

# OSPF、MTU和LSA打包技术说明

## 目录

[简介](#)

[OSPF数据包大小](#)

[DBD数据包中的MTU](#)

[OSPF行为和将LSA打包到LS更新数据包](#)

[在Cisco Bug ID CSCse01519之前](#)

[在Cisco Bug ID CSCse01519之后](#)

[思科漏洞ID CSCse01519](#)

[概述](#)

[场景](#)

## 简介

本文档介绍在Cisco Bug ID CSCse01519中，开放最短路径优先(OSPF)数据包、最大转换单元(MTU)、链路状态通告(LSA)和链路状态(LS)更新数据包的[交互](#)。

## OSPF数据包大小

路由器上的链路有MTU。传出数据包（如OSPF数据包）不能大于接口MTU。

[请求注解\(RFC\)2328记录](#)OSPF协议第2版。RFC 2328的附录A.1以下列方式描述了OSPF数据包的封装：

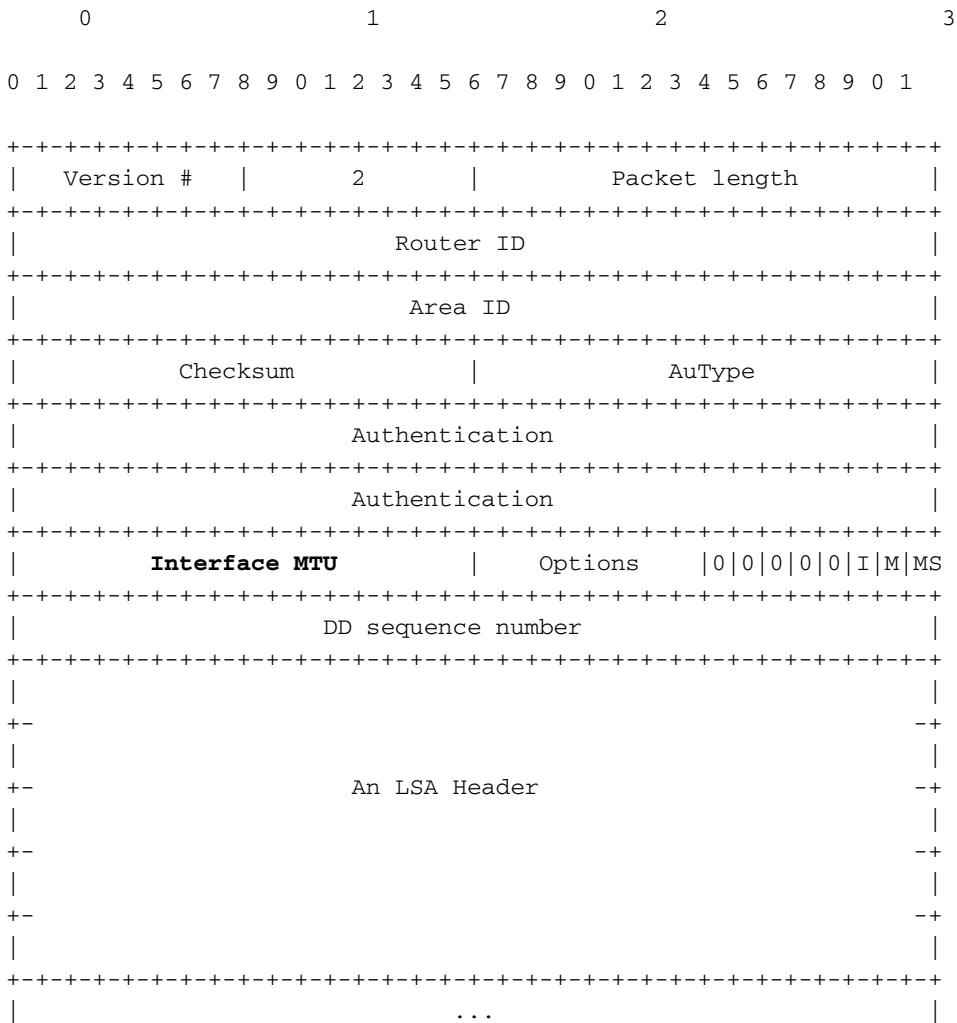
OSPF直接在Internet协议的网络层上运行。因此，OSPF数据包仅由IP和本地数据链路报头封装。

OSPF不定义分段其协议数据包的方法，并且在传输大于网络MTU的数据包时，它取决于IP分段。如有必要，OSPF数据包的长度最多可达65,535字节（包括IP报头）。可能较大的OSPF数据包类型（数据库描述数据包、链路状态请求、链路状态更新和链路状态确认数据包）通常可以拆分为多个单独的协议数据包，而不会丢失功能。建议使用；应尽可能避免IP分段。

LS更新数据包中可以有一个或多个LSA。一个LS更新数据包中的许多LSA称为将LSA打包到LS更新数据包中。

## DBD数据包中的MTU

RFC 2328中也指定了Database Description(DBD)数据包，该数据包描述了OSPF链路状态数据库的内容：



RFC 2328的附录A.3.3.将接口MTU描述为：

最大IP数据报的大小（以字节为单位），可从关联接口发送，而不进行分段。

当初始化OSPF邻接关系时，连接到链路的路由器会在DBD数据包中交换其接口MTU值。

RFC 2328第10.6节规定：

如果Database Description数据包中的Interface MTU字段指示的IP数据报大小大于路由器在接收接口上可以接受的大小，而不进行分段，则Database Description数据包将被拒绝。

使用debug ip ospf adj命令时，您可以看到这些DBD数据包的到达。

在本例中，两个OSPF邻居之间的MTU值不匹配。此路由器的MTU 1600:

```
OSPF: Rcv DBD from 10.100.1.2 on GigabitEthernet0/1 seq 0x2124 opt 0x52 flag 0x2
len 1452 mtu 2000 state EXSTART
OSPF: Nbr 10.100.1.2 has larger interface MTU
另一台OSPF路由器的接口为MTU 2000:
```

```
OSPF: Rcv DBD from 10.100.100.1 on GigabitEthernet0/1 seq 0x89E opt 0x52 flag 0x7
len 32 mtu 1600 state EXCHANGE
OSPF: Nbr 10.100.100.1 has smaller interface MTU
DBD数据包会持续重新传输，直到OSPF邻接关系最终被破坏。
```

```
OSPF: Send DBD to 10.100.1.2 on GigabitEthernet0/1 seq 0x9E6 opt 0x52 flag 0x7
len 32
OSPF: Retransmitting DBD to 10.100.1.2 on GigabitEthernet0/1 [10]
OSPF: Send DBD to 10.100.1.2 on GigabitEthernet0/1 seq 0x9E6 opt 0x52 flag 0x7
len 32
OSPF: Retransmitting DBD to 10.100.1.2 on GigabitEthernet0/1 [11]
%OSPF-5-ADJCHG: Process 1, Nbr 10.100.1.2 on GigabitEthernet0/1 from EXSTART to
DOWN, Neighbor Down: Too many retransmissions
```

## OSPF行为和将LSA打包到LS更新数据包

### 在Cisco Bug ID CSCse01519之前

在Cisco Bug ID [CSCse01519](#)之前，Cisco IOS®软件中的OSPF<sup>构建</sup>的OSPF数据包不超过1500字节，与接口MTU无关。因此，如果接口MTU大于1500字节，OSPF仍然只将最多1500字节打包到OSPF数据包中。这有些低效，因为OSPF可以在链路上发送更大的数据包并实现更大的吞吐量。

**注意：**此场景有一个例外。如果一个LSA的长度超过1500个字节，则OSPF会构建该数据包，无论其大小如何，因为OSPF无法对一个LSA进行分段。然后，路由器的IP堆栈对数据包进行分段，以适应传出接口的MTU。当OSPF路由器有许多链路，并且路由器LSA变得大于链路MTU时，通常会发生这种情况。

同样，如果传出接口的MTU小于1500字节，则OSPF进程仍会构建或封装长达1500字节的OSPF数据包，而路由器的IP堆栈将数据包分段为较小的IP数据包，以适应传出链路的MTU。这通常发生在运行OSPF的两台路由器之间的IPSec隧道。隧道封装字节增加的开销导致MTU小于1500字节。OSPF构建的OSPF数据包长达1500字节，然后在路由器传输数据包之前对数据包进行分段。这是一个额外的低效。

### 在Cisco Bug ID CSCse01519之后

在Cisco Bug ID [CSCse01519](#)之后，IOS中的OSPF可以将OSPF数据包打包为大于1500字节。如果传出接口的MTU大于1500字节，则会发生这种情况。传输效率更高，因为可以将更多信息打包到一个更大的数据包中。换句话说，如果一台OSPF路由器需要将许多外部LSA传输到OSPF邻居，则如果该路由器运行IOS且实施了Cisco Bug ID CSCse01519，它可以将更多外部LSA包装到一个LS更新数据包中。

Cisco Bug ID CSCse01519还允许OSPF生成小于1500字节的数据包。在某些情况下，两个OSPF邻居之间的MTU小于1500字节。在使用IPSec隧道的上一个示例中，OSPF传输小于1500字节的OSPF数据包，并避免IP分段；同样，LSA大于接口MTU的情况也是例外。

## 思科漏洞ID CSCse01519

升级OSPF路由器时，可能会发现由Cisco Bug ID CSCse01519引起的OSPF MTU[问题](#)。

### 概述

许多网络都有通过第2层(L2)交换网络或传输网络(由第2层VPN服务或同步数字层次结构/同步光纤网络(SDH/SONET)网络组成)连接的OSPF邻居。这些传输网络的MTU设置可能与运行OSPF的路由器不同。

虽然所有路由器的MTU设置都应正确，并应反映真实的MTU，但经常会有错误不为人知。

这是一个包含两台运行OSPF的路由器的示例网络。路由器1(R1)和路由器2(R2)通过L2交换机连接。

。

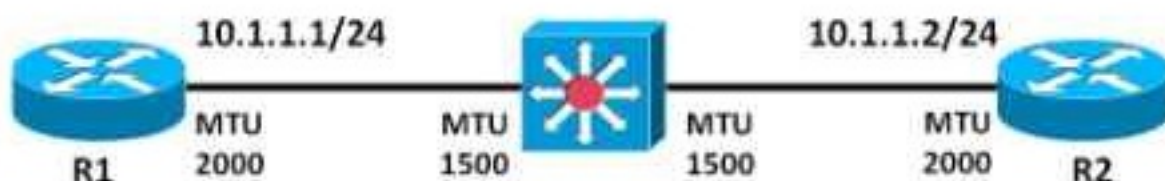


Figure 1 : Example network

在本例中，路由器的千兆以太网接口的MTU设置为2000。L2交换机的MTU仅为1500字节。

如果数据流量的大小从不大于1500字节，则可以使用IOS，而不使用Cisco Bug ID [CSCse01519](#)，因为OSPF数据包从不大于1500字节。但是，如果LSA为1800字节，例如，R1或R2上的OSPF进程会构建一个大于1500字节的LS更新数据包并进行传输，但该数据包会被路由器之间的L2交换机丢弃。

如果R2上的OSPF数据库具有足够的网络，则本地始发的LSA太大，LS更新数据包可能大于接口MTU。

- 如果这些网络是由covering network命令发起的，则这些网络会显示在R2的路由器LSA中。R2会构建一个大于2000字节的路由器LSA并传输它，但IP会将其分段为2000字节，即接口MTU。但L2交换机会丢弃这些数据包。然后，OSPF会无休止地重新传输此数据包，而且OSPF邻接状态从未完全。因此，即使您运行IOS时没有Cisco Bug ID CSCse01519，问题也会立即发现。
- 如果这些网络是由redistribute connected命令发起的，则网络会显示在外部LSA中。OSPF尝试将外部LSA打包到一个LS更新数据包中，该数据包的大小最大为1500字节。在这种情况下，由于接口MTU为2000字节，因此OSPF邻接关系达到“FULL”状态。未立即发现基础MTU不足的问题。当一台路由器升级到IOS时，Cisco Bug ID CSCse01519 ( 仅限注册用户 ) 会发现此问题。

## 场景

假设两台路由器都运行IOS版本，且没有Cisco Bug ID [CSCse01519](#)。

当OSPF邻接关系建立时，请注意R1从未收到大于1500字节的OSPF数据包，尽管接口的MTU为2000。

## 启用debug ip ospf packets命令。

```
OSPF: rcv. v:2 t:1 l:48 rid:10.100.1.2
      aid:0.0.0.0 chk:72CF aut:0 auk: from GigabitEthernet0/1
...
OSPF: rcv. v:2 t:4 l:1468 rid:10.100.1.2
      aid:0.0.0.0 chk:8389 aut:0 auk: from GigabitEthernet0/1
OSPF: rcv. v:2 t:4 l:136 rid:10.100.1.2
...
```

在此调试输出中，“l:1468”是OSPF数据包的长度，因此您可以看到最大的OSPF数据包是1468字节。“t:4”表示OSPF数据包为第4类，即链路状态更新数据包。RFC 2328第4.3节的下表定义了不同的OSPF数据包类型：

类型	数据包名称	协议功能
1	hello	发现/维护邻居
2	数据库说明	汇总数据库内容
3	链路状态请求	数据库下载
4	链路状态更新	数据库更新
5	链路状态确认	泛洪确认

## OSPF邻接关系达到“FULL”状态。

```
R1#show ip ospf neighbor gigabitEthernet 0/1
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
10.100.1.2	0	<b>FULL/</b> -	00:00:34	10.1.1.2	GigabitEthernet0/1

```
R2#show ip ospf neighbor gigabitEthernet 0/1
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
10.100.100.1	0	<b>FULL/</b> -	00:00:34	10.1.1.1	GigabitEthernet0/1

接下来，将R2上的IOS升级到IOS版本，并使用Cisco Bug ID CSCse01519。

```
R2#show ip ospf neighbor gigabitEthernet 0/1
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
10.100.100.1	0	<b>LOADING/</b> -	00:00:33	10.1.1.1	GigabitEthernet0/1

```
R2#show ip ospf neighbor gigabitEthernet 0/1 detail
```

```
Neighbor 10.100.100.1, interface address 10.1.1.1
  In the area 0 via interface GigabitEthernet0/1
  Neighbor priority is 0, State is LOADING, 5 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
  Options is 0x12 in Hello (E-bit L-bit )
  Options is 0x52 in DBD (E-bit L-bit O-bit)
  LLS Options is 0x1 (LR)
  Dead timer due in 00:00:39
  Neighbor is up for 00:00:49
  Index 1/1, retransmission queue length 0, number of retransmission 0
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 0, maximum is 0
  Last retransmission scan time is 0 msec, maximum is 0 msec
Number of retransmissions for last link state request packet 9
  Poll due in 00:00:00
```

```
R2#show ip ospf neighbor gigabitEthernet 0/1 detail
```

```
Neighbor 10.100.100.1, interface address 10.1.1.1
```

```

In the area 0 via interface GigabitEthernet0/1
Neighbor priority is 0, State is LOADING, 5 state changes
DR is 0.0.0.0 BDR is 0.0.0.0
Options is 0x12 in Hello (E-bit L-bit )
Options is 0x52 in DBD (E-bit L-bit O-bit)
LLS Options is 0x1 (LR)
Dead timer due in 00:00:33
Neighbor is up for 00:02:06
Index 1/1, retransmission queue length 0, number of retransmission 0
First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
Last retransmission scan length is 0, maximum is 0
Last retransmission scan time is 0 msec, maximum is 0 msec
Number of retransmissions for last link state request packet 25
Poll due in 00:00:03

```

```

%OSPF-5-ADJCHG: Process 1, Nbr 10.100.100.1 on GigabitEthernet0/1 from LOADING
to DOWN, Neighbor Down: Too many retransmissions

```

OSPF邻接关系停滞在“LOADING”状态，未达到“FULL”状态。重新传输直到OSPF达到其25次重新传输的限制。OSPF尝试再次建立邻接关系，同一问题再次出现，循环无休止地继续。

因此，R2的升级发现了以前隐藏的问题：底层MTU小于OSPF路由器使用的MTU。

当交换机将MTU更改为2000时，传输的OSPF数据包大于1500字节(“l:1980”)，无问题。

```

R1#
OSPF: rcv. v:2 t:3 l:1980 rid:10.100.1.2
aid:0.0.0.0 chk:AC5B aut:0 auk: from GigabitEthernet0/1

```

要检查基本MTU问题，请始终ping大小等于MTU和DF（不分段）位集的OSPF邻居IP地址。

要发现基础MTU的值，请执行ping操作并扫描大小。计算输出中感叹号(!)的数量，以确定正确的MTU。在本示例中，来自ping命令的最后一个回应应答的大小为1500字节。

```

R2#ping
Protocol [ip]:
Target IP address: 10.1.1.1
Repeat count [5]: 1
Datagram size [100]:
Timeout in seconds [2]:
Extended commands [n]: yes
Source address or interface:
Type of service [0]:
Set DF bit in IP header? [no]: yes
Validate reply data? [no]:
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes [n]: yes
Sweep min size [36]: 1460
Sweep max size [18024]: 1540
Sweep interval [1]:
Type escape sequence to abort.
Sending 81, [1460..1540]-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
Packet sent with the DF bit set
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
.....
Success rate is 49 percent (40/81), round-trip min/avg/max = 1/1/4 ms

```