

# [UNIX] Bug in Linux 2.4 and IPTables MAC Match Module

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2001-10/0043.html>

---

*From:* [support@securiteam.com](mailto:support@securiteam.com)

*Date:* 10/16/01

From: [support@securiteam.com](mailto:support@securiteam.com)

To: [list@securiteam.com](mailto:list@securiteam.com)

Subject: [UNIX] Bug in Linux 2.4 and IPTables MAC Match Module

Message-Id: <20011016121550.D5855138C1@mail.der-keiler.de>

Date: Tue, 16 Oct 2001 14:15:50 +0200 (CEST)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> – Know that you're safe.

-----

## Bug in Linux 2.4 and IPTables MAC Match Module

---

### SUMMARY

The Linux 2.4 kernel includes a new and very powerful firewalling, NAT, and packet mangling architecture called Netfilter. The main component of Netfilter is iptables, a generic structure for allowing firewall rules to perform NAT, mangle packets, and access custom extensions for packet matching and mangling.

One of the extensions supplied by default is the MAC module, which matches packets traveling through the firewall based on their MAC (Ethernet hardware) address. This module offers administrators some protection against malicious internal (or directly connected) users who spoof or change their IP address.

The MAC module does not correctly match very small packets. For example, small ping packets can be generated by the UNIX command 'ping somehost -s 4', or similarly under Windows with 'ping somehost -l 4'. Netcat with the -u option can generate small UDP packets which exhibit the same problem.

### DETAILS

## Securiteam: [UNIX] Bug in Linux 2.4 and IPTables MAC Match Modu

Recreation:

To reproduce the problem, you will need 2 machines:

- Victim, which runs iptables.
- Attacker, which can generate small ICMP or UDP packets.

We have used the DNS names 'Victim' and 'Attacker' to represent the IP addresses of these machines, and AT:TA:CK:ER:00:00 as the MAC address of the attacker. Please substitute real values if attempting to reproduce this problem.

On Victim, at a root prompt:

```
victim# iptables -P INPUT ACCEPT
victim# iptables -F INPUT
victim# iptables -I INPUT -p icmp -m mac --mac-source AT:TA:CK:ER:00:00
-j DROP
victim# iptables -L INPUT -v
Chain INPUT (policy ACCEPT xxxx packets, xxxxxxxx bytes)
pkts bytes target prot opt in out source
destination
  0 0 DROP icmp -- any any anywhere
anywhere MAC AT:TA:CK:ER:00:00
```

[note that the packet and byte counters are zero]

On Attacker (assuming Attacker runs Linux or similar)

```
attacker# ping -s 8 -c 1 Victim
PING Victim (xx.xx.xx.xx) from xx.xx.xx.xx : 8(36) bytes of data.

---- xx.xx.xx.xx ping statistics ----
1 packets transmitted, 0 packets received, 100% packet loss
```

[the ping packets were dropped, correctly]

On Victim again:

```
victim# iptables -L INPUT -v
Chain INPUT (policy ACCEPT 231 packets, 39475 bytes)
pkts bytes target prot opt in out source
destination
  1 36 DROP icmp -- any any anywhere
anywhere MAC 00:03:47:87:BA:C5
```

[note that the packet and byte counters have increased, the packet counter is showing 1 packet which is correct]

Now back to Attacker:

## Securiteam: [UNIX] Bug in Linux 2.4 and IPTables MAC Match Modu

```
attacker# ping -s 4 -c 1 Victim
PING Victim (xx.xx.xx.xx) from xx.xx.xx.xx : 4(32) bytes of data.
12 bytes from xx.xx.xx.xx: icmp_seq=0 ttl=255
```

```
--- xx.xx.xx.xx ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
```

[note that this time, the ping packet was replied to, not dropped by the rule]

Finally, back to Victim:

```
victim# iptables -L INPUT -v
Chain INPUT (policy ACCEPT 231 packets, 39475 bytes)
 pkts bytes target prot opt in out source
destination
 1 32 DROP icmp -- any any anywhere
anywhere MAC AT:TA:CK:ER:00:00
```

[note that the packet counters have not increased, the packet did not match the drop rule]

### Implications:

From a security point of view:

- Malicious internal users may evade restrictions placed on their MAC address in some cases. For example, they might ping internal or external hosts to determine whether they are running, even though firewall rules disallow this.
- They may also use small ICMP or UDP packets to leak information through the firewall, if no other rule prevents them from doing so.
- Several applications use small ICMP or UDP packets, for example ping, netcat, and Symantec pcAnywhere. Administrators will not be able to restrict the use of these programs to certain known MAC addresses, because of this bug. This may result in lower overall security (especially as that are no complete workaround yet).

### Solution:

Harald Welte, Netfilter core developer, has released a patch that has been verified to fix the problem described. The patch is very small and can be applied by hand (note that the patch may be line-wrapped).

```
--- linux-2.4.9/net/ipv4/netfilter/ipt_mac.c Tue Oct 2 18:50:56 2001
+++ linux-2.4.9-ipt_mac-fix/net/ipv4/netfilter/ipt_mac.c Tue Oct 2
19:32:20 2001
@@ -20,7 +20,7 @@
```

```
    /* Is mac pointer valid? */
    return (skb->mac.raw >= skb->head
- && skb->mac.raw < skb->head + skb->len - ETH_HLEN
+ && (skb->mac.raw + ETH_HLEN) <= skb->data
    /* If so, compare... */
```

## Securiteam: [UNIX] Bug in Linux 2.4 and IPTables MAC Match Modu

```
&& ((memcmp(skb->mac.ethernet->h_source, info->srcaddr,  
ETH_ALEN  
== 0) ^ info->invert));
```

### Workaround:

The simplest, but least secure, workaround is to avoid matching by MAC address, but only match on IP address. This is common practice, but less secure than matching by MAC address.

Another workaround is to use the latest version of iptables (1.2.3) from <http://netfilter.samba.org>. This includes a module called "length" which can be used to match small packets. Some administrators might like to allow ICMP and/or UDP packets below a certain size with a command like this (UNTESTED):

```
iptables -I INPUT -p icmp -m length --length 0:4 -j ACCEPT
```

Note that using such a command will reduce the security of your iptables-protected hosts.

In any case, a new version of iptables should be available soon, which fixes this bug.

### ADDITIONAL INFORMATION

The information has been provided by <mailto:chris@netservers.co.uk>  
Chris Wilson.

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

[list-unsubscribe@securiteam.com](mailto:list-unsubscribe@securiteam.com)

In order to subscribe to the mailing list, simply forward this email to: [list-subscribe@securiteam.com](mailto:list-subscribe@securiteam.com)

=====  
=====

### DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

- 
- **Previous message:** [support@securiteam.com](mailto:support@securiteam.com): "[NEWS] Cisco PIX Firewall Manager Password Disclosure Vulnerability"
  - **Messages sorted by:** [\[ date \]](#) [\[ thread \]](#) [\[ subject \]](#) [\[ author \]](#) [\[ attachment \]](#)