

pfSense - Bug #511

pkg_generate_rules

04/15/2010 09:56 AM - Warren Baker

Status:	Resolved	Start date:	04/15/2010
Priority:	Normal	Due date:	
Assignee:	Ermal Luçi	% Done:	0%
Category:	Package System	Estimated time:	0.00 hour
Target version:	2.0	Affected Architecture:	
Affected Version:	2.0		

Description

Hi Ermal

Found a bug with the way the discover_pkg_rules() function handles \$pkg_generate_rules functions within each package that is installed.

If you have squid and HAVP installed, only HAVP's generate_rules function is executed, leaving any other packages rules not been applied.

I found by moving 'return \$rules' to outside the foreach fixes the problem and allows for the discover_pkg_rules function to move onto the next package in the foreach loop.

Also having 'return ""' for a package that has incorrect PF syntax penalizes the other packages from been setup with PF correctly - which i don't think is correct or the intended behaviour(?).

Example of the change is below:

```
function discover_pkg_rules($ruletype) {
    global $config, $g;
    if(!is_dir("/usr/local/pkg"))
        return "";
    $files = split("\n", trim(`ls /usr/local/pkg/*.inc`));
    foreach($files as $pkg_inc) {
        if($pkg_inc == "ls: No match.")
            continue;
        update_filter_reload_status("Checking for {$ruletype} PF hooks in package {$pkg_inc}");
        require_once($pkg_inc);
        $pkg = basename($pkg_inc, ".inc");
        $pkg_generate_rules = "{$pkg}_generate_rules";
        if(function_exists($pkg_generate_rules)) {
            update_filter_reload_status("Processing early {$ruletype} rules for package {$pkg_inc}");
            log_error("Processing early {$ruletype} rules for package {$pkg_inc}");
            $rules .= $pkg_generate_rules("$ruletype");
            file_put_contents("${$g['tmp_path']}/rules.packages", $rules);
            $status = mwexec("/sbin/pfctl -nf {$g['tmp_path']}/rules.packages");
            if ($status <> 0) {
                $errrules = "There was an error while parsing the package filter rules for {$pkg_inc}.\n";
                log_error($errrules);
                file_put_contents("${$g['tmp_path']}/rules.packages.{$pkg_inc}", "#{$errrules}\n");
            }
            return "";
        }
    }
    return $rules;
}
```

Associated revisions

Revision 09bbbebb - 04/16/2010 03:38 PM - Ermal Luçi

Ticket #511. Actually return the rules from all the packages not just the first one.

Revision 7a31647e - 05/05/2010 04:56 PM - Ermal Luçi

Ticket #511. Do not penalize other packages if rules of one package are erroneous during package rule generation.

History

#1 - 04/16/2010 03:59 PM - Ermal Luçi

- Status changed from New to Feedback

#2 - 05/02/2010 02:24 AM - Chris Buechler

- Affected Version set to 2.0

Warren: can you confirm if this is now fixed for the scenario you could replicate?

#3 - 05/03/2010 10:25 AM - Warren Baker

Hey Chris

Chris Buechler wrote:

Warren: can you confirm if this is now fixed for the scenario you could replicate?

Yeah it is perfect. Although I have one recommendation, in the current setup if one package has errors in it's filter rules then it causes the rest of the packages, who have filter rules, to be penalized. See below for the 'return ""';. Should that return statement simply be removed and allow the rest of the packages to continue to load? The package error is been logged.

```
if ($status <> 0) {
    $errorrules = "There was an error while parsing the package filter rules for {
$pkg_inc}.\n";
    log_error($errorrules);
    file_put_contents("${g['tmp_path']}/rules.packages.${$pkg_inc}", "#{$errorrules
}\n");
    return "";
}
```

Thanks,
Warren

#4 - 05/03/2010 10:30 AM - Warren Baker

Hey Chris

Chris Buechler wrote:

Warren: can you confirm if this is now fixed for the scenario you could replicate?

Yeah it is perfect. Although I have one recommendation, in the current setup if one package has errors in it's filter rules then it causes the rest of the packages, who have filter rules, to be penalized. See below for the 'return "";'. Should that return statement simply be removed and allow the rest of the packages to continue to load? The package error is been logged so a user would be notified by the system logs.

```
if ($status <> 0) {
    $errorrules = "There was an error while parsing the package filter rules for {
$pkg_inc}.\n";
    log_error($errorrules);
    file_put_contents("${g['tmp_path']}/rules.packages.{pkg_inc}", "#{errorrules
}\n");
    return "";
}
```

Thanks,
Warren

#5 - 05/06/2010 07:56 AM - Ermal Luçi

I merged fixes yesterday to not penalize the other packages.

#6 - 05/06/2010 11:21 AM - Warren Baker

Ermal Luçi wrote:

I merged fixes yesterday to not penalize the other packages.

Thanks Ermal - that is perfect.

.warren

#7 - 05/06/2010 04:08 PM - Chris Buechler

- *Status changed from Feedback to Resolved*