3.	检查A: 当端点A每收到一个Ubit=1的DATA消息后, 就向高层递交数据。																					



测试编号：8.12	
参考：RFC 2960 第6.6节	
项目：DATA消息	
分项目：收到来自其他传送地址的SACK	
目的：验证如果消息发送到一个多归属的端点，则端点A应当接收来自另外传送地址的SACK	
测试预置条件：在端点B设置数据，端点B是一个多归属的端点，在收到端点A发送的数据后，端点B用另外的传送地址发送SACK予以证实	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
	<===== Send
	<----- DATA (TSN=1 )
	IP地址=X
SACK ( TSN ACK=1 )	-----> 不需要重传DATA
来自IP地址Y	
	<===== Send
	<----- DATA (TSN=2 )
	IP地址=Y
SACK ( TSN ACK=1 )	-----> 不需要重传DATA
来自IP地址X	
测试说明：	
1.	由端点A向IP地址X发送DATA消息，端点B由IP地址Y予以证实。
2.	由端点A向IP地址Y发送DATA消息，端点B由IP地址X予以证实。
3.	检查A：端点A正确接收该SACK，且不需要重发DATA。

测试编号：8.13	
参考：RFC 2960 第6.6节	
项目：DATA消息	
分项目：DATA数据块中未包含用户数据	
目的：验证如果收到的DATA数据块中未包含用户数据，则系统发送ABORT	
测试预置条件：端点A和B的偶联已建立，在端点B设置数据，发送的DATA数据块中未包括用户数据	
结构：1	测试类型：VAT
<p>预期的信号单元顺序：</p> <p style="text-align: center;"> <span style="margin-right: 150px;">端点 B</span> <span style="margin-right: 150px;">端点 A</span> <span>高层应用(ULP)</span> </p> <p style="text-align: center;"> DATA -----&gt; </p> <p style="text-align: center;"> 未包括用户数据 </p> <p style="text-align: center;"> &lt;----- ABORT </p>	
测试说明：	
1.	端点A和端点B的偶联已建立，由端点B发送的DATA数据块中未包含用户数据。即长度等于16。
2.	检查A：端点A用ABORT消息响应，ABORT中的差错原因为“无用户数据”。

测试编号: 8.14	
参考: RFC 2960 第6.6节	
项目: DATA消息	
分项目: SACK中的累积证实的TSN小于上一个收到的SACK的累积TSN的值	
目的: 如果系统收到的SACK的累积证实的TSN小于上一个收到的SACK的累积TSN的值, 则系统应当舍弃该SACK	
测试前置条件: 端点A和B的偶联已建立, 在端点B设置数据, 发送的SACK中累积证实的TSN小于上一个SACK中的值	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A                      高层应用(ULP)
	←===== Send
	DATA ( TSN=1 )
SACK ( TSN ACK=1 )	----->
	←===== Send
	DATA ( TSN=1 )
SACK ( TSN ACK=0 )	----->
测试说明:	
1.	端点A和端点B的偶联已建立, 由端点B发送的第二个SACK中的累积证实的TSN值小于第一个SACK中的值。
2.	检查A: 端点A舍弃该消息。
3.	重复测试: 用TSN ACK=1重新测试。

测试编号：8.15	
参考：RFC 2960 第6.2.1节	
项目：DATA消息	
分项目：在收到的SACK指示丢失的TSN是先前已经用GAP方式证实过的TSN	
目的：验证如果SACK指示丢失的TSN是先前已经用GAP方式证实过的TSN，则系统应重发该数据块	
测试预置条件：端点A和B的偶联已建立，在端点B设置数据，发送的SACK中指示丢失的TSN是先前用GAP方式证实过的TSN	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
	<===== Send
<-----	DATA ( TSN=1 )
	<===== Send
<-----	DATA ( TSN=2 )
	<===== Send
<-----	DATA ( TSN=3 )
SACK ( TSN ACK=1, GAP Star = 2, End = 2 )	<===== Send
<-----	DATA ( TSN=4 )
SACK ( TSN ACK=2, GAP Star = 2, End = 2 )	<===== Send
<-----	DATA ( TSN=5 )
SACK ( TSN ACK=2, GAP Star = 2, End = 3 )	在Rtx定时器超时前收到了4次该SACK
<-----	DATA ( TSN=3 ) 重发
测试说明：	
1.	端点A和端点B的偶联已建立，由端点B发送的SACK指示丢失的TSN是先前已经用GAP方式证实过的TSN。
2.	检查A：端点A重发该丢失的TSN对应的DATA。

#### 4.5.9 数据块的证实

测试编号：9.1		
参考：RFC 2960 第3.3.3节和第6.2.1节		
项目：证实		
分项目：正常的证实方法（立即证实）		
目的：验证系统可以对收到的DATA数据块进行证实		
测试前置条件：在端点B设置数据发送第一个DATA数据块或重发数据块		
结构：1	测试类型：VAT	
预期的信号单元顺序：		
端点 B	端点 A	高层应用(ULP)
第一个DATA (TSN =1)	----->	
	<-----	SACK (TSN ACK=1)
		DATA Arrive =====>
重发的DATA (TSN =1)	----->	
	<-----	SACK (TSN ACK=1)
		DATA Arrive =====>
测试说明：		
1.	端点A和端点B的偶联已建立，由端点B发送的每个DATA都由端点A用SACK予以证实。	
2.	所有的DATA都向高层发送。	
	注：也可以用以一个SACK消息证实多个收到的DATA数据块。	

测试编号: 9.2		
参考: RFC 2960 第6.2.1节		
项目: 证实		
分项目: 正常的证实方法 (延迟证实)		
目的: 验证系统可以延迟对收到的DTAT数据块用一个SACK进行证实		
测试预置条件: 在端点B设置数据发送DATA		
结构: 1	测试类型: VAT	
预期的信号单元顺序:		
端点 B	端点 A	高层应用(ULP)
DATA ( TSN =1 ) ----->		
小于200ms		DATA Arrive =====>
DATA ( TSN =2 ) ----->		
		DATA Arrive =====>
	<----- SACK ( TSN ACK=2 )	
测试说明:		
1.	端点A和端点B的偶联已建立, 由端点B发送的多个DATA都由端点A可以用一个SACK予以证实。	
2.	所有的DATA可以在发送SACK前向高层发送。	
注: 如果端点A使用延迟证实方式, 且在发送SACK前有需要发送的数据, 则可以把SACK与要发送的数据捆绑发送。		

测试编号: 9.3		
参考: RFC 2960 第6.2.1节		
项目: 证实		
分项目: 累积证实的TSN		
目的: 如果接收方在收到的数据中检出GAP, 则应立即发送SACK		
测试预置条件: 端点B设置数据发送的DATA的TSN中包括GAP		
结构: 1	测试类型: VAT	
预期的信号单元顺序:		
端点 B	端点 A	高层应用(ULP)
DATA ( TSN =1 )	----->	DATA Arrive =====>
DATA ( TSN =2 )	----->	DATA Arrive =====>
DATA ( TSN =3 )	----->	DATA Arrive =====>
DATA ( TSN =5 )	----->	
	<-----	SACK ( TSN ACK=3, GAP Star = 2, End = 2 )
测试说明:		
1.	端点A和端点B的偶联已建立, 由端点B发送的多个DATA中出现GAP。	
2.	检查A: 端点A在检出GAP后应当立即发送SACK来报告GAP。	

## 4.5.10 异常情况的测试

测试编号：10.1	
参考：RFC 2960 第6.2.1节	
项目：异常情况的测试	
分项目：数据块类型的编码	
目的：在收到不能识别的数据块时，系统应根据接收到的该字段的高2比特决定所采取的动作	
测试预置条件：端点A和端点B之间的偶联已建立，在端点B设置数据，发送的数据块的类型编码不存在，并把该字段的高两比特设置为不同取值，进行测试	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A                      高层应用(ULP)
1)	
DATA+数据块类型编码为X 高两比特的编码为11	----->
	<----- ERROR
	<----- SACK
2)	
DATA+数据块类型编码为X 高两比特的编码为00	-----> 端点A舍弃整个分组
3)	
DATA+数据块类型编码为X 高两比特的编码为01	-----> 端点A舍弃整个分组
	<----- ERROR
4)	
DATA+数据块类型编码为X 高两比特的编码为10	----->
	<----- SACK
测试说明：	
1.	端点A和B之间的偶联已建立，端点B发送的分组中包含DATA和一个数据块类型未定义的数据块。
2.	检查A：当高2比特为11时，端点A响应SACK，并发送ERROR消息（不识别的数据块类型）。
3.	检查B：当高2比特为00时，端点A直接舍弃该分组。
4.	检查C：当高2比特为01时，端点A舍弃该分组。并发送ERROR消息（不识别的数据块类型）。
5.	检查D：当高2比特为10时，端点A只响应SACK，不发送ERROR消息。



测试编号: 10.2 (1)		
参考: RFC 2960 第6.2.1节		
项目: 异常情况的测试		
分项目: 参数类型的编码		
目的: 在收到不能识别的参数时, 系统应根据接收到的该字段的高2比特决定所采取的动作		
测试预置条件: 端点A和端点B之间的偶联未建立, 在端点B设置数据, 发送的INIT中包含不存在的参数类型编码, 并把该字段的高两比特设置为不同取值, 重复测试		
结构: 1	测试类型: VAT	
预期的信号单元顺序:		
端点 B	端点 A	高层应用(ULP)
1)		
INIT	----->	
不识别参数的高两比特的 编码为11		
	<-----	INIT ACK(不识别的参数)
COOKIE ECHO	----->	
	<-----	COOKIE ACK
		Communication UP =====>
2)		
INIT	----->	
不识别参数的高两比特的 编码为10		
	<-----	INIT ACK
COOKIE ECHO	----->	
	<-----	COOKIE ACK
		Communication UP =====>
测试说明:		
1.	端点A和B之间的偶联未建立, 端点B发送的INIT中包含一个未定义的参数类型。	
2.	检查A: 当高2比特为11时, 端点A响应的INIT ACK消息中应包含不识别的数据块类型。	
3.	偶联应当正常建立。	
4.	检查B: 当高2比特为10时, 端点A正常响应INIT ACK消息, 偶联正常建立。	

测试编号：10.2（2）	
参考：RFC 2960 第6.2.1节	
项目：异常情况的测试	
分项目：参数类型的编码	
目的：在收到不能识别的参数时，系统应根据接收到的该字段的高2比特决定所采取的动作	
测试预置条件：端点A和端点B之间的偶联未建立，在端点B设置数据，发送的INIT中包含不存在的参数类型编码，并把该字段的高两比特设置为不同取值，重复测试	
结构：1	测试类型：VAT
预期的信号单元顺序：	
端点 B	端点 A      ✓      高层应用(ULP)
1)	
INIT	-----> 端点A舍弃该INIT分组
不识别参数的高两比特的 编码为00	
2)	
INIT	-----> 端点A舍弃该INIT分组
不识别参数的高两比特的 编码为01	<----- ERROR（不识别的参数）
测试说明：	
1.	端点A和B之间的偶联未建立，端点B发送的INIT中包含一个未定义的参数类型。
2.	检查A：当高2比特为00时，端点A直接舍弃该分组。
3.	检查B：当高2比特为01时，端点A舍弃该分组，并发送ERROR消息（不识别的数据块类型）。

#### 4.5.11 重传定时器

测试编号：11.1

参考：RFC 2960 第6.3.3节

项目：重传定时器

分项目：如果到某个传送地址的T3-rxt定时器超时，则应当增加T3-rxt定时器的值

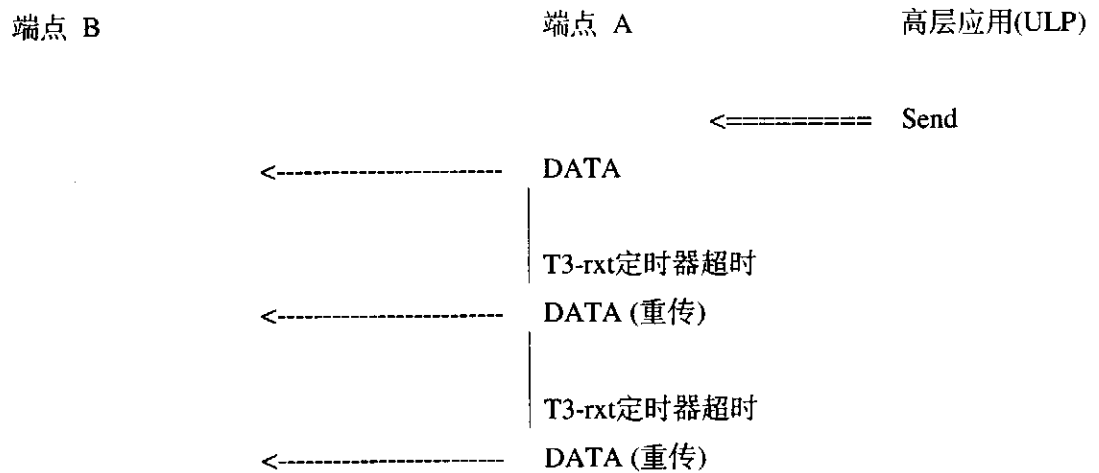
目的：如果到某个传送地址的T3-rxt定时器超时，则应当增加到某个地址的T3-rxt定时器的值

测试预置条件：端点A和端点B之间的偶联已建立，在端点B设置数据，不用SACK响应端点A发送的DATA

结构：1

测试类型：VAT

预期的信号单元顺序：



测试说明：

1. 由端点A发送DATA到端点B，端点B不响应SACK，第一次T3-rxt定时器超时后，端点A重发该DATA；第二次T3-rxt定时器超时后，端点A再次重发该DATA。
2. 检查A：第二个T3-rxt定时器值应当比第一个T3-rxt定时器的值大。
3. 重复测试：端点B设置为多归属的端点，当到所有的传送地址的T3-rxt定时器超时后，才增加T3-rxt定时器的值。

测试编号: 11.2	
参考: RFC 2960 第6.3.3节	
项目: 重传定时器	
分项目: 如果到某个传送地址的T3-rxt定时器超时, 则应当增加RTO的值	
目的: 如果到某个传送地址的T3-rxt定时器超时, 则应当增加到某个地址的T3-rxt定时器的值	
测试预置条件: 端点A和端点B之间的偶联已建立, 在端点B设置数据, 不用SACK响应端点A发送的DATA	
结构: 1	测试类型: VAT
预期的信号单元顺序:	
端点 B	端点 A                      高层应用(ULP)
	<===== Send
<-----	DATA
	T3-rxt定时器超时
<-----	DATA (重传)
	T3-rxt定时器超时
<-----	DATA (重传)
测试说明:	
1.	由端点A发送DATA到端点B, 端点B不响应SACK, 第一次T3-rxt定时器超时后, 端点A重发该DATA; 第二次T3-rxt定时器超时后, 端点A再次重发该DATA。
2.	检查A: 第二个T3-rxt定时器值应当比第一个T3-rxt定时器的值大。