Pass the SALT

When Priority Isn’t Enough:

Exploiting the VRRP Tie-Breaking IP Mechanism

Geoffrey Sauvageot-Berland – pentester, security researcher, and occasional lecturer

Slides :

https://urlr.me/UY7Dnm

orange™ Cyberdefense

Table of contents

01

Introduction

What is VRRP ?

~> Network protocol (OSI layer 3)

~> Used to guarantee high availability of several devices (routers, servers…)

~> Warning : High Availability (Failover, ~~Load balancer~~)

~> Open-standard

Why Use VRRP?

~> Interoperability across several devices, unlike HSRP or GLBP (cisco ownership)

~> Easy to configure

~> Enables transparent failover between devices (“automatic failover”)

How VRRP works ? Quick reminder

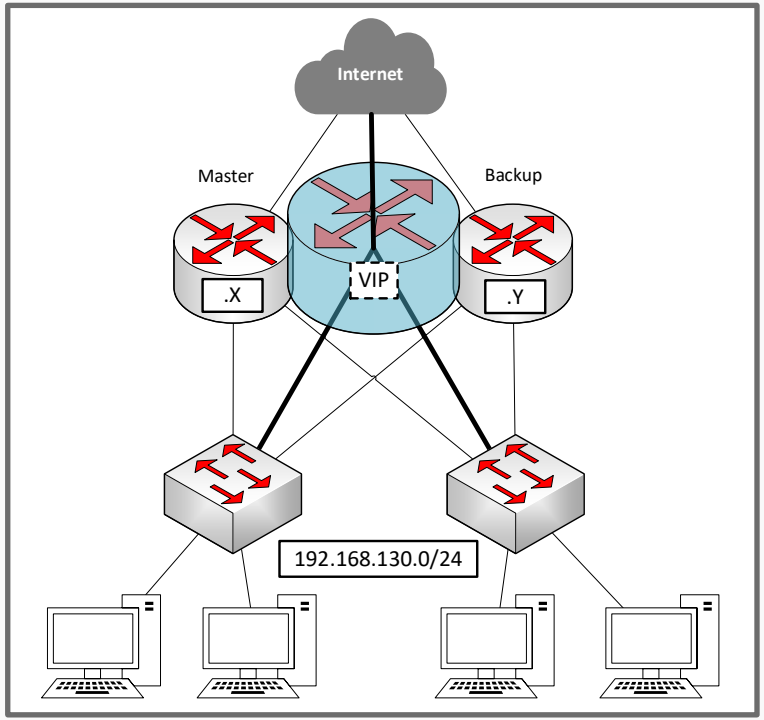
~> Creation of a VIP (Virtual IP address)

~> Shared among a group of nodes identified by a “VRID”

~> Only one node is elected as the Master

~> Priority values (0–255) are used for the election process

~> In case of a crash, a backup automatically takes over



02

VRRP priority conflict dilemma

VRRP priority conflict dilemma – Lab

~> 3 nodes

~> 2 legit and 1 rogue (attacker) with the same VRRP conf (VRID, priority, etc.) of the master

~> In this case, the IP tie-breaking mechanism will be triggered

```
                    +-----------------------------+
                    |     VIP: 192.168.130.180    |
                    +-----------------------------+
                                  |
+-------------------------------------------------------------------------+
| - VRRPv3                                                                |
| - VRID 51                                                               |
|     +-------------------------+       +-------------------------+       |
|     |          Master         |       |          Backup         |       |
|     |  IP: 192.168.130.100    |-------|  IP: 192.168.130.99     |       |
|     |  VRRP Priority: 255     |       |  VRRP Priority: 254     |       |
|     +-------------------------+       +-------------------------+       |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
|                                                                         |
+-------------------------------------------------------------------------+
```

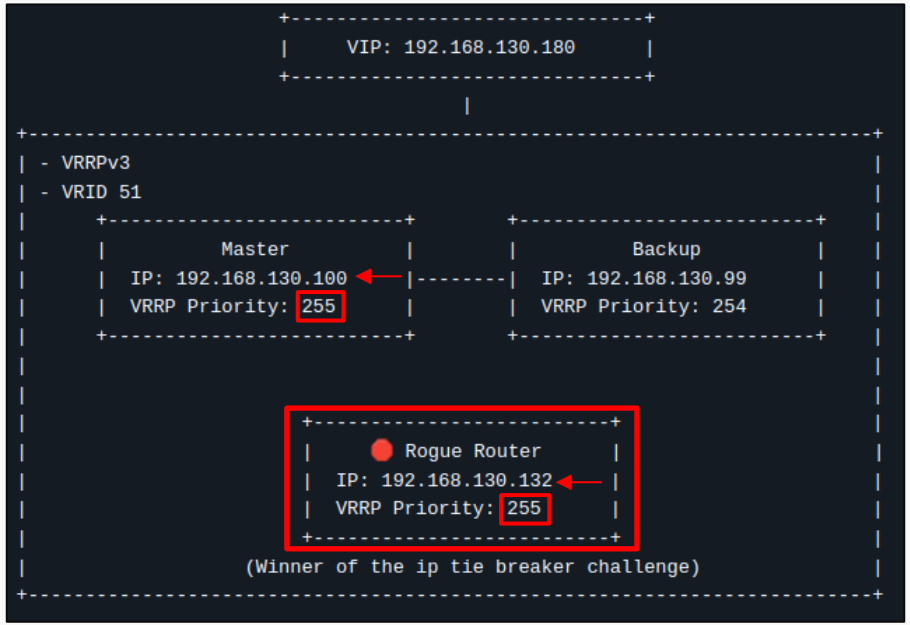
Prerequisites : in the same subnet as the VRRP nodes

VRRP priority conflict dilemma

~> The node with the “highest IP*” address wins the challenge (Rogue)

~> The legitimate master node will also become a backup node

~> Because **the Master’s IP address is lower than that of the rogue router, that’s normal !**



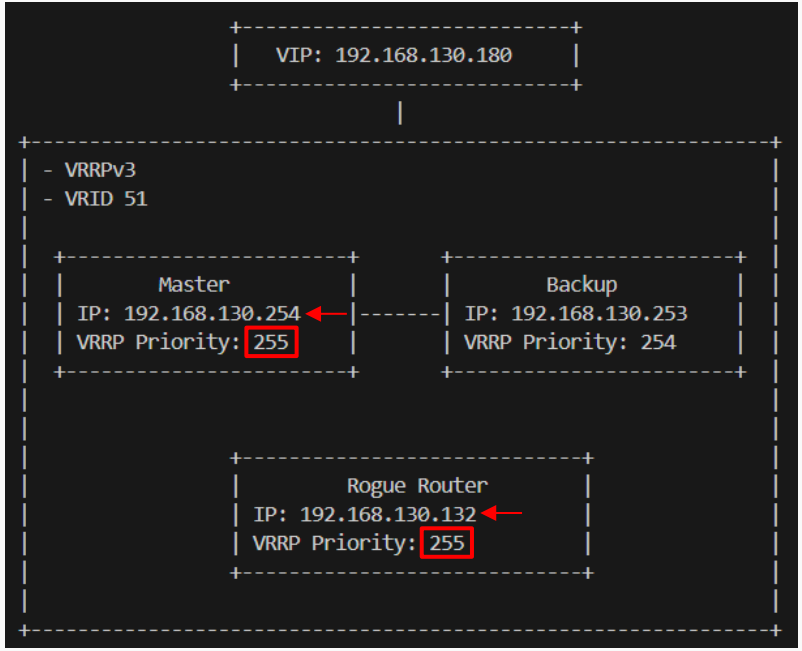
*On the last byte

03

Research Problem

Research problem (Keepalived project)

Is it possible to become master (take over the VIP) ***even if my rogue node has a "lower IP" than the current master with a SOTA* implementation ?***



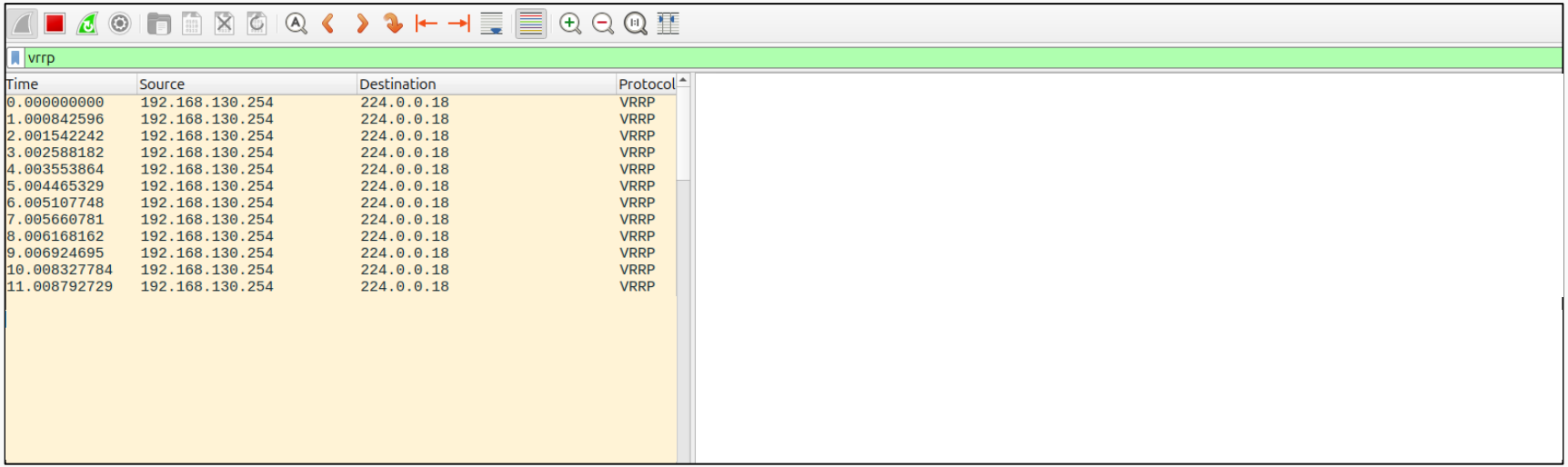
*state-of-the-art

Strange behavior (Keepalived project)

~> In the event of equal VRRP priority (255), a rogue router (192.168.130.132) could take over the master

~> **Even if the rogue router has a lower IP address on the last byte**

~> This led me to conclude that, by default, the ip tie-breaking mechanism did not work



vrrp

Time	Source	Destination	Protocol
0.000000000	192.168.130.254	224.0.0.18	VRRP
1.000842596	192.168.130.254	224.0.0.18	VRRP
2.001542242	192.168.130.254	224.0.0.18	VRRP
3.002588182	192.168.130.254	224.0.0.18	VRRP
4.003553864	192.168.130.254	224.0.0.18	VRRP
5.004465329	192.168.130.254	224.0.0.18	VRRP
6.005107748	192.168.130.254	224.0.0.18	VRRP
7.005660781	192.168.130.254	224.0.0.18	VRRP
8.006168162	192.168.130.254	224.0.0.18	VRRP
9.006924695	192.168.130.254	224.0.0.18	VRRP
10.008327784	192.168.130.254	224.0.0.18	VRRP
11.008792729	192.168.130.254	224.0.0.18	VRRP

Strange behavior (Keepalived project)

~> Stopping the attack shows the master's priority was successfully decreased

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Current filter: vrrp

Time	Source	Destination	Protocol	Info
11.735285573	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
12.735627466	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
13.737730633	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
14.724236196	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
15.692442768	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
16.434528946	192.168.130.132	224.0.0.18	VRRP	Announcement (v3)
16.442997203	192.168.130.254	224.0.0.18	VRRP	Announcement (v3)
16.444568561	192.168.130.254	224.0.0.18	VRRP	Announcement (v3)
17.445590190	192.168.130.254	224.0.0.18	VRRP	Announcement (v3)
18.445806554	192.168.130.254	224.0.0.18	VRRP	Announcement (v3)
19.446225627	192.168.130.254	224.0.0.18	VRRP	Announcement (v3)

```
▸ Frame 15953: 46 bytes on wire (368 bits), 46 bytes captured (368 bits) on interface vmnet8,
▸ Ethernet II, Src: VMware_2c:b5:c7 (00:50:56:2c:b5:c7), Dst: IPv4mcast_12 (01:00:5e:00:00:12)
▸ Internet Protocol Version 4, Src: 192.168.130.254, Dst: 224.0.0.18
▾ Virtual Router Redundancy Protocol
  ▸ Version 3, Packet type 1 (Advertisement)
    Virtual Rtr ID: 51
    Priority: 254 (Non-default backup priority)
    Addr Count: 1
    0000 .... = Reserved: 0
    .... 0000 0110 0100 = Adver Int: 100
    Checksum: 0x68d3 [correct]
    [Checksum Status: Good]
    IP Address: 192.168.130.180
```

~> Race condition ?

Strange behavior (Keepalived project)

```
vagrant@master-vrrp:~$ sudo journalctl -u keepalived.service -f
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: (VI_1) Sending/queueing gratuitous ARPs on ens33 for 192.168.130.180
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: Sending gratuitous ARP on ens33 for 192.168.130.180
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: Sending gratuitous ARP on ens33 for 192.168.130.180
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: Sending gratuitous ARP on ens33 for 192.168.130.180
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: Sending gratuitous ARP on ens33 for 192.168.130.180
Apr 29 10:25:50 master-vrrp Keepalived_vrrp[1021]: Sending gratuitous ARP on ens33 for 192.168.130.180
Apr 29 10:28:12 master-vrrp Keepalived_vrrp[1021]: (VI_1) CONFIGURATION ERROR: local instance and a remote instance are both configured as address owner, please fix - reducing local priority
Apr 29 10:28:12 master-vrrp Keepalived_vrrp[1021]: (VI_1) Master received advert from 192.168.130.132 with higher priority 255, ours 254
Apr 29 10:28:12 master-vrrp Keepalived_vrrp[1021]: (VI_1) Entering BACKUP STATE
Apr 29 10:28:12 master-vrrp Keepalived_vrrp[1021]: (VI_1) removing VIPs.
```

Keepalived backend logs (master)

vrrp.c

keepalived > vrrp > vrrp.c > vrrp_state_master_rx

```c
vrrp_state_master_rx(vrrp_t * vrrp, const vrrphdr_t *hd, const char *buf, ssize_t buflen)

    if (hd->priority == vrrp->effective_priority) {
        if (addr_cmp == 0)
            log_message(LOG_INFO, "(%s) WARNING - equal priority advert received from remote host with our IP address.", vrrp->iname);
        else if (vrrp->effective_priority == VRRP_PRIO_OWNER) {
            /* If we are configured as the address owner (priority == 255), and we receive an advertisement
             * from another system indicating it is also the address owner, then there is a clear conflict.
             * Report a configuration error, and drop our priority as a workaround. */
            log_message(LOG_INFO, "(%s) CONFIGURATION ERROR: local instance and a remote instance are both configured as address owner, please fix - reducing local priority", vrrp->iname);
            vrrp->effective_priority = VRRP_PRIO_OWNER - 1;
            vrrp->base_priority = VRRP_PRIO_OWNER - 1;
        }
    }
```

Strange behavior (Keepalived project)

When does the attack work? (Keepalived project)

	VRRPv2	VRRPv3
Auth Type	No Authentication Simple Text Password *(sniff the network to get the pwd before)* IP AH *(sniff the network and **try to crack the secret before**)*	No Authentication
Diffusion mode	Multicast & Unicast *(hypothetical attack but in practice unfeasible on a real case)*	Multicast & Unicast *(...)*

CVE ?

~> Lab-tested with keepalived and cisco implementation

~> Only keepalived seemed vulnerable, not cisco

~> **First conclusion:** keepalived implements ip-tie breaking incorrectly, so it's CVE.

~> Keepalived is making a patch, but they are not convinced that they are the problem

~> **Is this a CVE on keepalived or the RFC 9568 (latest) that keepalived follows ?**

acassen / **keepalived**

Code | Issues 25 | Pull requests 4 | Discussions | Actions | Security | Insights

Commit 8419149

pqarmitage committed on Feb 17 Verified

```
vrrp: Restore priority 255 if duplicate address owner detected

The VRRP RFCs assume that only one device is configured as the address
owned for any VRID.

keepalived has extended functionality which detects if two (or more)
systems are configured as the address owner (this is completely
invalid configuration). To avoid multiple systems acting as address
owner, and hence all of them remaining in master mode, keepalived will
reduce an address owner's priority to 254 if the other device configured
as address owner does not go away.

This commit restores the priority of a vrrp instance to 255 if it had
reduced it to 254 to avoid multiple VRRP instances simultaneously
advertising that they are the address owner.

Signed-off-by: Quentin Armitage <quentin@armitage.org.uk>
```

```
Internet Engineering Task Force (IETF)                        A. Lindem
Request for Comments: 9568                        LabN Consulting, L.L.C.
Obsoletes: 5798                                                A. Dogra
Category: Standards Track                                  Cisco Systems
Published: April 2024
ISSN: 2070-1721


  Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6

Abstract

   This document defines version 3 of the Virtual Router Redundancy
   Protocol (VRRP) for IPv4 and IPv6. It obsoletes RFC 5798, which
   previously specified VRRP (version 3). RFC 5798 obsoleted RFC 3768,
   which specified VRRP (version 2) for IPv4. VRRP specifies an election
   protocol that dynamically assigns responsibility for a Virtual Router
```


YOU GET AN EMAIL!
2009
AND YOU GET AN EMAIL!
AND YOU GET AN EMAIL!
EVERYONE GETS AN EMAIL!

RFC 9568 / Analyse

~> After several tests with Keepalived, we agreed the issue likely stemmed from the RFC

~> RFC 9568 requires the master to *"**discard**"* all received VRRP packets with a priority of 255

~> The packets was "***dropped before processing***", preventing IP tie-breaking → Bug

~> The priority of the initial master was therefore decreased in favor of the rogue node

RFC 9568 / Erratum 8298

~> A request to modify the RFC was therefore made by Quentin, maintainer of keepalived

~> Erratum 8298 allow node with priority 255 to process received VRRP packets normally

Errata ID: 8298
Status: Verified
Type: Technical
Publication Format(s) : TEXT, PDF, HTML
Reported By: Quentin Armitage
Date Reported: 2025-02-17
Verifier Name: Jim Guichard
Date Verified: 2025-03-06

Section 7.1 says:

```
   It MUST verify that the VRID is configured on the receiving
   interface and the local router is not the IPvX address owner
   (Priority = 255 (decimal)).

If any one of the above checks fails, the receiver MUST discard the
packet, SHOULD log the event (subject to rate-limiting), and MAY
indicate via network management that an error occurred.
```

RFC 9568 / Consequences of the Erratum

Tie-breaking now possible:

~> IP-based tie-breaking can now apply even for nodes with priority 255.

Impact on implementations:

~> Developers must update VRRP implementations to reflect this change.

~> Incompatibility with RFC 5798 (*The Cisco VRRP I tested wasn't vulnerable as it followed this old RFC.*)

Incompatibility with RFC 5798

Conflicting rules: ⚠️

~> RFC 5798 mandates ignoring lower-priority advert, while RFC 9568 requires responding with an advert

Result:

~> It is no longer possible to be compliant with both RFC 9568 and RFC 5798 at the same time

~> Keepalived decided to follow the latest RFC (9568 with the erratum)

04

Take aways

Key facts & Advices

~> It wasn't an CVE on keepalived, but a problem in the RFC itself imho

~> Misinterpretation of RFC 9568 led to incorrect behavior in some implementations

~> In any case, a hardened configuration is essential for VRRP

~> Hardened configuration is essential:

- Explicit priority settings (255 for the master 254,253… for the backup(s))
- Strict IP addressing and network segmentation
- Prefer unicast mode over multicast

Resources

~> Article in MISC magazine (No. 140): *The Security of the VRRP Protocol (Sept/Oct 2025)*

~> Anonymized Study on the Security of the VRRP Protocol (20 online articles/tutorials)

~> Keepalived Project

~> RFC 9568 – 5798

Thanks to

~> Claire Vacherot (@non_curat_lex), Théo Lorette-Froidevaux (@tolfsh), Laurent Levron

~> Keepalived team (keepalived.org)

~> Orange Cyberdefense (@OrangeCyberFR)

~> Pass the SALT

~> Family and closes friends

When Priority Isn’t Enough:

Exploiting the VRRP Tie-Breaking IP Mechanism

Pass the SALT

Q&A

geoffrey.sauvageotberland@orangecyberdefense.com

linkedin.com/in/geoffrey-sb

https://urlr.me/UY7Dnm