Protecting your Family

with pfSense and IPSec

As the family tech guy, they started to call me more and more for removing viruses and botnet infections. I fixed this by installing pfSense boxes as their primary Internet gateway. This gave them a more secure Internet connection and me some remote pfSense boxes to play with. All firewalls are connected via IPsec tunnels for remote management of the firewall and remote management of their home networks.

What you will learn…

- pfBlocker
- IPsec connections in pfSense

What you should know…

- Basic networking
- How to install pfSense
- Your way around its GUI
- (See BSD Magazine 2011/02)

I really love the ALIX embedded boards and pfSense has a special NanoBSD build for them. The board is a 2D13 model with a Geode 800 LX 500 MHz processor, 256MB RAM, on-board crypto accelerator, three ethernet ports and a Compact Flash socket. A 1GB Compact Flash card provides more than enough space for pfSense with some additional packages. The complete firewall will only draw about 7 Watt of power (Figure 1).

pfSense

From the website: "pfSense is a free, open source customized distribution of FreeBSD tailored for use as a firewall and router. In addition to being a powerful, flexible firewalling and routing platform, it includes a long list of related features and a package system allowing further expandability without adding bloat and potential security vulnerabilities to the base distribution."

pfSense is a fork of the M0n0wall project and the interface looks similar in many, many places. I switched from M0n0wall to pfSense for one feature: fail-over capabilities. Both M0n0wall and pfSense are actively maintained distributions and my setup could also be created with M0n0wall in the same way.

The version I am using is 2.1-BETA0, available at the snapshot server of the pfSense project.

PfSense comes with a pretty decent working default installation. Vr0 (left network socket) is defined as LAN interface with 192.168.1.1 as default address and a running DHCP server. Vr1 (middle network socket) is defined as WAN interface and will be listening for DHCP servers offering it an IP address –This Paragraph sounds irrelevant because the article doesn't talk about hardware.

For my setup, I changed the LAN IP addresses and DHCP server to avoid conflicting IP ranges in different

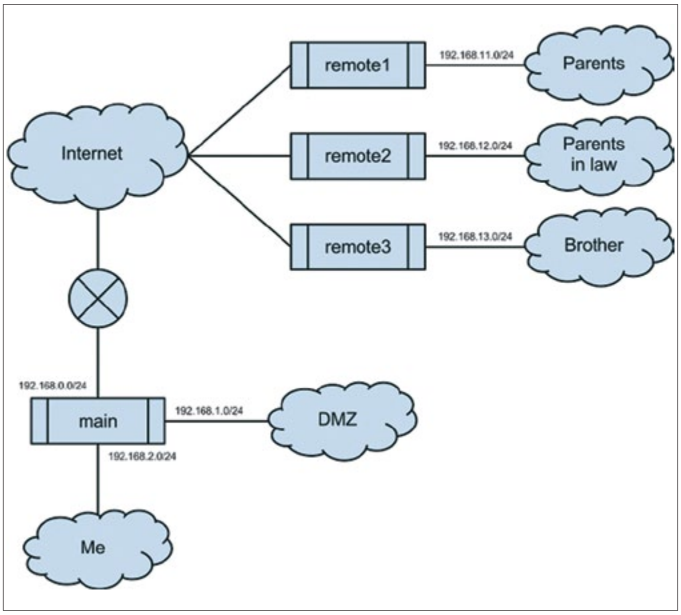

Figure 1. *A scheme of the network*

networks. This can be done through the serial console of through the web GUI. Please take note that the dhcp server will be disabled if you change the LAN IP through the web GUI. You need to – assign yourself a static IP address in the new IP range, in order to reconnect to the firewall's LAN address.

pfBlocker

I choose the pfBlocker package with IP lists from Emerging Threats and ZeuS tracker to actively block known compromised IP addresses, both from entering the network or leaving the network.

If my family becomes infected with a botnet (eg. by browsing to a site with a Java exploit), their infected computer cannot access the Command&Control server on the Internet and cannot be used or misused by the botnet controller.

pfBlocker is a new 2.x package that merged the older 1.x Countryblock and IPblocklist packages. It creates an alias and a firewall rule to block traffic to and/or from the IP addresses in the alias. It is installed under System > Packages by clicking the + next to the pfBlocker package.

After installation, pfBlocker can be configured under Firewall > pfBlocker.

I do not use the features to block known spammer IP's, as my family does not run SMTP servers and use the POP/IMAP/SMTP servers provided by their ISP, instead aFirewall rules drop all SMTP traffic arriving at the firewall.

The IP lists I configured are publicly available from Emerging Threats and iBlocklist. After enabling pfBlocker on the General tab, lists can be configured on the List tab. Click on the + to create a new list.

```
Alias name    ET
List description Emerging Threats
Lists  txt + http://rules.emergingthreats.net/fwrules/
                     emerging-Block-IPs.txt
       txt + http://rules.emergingthreats.net/blockrules/
                     compromised-ips.txt
List action      Deny both
Update freq      Every 12 hours
```

This will create a new list called "ET" with two sources of IP addresses. You can look into iblocklist.com for more publicly available lists. I recommend blocking DROP (*Don't Route On Peer*), ZeuS and DShield as minimum lists. iblocklist.com will offer p2p-style lists without subscription. pfBlocker can read these if they are added as txt lists. If you would like .gz or native .txt lists, consider taking a $9.99 yearly subscription.

Finally, pfBlocker has a widget for the dashboard to show the status and hits for each configured list.

IPsec for Remote Management

pfSense is managed through the GUI (via http(s)) or the console (ssh). Both protocols are secure and can be used for management over the Internet.

To decrease the number of interesting open ports from port scans (and for fun), I choose to perform management over a vpn connection. I used OpenVPN at first, but at crucial times (like family calling for support) it dropped the connection and would not reconnect. Therefore I decided to give IPsec a try. pfSense uses IPsec-Tools, a port of KAME's libipsec, setkey, and racoon. These tools are fully integrated in the GUI and have proven to be rock-solid.

IPsec terms

Internet Protocol Security (IPsec) is a protocol suite for securing *Internet Protocol* (IP) communications by authenticating and encrypting each IP packet. It operates at the IP layer (OSI layer 3). Some important terms are:

Security Associations (SA)

A SA is a one-way encrypted tunnel. For bi-directional traffic (like a TCP connection), we need two tunnels, one for each direction. The tunnel is created between the public IP addresses of the tunnel endpoints.

Listing 1. *Phase 1 main site*

```
Interface    WAN    (on which interface should pfSense establish IPsec connections)
Remote Gateway  <IP address> (public IP address or fqdn of remote site)
Auth. Method    Mutual PSK    (I will be using pre-shared symmetric keys)
My Identifier   Distinguished Name + “Main”                (unique name of main site)
Peer Identifier     Distinguished Name + “Remote1” (unique name of remote site)
Pre-shared Key  <the key> (mine is 256 bits in hex)
Policy Generation   Default    (the default)
Encryption Algorithm   AES-128    (AES-128 is offloaded to the on-board crypto accelerator)
Hash Algorithm  SHA-1
DH Key Group    2 (1024 bits)
NAT Traversal   Enable    (the main site is behind a NAT router)
```

Listing 2. *Phase 1 remote site*

```
Most settings are the same, but in reverse. So:
Remote Gateway  <IP address>     (public IP address or fqdn of main site)
My Identifier   Distinguished Name + “Remote1”    (unique name of remote site)
Peer Identifier     Distinguished Name + “Main”    (unique name of main site)
Policy Generation   Unique         (I found this to work reliable with Default at the main site)
All other settings like key and algorithm must be identical to phase 1 on the main site.
```

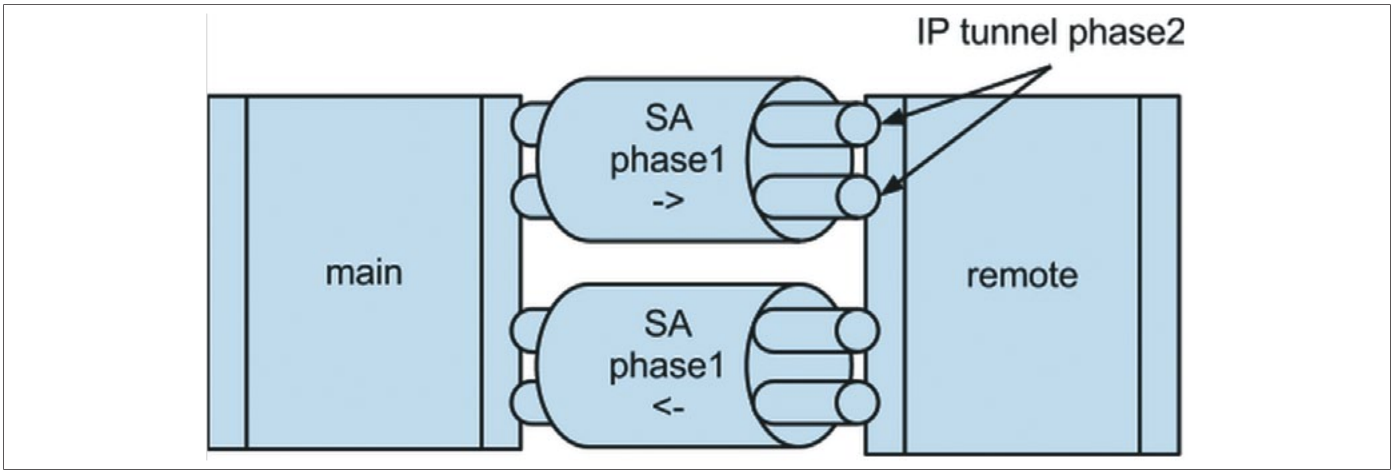

Figure 2. *IP tunnel phase2*

Security Policy (SP)

The SP will determine the specifications of the IPsec tunnel, like encryption algorithms and lifetime.

Authentication Header (AH)

The AH provides integrity and data origin authentication for IP datagrams by calculating a hash value of the header of the packet.

Encapsulation Security Payload (ESP)

ESP provides mainly confidentiality, integrity and data-origin authentication by encrypting the payload of the IP packet.

Internet Security Association and Key Management Protocol (ISAKMP)

The ISAKMP provides a framework for authentication and key exchange, with authenticated keying material provided often by pre-shared keys or Internet Key Exchange (IKE and IKEv2).

Setup in pfSense

IPsec connections are negotiated in two phases. In phase 1, a SA is created using the ISAKMP.

Upon successful completion, both endpoints have authenticated each other, exchanged keys and can communicate securely. In phase 2, the actual tunnel is created for transferring data between hosts and routed networks. This tunnel is created based on the SP for the specific tunnel (Figure 2).

I will only list the relevant fields that will create the connection between the main site and remote1.

Remote2 and Remote3 are done in the same way, but with their own IP addresses and their own preshared keys. IPsec connections are defined under VPN > IPsec in the GUI. First, check the box and “Save” to enable IPsec. This will start the necessary tools. Then click the + to add connections. (Listings 1-2)

Now that phase 1 has been setup, we can decide what traffic we want to send over it. Click on “+ Show 0 Phase-2

Listing 3. *Phase 2 main site*

```
Mode          Tunnel IPv4     (tunnel mode will encapsulate the original IP packet)
Local network    192.168.2.0/24   (IP range of the main site)
Remote network   192.168.11.0/24  (IP range of the remote site)
Protocol      ESP       (Encapsulate Security Payload, encrypt and authenticate packet)
Encryption Algorithm   AES-128
Hash algorithm   SHA-1
PFS group     2 (1024 bit)     (Perfect Forward Secrecy)
```

Listing 4. *Phase 2 remote site*

```
Most settings are the same, but in reverse. So:
Local network    192.168.11.0/24  (IP range of the remote site)
Remote network   192.168.2.0/24   (IP range of the main site)
All other settings like algorithm and PFS group must be identical to phase 2 on the main site.
```

Listing 5. *Routing firewall traffic through the IPsec tunnel*

```
Interface        LAN (local traffic originates from this interface before being routed)
Address family      IPv4
Name             GW_IPsec
Gateway          192.168.11.254   (the IP address of the LAN interface)
Default Gateway     No         (regular Internet traffic is sent to the Internet)
Disable Gateway MonitoringYes          (no need to ping the local LAN interface, its up...)

Now the gateway is defined, we can use it in a static route. Go back to System > Routing in the GUI.
On the Routing tab, click the + button to add a new static route
Destination Network    192.168.2.0/24   (the IP range of the main site)
Gateway          GW_IPsec
```

entries" to show any phase 2 entries and click on its + to add the phase-2 entry. (Listings 3-4)

If you want to route more networks over the same IPsec connection, you will have to define them as separate phase 2 definitions under the existing phase 1. My DMZ for example has its own phase 2 definition on the main site and each remote site with the IP ranges of the DMZ, 192.168.1.0/24 instead of the 192.168.2.0/24.

Remember to Clic on "Apply Changes" at the top of the Page in order for them to take effect.

Routing Firewall

Because of the way IPsec is implemented in the BSD kernel, it is not possible to route traffic that originates from the firewall directly into the IPsec tunnel. It simply does not know where to go. This can be fixed by adding a static route. Traffic from the remote network arriving at the LAN interface has no problems and will be routed directly through the IPsec tunnel.

First add a new gateway. This is done under System > Routing in the GUI.

On the Gateways tab, click the + button and add a new gateway (Listing 5).

Now the gateway is defined, we can use it in a static route. Go back to System > Routing in the GUI.

On the Routing tab, click the + button to add a new static route

```
Destination Network 192.168.2.0/24   (the IP range of the
                    main site)
Gateway             GW_IPsec
```

Now pfSense knows where the local packets destined for the main site should be delivered to (LAN interface) and how they should be routed (through the IPsec tunnel).

NAT Ports

If your box is is behind a firewall, you should open two ports for IPsec traffic.

```
500 udp ISAKMP
4500   udp IPsec NAT Traversal
```

My main site is behind NAT, so I added these ports to its NAT gateway.

Firewall Rules

PfSense creates hidden firewall rules on the incoming interface for udp ports 500 and 4500 to allow incoming SA's.

PfSense creates one new firewall interface for all IPsec connections. In order to allow traffic to pass through the tunnel, you will have to add relevant firewall rules to this new interface. Allowing ssh and https from the main network 192.168.2.0/24 to the remote firewall 192.168.11.254 is a start.

Result

Connected to my network 192.168.2.0/24, I can browse to the address for a remote firewall (eg. *https://192.168.11.254*), a IPsec tunnel will be created over the Internet and the traffic is routed through the remote firewall and back as if they were connected to my own network. Nice.

Where do we go from here?

In future Articles i will show you how to send all logfiles from the remote firewalls to a Splunk server that is located in the DMZ. For this reason, I added new phase 2 entries to all remote sites, so the remote sites can route traffic to this DMZ server via the existing SA.

Managing three remote firewalls is doable, but more will become a hassle. pfSense requires very little maintenance, but doing the same task more than three times should be automated. One option is to script those changes. This will also allow to automate defenses. If one firewall detects a port scan, the others should add the offending IP in a drop list for 12 hours.

Special Thanks

- pfSense for getting me interested in NanoBSD and for providing all my firewall needs: *http://www.pfsense.org*, snapshots can be found at *http://snapshots.pfsense.org*
- Emerging Threats for their research and blocklists: *http://www.emergingthreats.net*
- iBlocklist for their aggregation of blocklists: *http://www.iblocklist.com*
- PC-Engines for their nifty ALIX boards: *http://www.pcengines.ch/*
- Varia Store for distributing ALIX boards with useful cases: *http://www.varia-store.de*

ERWIN KOOI

Erwin Kooi is an information security manager for a large grid operator. He started with FreeBSD 4.5 and is an avid fan ever since.