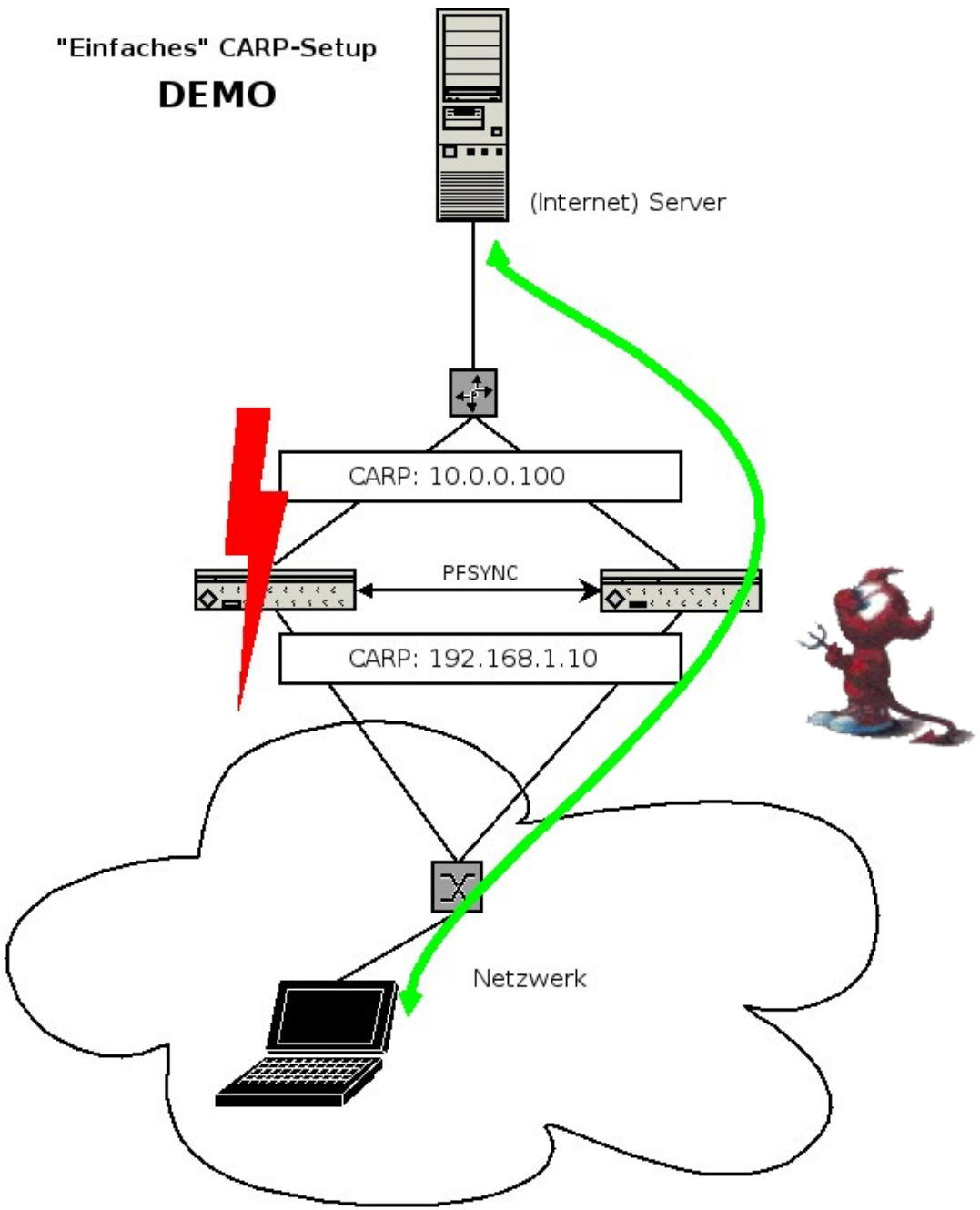
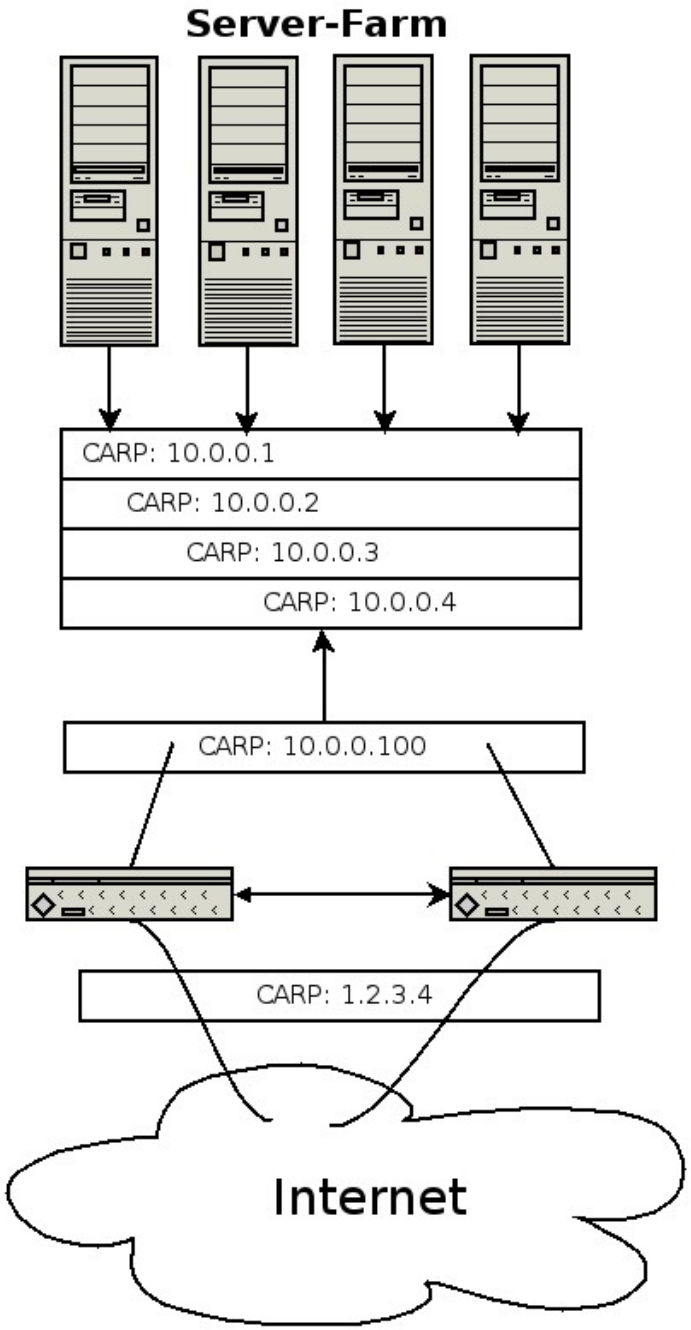
Redundanz unter FreeBSD mit CARP und PF





Redundanz unter FreeBSD mit CARP und PF

pf

Das OpenBSD Firewall Paket „packet filter“ (**pf**) ist seit der Version 5.3 Bestandteil von FreeBSD.

Es wurde Ende 2001 in OpenBSD als Ersatz für das in die Jahre gekommene IPFilter Paket vorgestellt und wird seither aktiv weiterentwickelt. **pf** steht mittlerweile in allen BSDs zur Verfügung und erfreut sich höchster Beliebtheit.

Eine Reihe von Funktionen machen **pf** zur derzeit wohl umfangreichsten Firewall-Software. Darunter viele Features, die sonst nur (wenn überhaupt) mit kostspieliger, kommerzieller Hardware zur Verfügung stehen.

Im Zusammenspiel mit **pfsync** und **CARP** eröffnet **pf** absolut einmalige Fähigkeiten ein Netzwerk zu schützen. Mit vergleichsweise wenig Aufwand kann ein Unternehmensnetzwerk vollkommen redundant und verbindungsorientiert an das Internet angeschlossen werden.

Besondere Fähigkeiten hat **pf** auch im Umgang mit NAT-Installationen und im Rahmen von IPv6. Durch eine einfach zu verstehende, mächtige Ruleset Syntax ist **pf** leicht zu erlernen und verwalten. Mit wenigen Zeilen lassen sich komplexe Zusammenhänge ausdrücken und selbst Lastbalancierung oder das Verwalten von großen IP-Listen (SPAM) ist kinderleicht.

pfsync

Bei **pf** handelt es sich um eine „stateful“-Firewall. Das heißt für jede Verbindung (TCP und alle anderen auf IP aufsetzenden Protokolle) führt **pf** Statusinformationen in einem sog. „state-entry“ mit. Dort sind alle Eigenschaften der Verbindung gespeichert und die Firewall kann mit Hilfe dieser Informationen eine Reihe von Angriffen bereits im Keim ersticken. Insbesondere Clients mit einer „schwachen“ TCP-Stack Implementierung können so vor „**I**nitial **S**equence **N**umber“ - Attacken geschützt werden.

Will man redundante Firewalls aufsetzen, so müssen diese Informationen allen Knoten im Verbund zur Verfügung stehen, da sonst die Verbindung beim Ausfall einer Komponente nicht aufrecht erhalten werden könnte. Für diese Aufgabe wurde das **pfsync** Protokoll entwickelt, das zum Austausch der Verbindungsstatusinformationen dient.

Da **pf** auch NAT-Informationen in „state-entries“ ablegt ist es sogar möglich Clients hinter einer NAT-Firewall (also Clients ohne öffentliche IP) ohne Zurücksetzen der offenen Verbindungen redundant anzuschließen.

CARP

Das **C**ommon **A**ddress **R**edundancy **P**rotocol wurde 2004 als freier Ersatz für das von Cisco patentierte VRRP entwickelt und ist momentan in Open- und FreeBSD verfügber. Es bietet im Prinzip die gleiche Funktionalität, hat aber einige zusätzliche Funktionen, die beim Entwurf von VRRP vergessen wurden. So bietet **CARP** beispielsweise volle Unterstützung für IPv6 und ist durch Verschlüsselung gegen eine Reihe von Angriffen gegen VRRP geschützt.

Mit **CARP** ist es (einfach ausgedrückt) möglich eine Reihe von Rechnern hinter der gleichen IP-Adresse zu verstecken. Fällt einer der Rechner aus, so übernimmt ein anderer weitere Anfragen. Zusätzlich ist es möglich die Last über alle vorhandenen Knoten zu verteilen und auch beim Ausfall eines oder mehrerer Teilnehmer die verbleibenden Knoten (siehe Abb. umseitig) zu übernehmen.

Die Konfiguration von **CARP** ist überraschend einfach. **CARP** stellt sich dem Benutzer als virtuelles Interface dar und ist somit so einfach zu verwalten wie ein VLAN-Adapter.