

# Kapitel 12

## Hochverfügbarkeit

Firewalls fallen manchmal aus. Und dann erfüllen sie ihre fundamentalste Aufgabe nicht mehr, die darin besteht, Netzwerke zu beschützen.

Und Firewalls fallen genauso gerne aus wie andere elektronische Geräte. Das ist eine akzeptierte Tatsache und aus diesem Grund haben High-End-Firewalls zusätzliche Netzteile, Lüfter, CPUs oder Uplinks. In den unteren Preissegmenten hilft man sich meist damit, dass mehrere Firewalls als Gruppe (Cluster) zum Einsatz kommen. Dann entsteht ein Cluster für Hochverfügbarkeit und Ausfallschutz.

Innerhalb der Gruppe einigen sich die Geräte darauf, dass *eine* Firewall die Arbeit verrichtet und die andere zuschaut und beobachtet. Die Beobachtungen des passiven Geräts sind wichtig, denn dieses übernimmt die Geschäfte, sobald es bemerkt, dass sein Partner hinüber ist.

### Grundlagen

Technisch läuft das in geordneten Bahnen ab, denn alle Firewalls der *Redundanz-Gruppe* müssen sich an den gemeinsamen Standard *Common Address Redundancy Protocol* (CARP) halten.

Sobald CARP auf einer Firewall eingerichtet ist, horcht diese an ihren Netzwerkkinterfaces auf Lebenszeichen anderer CARP-Gateways. Der erste Teilnehmer der Gruppe macht sich selbst zum Master und sendet Lebenszeichen im Sekundentakt ins Netz. Die zweite Firewall derselben Gruppe

empfängt diese Keepalives und bleibt im Backup-Modus: nichts tun und warten.

### **Hinweis**

Für die Funktionsweise von CARP ist es irrelevant, ob die Teilnehmer aus Firewalls, Routern, Servern oder sonstigen Gateways bestehen. Daher werden die Bezeichnungen in diesem Kapitel synonym verwendet.

Sobald der Backup-Rechner drei Takte lang nichts von seinem Meister hört, muss er von einer Havarie ausgehen und macht sich selbst zum Master. Dann beginnt die Arbeit, denn er muss alle Aufgaben vom ehemaligen Chef übernehmen. Und das so schnell wie möglich, damit das Tagesgeschäft normal weitergehen kann.

Wer erzählt jetzt den anderen Geräten im Netz, dass eine neue Firewall am Start ist? Niemand, denn diese neue Firewall übernimmt auch die IP- und MAC-Adresse der Redundanzgruppe. Für die anderen Teilnehmer im Netz hat sich (außer einer kurzen Unterbrechung) nichts verändert.

Die Lebenszeichen, Heartbeats oder Keepalives, sind IPv4-Pakete an die Multicast-Adresse 224.0.0.18. In diesen Paketen steht die virtuelle IPv4-Adresse, die sich alle CARP-Router teilen. Außerdem hat jede Redundanzgruppe eine eigene Gruppennummer, damit mehrere CARP-Gruppen im selben Netzsegment aktiv sein können.

Die Beschreibung und Funktionsweise erinnert an das Redundanzprotokoll *Virtual Router Redundancy Protocol* (VRRP). Die Ähnlichkeit besteht darin, dass CARP in weiten Teilen eine Kopie von VRRP ist. Die Lizenzpolitik vom Betriebssystem BSD macht die Nutzung von VRRP unmöglich, also haben die BSD-Entwickler eine funktionsähnliche Variante unter dem Namen CARP hervorgebracht.

## **Labor**

Das Demo-Lab stellt drei Firewalls, von denen zwei (RT-1 und RT-2) zusammen ein CARP-Cluster bilden. Abbildung 12.1 zeigt den Aufbau als Netzdiagramm.

Alle Teilnehmer von Standort-1 nutzen als Standardgateway weder die IPv4-Adresse von RT-1, noch die von RT-2, sondern die *zusätzliche* Adresse 10.1.1.5, die der CARP-Gruppe gehört. Das ist die LAN-Seite der Geräte – auf der WAN-Seite bilden die Firewalls neben ihren bekannten IP-Adressen ebenfalls eine zusätzliche CARP-Adresse. Damit der Ausfallschutz funktioniert, muss das CARP-Cluster aus beiden Richtungen über die virtuellen Adressen angesprochen werden.

Auf der WAN-Seite kommuniziert das CARP-Pärchen mit Firewall RT-core, welche zum Ziel der Verbindungsanfragen von Standort-1 wird.

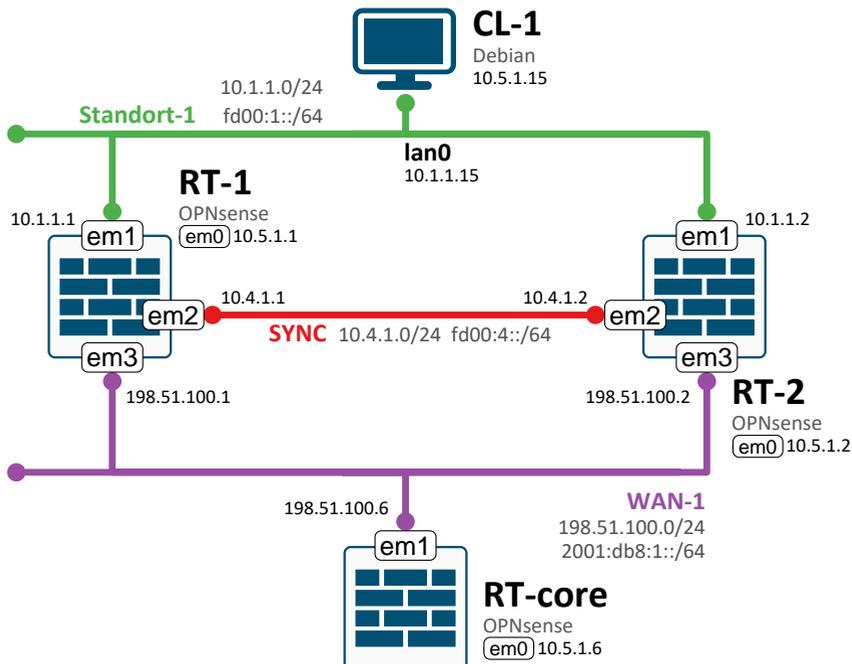


Abbildung 12.1: Laboraufbau mit Hochverfügbarkeit

Damit der Netzverkehr zwischen den Geräten problemlos fließen kann, sind die Regelwerke der beteiligten Firewalls sehr offen gestaltet und erlauben alles. Sobald alle Aspekte der Hochverfügbarkeit funktionieren, sollten die Regeln an den Schutzbedarf der Umgebung angepasst werden.

Das Netzsegment SYNC wird erst später im Kapitel benötigt und dort erklärt.

## CARP-Gruppe

OPNsense wird im Bereich *Schnittstellen* → *Virtuelle IPs* → *Einstellungen* zum CARP-Router. Die Einrichtung über den *Plus*-Button zum Hinzufügen erfordert eine eindeutige Gruppennummer und eine virtuelle Adresse mit entsprechender Netzmaske. Tabelle 12.1 listet die Einstellungen für die primäre Firewall RT-1.

Eigenschaft	LAN	WAN
Modus	CARP	CARP
Schnittstelle	LAN	WAN1
Network / Address	10.1.1.5/24	198.51.100.12/24
Passwort	<i>beliebig</i>	<i>beliebig</i>
VHID Gruppe	1	2
advbase	1	1

Tabelle 12.1: Einstellungen der virtuellen IPv4-Adressen

Der Wert von *advbase* gibt die Dauer (in Sekunden) zwischen zwei Lebenszeichen an. Damit symbolisiert *advbase* gleichzeitig eine Priorität der aussendenden Firewall, wobei ein niedriger Wert eine höhere Bedeutung hat. Ein *advbase*-Intervall von 1 garantiert dem Sender seine Rolle als Master. Die passive Firewall muss einen größeren Wert haben. Das Passwort schützt die CARP-Gruppe vor ungewollten Mitgliedern und muss auf allen Geräten identisch sein.

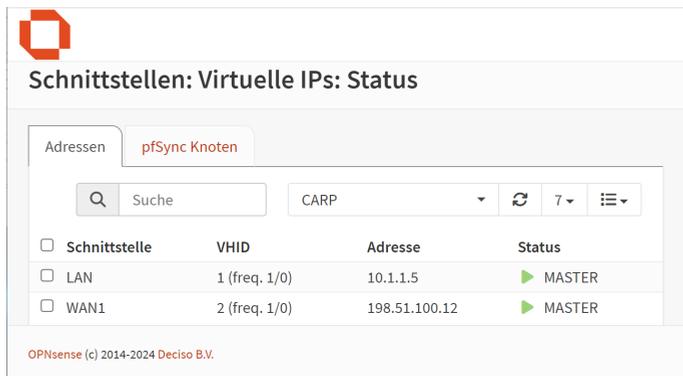


Abbildung 12.2: Firewall RT-1 wird CARP Master

Nach einem anschließenden *Speichern* beginnt RT-1 mit dem Aussenden von Heartbeats auf seiner LAN- und WAN-Schnittstelle. Für RT-2 ist die Vorgehensweise identisch, bis auf den höheren *advbase*-Wert von beispielsweise 3.

Wenige Sekunden danach haben sich die beiden CARP-Kandidaten darauf geeinigt, wer der Chef ist und wer der Assistent. In diesem Beispiel hat RT-1 gewonnen und übernimmt die Masterrolle (Abbildung 12.2).

Bei Firewall RT-2 sieht die Situation ganz ähnlich aus, nur sollte der Status auf BACKUP stehen (Abbildung 12.3).

Falls dort ebenfalls MASTER angegeben ist, sind beide Firewalls in der Masterposition und streiten sich um die virtuelle IP-Adresse. Dieser Zustand darf im normalen Betrieb nicht vorkommen, da es auf der Clientseite meist zu Programmabbrüchen führt.

Beide Teilnehmer werden zum MASTER, wenn sie die Lebenszeichen des anderen *nicht* hören. Die Fehlersuche beginnt bei der Kommunikation der Geräte untereinander mithilfe von *ping* auf die physische IPv4-Adresse.

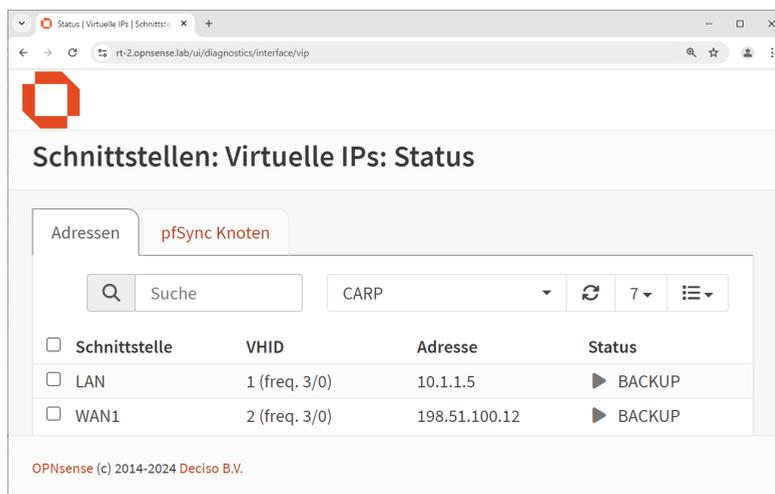


Abbildung 12.3: Firewall RT-2 wird CARP Backup

Sobald sich MASTER und BACKUP geeinigt haben, kann die Ende-zu-Ende-Verbindung von Client CL-1 durch die Firewalls zum Zielhost RT-core mit *traceroute* anschaulich geprüft werden. Denn *traceroute* ermittelt, wel-

chen Weg ein Paket durchs Netz nimmt und zeigt in der folgenden Ausgabe, dass RT-1 für die Weiterleitung zuständig ist.

```
root@cl-1:~# traceroute -I 198.51.100.6
traceroute to 198.51.100.6 (198.51.100.6), 30 hops max [...]
 1  10.1.1.1 (10.1.1.1)  0.931 ms  22.727 ms  *
 2  198.51.100.6 (198.51.100.6)  23.003 ms  43.694 ms  43.712 ms
```

Nachdem dieser Normalzustand herrscht, passiert ein erster simulierter Ausfall. Die Master-Firewall RT-1 erfährt einen plötzlichen Stromausfall oder die virtuelle Maschine wird gestoppt.

Was passiert? RT-2 empfängt keine Lebenszeichen mehr und ernennt sich nach wenigen Sekunden zum Master. Dasselbe traceroute-Kommando auf Client CL-1 zeigt nun den geänderten Pfad durch RT-2 bis zum Ziel.

```
root@cl-1:~# traceroute -I 198.51.100.6
traceroute to 198.51.100.6 (198.51.100.6), 30 hops max [...]
 1  10.1.1.2 (10.1.1.2)  0.670 ms  20.554 ms  *
 2  198.51.100.6 (198.51.100.6)  20.844 ms  22.596 ms  22.652 ms
```

## Zustandslos

Wenn kein Traffic im Netz ist, bemerkt auch kein Client den Ausfall von Firewall RT-1. Aber was passiert bei einem Dateitransfer?

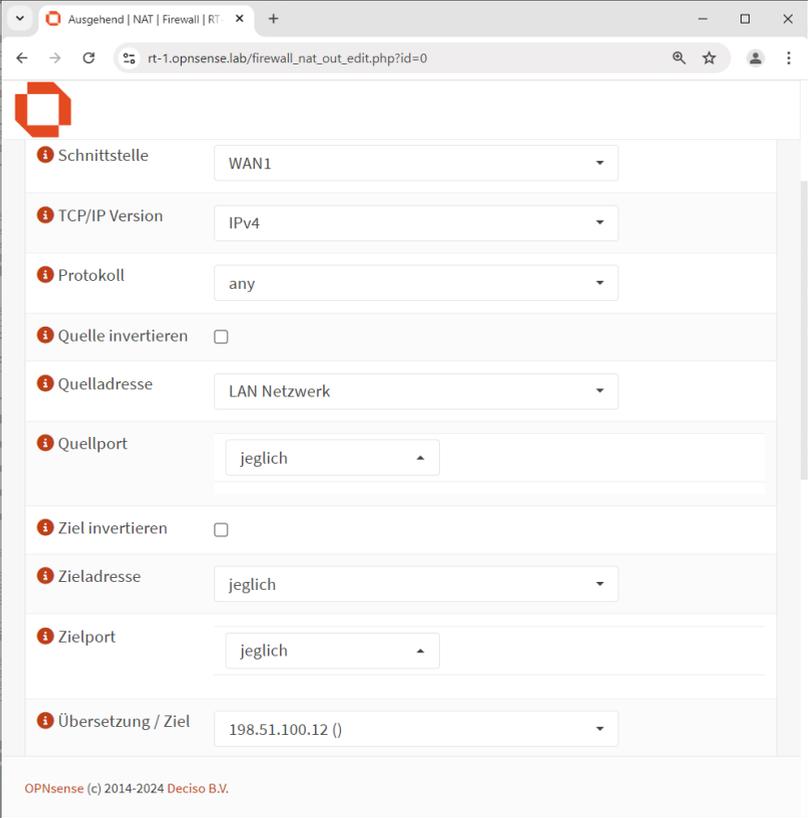
Wie sich ein unterbrochener Transfer verhält, hängt ganz von der Anwendung und den Timeouts ab. Ein beispielhafter Webdownload von CL-1, der auf einen öffentlichen Webserver zugreift, kommt während des Ausfalls ins Stocken, läuft aber nach circa vier Sekunden weiter.

Im Moment ist die Konfiguration der Laborgeräte noch ziemlich weltfremd, da die Firewalls den Traffic vom LAN ungehindert weiterreichen und zustandslos arbeiten. In Unternehmensnetzen gibt es jede Menge Hindernisse, wie Adressumsetzung (NAT) oder strikte Firewallregeln, die sich den Zustand jeder Verbindung merken.

## Adressumsetzung

Ein Router mit Kontakt zum Internet hat normalerweise einen Paketfilter an Bord. Höchstwahrscheinlich ist auch noch eine Adressumsetzung (NAT, vgl. Kap. 8) dabei, um von privaten Adressen in öffentliche zu übersetzen.

Um das Labornetz etwas realitätsnäher zu gestalten, vollführen die CARP-Geräte eine Adressumsetzung vom internen Netz 10.1.1.0/24 in die passende öffentliche IPv4-Adresse. Abbildung 12.4 zeigt die Einrichtung von NAT bei Firewall RT-1.



The screenshot shows the OPNsense web interface for configuring NAT on Firewall RT-1. The browser address bar shows the URL `rt-1.opnsense.lab/firewall_nat_out_edit.php?id=0`. The configuration form includes the following fields:

- Schnittstelle:** WAN1
- TCP/IP Version:** IPv4
- Protokoll:** any
- Quelle invertieren:**
- Quelladresse:** LAN Netzwerk
- Quellport:** jeglich
- Ziel invertieren:**
- Zieladresse:** jeglich
- Zielport:** jeglich
- Übersetzung / Ziel:** 198.51.100.12 ()

At the bottom of the interface, the copyright notice reads: OPNsense (c) 2014-2024 Deciso B.V.

Abbildung 12.4: Firewall RT-1 soll IPv4-Adressen umsetzen

Wichtig ist hierbei, dass der OPNsense-Rechner für die Adressumsetzung *nicht* seine eigene physische IPv4-Adresse am WAN-Adapter verwendet (198.51.100.1), sondern die gemeinsame CARP-Adresse (198.51.100.12). Ansonsten ist die Hochverfügbarkeit nicht mehr gegeben, denn mit dem Ausfall von Firewall RT-1 ist auch die physische Adresse von RT-1 nicht erreichbar. Und genau diese Adresse würde RT-core für die Antwortpakete verwenden.

Also übersetzt RT-1 ausgehende Pakete korrekterweise in die CARP-Adresse, die im Fehlerfall an RT-2 vererbt wird und die Antwortpakete fließen durch RT-2 zurück zum Client.

Jetzt müssen die Firewalls genau Buch führen: Welches (Antwort-)Paket muss der Paketfilter akzeptieren und welche IP mit welchem Port wird wie übersetzt?

Beim Ausfall des CARP-Masters RT-1 wird ein Datentransfer von CL-1 erst stocken und nach dem Schwenk auf die Backup-Firewall abbrechen. Die Ursache liegt in den Firewall- und NAT-Tabellen des Backup-Systems. Denn diese sind leer.

### Zustandstabellen

Eine Zustandstabelle ist grundsätzlich eine feine Sache: Sie listet alle bestehenden Verbindungen auf, die durch die Firewall fließen. Paketfilter und Adressumsetzer schauen für jedes Paket in diese Tabelle, um zu erfahren, ob das Paket zu einer bestehenden Verbindung gehört und weiter behandelt werden darf.

Bei *einer* Firewall ist das eine Verbesserung der Sicherheit. Bei mehreren Firewalls besteht das Problem, dass jedes Gerät seine eigene Tabelle pflegt. Bei CARP hat der Master-Router eine volle Tabelle und die Tabelle des Backup-Routers ist leer, denn er hat noch keine einzige Verbindung gesehen.

### Synchronisation der Tabellen

OPNsense löst das Problem mit den unterschiedlichen Tabelleninhalten durch eine Methode des Betriebssystems. Denn FreeBSD hat seit über zehn Jahren das Protokoll *pfsync* zur Synchronisation von Paketfiltern dabei. *pfsync* ist zwar unabhängig von CARP, aber zusammen sind sie ein gutes Team.

Damit teilt der CARP-Master sein Wissen über die Zustandstabelle mit dem Backup-Router. In kurzen Abständen sendet der Master-Router Änderungen seiner Tabelle an eine frei wählbare IPv4-Adresse oder an eine Multicast-Adresse, sodass der Backup-Router seine lokale Tabelle entsprechend ergänzen kann. Das Ziel ist, dass alle Geräte im CARP-Verbund denselben Inhalt in den Firewall- und NAT-Tabellen haben.

Falls für die Synchronisation ein eigenes Netzsegment zur Verfügung steht, umso besser. Denn der Abgleich zwischen den Teilnehmern muss in Echtzeit passieren. Eine zehn Sekunden alte Firewalltabelle hilft nicht viel bei Verbindungen, die innerhalb der letzten neun Sekunden aufgebaut wurden.

Die Synchronisation beginnt bei *System* → *Hochverfügbarkeit* → *Einstellungen*. Im Bereich *Allgemeine Einstellungen* erwartet die Konfigurationsoberfläche eine Synchronisierungsschnittstelle, die in diesem Laborszenario *SYNC* heißt und das IP-Netz 10.4.1.0/24 belegt. Die *Peer-IP* ist die IPv4-Adresse der Gegenstelle; RT-1 (10.4.1.1) sendet seine Berichte an die IPv4-Adresse 10.4.1.2 und erreicht damit RT-2 (Abbildung 12.5). Bei RT-2 ist es genau umgekehrt.

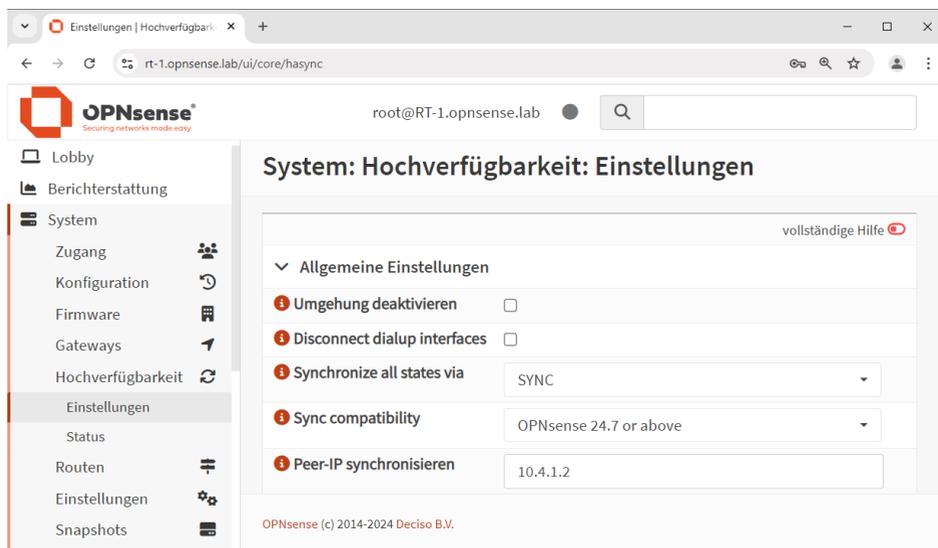


Abbildung 12.5: RT-1 sendet alle Statusänderungen an RT-2

Die *Synchronisierungseinstellungen*, im unteren Bereich der Webseite, senden Konfigurationsänderungen an die Partner-Firewall und werden im Abschnitt *Synchronisation der Konfiguration* beleuchtet.

Der Austausch von Tabelleninhalten passiert über das Interface *em2*, welches eine direkte, aber unabhängige Verbindung zwischen RT-1 und RT-2 darstellt. Bei zwei Firewalls sollte die Zieladresse die IPv4 des Partners sein.

Bei mehreren Firewalls muss das Feld frei bleiben, damit *pfsync* die vordefinierte Multicast-Adresse 224.0.0.240 verwendet und damit alle Teilnehmer im Netzsegment erreicht.

Jetzt lernen die CARP-Enthusiasten gegenseitig ihre Tabelleninhalte. Für CARP würde es ausreichen, wenn nur der Backup-Router vom Master lernt, aber die Synchronisation verläuft in beide Richtungen.

Wenn jetzt wieder ein unerwartetes Ereignis die primäre Firewall RT-1 zur Strecke bringt, übernimmt RT-2 die CARP-Rolle und das Routing der Verbindungen. Der Failover-Prozess dauert ein paar Sekunden, aber dann läuft der Datentransfer von CL-1 weiter, denn er steht bereits *vor* der Havarie in der Sessiontabelle von RT-2.

```
root@RT-2:~ # pfctl -s state | grep 10.1.1.15
all tcp 198.51.100.6:443 <- 10.1.1.15:60616 ESTABLISHED[...]
all tcp 198.51.100.12:39870 (10.1.1.15:60616) -> \
  198.51.100.6:443 ESTABLISHED:ESTABLISHED
```

### Synchronisation der Konfiguration

Neben den Inhalten der Sessiontabelle kann OPNsense auch Konfigurationsänderungen von der primären Firewall auf die Backup-Maschine übertragen. Damit müssen Änderungen an einem Firewall-Cluster nur an *einer* Maschine durchgeführt werden. Das spart Zeit und verhindert Tippfehler.

OPNsense hat dafür kein neues Protokoll erfunden, sondern führt die Konfigurationsbefehle in Echtzeit auf beiden Maschinen aus. Die Kommunikation mit der entfernten Firewall läuft über HTTPS. Aus diesem Grund benötigt die Einrichtung der *Synchronisierungseinstellungen (XMLRPC Sync)* unter *System* → *Hochverfügbarkeit* → *Einstellungen* auch Benutzername, Kennwort und die IP-Adresse der sekundären Maschine. Weiterhin benötigt die empfangende Firewall eine Regel, die den Webzugriff des Partners akzeptiert. Wenn die Konfigurationsoberfläche zusätzlich durch die Maßnahmen in Abschnitt *Verwaltungszugang anbinden* aus Kapitel 9 auf Seite 108 gesichert ist, muss hier auch die jeweilige Schnittstelle hinzugefügt werden.

## Best Practice

Bei der redundanten Auslegung von Firewalls gibt es verschiedene Methoden, welche den Ablauf harmonisch gestalten und die Verfügbarkeit verbessern.

### Asymmetrisches Routing

Wenn ein Paket auf dem Hinweg zum Server einen anderen Pfad nimmt als auf dem Rückweg, ist das Routing asymmetrisch. Theoretisch ist das kein Problem, aber in der Praxis verhindern zustandsorientierte Firewalls, NAT-Gateways oder IDS-Systeme eine erfolgreiche Verbindung.

Mit CARP passiert sehr leicht ein asymmetrisches Routing. Diese Asymmetrie entsteht sogar im Labornetz, wenn RT-1 Master für die LAN-Seite ist und RT-2 Master für die WAN-Seite ist.

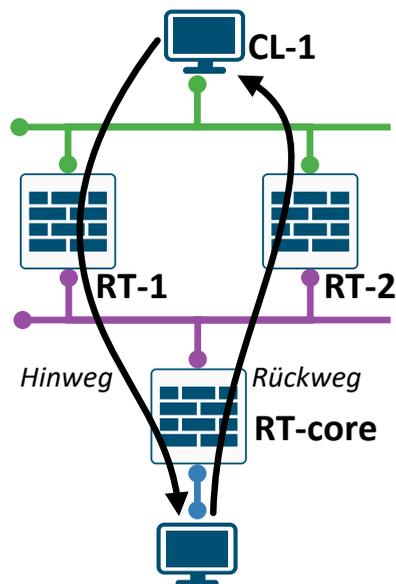


Abbildung 12.6: CARP kann asymmetrisches Routing hervorrufen

In Abbildung 12.6 sendet Client CL-1 Netzpakete an sein Default-Gateway, welche von RT-1 angenommen werden. Über RT-core gelangt das Paket an

sein Ziel. Der Weg zurück beginnt bei RT-core. Dieser Router sendet weiter an die CARP-Adresse und wird von RT-2 beantwortet. RT-2 weiß von dieser Verbindung nichts, weil er das erste Paket gar nicht gesehen hat, welches über RT-1 geroutet wurde. Wenn RT-2 als „dummer“ Router agiert, leitet er die Pakete weiter zu CL-1 und alles ist gut. Falls RT-2 aber *stateful* arbeitet, wird er alle unbekanntenen Pakete verwerfen. Dann verhindert asymmetrisches Routing die erfolgreiche Kommunikation von CL-1.

OPNsense kann den CARP-Prozessen eine Priorität mitgeben, sodass *eine* Firewall für alle CARP-Gruppen Master wird. Damit ist und bleibt das Routing symmetrisch. Die Einrichtung von Prioritäten geschieht bei CARP über die Advertising-Frequenz, die im Abschnitt *CARP-Gruppe* auf Seite 164 beschrieben ist.

### Wahl zum Master

Grundsätzlich gewinnt der CARP-Router mit der höchsten Advertising-Frequenz. Diese Frequenz errechnet sich aus den Werten für *advbase* (Basis) und *advskew* (Zeitversatz). Je höher die Frequenz, desto häufiger wird ein Keepalive ins Netz gesendet. Das Intervall zwischen zwei Lebenszeichen berechnet CARP mit Formel 12.1. Also: Je größer die Zahlen, umso niedriger die Advertising-Frequenz und desto unwahrscheinlicher wird die Firewall zum Master.

$$\text{Intervall} = \text{Basis} (\text{advbase}) + \frac{\text{Zeitversatz} (\text{advskew})}{256} \quad (12.1)$$

Wenn beide Kandidaten die voreingestellten Werte von Basis=1 und Zeitversatz=0 haben, gewinnt der Router mit der größeren IP-Adresse. In diesem Fall wird RT-2 der Master, weil seine IPv4 10.1.1.2 numerisch größer ist, als die von seinem Gegenkandidaten RT-1 mit 10.1.1.1. Auf der WAN-Seite ist das genauso.

Wenn RT-1 die bevorzugte Firewall sein soll, weil beispielsweise die Hardware leistungsstärker oder neuer ist, muss RT-1 mit einer besseren Advertising-Frequenz punkten. Für ein schnelles Failover bleibt der Basiswert auf beiden Firewalls bei einer Sekunde und der Zeitversatz bei RT-2 findet einen hohen Wert von beispielsweise 100.

**Hinweis**

Der Wert von *advskew* wird sichtbar im erweiterten Modus einer virtuellen IP-Adresse bei *Schnittstellen* → *Virtuelle IPs* → *Einstellungen*.

Mit dieser Manipulation hat RT-2 die schlechteren Karten bei der Wahl. Als Folge schwenkt die Masterrolle von RT-2 zu RT-1.

**Synchronisation**

Bei den unterschiedlichen Techniken für die Synchronisation ist die Richtung entscheidend, um neue Inhalte nicht mit alten Werten zu überschreiben. Die Synchronisation von Tabellen arbeitet bevorzugt bidirektional, also sollte bei *Peer-IP synchronisieren* der Statussynchronisation stets die IP-Adresse der Partnerfirewall eingetragen sein. Damit sind sogar Verbindungen abgesichert, die aus Versehen über die Backup-Firewall laufen.

Die Synchronisation der Konfiguration ist ein unidirektionales Geschäft: Die primäre Firewall gibt die Vorgaben an die sekundäre Firewall – nicht umgekehrt. Das Feld *Synchronisiere Konfiguration zur IP* ist auf RT-1 mit der IP-Adresse von RT-2 gefüllt. Auf RT-2 bleibt dieses Feld leer. Damit verbleiben versehentliche Änderungen auf RT-2 lokal und gefährden nicht die Einstellungen des Partners.

**Schnelleres Failover**

Andere Redundanzprotokolle für Gateway-Failover erreichen Umschaltzeiten von unter einer Sekunde. Das Intervall für die Keepalives liegt dann im Bereich von wenigen Hundert Millisekunden mit einem Timeout von einer knappen Sekunde.

Dieser Luxus ist bei OPNsense nicht möglich. Die vorgegebene Dauer zwischen zwei Lebenszeichen-Paketen ist gleichzeitig der Minimalwert: eine Sekunde. Höhere Werte lassen sich konfigurieren, aber der Timeout ist stets die dreifache Dauer. Per Voreinstellung sind das etwa 3–4 Sekunden.

Ein flotteres Umschalten ist mit CARP zwar machbar, aber die Implementierung unter FreeBSD legt als Minimum eine Sekunde fest. Failover im Millisekundenbereich leistet CARP nur bei OpenBSD, welches nicht als Plattform für OPNsense auserwählt wurde.

## Lastverteilung

Bei CARP ist immer nur *eine* Firewall der aktive Master. Eine Verteilung der Netzlast auf mehrere Geräte ist im Protokoll mit Tricks möglich.

Für eine „Lastverteilung des kleinen Mannes“ (Abbildung 12.7) bekommen die Firewalls eine weitere CARP-Gruppe pro Interface. In dieser neuen Gruppe ist genau die Firewall Master, die in der ersten Gruppe Backup ist. Dem obigen Beispiel folgend ist RT-1 Master und RT-2 Backup der CARP-Gruppe 1. In der neuen CARP-Gruppe 2 ist RT-1 Backup und RT-2 der Master. Während Gruppe 1 die IPv4-Adresse 10.1.1.5 bedient, könnte Gruppe zwei zur Adresse 10.1.1.6 gehören. Der Trick besteht darin, dass die Hälfte der Clients in diesem Netzsegment ihr Standardgateway auf 10.1.1.5 stellen und die andere Hälfte 10.1.1.6 als Gateway nutzen.

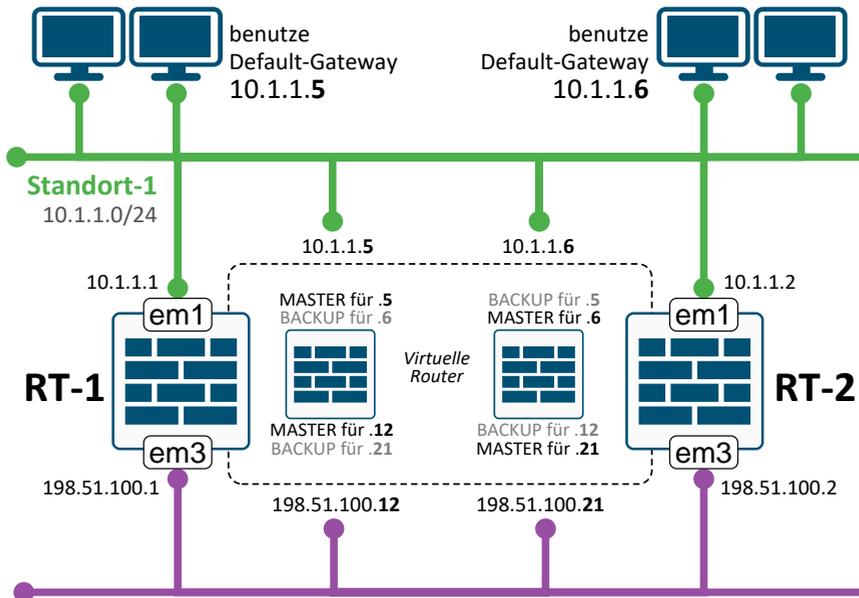


Abbildung 12.7: Lastverteilung mit CARP

Ob eine Firewall Master oder Backup wird, hängt an ihrer Advertising-Frequenz, die über die Werte *advbase* und *advskew* voreingestellt werden. Die Ausgangswerte sind 1 und 0. Die Einstellungen in Tabelle 12.2 machen RT-1 zum Master für Gruppe 1 und zum Backup von Gruppe 2. Genau

andershherum wird RT-2 der Backup-Router für Gruppe 1 und der Master von Gruppe 2. Das Wahlergebnis von RT-1 ist in Abbildung 12.8 dargestellt.

Firewall	Interface	VHID-Gruppe	adv-base	adv-skew	Rolle	IPv4-Adresse
RT-1	LAN	1	1	0	Master	10.1.1.5
RT-1	LAN	2	1	100	Backup	10.1.1.6
RT-1	WAN	7	1	0	Master	198.51.100.12
RT-1	WAN	8	1	100	Backup	198.51.100.21
RT-2	LAN	1	1	100	Backup	10.1.1.5
RT-2	LAN	2	1	0	Master	10.1.1.6
RT-2	WAN	7	1	100	Backup	198.51.100.12
RT-2	WAN	8	1	0	Master	198.51.100.21

Tabelle 12.2: Advertising-Frequenz für eine Lastverteilung mit CARP

Die genauen Zahlen für den Zeitversatz sind nicht entscheidend. Hauptsache die eine Firewall hat einen höheren Wert als die andere.

Damit teilen sich beide Firewalls die Netzlast. Der Anteil jedes Geräts ist nicht kontrollierbar: Im besten Fall arbeitet jedes Gateway genau 50 % der Pakete ab, im ungünstigsten Fall erhält RT-1 über 99 % aller Verbindungen und RT-2 langweilt sich mit dem verbleibenden Prozent.

### Schnittstellen: Virtuelle IPs: Status

<input type="checkbox"/>	Schnittstelle	VHID	Adresse	Status
<input type="checkbox"/>	LAN	1 (freq. 1/0)	10.1.1.5	▶ MASTER
<input type="checkbox"/>	LAN	2 (freq. 1/100)	10.1.1.6	▶ BACKUP
<input type="checkbox"/>	WAN1	7 (freq. 1/0)	198.51.100.12	▶ MASTER
<input type="checkbox"/>	WAN1	8 (freq. 1/100)	198.51.100.21	▶ BACKUP

Abbildung 12.8: Firewalls RT-1 und RT-2 teilen sich die Arbeit

### IP Version 6

CARP ist durchgängig bereit für IPv6. Aber das ist nur die halbe Miete, denn die Synchronisation der Tabellen hat bei der neueren IP-Version leichte Schwierigkeiten. Das *pfsync*-Protokoll ist für IPv6 ausgelegt und die Web-Oberfläche akzeptiert eine IPv6-Adresse bei *Peer-IP synchronisieren*. Unter der Haube nutzt *pfsync* jedoch IPv4 für die Kommunikation.

Wenn die Synchronisation tatsächlich sein eigenes Netzsegment belegt, dann fällt die IPv4-Verbindung für den Austausch im restlichen IPv6-Netz nicht weiter auf. Wer im *pfsync*-Segment eine IPv6-Verbindung benutzen möchte, kann dies mit dem Befehl `ifconfig` auf der Kommandozeile erreichen:

```
ifconfig pfsync0 syncpeer fd00:4::2
```

Glücklicherweise ist die Synchronisation der Konfiguration (XMLRPC Sync) weniger stur und zeigt sich vollständig „IPv6-ready“.

### Technischer Hintergrund

OPNsense verrät in der Web-Oberfläche zum Konfigurieren der Firewall bereits viel über die eingesetzten Protokolle CARP, *pfsync* und XMLRPC.

CARP erstellt für den ausgewählten Netzadapter eine zusätzliche IP-Adresse, die als *Virtuelle IP* konfiguriert wurde. Zu dieser frei wählbaren IP-Adresse gehört die feste MAC-Adresse `00:00:5e:00:01:NN`, wobei *NN* die Gruppennummer ist. Mit diesem Trick sind in einem Netzsegment mehrere CARP-Gruppen möglich, die sich gegenseitig nicht stören.

Die Lebenszeichen des CARP-Masters sind schlanke Pakete an die Multicast-Adresse `224.0.0.18` oder `FF02::12` bei IPv6. Damit haben alle Teilnehmer im selben Netz ungefragt die Chance, die Multicast-Pakete zu empfangen.

Auf der Kommandozeile sind die Einstellungen und das Wahlergebnis mit dem bekannten Kommando `ifconfig` sichtbar und konfigurierbar. Der LAN-Adapter von RT-2 berichtet über seine CARP-Erfahrungen:

```
root@RT-2:~ # ifconfig em1
em1: flags=1008943<UP,BROADCAST,RUNNING,PROMISC,SIMPLEX, [...]
```

```

description: LAN (opt1)
options=48500b8<VLAN_MTU,VLAN_HWTAGGING,JUMBO_MTU, [...]
ether 00:15:16:02:01:02
inet 10.1.1.2 netmask 0xffffffff broadcast 10.1.1.255
inet 10.1.1.5 netmask 0xffffffff broadcast 10.1.1.255 vhid 1
inet 10.1.1.6 netmask 0xffffffff broadcast 10.1.1.255 vhid 2
inet6 fe80::215:16ff:fe02:102%em1 prefixlen 64 scopeid 0x2
inet6 fd00:1::2 prefixlen 64
inet6 fd00:1::5 prefixlen 64 vhid 3
carp: BACKUP vhid 1 advbase 3 advskew 0
      peer 224.0.0.18 peer6 ff02::12
carp: MASTER vhid 2 advbase 1 advskew 0
      peer 224.0.0.18 peer6 ff02::12
carp: BACKUP vhid 3 advbase 3 advskew 0
      peer 224.0.0.18 peer6 ff02::12
media: Ethernet autoselect (1000baseT <full-duplex>)
status: active
nd6 options=121<PERFORMNUD,AUTO_LINKLOCAL,NO_DAD>

```

Die Synchronisation von pfsync gehört irgendwie zu CARP dazu. Daher ist die Bedienung ähnlich und auch hier ist ifconfig das Werkzeug der Wahl. Anders als CARP wird pfsync nicht an einen Netzadapter angehängt, sondern ist ein eigener Adapter mit dem treffenden Namen *pfsync0*. Die Einstellungen lassen sich also mit dem folgenden Kommando auf RT-2 auslesen:

```

root@RT-2:~ # ifconfig pfsync0
pfsync0: flags=1000041<UP,RUNNING,LOWER_UP> metric 0 mtu 1500
      options=0
      syncdev: em2 syncpeer: 10.4.1.1 maxupd: 128 defer: off version: 1400
      syncok: 1
      groups: pfsync

```

Für beide Methoden zur Hochverfügbarkeit ist kein Prozess nötig, der im Hintergrund schnurrt, denn die Implementierung läuft im Kernel ab.

## Zusammenfassung

Die Hochverfügbarkeit von Firewalls mit CARP und pfsync ist eine stabile und einfache Möglichkeit, die Ausfallzeit von Systemen gering zu halten. Der Zauber liegt in einem Firewallpäarchen: *Zwei* identische Geräte bilden

die Firewall, wobei Gerät-1 die Arbeit erledigt und Gerät-2 übernimmt, sobald sein Kollege schlappmacht.

Der Konfigurationsaufwand ist nur minimal höher, denn mit XMLRPC-Sync gleichen die Teilnehmer der Firewallgruppe sogar ihre Konfiguration selbstständig ab.