

Re: ipfw plus authentication (authpf is cool but....)

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/questions/2006-04/msg00407.html>

- *From:* Nikos Vassiliadis <nvass@xxxxxxxxxxxxxxxx>
 - *Date:* Wed, 5 Apr 2006 11:00:42 +0300
-

On Wednesday 05 April 2006 04:22, Mark Jayson Alvarez wrote:

Hi Nikos

Nikos Vassiliadis <nvass@xxxxxxxxxxxxxxxx> wrote: On Monday 03 April 2006

10:34, Mark Jayson Alvarez wrote:

Hi

I am looking for ways to manage our LAN by having each user register their ipaddress, mac address, workstation os, etc. in our ldap directory. Now in our pcrouter, the users will first send his login credentials to the pcrouter, and then the pcrouter will check against ldap if this login is correct, and if it is, then it will now do an ldapsearch/compare operation to see if the source address (ip/mac) of the user trying to gain network access is indeed belongs to that user. Only then, the ipfw ruleset will be changed to allow traffic originating from this source address...

<snip>

Does it have to be LDAP and ipfw?
there is authpf which..

Ofcourse this does not cover the IP|MAC address checking you mentioned, but I don't see how this enhances security. It will be easy for a user to change his IP|MAC address.

</snip>

Our main problem is that in our company, each user has his own workstation(no one else uses it).. However, due to poor implementation of ip allocation strategy, any user can change his ip to whatever ip address he wants, thus it would be hard for us to really monitor who is doing this and who is doing that (because it would be useless to see the ip address of the one who's eating up or bandwidth or doing p2p when we cannot determine who is this user this ip belongs to. This leads us to our decision to have every user assigned a static ip address and have him register his mac address, all stored in ldap directory, and have him authenticate to the pc

Re: ipfw plus authentication (authpf is cool but....)

router first before being allowed to access any server.

All these are not problems with the solution I suggest below

Authpf is somewhat close to this idea but perhaps it was designed for environment wherein users have no permanent workstation, or user can come from any location, even outside the company(at home)....

I have created a draft of my proposed solution:

First, user will authenticate to a web based login form which is tied up against the ip[f|fw|tables] ruleset.

When the user submits the form, the cgi will then verify if the user is really who he claims to be by doing an ldapbind using the credentials provided. Also, the script will check if the request is coming from an ip address that is assigned to that user, by comparing it to his ldap attributes (somewhat prevents users from using other user's ip address).

If everything goes well, the script will happily change the router's firewall ruleset to allow the user to pass thru. (note that in our setup, we have allocated a single class C ip block for all the staffs(120) (no need to have separate blocks since all policies applies to all). Also, we have placed all the servers (mail, proxy, file, printer, im etc) in a different block to make sure that authentication will happen first before a user is allowed to access any of those servers.

Next, we will also provide a logout form(the same as logging out from ssh session in authpf) so that the ruleset can be reverted back when the user does not want to access any network server anymore. The problem with this is that users may be too lazy to logout to the network authentication.. In authpf, even the user did not logout from his ssh session, when he turns off his computer, the ssh session will automatically be terminated. I'm thinking perhaps I can have a nagios server constantly monitoring each user's network connectivity and then changing the firewall ruleset once the user's machine is unreachable...

Another problem I am thinking is that, when a user has already authenticated to the router and have his ip address verified and has been allowed in the firewall, another smart user might immediately change his ip/mac address to that of the authenticated user, and thus making it hard to track his network activity again.. I'm still going to investigate if arpwatch can fill this need....

What do you think???

Re: ipfw plus authentication (authpf is cool but....)

I think all these can be addressed with mpd & RADIUS... read bellow

If it isn't too much trouble you can use PPPoE and/or PPTP with mpd & RADIUS.

You'll then have:

- 1) username/password authorization
- 2) dynamic or static IP address assignment from predefined ranges
- 3) accounting per username
- 4) traffic control per IP address(ipfw dummynet)*
- 5) other interesting RADIUS or PPP features. For example:
 - a) idle-timeout(the user is not using the network and will be logged off).
 - b) session-timeout(the user will have a forced log-out after, let's say, 10 hours).
 - c) session control. You will know if the connection is "up", no matter the traffic.

*) actually very few ipfw rules. Two rules per bandwidth category (384/128, 512/256 etc). I used ipfw tables for that. mpd can also do per-user ipfw-rules if you want that

Something that might interest you, is that FreeRadius can use LDAP backend.

PPPOE clients exist in every Unix-like OpenSourceSoft OS

PPTP is the windows native VPN

I would choose PPPoE, it's lightweight compared to PPTP.

You mentioned that the staff is on its own LAN segment, right??

I don't know if DHCP can cover your needs, in an all-super-user enviroment. I think this can. But its complicated.

I have configuration I can share if you want. Infact I was planning to write a howto...

HTH, Nikos

Re: ipfw plus authentication (authpf is cool but....)

Re: ipfw plus authentication (authpf is cool but....)

Anyone have gone with this solution before??

Thanks

Blab-away for as little as 1¢/min. Make PC-to-Phone Calls using Yahoo! Messenger with Voice.

freebsd-questions@xxxxxxxxxxx mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-questions>
To unsubscribe, send any mail to
"freebsd-questions-unsubscribe@xxxxxxxxxxx"

freebsd-questions@xxxxxxxxxxx mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-questions>
To unsubscribe, send any mail to
"freebsd-questions-unsubscribe@xxxxxxxxxxx"

New Yahoo! Messenger with Voice. Call regular phones from your PC and save big.

freebsd-questions@xxxxxxxxxxx mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-questions>
To unsubscribe, send any mail to
"freebsd-questions-unsubscribe@xxxxxxxxxxx"

freebsd-questions@xxxxxxxxxxx mailing list
<http://lists.freebsd.org/mailman/listinfo/freebsd-questions>
To unsubscribe, send any mail to "freebsd-questions-unsubscribe@xxxxxxxxxxx"