

# Dealing with mallocfail and High CPU Utilization Resulting From the "Code Red" Worm

Document ID: 12808

---

## Introduction

### Prerequisites

- Requirements
- Components Used
- Conventions

### How the "Code Red" Worm Infects Other Systems

### Advisories that Discuss the "Code Red" Worm

### Symptoms

### Identify the Infected Device

### Prevention Techniques

- Block Traffic to Port 80
- Reduce ARP Input Memory Usage
- Use Cisco Express Forwarding (CEF) Switching

### Cisco Express Forwarding vs Fast Switching

- Fast Switching Behavior and Implications
- Advantages of CEF
- Sample Output: CEF
- Things to Consider

### "Code Red" Frequently Asked Questions and Their Answers

Q. I use NAT, and experience 100 percent CPU utilization in IP Input. When I execute show proc cpu, my CPU utilization is high in interrupt level – 100/99 or 99/98. Can this be related to "Code Red"?

Q. I run IRB, and encounter high CPU utilization in the HyBridge Input process. Why does this happen? Is it related to "Code Red"?

Q. My CPU utilization is high at interrupt level, and I receive flushes if I try a show log. The traffic rate is also only somewhat higher than normal. What is the reason for this?

Q. I can see numerous HTTP connection attempts on my IOS router that runs an ip http-server. Is this because of the "Code Red" worm scan?

### Workarounds

### Related Information

---

## Introduction

This document describes the "Code Red" worm and the problems the worm can cause in a Cisco routing environment. This document also discusses techniques to prevent infestation of the worm, and provides links to related advisories that discuss solutions for worm-related problems.

The "Code Red" worm exploits a vulnerability in the Index Service of the Microsoft Internet Information Server (IIS) version 5.0. When the "Code Red" worm infects a host, it causes the host to probe and infect a random series of IP addresses, which causes a sharp increase in network traffic. This is especially problematic if there are redundant links in the network and/or Cisco Express Forwarding (CEF) is not used to switch packets.

# Prerequisites

## Requirements

There are no specific requirements for this document.

## Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

## Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

## How the "Code Red" Worm Infects Other Systems

The "Code Red" worm attempts to connect to randomly generated IP addresses. Every infected IIS server can attempt to infect the same set of devices. You can trace the source IP address and TCP port of the worm because it is not spoofed. Unicast Reverse Path Forwarding (URPF) cannot suppress a worm attack because the source address is legal.

## Advisories that Discuss the "Code Red" Worm

These advisories discuss the "Code Red" worm, and explain how to patch software affected by the worm:

- Cisco Security Advisory: "'Code Red'" Worm – Customer Impact
- Remote IIS Index Server ISAPI Extension Buffer Overflow
- .ida "Code Red" Worm
- CERT? Advisory CA-2001-19 "Code Red" Worm Exploiting Buffer Overflow In IIS Indexing Service DLL

## Symptoms

Here are some symptoms, which indicate that a Cisco router is affected by the "Code Red" worm:

- Large number of flows in NAT or PAT tables (if you use NAT or PAT).
- Large number of ARP requests or ARP storms in the network (caused by the IP address scan).
- Excessive memory use by IP Input, ARP Input, IP Cache Ager and CEF processes.
- High CPU utilization in ARP, IP Input, CEF and IPC.
- High CPU utilization at interrupt level at low traffic rates, or high CPU utilization at process level in IP Input, if you use NAT.

A low memory condition or sustained high CPU utilization (100 percent) at interrupt level can cause an Cisco IOS® router to reload. The reload is caused by a process that misbehaves due to the stress conditions.

If you do not suspect that devices in your site are infected by or are the target of the "Code Red" worm, see the Related Information section for additional URLs on how to troubleshoot any issues you encounter.

# Identify the Infected Device

Use flow switching to identify the source IP address of the affected device. Configure **ip route-cache flow** on all the interfaces to record all the flows switched by the router.

After a few minutes, issue the **show ip cache flow** command to view the recorded entries. During the initial phase of the "Code Red" worm infection, the worm tries to replicate itself. The replication occurs when the worm sends HT requests to random IP addresses. Therefore, you must look for cache flow entries with destination port 80 (HT., 0050 in hex).

The **show ip cache flow | include 0050** command displays all the cache entries with a TCP port 80 (0050 in hex):

```
Router#show ip cache flow | include 0050
...

scram      scrappers  dative      DstIPAddress  Pr SrcP  DstP  Pkts
V11        193.23.45.35  V13         2.34.56.12    06 0F9F  0050  2
V11        211.101.189.208  Null        158.36.179.59  06 0457  0050  1
V11        193.23.45.35  V13         34.56.233.233  06 3000  0050  1
V11        61.146.138.212  Null        158.36.175.45  06 B301  0050  1
V11        193.23.45.35  V13         98.64.167.174  06 0EED  0050  1
V11        202.96.242.110  Null        158.36.171.82  06 0E71  0050  1
V11        193.23.45.35  V13         123.231.23.45  06 121F  0050  1
V11        193.23.45.35  V13         9.54.33.121    06 1000  0050  1
V11        193.23.45.35  V13         78.124.65.32   06 09B6  0050  1
V11        24.180.26.253  Null        158.36.179.166  06 1132  0050  1
```

If you find an abnormally high number of entries with the same source IP Address, random destination IP Address<sup>1</sup>, DstP = 0050 (HTTP), and Pr = 06 (TCP), you have probably located an infected device. In this output example, the source IP address is 193.23.45.35 and comes from VLAN1.

<sup>1</sup>Another version of the "Code Red" worm, called "Code Red II", does not choose a totally random destination IP address. Instead, "Code Red II" keeps the network portion of the IP address, and chooses a random host portion of the IP address in order to propagate. This allows the worm to spread itself faster within the same network.

"Code Red II " uses these networks and masks:

Mask	Probability of Infection
0.0.0.0	12.5% (random)
255.0.0.0	50.0% (same class A)
255.255.0.0	37.5% (same class B)

Target IP addresses that are excluded are 127.X.X.X and 224.X.X.X, and no octet is allowed to be 0 or 255. In addition, the host does not attempt to re-infect itself.

For more information, refer to Code Red (II) .

Sometimes, you cannot run netflow to detect a "Code Red" infestation attempt. This can be because you run a version of code that does not support netflow, or because the router has insufficient or excessively fragmented memory to enable netflow. Cisco recommends that you do not enable netflow when there are multiple ingress interfaces and only one egress interface on the router, because netflow accounting is performed on the ingress path. In this case, it is better to enable IP accounting on the lone egress interface.

**Note:** The **ip accounting** command disables DCEF. Do not enable IP accounting on any platform where you want to use DCEF switching.

```
Router(config)#interface vlan 1000
Router(config-if)#ip accounting
```

```
Router#show ip accounting
Source          Destination          Packets          Bytes
20.1.145.49     75.246.253.88       2                96
20.1.145.43     17.152.178.57       1                48
20.1.145.49     20.1.49.132         1                48
20.1.104.194    169.187.190.170     2                96
20.1.196.207    20.1.1.11           3               213
20.1.145.43     43.129.220.118     1                48
20.1.25.73      43.209.226.231     1                48
20.1.104.194    169.45.103.230     2                96
20.1.25.73      223.179.8.154      2                96
20.1.104.194    169.85.92.164      2                96
20.1.81.88      20.1.1.11           3               204
20.1.104.194    169.252.106.60     2                96
20.1.145.43     126.60.86.19       2                96
20.1.145.49     43.134.116.199     2                96
20.1.104.194    169.234.36.102     2                96
20.1.145.49     15.159.146.29      2                96
```

In the **show ip accounting** command output, look for source addresses that attempt to send packets to multiple destination addresses. If the infected host is in the scan phase, it attempts to establish HTTP connections to other routers. So you will see attempts to reach multiple IP addresses. Most of these connection attempts normally fail. Therefore, you see only a small number of packets transferred, each with a small byte count. In this example, it is likely that 20.1.145.49 and 20.1.104.194 are infected.

When you run Multi-Layer Switching (MLS) on the Catalyst 5000 Series and the Catalyst 6000 Series, you must take different steps to enable netflow accounting and track down the infestation. In a Cat6000 switch equipped with Supervisor 1 Multilayer Switch Feature Card (MSFC1) or SUP I/MSFC2, netflow-based MLS is enabled by default, but the flow-mode is destination-only. Therefore, the source IP address is not cached. You can enable "full-flow" mode to track down infected hosts with the help of the **set mls flow full** command on the supervisor.

For Hybrid mode, use the **set mls flow full** command:

```
6500-sup(enable)#set mls flow full
Configured IP flowmask is set to full flow.
Warning: Configuring more specific flow mask may dramatically
increase the number of MLS entries.
```

For Native IOS mode, use the **mls flow ip full** command:

```
Router(config)#mls flow ip full
```

When you enable "full-flow" mode, a warning is displayed to indicate a dramatic increase in MLS entries. The impact of the increased MLS entries is justifiable for a short duration if your network is already infested with the "Code Red" worm. The worm causes your MLS entries to be excessive and on the rise.

To view the information collected, use these commands:

For Hybrid mode, use the **set mls flow full** command:

```
6500-sup(enable)#set mls flow full
Configured IP flowmask is set to full flow.
Warning: Configuring more specific flow mask may dramatically
increase the number of MLS entries.
```

For Native IOS mode, use the **mls flow ip full** command:

```
Router(config)#mls flow ip full
```

When you enable "full-flow" mode, a warning is displayed to indicate a dramatic increase in MLS entries. The impact of the increased MLS entries is justifiable for a short duration if your network is already infested with the "Code Red" worm. The worm causes your MLS entries to be excessive and on the rise.

To view the information collected, use these commands:

For Hybrid mode, use the **show mls ent** command:

```
6500-sup(enable)#show mls ent
Destination-IP  Source-IP      Prot  DstPrt  SrcPrt  Destination-Mac  Vlan  EDst
ESrc  DPort      SPort      Stat-Pkts  Stat-Bytes  Uptime  Age
-----
-----
```

**Note:** All these fields are filled in when they are in "full-flow" mode.

For Native IOS mode, use the **show mls ip** command:

```
Router#show mls ip
DstIP          SrcIP          Prot:SrcPort:DstPort  Dst i/f:DstMAC
-----
Pkts          Bytes          SrcDstPorts          SrcDstEncap Age  LastSeen
-----
```

When you determine the source IP address and destination port involved in the attack, you can set MLS back to "destination-only" mode.

For Hybrid mode use the **set mls flow destination** command:

```
6500-sup(enable) set mls flow destination
Usage: set mls flow <destination|destination-source|full>
```

For Native IOS mode, use the **mls flow ip destination** command:

```
Router(config)#mls flow ip destination
```

The Supervisor (SUP) II/MSFC2 combination is protected from attack because CEF switching is performed in the hardware, and netflow statistics are maintained. So, even during a "Code Red" attack, if you enable full-flow mode, the router is not swamped, because of the faster switching mechanism. The commands to enable full-flow mode and display the statistics are the same on both the SUP I/MSFC1 and the SUP II/MSFC2.

## Prevention Techniques

Use the techniques listed in this section to minimize the impact of the "Code Red" worm on the router.

### Block Traffic to Port 80

If it is feasible in your network, the easiest way to prevent the "Code Red" attack is to block all traffic to port 80, which is the well known port for WWW. Build an access-list to deny IP packets destined to port 80 and apply it inbound on the interface that faces the infection source.

## Reduce ARP Input Memory Usage

ARP Input uses up huge amounts of memory when a static route points to a broadcast interface, like this:

```
ip route 0.0.0.0 0.0.0.0 Vlan3
```

Every packet for the default route is sent to the VLAN3. However, there is no next hop IP address specified, and so, the router sends an ARP request for the destination IP address. The next hop router for that destination replies with its own MAC address, unless Proxy ARP is disabled. The reply from the router creates an additional entry in the ARP table where the destination IP address of the packet is mapped to the next-hop MAC address. The "Code Red" worm sends packets to random IP addresses, which adds a new ARP entry for each random destination address. Each new ARP entry consumes more and more memory under the ARP Input process.

Do not create a static default route to an interface, especially if the interface is broadcast (Ethernet/Fast Ethernet/GE/SMDS) or multipoint (Frame Relay/ATM). Any static default route must point to the IP address of the next hop router. After you change the default route to point to the next hop IP address, use the **clear arp-cache** command to clear all the ARP entries. This command fixes the memory utilization problem.

## Use Cisco Express Forwarding (CEF) Switching

In order to lower CPU utilization on an IOS router, change from Fast/Optimum/Netflow switching to CEF switching. There are a few caveats to enable CEF. The next section discusses the difference between CEF and fast switching, and explains the implications when you enable CEF.

## Cisco Express Forwarding vs Fast Switching

Enable CEF to alleviate the increased traffic load caused by the "Code Red" worm. Cisco IOS® Software Releases 11.1( )CC, 12.0, and later support CEF on the Cisco 7200/7500/GSR platforms. Support for CEF on other platforms is available in Cisco IOS Software Release 12.0 or later. You can investigate further with the Software Advisor ( registered customers only) tool.

Sometimes, you cannot enable CEF on all routers due to one of these reasons:

- Insufficient memory
- Unsupported platform architectures
- Unsupported interface encapsulations

## Fast Switching Behavior and Implications

Here are the implications when you use fast switching:

- Traffic driven cache The cache is empty until the router switches packets and populates the cache.
- First packet is process switched The first packet is process-switched, because the cache is initially empty.
- Granular cache The cache is built at a granularity of the most specific Routing Information Base (RIB) entry part of a major net. If RIB has /24s for major net 131.108.0.0, the cache is built with /24s for this major network.
- /32 cache is used /32 cache is used to balance the load for each destination. When the cache balances load, the cache is built with /32s for that major net.

**Note:** These last two issues can potentially cause a huge cache that would consume all memory.

- Caching at major network boundaries With default route, caching is performed at major network boundaries.
- The Cache Ager The cache ager runs every minute and checks 1/20th (5 percent) of the cache for unused entries under normal memory conditions, and 1/4th (25 percent) of the cache in a low memory condition (200k).

In order to change the above values, use the **ip cache-ager-interval X Y Z** command, where:

- X is <0–2147483> number of seconds between ager runs. Default = 60 seconds.
- Y is <2–50> 1/(Y+1) of cache to age per run (low memory). Default = 4.
- Z is <3–100> 1/(Z+1) of cache to age per run (normal). Default = 20.

Here is a sample configuration that uses **ip cache-ager 60 5 25**.

```
Router#show ip cache
IP routing cache 2 entries, 332 bytes
  27 adds, 25 invalidates, 0 refcounts
Cache aged by 1/25 every 60 seconds (1/5 when memory is low).
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 03:55:12 ago

Prefix/Length      Age      Interface      Next Hop
4.4.4.1/32         03:44:53   Serial1        4.4.4.1
192.168.9.0/24    00:03:15   Ethernet1      20.4.4.1

Router#show ip cache verbose
IP routing cache 2 entries, 332 bytes
  27 adds, 25 invalidates, 0 refcounts
Cache aged by 1/25 every 60 seconds (1/5 when memory is low).
Minimum invalidation interval 2 seconds, maximum interval 5 seconds,
  quiet interval 3 seconds, threshold 0 requests
Invalidation rate 0 in last second, 0 in last 3 seconds
Last full cache invalidation occurred 03:57:31 ago
Prefix/Length      Age      Interface      Next Hop
4.4.4.1/32-24      4        0F000800
192.168.9.0/24-0   00:05:35 Ethernet1      20.4.4.1
                   14      00000C34A7FC00000C13DBA90800
```

Based on the setting of your cache ager, some percentage of your cache entries age out of your fast-cache table. When entries age quickly, a larger percentage of the fast-cache table ages, and the cache table becomes smaller. As a result, memory consumption on the router reduces. A disadvantage is that traffic continues to flow for the entries that were aged out of the cache table. Initial packets are process-switched, which causes a short spike in CPU consumption in **IP Input** until a new cache entry is built for the flow.

From Cisco IOS Software Releases 10.3(8), 11.0(3) and later, the IP cache ager is handled differently, as explained here:

- The **ip cache-ager-interval** and **ip cache-invalidate-delay** commands are available only if the **service internal** command is defined in the configuration.
- If the period between ager invalidation runs is set to 0, the ager process is disabled entirely.
- Time is expressed in seconds.

**Note:** When you execute these commands, the CPU utilization of the router increases. Use these commands only when absolutely necessary.

```
Router#clear ip cache ?
```

A.B.C.D Address prefix  
<CR>--> will clear the entire cache and free the memory used by it!

```
Router#debug ip cache
IP cache debugging is on
```

## Advantages of CEF

- The Forwarding Information Base (FIB) table is built based on the routing table. Therefore forwarding information exists before the first packet is forwarded. The FIB also contains /32 entries for directly connected LAN hosts.
- The Adjacency (ADJ) table contains the Layer 2 rewrite information for next-hops and directly-connected hosts (an ARP entry creates a CEF adjacency).
- There is no cache ager concept with CEF to spike CPU utilization. A FIB entry is deleted if a routing table entry is deleted.



**Caution:** Again, a default route that points to a broadcast or multipoint interface means that the router sends ARP requests for every new destination. ARP requests from the router potentially create a huge adjacency table until the router runs out of memory. If CEF fails to allocate memory CEF/DCEF disables itself. You will need to manually enable CEF/DCEF again.

## Sample Output: CEF

Here is some sample output of the **show ip cef summary** command, that shows memory usage. This output is a snapshot from a Cisco 7200 route server with Cisco IOS Software Release 12.0.

```
Router>show ip cef summary
IP CEF with switching (Table Version 2620746)
 109212 routes, 0 reresolve, 0 unresolved (0 old, 0 new), peak 84625
 109212 leaves, 8000 nodes, 22299136 bytes, 2620745 inserts, 2511533
 invalidations
 17 load sharing elements, 5712 bytes, 109202 references
 universal per-destination load sharing algorithm, id 6886D006
 1 CEF resets, 1 revisions of existing leaves
 1 in-place/0 aborted modifications
 Resolution Timer: Exponential (currently 1s, peak 16s)
 refcounts: 2258679 leaf, 2048256 node
```

Adjacency Table has 16 adjacencies

```
Router>show processes memory | include CEF
PID TTY Allocated Freed Holding Getbufs Retbufs Process
 73 0 147300 1700 146708 0 0 CEF process
 84 0 608 0 7404 0 0 CEF Scanner
```

```
Router>show processes memory | include BGP
 2 0 6891444 6891444 6864 0 0 BGP Open
 80 0 3444 2296 8028 0 0 BGP Open
 86 0 477568 476420 7944 0 0 BGP Open
 87 0 2969013892 102734200 338145696 0 0 BGP Router
 88 0 56693560 2517286276 7440 131160 4954624 BGP I/O
 89 0 69280 68633812 75308 0 0 BGP Scanner
 91 0 6564264 6564264 6876 0 0 BGP Open
 101 0 7635944 7633052 6796 780 0 BGP Open
 104 0 7591724 7591724 6796 0 0 BGP Open
 105 0 7269732 7266840 6796 780 0 BGP Open
 109 0 7600908 7600908 6796 0 0 BGP Open
```

110 0 7268584 7265692 6796 780 0 BGP Open

Router>show memory summary | include FIB

Alloc PC	Size	Blocks	Bytes	What
0x60B8821C	448	7	3136	FIB: FIBIDB
0x60B88610	12000	1	12000	FIB: HWIDB MAP TABLE
0x60B88780	472	6	2832	FIB: FIBHWIDB
0x60B88780	508	1	508	FIB: FIBHWIDB
0x60B8CF9C	1904	1	1904	FIB 1 path chunk pool
0x60B8CF9C	65540	1	65540	FIB 1 path chunk pool
0x60BAC004	1904	252	479808	FIB 1 path chun
0x60BAC004	65540	252	16516080	FIB 1 path chun

Router>show memory summary | include CEF

0x60B8CD84	4884	1	4884	CEF traffic info
0x60B8CF7C	44	1	44	CEF process
0x60B9D12C	14084	1	14084	CEF arp throttle chunk
0x60B9D158	828	1	828	CEF loadinfo chunk
0x60B9D158	65540	1	65540	CEF loadinfo chunk
0x60B9D180	128	1	128	CEF walker chunk
0x60B9D180	368	1	368	CEF walker chunk
0x60BA139C	24	5	120	CEF process
0x60BA139C	40	1	40	CEF process
0x60BA13A8	24	4	96	CEF process
0x60BA13A8	40	1	40	CEF process
0x60BA13A8	72	1	72	CEF process
0x60BA245C	80	1	80	CEF process
0x60BA2468	60	1	60	CEF process
0x60BA65A8	65488	1	65488	CEF up event chunk

Router>show memory summary | include adj

0x60B9F6C0	280	1	280	NULL adjacency
0x60B9F734	280	1	280	PUNT adjacency
0x60B9F7A4	280	1	280	DROP adjacency
0x60B9F814	280	1	280	Glean adjacency
0x60B9F884	280	1	280	Discard adjacency
0x60B9F9F8	65488	1	65488	Protocol adjacency chunk

## Things to Consider

When the number of flows is large, CEF typically consumes less memory than fast switching. If memory is already consumed by a fast switching cache, you must clear the ARP cache (through the **clear ip arp** command) before you enable CEF.

**Note:** When you clear the cache, a spike is caused in the CPU utilization of the router.

## "Code Red" Frequently Asked Questions and Their Answers

**Q. I use NAT, and experience 100 percent CPU utilization in IP Input. When I execute show proc cpu, my CPU utilization is high in interrupt level – 100/99 or 99/98. Can this be related to "Code Red"?**

A. There is recently fixed a NAT Cisco bug (CSCdu63623 ( registered customers only) ) that involves scalability. When there are tens of thousands of NAT flows (based on the platform type), the bug causes 100 percent CPU utilization at process or interrupt level.

In order to determine whether this bug is the reason, issue the **show align** command, and verify whether the router faces alignment errors. If you do see alignment errors or spurious memory accesses, issue the **show align** command a couple of times and see if the errors are on the rise. If the number of errors is on the rise, alignment errors can be the cause of high CPU utilization at interrupt level, and not Cisco bug CSCdu63623 ( registered customers only) . For more information, refer to Troubleshooting Spurious Accesses and Alignment Errors.

The **show ip nat translation** command displays the number of active translations. The meltdown point for an NPE-300 class processor is about 20,000 to 40,000 translations. This number varies based on the platform.

This meltdown problem was observed previously by a couple of customers, but after "Code Red", more customers have experienced this problem. The only workaround is to run NAT (instead of PAT), so that there are fewer active translations. If you have a 7200, use an NSE-1, and lower the NAT timeout values.

## Q. I run IRB, and encounter high CPU utilization in the HyBridge Input process. Why does this happen? Is it related to "Code Red"?

A. The HyBridge Input process handles any packets that cannot be fast-switched by the IRB process. The inability of the IRB process to fast-switch a packet can be because:

- The packet is a broadcast packet.
- The packet is a multicast packet.
- The destination is unknown, and ARP needs to be triggered.
- There are spanning tree BPDUs.

HyBridge Input encounters problems if there are thousands of point-to-point interfaces in the same bridge group. HyBridge Input also encounters issues (but to a lesser extent) if there are thousands of VSSs in the same multipoint interface.

What are possible reasons for problems with IRB? Assume that a device infected with "Code red" scans IP addresses.

- The router needs to send an ARP request for each destination IP address. A flood of ARP requests result on every VC in the bridge group for each address that is scanned. The normal ARP process does not cause a CPU problem. However, if there is an ARP entry without a bridge entry, the router floods packets destined for addresses for which ARP entries already exist. This can cause high CPU utilization because the traffic is process-switched. To avoid the problem, increase the bridge-aging time (default 300 seconds or 5 minutes) to match or exceed the ARP timeout (default 4 hours) so that the two timers are synchronized.
- The address that the end host attempts to infect is a broadcast address. The router does the equivalent of a subnet broadcast that needs to be replicated by the HyBridge Input process. This does not happen if the **no ip directed-broadcast** command is configured. From Cisco IOS Software Release 12.0, the **ip directed-broadcast** command is disabled by default, which causes all IP-directed broadcasts to be dropped.
- Here is a side note, unrelated to "Code Red", and related to IRB architectures:

Layer 2 multicast and broadcast packets need to be replicated. Therefore, a problem with IPX servers that run on a broadcast segment can bring the link down. You can use subscriber policies to avoid the problem. For more information, refer to xDSL Bridge Support. You must

- also consider bridge access-lists, which limit the type of traffic allowed to pass through the router.
- In order to alleviate this IRB problem, you can use multiple bridge groups, and ensure that there is a one-to-one mapping for BVIs, sub-interfaces and VCs.
  - RBE is superior to IRB because it avoids the bridging stack altogether. You can migrate to RBE from IRB. These Cisco bugs inspire such migration:

- ◆ CSCdr11146 ( registered customers only)
- ◆ CSCdp18572 ( registered customers only)
- ◆ CSCds40806 ( registered customers only)

## **Q.My CPU utilization is high at interrupt level, and I receive flushes if I try a show log. The traffic rate is also only somewhat higher than normal. What is the reason for this?**

A. Here is an example of the **show logging** command output:

```
Router#show logging
  Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
                                     ^
                                     this value is non-zero
  Console logging: level debugging, 9 messages logged
```

Check whether you log to the console. If so, check whether there are traffic HTTP requests. Next, check whether there are any access-lists with log keywords or debugs that watch particular IP flows. If flushes are on the rise, it can be because the console, usually a 9600 baud device, is unable to handle the amount of information received. In this scenario, the router disables interrupts and does nothing but process console messages. The solution is to disable console logging or remove whatever type of logging you perform.

## **Q. I can see numerous HTTP connection attempts on my IOS router that runs an ip http-server. Is this because of the "Code Red" worm scan?**

A."Code Red" can be the reason here. Cisco recommends that you disable the **ip http server** command on the IOS router so that it need not deal with numerous connection attempts from infected hosts.

## **Workarounds**

There are various workarounds that are discussed in the Advisories that Discuss the "Code Red" Worm section. Refer to the advisories for the workarounds.

Another method to block the "Code Red" worm at network ingress points uses Network-Based Application Recognition (NBAR) and Access Control Lists (ACLs) within IOS software on Cisco routers. Use this method in conjunction with the recommended patches for IIS servers from Microsoft. For more information on this method, refer to Using NBAR and ACLs for Blocking the "Code Red" Worm at Network Ingress Points.

---

## **Related Information**

- **Troubleshooting Memory Problems**
- **Troubleshooting Buffer Leaks**
- **Troubleshooting High CPU Utilization on Cisco Routers**
- **Troubleshooting Router Crashes**
- **Troubleshooting TechNotes – Routers**

- **Troubleshooting the Router**
  - **Technical Support & Documentation – Cisco Systems**
- 

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

---

Updated: Jun 20, 2008

Document ID: 12808

---