

Polarizzazione CEF

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Come evitare la polarizzazione CEF](#)

Introduzione

Questo documento descrive come la polarizzazione Cisco Express Forwarding (CEF) possa causare un uso non ottimale dei percorsi ridondanti a una rete di destinazione. La polarizzazione CEF è l'effetto che si ha quando un algoritmo hash sceglie un particolare percorso e i percorsi ridondanti rimangono completamente inutilizzati.

Prerequisiti

Requisiti

Nessun requisito specifico previsto per questo documento.

Componenti usati

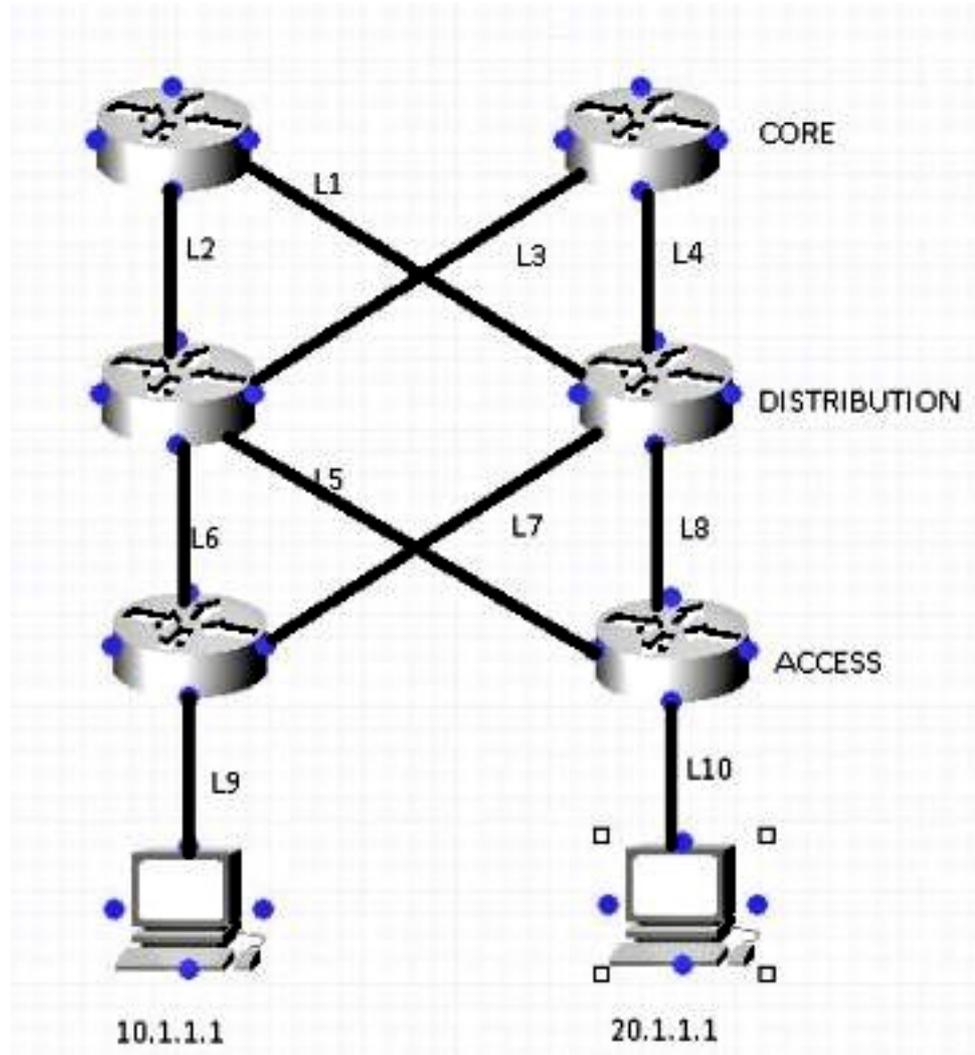
Per questo documento, è stato usato uno switch Cisco Catalyst 6500 con Supervisor Engine 720.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

CEF commuta i pacchetti in base alla tabella di routing popolata dai protocolli di routing, ad esempio Enhanced Interior Gateway Routing Protocol (EIGRP) e Open Shortest Path First (OSPF). Il CEF esegue il bilanciamento del carico una volta calcolata la tabella di routing (RIB, Routing Table). In una struttura di rete gerarchica, possono esistere molti percorsi ridondanti di

livello 3 (L3) a costo uguale. Si consideri questa topologia in cui il traffico passa dal livello di accesso attraverso la distribuzione e il core e nel centro dati.



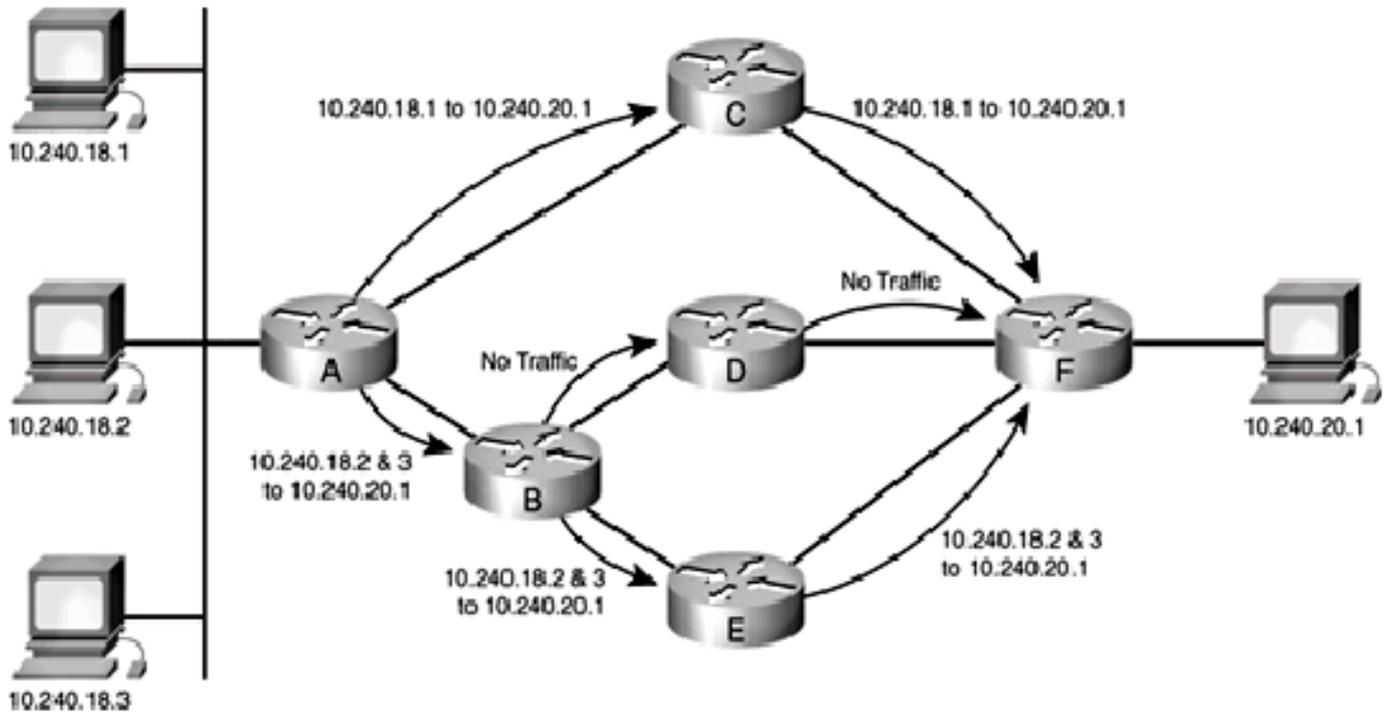
Si supponga che per raggiungere la rete 10.1.1.1 dal router 1 (R1) [in alto a sinistra], vi siano due percorsi con costo uguale (L1, L2). La decisione su quale dei due collegamenti viene utilizzato viene presa da un algoritmo di hashing. Per impostazione predefinita, nell'algoritmo hash vengono utilizzati l'indirizzo IP di origine (SIP) e l'indirizzo IP di destinazione (DIP).

Di seguito è riportata una descrizione del funzionamento dell'algoritmo di hash:

Se i percorsi sono solo due, lo switch/router esegue un'operazione esclusiva-OR (XOR) sui bit dell'ordine inferiore (un bit quando occorre selezionare uno dei due collegamenti, due bit per i collegamenti 3-4 e così via) del SIP e del DIP. L'operazione XOR dello stesso SIP e DIP determina sempre l'utilizzo dello stesso collegamento da parte del pacchetto.

Il pacchetto passa quindi al livello di distribuzione, dove lo stesso algoritmo hash viene utilizzato insieme allo stesso input hash, e sceglie un singolo collegamento per tutti i flussi, lasciando l'altro sottoutilizzato. Questo processo è denominato polarizzazione CEF (utilizzo dello stesso algoritmo hash e dello stesso input hash che determina l'utilizzo di un singolo collegamento ECMP (Equal-Cost Multi-Path) per TUTTI i flussi).

Questo esempio illustra più dettagliatamente questo processo:



1. Il traffico proveniente dalla versione 10.240.18.1 e destinato alla versione 10.240.20.1 entra nella rete sul router A e viene commutato a livello di CEF. Poiché esistono due percorsi uguali alla rete 10.240.20.0/24, gli indirizzi di origine e di destinazione nel pacchetto passano attraverso l'algoritmo hash e il risultato è un percorso specifico utilizzato per raggiungere la destinazione. In questo caso, il percorso dei pacchetti è verso il router C. Da lì, i pacchetti vanno al router F e alla destinazione finale.
- 2.
3. Il traffico proveniente dalla versione 10.240.18.2 e destinato alla versione 10.240.20.1 entra nella rete sul router A e viene anche commutato a livello di CEF. Poiché esistono due percorsi uguali alla rete 10.240.20.0/24, gli indirizzi di origine e di destinazione nel pacchetto passano attraverso l'algoritmo hash e il CEF sceglie un percorso. In questo caso, il percorso dei pacchetti è verso il router B.
- 4.
5. Il traffico proveniente dalla versione 10.240.18.3 e destinato alla versione 10.240.20.1 entra nella rete sul router A e viene anche commutato a livello di CEF. Poiché esistono due percorsi uguali alla rete 10.240.20.0/24, gli indirizzi di origine e di destinazione nel pacchetto passano attraverso l'algoritmo hash e il CEF sceglie un percorso. In questo caso, il percorso dei pacchetti è verso il router B.
- 6.
7. I pacchetti originati da 10.240.18.2 e 10.240.18.3 arrivano entrambi al router B, che a sua volta ha due percorsi pari costo per raggiungere la modalità 10.240.20.1. Esegue nuovamente questi set di coppie di origine e destinazione tramite l'algoritmo hash, che produce gli stessi risultati prodotti dall'algoritmo hash sul router A. Ciò significa che entrambi i flussi di pacchetti passano lungo un percorso - in questo caso, il collegamento verso il router E. Il collegamento al router D non riceve traffico.
- 8.
9. Dopo aver ricevuto il traffico proveniente dalle versioni 10.240.18.2 e 10.240.18.3 sul router E, lo switch viene indirizzato al router F e quindi alla destinazione finale.

Come evitare la polarizzazione CEF

1. Alternare la configurazione degli input di hashing **predefiniti** (SIP e DIP) e **completi** (SIP + DIP + layer4) a ciascun layer della rete.

Catalyst 6500 offre alcune scelte per l'algoritmo di hashing:

Predefinito: utilizzare l'indirizzo IP di origine e di destinazione, assegnando un peso diverso a ciascun collegamento in modo da evitare la polarizzazione. **Semplice:** utilizzare l'indirizzo IP di origine e di destinazione, attribuendo lo stesso peso a ciascun collegamento. **Full (Completo)** - Consente di utilizzare l'indirizzo IP di origine e di destinazione e il numero di porta di livello 4, con pesi diversi. **Full Simple** - Utilizza l'indirizzo IP di origine e di destinazione e il numero di porta di livello 4, con lo stesso peso assegnato a ciascun collegamento.

```
6500(config)#mls ip cef load-sharing ?
full      load balancing algorithm to include L4 ports
simple     load balancing algorithm recommended for a single-stage CEF router
```

```
6500(config)#mls ip cef load-sharing full ?
simple     load balancing algorithm recommended for a single-stage CEF router
<cr>
```

Al momento non esistono comandi per controllare l'algoritmo di condivisione del carico in uso. Il modo migliore per scoprire quale metodo è in uso è controllare la configurazione corrente con il comando **show running-config**. Se non è presente alcuna configurazione a partire da **mls ip cef load-sharing**, viene utilizzato l'algoritmo di ponderazione ineguale di origine e destinazione predefinito.

Nota: 1) Catalyst 6500 non supporta la condivisione del carico per pacchetto. 2) L'opzione **full** NON include un ID universale nell'hash. Se viene utilizzato a ogni livello di una topologia multilivello, è possibile la polarizzazione. Si consiglia di utilizzare l'opzione **simple** con questo comando per ottenere una migliore condivisione del carico e utilizzare un numero inferiore di adiacenze hardware.

2. Alternare tra un numero pari e uno dispari di collegamenti ECMP a ciascun livello della rete. Il bilanciamento del carico CEF non dipende dal modo in cui le route del protocollo vengono inserite nella tabella di routing. Pertanto, le route OSPF presentano lo stesso comportamento dell'EIGRP. In una rete gerarchica in cui sono presenti più router che eseguono la condivisione del carico in una riga, tutti utilizzano lo stesso algoritmo per la condivisione del carico.

L'algoritmo hash esegue il bilanciamento del carico in questo modo per impostazione predefinita:

```
1: 1
2: 7-8
3: 1-1-1
4: 1-1-1-2
5: 1-1-1-1-1
6: 1-2-2-2-2-2
7: 1-1-1-1-1-1-1
8: 1-1-1-2-2-2-2-2
```

Il numero prima dei due punti rappresenta il numero di percorsi pari costo. Il numero che segue i due punti rappresenta la percentuale di traffico inoltrata per percorso.

Ciò significa che:

Per due percorsi di costo uguali, la condivisione del carico è 46,666%-53,333%, non 50%-50%. Per tre percorsi di costo uguali, la condivisione del carico è 33,33%-33,33%-33,33% (come previsto). Per quattro percorsi di costo uguali, la condivisione del carico è del 20%-20%-20%-40% e non del 25%-25%-25%-25%.

Ciò mostra che, quando è presente un numero pari di collegamenti ECMP, il traffico non è con carico bilanciato

Un modo per disabilitare la polarizzazione CEF è il **peso antipolarizzazione**, introdotto nella versione 12.2(17d)SXB2.

Per abilitare il **peso anti-polarizzazione**, immettere questo comando:

```
6500(config)# mls ip cef load-sharing full simple
```

Utilizzare questo comando se esistono due percorsi di costo uguali ed entrambi devono essere utilizzati allo stesso modo. L'aggiunta della parola chiave **simple** consente all'hardware di utilizzare lo stesso numero di adiacenze dell'adiacenza CEF di Cisco IOS®. Senza la parola chiave **simple**, l'hardware installa ulteriori voci adiacenti per evitare la polarizzazione della piattaforma.

3.

4. Cisco IOS ha introdotto il concetto **unique-ID/universal-ID** che aiuta a evitare la polarizzazione CEF. Questo algoritmo, denominato algoritmo universale (il valore predefinito nelle versioni correnti di Cisco IOS), aggiunge un valore specifico del router a 32 bit alla funzione hash (chiamato ID universale - è un valore generato in modo casuale al momento dell'avvio dello switch che può essere controllato manualmente). In questo modo, la funzione hash viene associata a ciascun router con un ID univoco, che assicura che l'hash della stessa coppia origine/destinazione venga eseguito su un valore diverso nei diversi router del percorso. Questo processo offre una migliore condivisione del carico a livello di rete ed evita il problema della polarizzazione. Questo concetto di ID univoco non funziona per un numero pari di percorsi pari costo a causa di una limitazione hardware, ma funziona perfettamente per un numero dispari di percorsi pari costo. Per risolvere questo problema, Cisco IOS aggiunge un collegamento alla tabella delle adiacenze hardware quando è presente un numero pari di percorsi pari al costo, in modo che il sistema creda che vi sia un numero dispari di collegamenti pari al costo.

Per configurare un valore personalizzato per l'ID universale, utilizzare:

```
6500(config)# ip cef load-sharing algorithm universal
```