
PacketFence

PacketFence Upgrade Guide

PacketFence v12.1.0

Version 12.1.0 - February 2023

Table of Contents

1. About this Guide . . . 2
   1.1. Other sources of information . . . 2
2. General Upgrade Tips . . . 3
   2.1. Prerequisites . . . 3
   2.2. Database and configurations backup . . . 3
   2.3. Disable monit alerts (only if monit is installed) . . . 3
3. Type of upgrades . . . 4
4. Apply maintenance patches . . . 5
   4.1. Important note for cluster environments . . . 5
   4.2. Disable monit alerts (only if monit is installed) . . . 5
   4.3. Stop all PacketFence services . . . 5
   4.4. Upgrade packages . . . 5
   4.5. New versions of config files . . . 6
   4.6. Rebooting after services have been stopped (optional) . . . 7
   4.7. Restart PacketFence services . . . 7
5. Upgrade to another version (major or minor) . . . 8
   5.1. For a standalone server . . . 8
   5.2. For a cluster . . . 8
6. Upgrading from a version prior to 8.0.0 . . . 9
   6.1. Realms upgrade . . . 9
   6.2. Fingerbank v2 . . . 9
   6.3. Changes to the default switch roles . . . 10
   6.4. Removal of the graphite database . . . 10
   6.5. Changes to DNS filters . . . 10
   6.6. Database schema update (all Linux distributions) . . . 11
7. Upgrading from a version prior to 8.1.0 . . . 12
   7.1. Changes on unreg_on_accounting_stop parameter . . . 12
   7.2. Database schema update (all Linux distributions) . . . 12
8. Upgrading from a version prior to 8.2.0 . . . 13
   8.1. Queue Stats maintenance job removal . . . 13
   8.2. Upgrade pfdetect Perl regex to the go RE2 regex . . . 13
   8.3. Upgrade realm.conf to be tenant aware . . . 13
   8.4. The api_user table has been deprecated . . . 13
   8.5. Upgrade pf user privileges . . . 13
   8.6. Update connection_type from WIRED_MAC_AUTH to Ethernet-NoEAP . . . 14
   8.7. Database schema . . . 14
9. Upgrading from a version prior to 8.3.0 . . . 15
   9.1. Upgrade pf.conf to rename configuration parameters . . . 15
   9.2. Upgrade authentication.conf to add searchattributes parameter . . . 15
   9.3. Adjustment to the encoding of the configuration files and templates . . . 15
   9.4. Database schema . . . 15
10. Upgrading from a version prior to 9.0.0 . . . 17
   10.1. Support for Debian 8 dropped . . . 17
   10.2. Necessity to use MariaDB . . . 17
   10.3. Deprecate the classic dhcp filters . . . 17
   10.4. Violations have been renamed to Security Events . . . 17

10.5. Removed MAC detection setting . . . 17
10.6. Modifications to accounting cleanup . . . 18
10.7. Admin roles configuration . . . 18
10.8. Database schema . . . 18
11. Upgrading from a version prior to 9.1.0 . . . 19
11.1. Now possible to disable a domain . . . 19
11.2. pfperl-api port . . . 19
11.3. Linkedin OAuth2 . . . 19
11.4. VLAN pool configuration . . . 19
11.5. Remove Useragent Triggers . . . 19
11.6. Self service portal . . . 20
11.7. Password of the day rotation . . . 20
11.8. Database schema . . . 20
12. Upgrading from a version prior to 9.2.0 . . . 21
12.1. Merge of all RPM packages into one (RHEL / CentOS only) . . . 21
12.2. New GPG key for Debian installations (Debian only) . . . 21
12.3. Database schema . . . 22
13. Upgrading from a version prior to 9.3.0 . . . 23
13.1. Execute script action doesn't use sudo anymore . . . 23
13.2. Database schema . . . 23
14. Upgrading from a version prior to 10.0.0 . . . 24
14.1. Kernel development package . . . 24
14.2. Timezone . . . 24
14.3. Tracking configuration service enabled by default . . . 24
14.4. New PacketFence PKI in Golang . . . 25
14.5. New MariaDB Galera recovery service . . . 25
14.6. Removal of currently-at file and configurator display . . . 25
14.7. Database Privileges . . . 25
14.8. Filter Engine . . . 26
14.9. httpd.admin daemon disabled by default . . . 26
14.10. Database schema . . . 26
15. Upgrading from a version prior to 10.1.0 . . . 27
15.1. RADIUS attributes in authentication sources . . . 27
15.2. Changes in RADIUS configuration for better LDAP support . . . 27
15.3. RADIUS filter templates . . . 27
15.4. New EAP configuration parameter in realm.conf file . . . 27
15.5. Status of rules . . . 27
15.6. Support for CoA in Unifi controllers . . . 28
15.7. Database schema . . . 28
16. Upgrading from a version prior to 10.2.0 . . . 29
16.1. Backup of pfmon.conf (Debian-based systems only) . . . 29
16.2. Self registration portal . . . 29
16.3. Switch type needs to be defined . . . 29
16.4. Convert the pfmon configuration file to pfcron . . . 29
16.5. Rename PFMON* actions to PFCRON* . . . 30
16.6. Syslog parsers are now tenant aware . . . 30
16.7. Database schema . . . 30
17. Upgrading from a version prior to 10.3.0 . . . 31
17.1. MariaDB Upgrade to 10.2 . . . 31
17.2. Rename win_agent_download_uri → windows_agent_download_uri . . . 42
17.3. LDAP port per server has been deprecated . . . 42
17.4. Removal of `inline_accounting` table . . . 42
17.5. `pfdhcplistener` is now tenant aware . . . 42

17.6. Regenerate domain(s) configuration . . . 42
17.7. Database schema . . . 42
17.8. Restart of `packetfence-mariadb` service (standalone installations **only**) . . . 43
18. Upgrading from a version prior to 11.0.0 . . . 44
18.1. Export (on current installation) . . . 44
18.2. Import (on new installation) . . . 44
18.3. Instructions for upgrades without import . . . 44
18.4. NTLM cache background job deprecated in Active Directory Domains . . . 45
18.5. pf-maint.pl script deprecated . . . 45
18.6. WMI scan engine deprecated . . . 45
18.7. TLS 1.0 and 1.1 are disabled by default in FreeRADIUS . . . 45
19. Upgrading from a version prior to 11.1.0 . . . 46
19.1. Automation of upgrades for standalone servers . . . 46
19.2. Support of custom rules in iptables.conf . . . 46
19.3. Support of local authentication for 802.1X in web admin . . . 46
19.4. Support of Monit configuration in pf.conf . . . 46
19.5. Note for cluster upgrades . . . 46
20. Upgrading from a version prior to 11.2.0 . . . 48
20.1. Automation of upgrades for standalone servers . . . 48
20.2. Note for cluster upgrades . . . 48
20.3. Change of behavior for filter engines not_equals operator . . . 48
20.4. Notification on certificates expiration in pfpki . . . 48
21. Upgrading from a version prior to 12.0.0 . . . 50
21.1. Tenant code deprecated . . . 50
21.2. Clusters now use ProxySQL to load balance the DB connections . . . 50
21.3. Bandwidth accounting is now disabled by default. . . . 50
21.4. Fix permissions and checkups deprecated . . . 50
21.5. Change of behavior for the RADIUS source NAS-IP-Address . . . 50
21.6. Log files names updated . . . 51
21.7. Remote database backups . . . 51
22. Upgrading from a version prior to 12.1.0 . . . 52
22.1. configreload deprecated on pfcmd service pf restart . . . 52
23. Archived upgrade notes . . . 53
23.1. Upgrading from a version prior to 4.0.0 . . . 53
23.2. Upgrading from a version prior to 4.0.1 . . . 54
23.3. Upgrading from a version prior to 4.0.3 . . . 54
23.4. Upgrading from a version prior to 4.0.4 . . . 54
23.5. Upgrading from a version prior to 4.0.5 . . . 54
23.6. Upgrading from a version prior to 4.0.6 . . . 55
23.7. Upgrading from a version prior to 4.1.0 . . . 55
23.8. Upgrading from a version prior to 4.2.0 . . . 55
23.9. Upgrading from a version prior to 4.3.0 . . . 56
23.10. Upgrading from a version prior to 4.4.0 . . . 57
23.11. Upgrading from a version prior to 4.5.0 . . . 57
23.12. Upgrading from a version prior to 4.6.0 . . . 57
23.13. Upgrading from a version prior to 4.7.0 . . . 58
23.14. Upgrading from a version prior to 5.0.0 . . . 58
23.15. Upgrading from a version prior to 5.1.0 . . . 60
23.16. Upgrading from a version prior to 5.2.0 . . . 61
23.17. Upgrading from a version prior to 5.3.0 . . . 61
23.18. Upgrading from a version prior to 5.4.0 . . . 62
23.19. Upgrading from a version prior to 5.5.0 . . . 64
23.20. Upgrading from a version prior to 5.6.0 . . . 66

23.21. Upgrading from a version prior to 5.7.0 . . . 67
23.22. Upgrading from a version prior to 6.0.0 . . . 67
23.23. Upgrading from a version prior to 6.1.0 . . . 70
23.24. Upgrading from a version prior to 6.2.0 . . . 71
23.25. Upgrading from a version prior to 6.2.1 . . . 71
23.26. Upgrading from a version prior to 6.3.0 . . . 71
23.27. Upgrading from a version prior to 6.4.0 . . . 73
23.28. Upgrading from a version prior to 6.5.0 . . . 74
23.29. Upgrading from a version prior to 7.0.0 . . . 76
23.30. Upgrading from a version prior to 7.1.0 . . . 82
23.31. Upgrading from a version prior to 7.2.0 . . . 83
23.32. Upgrading from a version prior to 7.3.0 . . . 84
23.33. Upgrading from a version prior to 7.4.0 . . . 85
24. Additional Information . . . 86
25. Commercial Support and Contact Information . . . 87
26. GNU Free Documentation License . . . 88

1. About this Guide

This guide covers procedures to upgrade PacketFence servers.

1.1. Other sources of information

Clustering Guide
: Covers installation in a clustered environment.

Developer's Guide
: Covers API, captive portal customization, application code customizations and instructions for supporting new equipment.

Installation Guide
: Covers installation and configuration of PacketFence.

Network Devices Configuration Guide
: Covers switches, WiFi controllers and access points configuration.

PacketFence News
: Covers noteworthy features, improvements and bug fixes by release.

These files are included in the package and release tarballs.

2. General Upgrade Tips

2.1. Prerequisites

The MariaDB root password that was provided during the initial configuration is required.

2.2. Database and configurations backup

NOTE Starting from PacketFence 11.0.0, this step is not necessary for doing an automated upgrade.

Taking a complete backup of the current installation is strongly recommended. Perform a backup using:

For PacketFence versions prior to 9.0.0:

```
/usr/local/pf/addons/database-backup-and-maintenance.sh
```

For PacketFence versions 9.0.0 and later:

```
/usr/local/pf/addons/backup-and-maintenance.sh
```

2.3. Disable monit alerts (only if monit is installed)

NOTE Starting from PacketFence 11.0.0, this step is not necessary for doing an automated upgrade.

If `monit` is installed and running, stop and disable it with:

```
systemctl stop monit
systemctl disable monit
```

3. Type of upgrades

Starting from PacketFence 11.0.0, the PacketFence installation can be upgraded in two ways:

- Apply maintenance patches
- Upgrade to another version (major or minor)

For all PacketFence versions prior to 11.0.0, follow the steps described in the Upgrade procedure.

4. Apply maintenance patches

4.1. Important note for cluster environments

In cluster environments, you need to perform following steps on **one server at a time**. To avoid multiple moves of the virtual IP addresses, you can start with nodes which don't own any virtual IP addresses first. You must ensure all services have been restarted correctly before moving to the next node.

4.2. Disable monit alerts (only if monit is installed)

If `monit` is installed and running, shut it down with:

```
systemctl stop monit
systemctl disable monit
```

4.3. Stop all PacketFence services

It is recommended to stop all PacketFence services that are currently running before proceeding any further:

```
/usr/local/pf/bin/pfcmd service pf stop
systemctl stop packetfence-config
```

4.4. Upgrade packages

WARNING	All non-configuration files will be overwritten by new packages. All changes made to any other files will be lost during the upgrade.

In order to upgrade your PacketFence packages to latest version, you can run following commands:

4.4.1. RHEL-based systems

```
yum clean all --enablerepo=packetfence
yum update --enablerepo=packetfence
```

4.4.2. Debian-based systems

- If `libmariadb-dev` is installed on your system at a version prior to 10.5.18

- If `packetfence-captive-portal-javascript` or `packetfence-doc` or `packetfence-pfappserver-javascript` are installed on your system **and** your PacketFence version is 12.0 or 12.1

You will need to run following commands:

```
apt update
apt install packetfence
apt autoremove
apt upgrade
```

In all other cases, you can simply run:

```
apt update
apt upgrade
```

NOTE In order to get `libmariadb-dev` package version, you can run following command: `dpkg -l | grep libmariadb-dev`. If previous command doesn't return anything, `libmariadb-dev` is not installed.

4.5. New versions of config files

Once packages are all upgraded, you should take care to review any changes to configuration files and merge them if required.

To find out which configuration files have changed run following command:

RHEL-based systems

```
find /usr/local/pf -name \*.rpmnew
```

Debian-based systems

```
find /usr/local/pf -name "*.dpkg-dist"
```

The list of files returned are the new versions shipped with PacketFence. Compare them with your installed versions and see if there are changes that should be merged into your existing configuration.

NOTE Debian-based systems should have interactively asked for existing modified files.

Then, once you are done make sure to delete these files so that there is no confusion the next time you upgrade PacketFence:

```
find /usr/local/pf -name \*.rpmnew -delete
find /usr/local/pf -name "*.dpkg-dist" -delete
```

4.6. Rebooting after services have been stopped (optional)

If you need to reboot a standalone server or a server from a cluster after services have been stopped, make sure you set the systemd target to `multi-user.target` before rebooting:

```
systemctl set-default multi-user.target
```

This will make sure your services don't start up after the reboot.

Set it back to previous target after it boots up:

Cluster

```
systemctl set-default packetfence-cluster.target
```

Standalone

```
systemctl set-default packetfence.target
```

4.7. Restart PacketFence services

```
/usr/local/pf/bin/pfcmd pfconfig clear_backend
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd service pf restart
```

5. Upgrade to another version (major or minor)

5.1. For a standalone server

Follow instructions related to automation of upgrades.

5.2. For a cluster

Please refer to the PacketFence Clustering Guide, more specifically the Performing an upgrade on a cluster.

6. Upgrading from a version prior to 8.0.0

6.1. Realms upgrade

The way PacketFence detects if the realm is stripped out of the username when performing authentication and authorisation has been moved to the realms. Moreover, it is now configurable based on the context (login on the captive portal or administration interface, as well as when performing authorization in RADIUS 802.1x)

In order to migrate the configuration, use the following script to help guide you through the migration:

```
/usr/local/pf/addons/upgrade/to-8.0-authentication-conf.pl
```

6.2. Fingerbank v2

Device names

Packetfence now uses Fingerbank v2 for improved device profiling. Since this new version brings new device names, a rename of the current data is necessary.

Rename the current data:

```
/usr/local/pf/addons/upgrade/to-8.0-fingerbank-db-data.pl
```

Mandatory Fingerbank API key

Fingerbank no longer releases a signature database and now uses an API for device profiling. In order for device profiling to continue working, there must be a Fingerbank API key configured in PacketFence.

In order to do so, you should make sure you have the following in `/usr/local/fingerbank/conf/fingerbank.conf`

NOTE

In order to request an API key, you can visit the following URL: https://api.fingerbank.org/users/register

```
[upstream]
api_key=YOUR_API_KEY_GOES_HERE
```

WARNING

Fingerbank v1 and v2 **do not** use the same infrastructure. The accounts (API keys) created on fingerbank.inverse.ca before the 8.0 release have been migrated to api.fingerbank.org. Still, you should make sure that you have the

correct API key configured in fingerbank.conf by looking at your profile on https://api.fingerbank.org/users/register. If you have a corporate account, then you can safely assume its been migrated, you can email fingerbank@inverse.ca for a confirmation. If you use a Github account and you have tried Fingerbank v2 prior to the PacketFence 8.0 release, **then your API key will be different**. Make sure you update fingerbank.conf in that case.

If you manage a large scale environment, you'll want to make sure your account can perform an unlimited amount of API requests on Fingerbank so that device profiling works correctly in a consistent way. In order to obtain this, contact fingerbank@inverse.ca. Note that most Inverse customers are entitled to free unlimited usage of the Fingerbank Cloud API.

6.3. Changes to the default switch roles

The default roles that were returned using "Role by Switch Role" have been removed. If you were relying on them to be returned in the RADIUS response, then you need to add them back in the default switch in the 'Roles' tab.

The previous values were:

- `registration`: `registration`
- `isolation`: `isolation`
- `macDetection`: `macDetection`
- `inline`: `inline`
- `voice`: `voice`

This is should only be necessary if you are using ACL assignment on your switches and using the default names that were there in PacketFence before.

6.4. Removal of the graphite database

PacketFence doesn't use graphite anymore for its dashboard. It is recommended to delete the graphite database although this is purely optional.

In order to do so, execute the following:

```
mysql -u root -p -e "drop database pf_graphite"
```

6.5. Changes to DNS filters

The $qname parameter need to be removed from dns_filters.conf

In order to do so, execute the following command:

```
sed -i -e 's/\$qname//g' /usr/local/pf/conf/dns_filters.conf
```

6.6. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.4 schema to 8.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-7.4.0-8.0.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 8.0.0).

7. Upgrading from a version prior to 8.1.0

7.1. Changes on unreg_on_accounting_stop parameter

The global configuration parameter unreg_on_acct_stop has been moved in the connection profile. So if you enabled it then make sure to enable it now in the connection profile.

7.2. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.4 schema to 8.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-8.0.0-8.1.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 8.1.0).

8. Upgrading from a version prior to 8.2.0

8.1. Queue Stats maintenance job removal

The queue_stats maintenance job has been deprecated in favor of using pfstats. In order to remove configuration related to this maintenance job, run:

```
/usr/local/pf/addons/upgrade/to-8.2-pfmon-conf.pl
```

8.2. Upgrade pfdetect Perl regex to the go RE2 regex

The pfdetect was moved from Perl to Go so all rule regexes have to be converted to the RE2 regex syntax. RE2 is mostly is compatiable the Perl regex syntax. More information on the RE2 syntax can be found here https://github.com/google/re2/wiki/Syntax. To upgrade the regex run:

```
/usr/local/pf/addons/upgrade/to-8.2-pfdetect-conf.pl
```

Any Perl regex that cannnot be convert will be displayed and should be fixed.

8.3. Upgrade realm.conf to be tenant aware

The realms are now multi-tenant aware, in order to upgrade your configuration to have the existing realms use the default tenant, execute the following script:

```
/usr/local/pf/addons/upgrade/to-8.2-realm-conf.pl
```

8.4. The api_user table has been deprecated

Any users in that were in the api_user table should be migrated to PacketFence local account (password table)

8.5. Upgrade pf user privileges

Starting from 8.2, stored routines will be dump **with** the PacketFence database. The user created at the installation ('pf' by default) in database need to have additional privileges to do that task.

To upgrade the privileges of that user, run the following command:

```
/usr/local/pf/addons/upgrade/to-8.2-upgrade-pf-privileges.sh
```

8.6. Update connection_type from WIRED_MAC_AUTH to Ethernet-NoEAP

We merged the WIRED_MAC_AUTH and Ethernet-NoEAP to Ethernet-NoEAP so the configuration needs to be updated, to do that run:

```
sed -i "s/WIRED_MAC_AUTH/Ethernet-NoEAP/g" /usr/local/pf/conf/profiles.conf
/usr/local/pf/conf/vlan_filters.conf /usr/local/pf/conf/radius_filters.conf
/usr/local/pf/conf/switch_filters.conf /usr/local/pf/conf/authentication.conf
```

8.7. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 8.1 schema to 8.2.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-8.1.0-8.2.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 8.2.0).

9. Upgrading from a version prior to 8.3.0

9.1. Upgrade pf.conf to rename configuration parameters

We moved radius_authentication_methods section to radius_configuration and moved all the radius configuration parameters in this new section. To upgrade your configuration execute the following script:

```
/usr/local/pf/addons/upgrade/to-8.3-rename-pf-conf-parameters.pl
```

9.2. Upgrade authentication.conf to add searchattributes parameter

We added a new parameter in AD and LDAP authentication sources to be able to do 802.1x authentication with any unique ldap attributes. This parameter "searchattributes" need to be added in the existing authentication sources. To apply this configuration execute the following script:

```
/usr/local/pf/addons/upgrade/to-8.3-authentication-searchattributes.pl
```

9.3. Adjustment to the encoding of the configuration files and templates

Configuration and templates in the admin were previously being saved as latin1 instead of utf8.

This script will convert all latin1 config file to utf8

```
/usr/local/pf/addons/upgrade/to-8.3-conf-latin1-to-utf8.sh
```

9.4. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 8.2 schema to 8.3.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-8.2.0-8.3.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 8.3.0).

10. Upgrading from a version prior to 9.0.0

10.1. Support for Debian 8 dropped

Debian 8 will not be supported anymore for versions 9.0.0 and above. You should instead use Debian 9 now as it is currently the only supported Debian version.

10.2. Necessity to use MariaDB

NOTE

This only applies to users using an external database server. If your database is hosted on the same server as PacketFence whether you are in cluster or standalone, this requires no attention.

Users hosting an external database for PacketFence will need to run a recent version of MariaDB as it will be the only supported database backend. Failure to use MariaDB may result in errors in the database migration script.

In order to migrate to MariaDB, it is suggested to create a new database server and perform an export of the data through mysqldump and import it in the new server.

The recommended MariaDB version for PacketFence is currently 10.1.21

A recent version of MySQL can also work but going forward, the only tested database engine will be MariaDB.

10.3. Deprecate the classic dhcp filters

The previous dhcp filters engine has been deprecated in favor of the new one who is able to modify the dhcp answer on the fly.

10.4. Violations have been renamed to Security Events

The violations have been renamed to security events. In order to make the appropriate changes in your configuration, execute the following script:

```
/usr/local/pf/addons/upgrade/to-9.0-security-events.sh
```

10.5. Removed MAC detection setting

The MAC detection setting in the switches has been removed. In order to cleanup the switches configuration for the removal of this setting, execute the following script:

```
/usr/local/pf/addons/upgrade/to-9.0-remove_mac_detection.sh
```

10.6. Modifications to accounting cleanup

Accounting cleanup is now done via a pfmon task (acct_cleanup) instead of the database backup and maintenance script. Make sure you adjust the cleanup window in pfmon's configuration (Configuration→System Maintenance→Maintenance) if necessary. Also note that the default retention for the accounting data has been lowered to 1 day instead of 1 week like it was before.

10.7. Admin roles configuration

In order to upgrade the Admin rights, run the following commands

```
cd /usr/local/pf
sed -i "s/SERVICES/SERVICES_READ/g" /usr/local/pf/conf/adminroles.conf
sed -i "s/REPORTS/REPORTS_READ/g" /usr/local/pf/conf/adminroles.conf
```

10.8. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 8.3 schema to 9.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-8.3.0-9.0.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 9.0.0).

11. Upgrading from a version prior to 9.1.0

11.1. Now possible to disable a domain

In order to add the necessary enabled flag to your existing domains, run the following command:

```
/usr/local/pf/addons/upgrade/to-9.1-add-domain-conf.pl
```

11.2. pfperl-api port

The port of the pfperl-api service has changed, in order to adjust the existing configuration, run the following command:

```
/usr/local/pf/addons/upgrade/to-9.1-update-api.conf.sh
```

11.3. Linkedin OAuth2

The LinkedIn API calls have changed drastically. On top of the new LinkedIn modules that are part of the update, you will need to change the following parameter in all your existing LinkedIn sources:

```
API URL of logged user ->
https://api.linkedin.com/v2/emailAddress?q=members&projection=(elements*(handle
~))
```

11.4. VLAN pool configuration

The VLAN pool strategy configuration has been moved to the connection profiles.

In order to migrate the current setting of pf.conf into profiles.conf, you will need to run the following command:

```
/usr/local/pf/addons/upgrade/to-9.1-move-vlan-pool-technique-parameter.pl
```

11.5. Remove Useragent Triggers

The useragent and user_agent security event triggers have been deprecated. Performing HTTP User-Agent based detection is extremelly inefficient given the very dynamic nature of HTTP User-Agents. You should instead be using the device trigger which leverages the device profiling

performed by Fingerbank. In order to remove any existing useragent trigger, execute the following script:

```
/usr/local/pf/addons/upgrade/to-9.1-security-events-remove-useragent.pl
```

11.6. Self service portal

The device registration configuration file has been removed in favor of using a configuration file for all the self service portal features (status page + device registration).

In order to migrate your configuration, run the following script:

```
/usr/local/pf/addons/upgrade/to-9.1-selfservice-conf.pl
```

11.7. Password of the day rotation

Password of the day source now uses access duration values to rotate password.

In order to migrate your configuration, run the following script:

```
/usr/local/pf/addons/upgrade/to-9.1-update-potd.pl
```

11.8. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 9.0 schema to 9.1.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-9.0.0-9.1.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 9.1.0).

12. Upgrading from a version prior to 9.2.0

12.1. Merge of all RPM packages into one (RHEL / CentOS only)

NOTE | This step needs to be done **before** packages upgrade.

Starting from now, PacketFence will be released as an unique RPM package for `x86_64` architectures. To remove properly older RPM packages, you need to follow these steps:

1. Follow instructions mentioned in Stop all PacketFence services section and stop before starting packages upgrades
2. Uninstall old RPM without running post-uninstallation steps:

```
rpm -e --nodeps --noscripts packetfence-config

# run only if packetfence-remote-arp-sensor has been installed
rpm -e --nodeps --noscripts packetfence-remote-arp-sensor
```

3. Recopy previous `pfconfig.conf` filename to its original location:

```
mv -f /usr/local/pf/conf/pfconfig.conf.rpmsave
/usr/local/pf/conf/pfconfig.conf
```

4. Upgrade PacketFence packages by following instructions in Packages upgrades section for RHEL / CentOS based systems
5. Continue upgrade procedure

At the end of upgrade procedure, you should have only one RPM package called `packetfence`. If you previously installed `packetfence-release` package in order to have PacketFence repository installed, this one has been upgraded to latest version.

12.2. New GPG key for Debian installations (Debian only)

NOTE | This step needs to be done **before** packages upgrade.

In order to install new versions of Debian packages, you will need to add a new GPG key to your system:

```
wget -O - https://inverse.ca/downloads/GPG_PUBLIC_KEY | sudo apt-key add -
```

You can safely remove the oldest one:

```
sudo apt-key del FE9E84327B18FF82B0378B6719CDA6A9810273C4
```

12.3. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 9.1 schema to 9.2.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-9.1.0-9.2.0.sql
```

Once completed, update the file `/usr/local/pf/conf/currently-at` to match the new release number (PacketFence 9.2.0):

```
cat /usr/local/pf/conf/pf-release > /usr/local/pf/conf/currently-at
```

13. Upgrading from a version prior to 9.3.0

13.1. Execute script action doesn't use sudo anymore

Execute script action in security events doesn't use `sudo` anymore to run scripts. Consequently, you should ensure that `pf` user is:

- able to read and execute these scripts
- able to run commands inside these scripts (without `sudo`)

13.2. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 9.2 schema to 9.3.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-9.2.0-9.3.0.sql
```

Once completed, update the file `/usr/local/pf/conf/currently-at` to match the new release number (PacketFence 9.3.0):

```
cat /usr/local/pf/conf/pf-release > /usr/local/pf/conf/currently-at
```

14. Upgrading from a version prior to 10.0.0

14.1. Kernel development package

NOTE | This step needs to be done **before** packages upgrade.

In this version we need to have the kernel development package that matches your current kernel version in order to build the Netflow kernel module.

14.1.1. RHEL / CentOS based systems

```
yum install kernel-devel-$(uname -r)
```

The headers for your specific kernel may not be published anymore in the CentOS repository. If that is the case, then perform the following prior to the upgrade:

```
yum update kernel
reboot
yum install kernel-devel-$(uname -r)
```

NOTE | Be sure to follow instructions in [_rebooting_after_services_have_been_stopped] section to ensure services will not restart.

14.1.2. Debian-based systems

```
apt install linux-headers-$(uname -r)
```

14.2. Timezone

The timezone set in pf.conf will be set on the operating system every time PacketFence reloads its configuration. For this reason, you should review the timezone setting in the general section of pf.conf (System Configuration → General Configuration in the admin). If its empty, PacketFence will use the timezone that is already set on the server and you don't have anything to do. Otherwise, it will set the timezone in this setting on the operating system layer for consistency which may modify the timezone setting of your operating system. In this case you should ensure that you reboot the server after completing all the steps of the upgrade so that the services start with the right timezone.

14.3. Tracking configuration service enabled by default

`packetfence-tracking-config` service is now enabled by default. It means that all manual

changes to configuration files will be recorded, including passwords.

You can disable this service from PacketFence web admin if you don't want such behavior.

14.4. New PacketFence PKI in Golang

NOTE If you do not use the PacketFence PKI, you can safely ignore this step

PacketFence-pki is deprecated in favour of the new PacketFence PKI written in Golang. If you previously used the PacketFence-pki you will need to migrate from the SQLite database to MariaDB. To migrate, be sure that the database is running and the new PKI too and do the following:

```
/usr/local/pf/addons/upgrade/to-10.0-packetfence-pki-migrate.pl
```

Next edit the PKI providers (Configuration → PKI Providers) and redefine the profile to use. Finally, if you use OCSP then change the URL to use this one: http://127.0.0.1:22225/api/v1/pki/ocsp

14.5. New MariaDB Galera recovery service

This release adds a new service that will automatically attempt to recover broken Galera cluster members and can also perform a full recovery of a Galera cluster. These automated decisions may lead to potential data loss. If this is not acceptable for you disable the galera-autofix service in pf.conf or in "System Configuration→Services". More details and documentation is available in the "The galera-autofix service" section of the clustering guide.

14.6. Removal of currently-at file and configurator display

The file `/usr/local/pf/conf/currently-at` is no longer needed, it can be removed:

```
rm /usr/local/pf/conf/currently-at
```

You also need to disable access to configurator by running:

```
printf '\n[advanced]\nconfigurator=disabled\n' >> /usr/local/pf/conf/pf.conf
```

14.7. Database Privileges

Some queries now need CREATE TEMPORARY TABLE privilege. You will be prompted for the MariaDB root password when running this script:

```
/usr/local/pf/addons/upgrade/to-10.0-upgrade-pf-privileges.sh
```

14.8. Filter Engine

We are now using a new format for the VLAN/DNS/DHCP/RADIUS/Switch filters. This script will convert the old format to the new one:

```
/usr/local/pf/addons/upgrade/to-10.0-filter_engines.pl
```

14.9. httpd.admin daemon disabled by default

Starting from now, `httpd.admin` daemon is disabled by default and web admin interface is managed by HAProxy using `haproxy-admin` daemon.

It means that if you use a dedicated SSL certificate (different from captive portal certificate) for web admin interface, this one has been replaced by your captive portal certificate. You can find it at `/usr/local/pf/conf/ssl/server.pem`.

14.10. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 9.3 schema to 10.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-9.3.0-10.0.0.sql
```

15. Upgrading from a version prior to 10.1.0

15.1. RADIUS attributes in authentication sources

RADIUS attributes used in rules of authentication sources are now prefixed by `radius_request`. This script will add the prefix:

```
/usr/local/pf/addons/upgrade/to-10.1-authentication-prefix.pl
```

15.2. Changes in RADIUS configuration for better LDAP support

In order to improve LDAP support when using RADIUS, new files and configuration parameters have been added. This script will update your current configuration:

```
/usr/local/pf/addons/upgrade/to-10.1-move-radius-configuration-parmeters.pl
```

15.3. RADIUS filter templates

RADIUS filters now support templated values like switch templates. This script will update your RADIUS filters to new format:

```
/usr/local/pf/addons/upgrade/to-10.1-radius-filter-template.pl
```

15.4. New EAP configuration parameter in realm.conf file

A new EAP parameter has been added to `realm.conf` file. This script will add this parameter to your current configuration file:

```
/usr/local/pf/addons/upgrade/to-10.1-realm-conf.pl
```

15.5. Status of rules

It's now possible to enable/disable rules in authentication sources. This script will add the new `status` parameter:

```
/usr/local/pf/addons/upgrade/to-10.1-rule-status.pl
```

15.6. Support for CoA in Unifi controllers

Support for CoA for Unifi AP is now supported but requires to have the latest controller and AP firmware available. Make sure you run the latest version of the controller and firmware if you use Ubiquiti equipment.

15.7. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 10.0.0 schema to 10.1.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-10.0.0-10.1.0.sql
```

16. Upgrading from a version prior to 10.2.0

16.1. Backup of pfmon.conf (Debian-based systems only)

NOTE | This step needs to be done **before** packages upgrade.

Debian packages upgrades will remove `/usr/local/pf/conf/pfmon.conf` file in favor of `/usr/local/pf/conf/pfcron.conf`. In order to keep your configuration in place, you need to make a backup of your `pfmon.conf` file **before** running packages upgrades:

```
cp /usr/local/pf/conf/pfmon.conf /root/pfmon.conf.rpmsave
```

After packages upgrades have been performed, you can move file to its original location:

```
mv /root/pfmon.conf.rpmsave /usr/local/pf/conf/pfmon.conf.rpmsave
```

Configuration will be moved to `/usr/local/pf/conf/pfcron.conf` file during configuration migration step.

WARNING | `rpmsave` extension is not an error, script `to-10.2-pfmon-maintenance.pl` will migrate configuration using this filename.

16.2. Self registration portal

The parameter device_registration_role has been renamed device_registration_roles, in order to apply the change run the following script:

```
/usr/local/pf/addons/upgrade/to-10.2-selfservice-conf.pl
```

16.3. Switch type needs to be defined

If switch type was not defined, this script will set it to `Generic`:

```
/usr/local/pf/addons/upgrade/to-10.2-default-switch-packetfence-standard.pl
```

16.4. Convert the pfmon configuration file to pfcron

Convert the pfmon configuration file to pfcron

```
/usr/local/pf/addons/upgrade/to-10.2-pfmon-maintenance.pl
```

16.5. Rename PFMON* actions to PFCRON*

Rename PFMON actions to the PFCRON actions

```
/usr/local/pf/addons/upgrade/to-10.2-adminroles-conf.pl
```

16.6. Syslog parsers are now tenant aware

Add the tenant_id to pfdetect

```
/usr/local/pf/addons/upgrade/to-10.2-pfdetect-conf.pl
```

16.7. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 10.1.0 schema to 10.2.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-10.1.0-10.2.0.sql
```

17. Upgrading from a version prior to 10.3.0

17.1. MariaDB Upgrade to 10.2

NOTE | This step needs to be done **before** packages upgrade.

PacketFence now depends of the MariaDB version 10.2. In order to upgrade the MariaDB version you need to execute the following steps before upgrading PacketFence.

17.1.1. Standalone MariaDB upgrade

In order to be able to work on the server, we first need to stop all the PacketFence application services on it, see Stop all PacketFence services section.

Now stop `packetfence-mariadb`:

```
systemctl stop packetfence-mariadb
```

Now proceed with the MariaDB upgrade

RHEL / CentOS based systems

```
rpm -e --nodeps MariaDB-client MariaDB-common MariaDB-server MariaDB-shared
yum install --enablerepo=packetfence MariaDB-server
```

Debian-based systems

```
dpkg -r --force-depends mariadb-server mariadb-client-10.1 mariadb-client-core-
10.1 \
mariadb-common mariadb-server-10.1 mariadb-server-core-10.1 libmariadbclient18
mysql-common
apt update
apt install mariadb-server mariadb-client-10.2 mariadb-client-core-10.2 \
mariadb-common mariadb-server-10.2 mariadb-server-core-10.2 libmariadbclient18
libmariadb3 \
mysql-common
```

NOTE | If you manually installed Percona XtraBackup to take your backups, you need to install `MariaDB-backup` (rpm) and `mariadb-backup-10.2` (deb) as a replacement.

NOTE | On Debian, ignore prompts related to change of `root` password during package upgrade.

At this moment you have the newest version of MariaDB installed on your system. Ensure

MariaDB is running:

RHEL / CentOS based systems **only**

```
systemctl unmask mariadb
systemctl start mariadb
```

You can check you are running MariaDB 10.2 version with following command:

```
mysql -u root -p -e "show variables where Variable_name='version';"
```

Next step is to upgrade your databases:

```
mysql_upgrade -u root -p
```

NOTE If the following error appears "Recovering after a crash using tc.log" then delete the file `/var/lib/mysql/tc.log`

After databases have been upgraded, you can disable default MariaDB service:

RHEL / CentOS based systems

```
systemctl stop mariadb
systemctl mask mariadb
```

Debian-based systems

```
systemctl stop mysql
pkill -u mysql
systemctl mask mysql
```

`packetfence-mariadb` service will be started later by upgrade of PacketFence package(s).

At this step you have now the MariaDB 10.2 database ready. You can now upgrade the PacketFence version by following instructions in Packages upgrades section.

17.1.2. Cluster MariaDB upgrade

CAUTION Performing a live upgrade on a PacketFence cluster is not a straightforward operation and should be done meticulously.

In this procedure, the 3 nodes will be named A, B and C and they are in this order in `cluster.conf`. When we referenced their hostnames, we speak about hostnames in `cluster.conf`.

Backups

First, ensure you have taken backups of your data. We highly encourage you to perform

snapshots of all the virtual machines prior to the upgrade. You should also take a backup of the database and the `/usr/local/pf` directory using database and configurations backup instructions

Disabling the auto-correction of configuration

The PacketFence clustering stack has a mechanism that allows configuration conflicts to be handled accross the servers. This will come in conflict with your upgrade, so you should disable it.

In order to do so, go in *Configuration→System Configuration→Maintenance* and disable the *Cluster Check* task.

Once this is done, restart `pfmon` or `pfcron` on all nodes using:

For PacketFence versions prior to 10.2

```
/usr/local/pf/bin/pfcmd service pfmon restart
```

For PacketFence version 10.2 and later

```
/usr/local/pf/bin/pfcmd service pfcron restart
```

Disabling galera-autofix (for PacketFence version 10.0 and later)

You should disable the `galera-autofix` service in the configuration to disable the automated resolution of cluster issues during the upgrade.

In order to do so, go in *Configuration→System Configuration→Services* and disable the `galera-autofix` service.

Once this is done, stop `galera-autofix` service on **all** nodes using:

```
/usr/local/pf/bin/pfcmd service galera-autofix updatesystemd
/usr/local/pf/bin/pfcmd service galera-autofix stop
```

Migrating service on node C

In order to be able to work on node C, we first need to stop all the PacketFence application services on it:

```
/usr/local/pf/bin/pfcmd service pf stop
```

`packetfence-config` needs to stay up to disable node A and B in configuration.

NOTE The steps below will cause a temporary loss of service.

Detach node C from the cluster

First, we need to tell A and B to ignore C in their cluster configuration. In order to do so, execute the following command **on A and B** while changing `node-C-hostname` with the actual hostname of node C:

```
/usr/local/pf/bin/cluster/node node-C-hostname disable
```

Once this is done proceed to restart the following services on nodes A and B **one at a time**. This will cause service failure during the restart on node A

```
/usr/local/pf/bin/pfcmd service radiusd restart
/usr/local/pf/bin/pfcmd service pfdhcplistener restart
/usr/local/pf/bin/pfcmd service haproxy-admin restart
/usr/local/pf/bin/pfcmd service haproxy-db restart
/usr/local/pf/bin/pfcmd service haproxy-portal restart
/usr/local/pf/bin/pfcmd service keepalived restart
```

Then, we should tell C to ignore A and B in their cluster configuration. In order to do so, execute the following commands on node C while changing `node-A-hostname` and `node-B-hostname` by the hostname of nodes A and B respectively.

```
/usr/local/pf/bin/cluster/node node-A-hostname disable
/usr/local/pf/bin/cluster/node node-B-hostname disable
```

The commands above will make sure that nodes A and B will not be forwarding requests to C even if it is alive. Same goes for C which won't be sending traffic to A and B. This means A and B will continue to have the same database informations while C will start to diverge from it when it goes live. We'll make sure to reconcile this data afterwards.

MariaDB upgrade on node C

Now stop `packetfence-mariadb` on node C:

```
systemctl stop packetfence-mariadb
```

Now proceed with the MariaDB upgrade

RHEL / CentOS based systems **only**

```
rpm -e --nodeps MariaDB-client MariaDB-common MariaDB-server MariaDB-shared
yum install --enablerepo=packetfence MariaDB-server MariaDB-backup
```

Debian-based systems

```
dpkg -r --force-depends mariadb-server mariadb-client-10.1 mariadb-client-core-
10.1 \
mariadb-common mariadb-server-10.1 mariadb-server-core-10.1 libmariadbclient18
\
mysql-common
apt update
```

```
apt install mariadb-server-10.2 mariadb-common mariadb-client-10.2 \
mariadb-client-core-10.2 mariadb-server-core-10.2 libmariadb3 \
libmariadbclient18 mariadb-server mariadb-backup-10.2 mysql-common
```

NOTE On Debian, ignore prompts related to change of `root` password during package upgrade.

At this moment you have the newest version of MariaDB installed on your system. Ensure MariaDB is running:

RHEL / CentOS based systems **only**

```
systemctl unmask mariadb
systemctl start mariadb
```

You can check you are running MariaDB 10.2 version with following command:

```
mysql -u root -p -e "show variables where Variable_name='version';"
```

Next step is to upgrade your databases:

```
mysql_upgrade -u root -p
```

NOTE If the following error appear "Recovering after a crash using tc.log" then delete the file /var/lib/mysql/tc.log

After databases have been upgraded, you can disable default MariaDB service:

RHEL / CentOS based systems **only**

```
systemctl stop mariadb
systemctl mask mariadb
```

Debian-based systems

```
systemctl stop mysql
pkill -u mysql
systemctl mask mysql
```

At this step you have now the MariaDB 10.2 database ready. In order to start MariaDB as standalone on node C, you need to regenerate MariaDB config (`packetfence-mariadb` service will be started later by upgrade of packetfence package(s))

```
/usr/local/pf/bin/pfcmd generatemariadbconfig
```

Upgrading node C

Next, you can upgrade your operating system and/or PacketFence on node C by following instructions of Packages upgrades section.

IMPORTANT | If you are on a RHEL/CentOS based systems, the command to install `packetfence-release` released with 10.3.0 version will be:

```
https://www.packetfence.org/downloads/PacketFence/RHEL7/packetfence-release-
7.stable.noarch.rpm
```

Maintenance patches (on node C)

In order to have latest bug fixes on your PacketFence version, you can apply maintenance patches by running:

```
/usr/local/pf/addons/pf-maint.pl
```

Configuration migration and database schema updates (on node C)

Now, make sure you follow the directives in the upgrade guide as you would on a standalone server **including** the database schema updates.

Start service on node C

Now, start the application service on node C using following instructions:

```
/usr/local/pf/bin/pfcmd fixpermissions
/usr/local/pf/bin/pfcmd pfconfig clear_backend
systemctl restart packetfence-config
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd service pf restart
```

Stop services on nodes A and B

Next, stop all application services on node A and B:

- Stop all PacketFence services:

```
/usr/local/pf/bin/pfcmd fixpermissions
/usr/local/pf/bin/pfcmd pfconfig clear_backend
systemctl restart packetfence-config
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd service pf stop
```

- Stop database:

```
systemctl stop packetfence-mariadb
```

Validate migration

You should now have full service on node C and should validate that all functionnalities are working as expected. Once you continue past this point, there will be no way to migrate back to nodes A and B in case of issues other than to use the snapshots taken prior to the upgrade.

If all goes wrong

If your migration to node C goes wrong, you can fail back to nodes A and B by stopping all services on node C and starting them on nodes A and B

On node C

```
systemctl stop packetfence-mariadb
/usr/local/pf/bin/pfcmd service pf stop
```

On nodes A and B

```
systemctl start packetfence-mariadb
/usr/local/pf/bin/pfcmd service pf start
```

Once you are feeling confident to try your failover to node C again, you can do the exact opposite of the commands above to try your upgrade again.

If all goes well

If you are happy about the state of your upgrade, you can continue on the steps below in order to complete the upgrade of the two remaining nodes.

MariaDB upgrade on nodes A and B

Now proceed with the MariaDB upgrade:

RHEL / CentOS based systems

```
rpm -e --nodeps MariaDB-client MariaDB-common MariaDB-server MariaDB-shared
yum install --enablerepo=packetfence MariaDB-server MariaDB-backup
```

Debian-based systems

```
dpkg -r --force-depends mariadb-server mariadb-client-10.1 mariadb-client-core-
10.1 \
mariadb-common mariadb-server-10.1 mariadb-server-core-10.1 libmariadbclient18
\
mysql-common
apt update
apt install mariadb-server-10.2 mariadb-common mariadb-client-10.2 \
```

```
mariadb-client-core-10.2 mariadb-server-core-10.2 libmariadb3 \
libmariadbclient18 mariadb-server mariadb-backup-10.2 mysql-common
```

NOTE On Debian, ignore prompts related to change of `root` password during package upgrade.

To let nodes A and B rejoin cluster **before** upgrading PacketFence packages, you need to update MariaDB configuration:

```
sed -i "s/xtrabackup/mariabackup/g" /usr/local/pf/conf/mariadb/mariadb.conf.tt
```

At this moment you have the newest version of MariaDB installed on nodes A and B.

On Debian-based systems **only**, you need to stop default `mysql` service:

Debian-based systems **only**

```
systemctl stop mysql
pkill -u mysql
systemctl mask mysql
```

At this step you have now the MariaDB 10.2 database ready.

Reintegrating nodes A and B

Optional step: Cleaning up data on node C

When you will re-establish a cluster using node C in the steps below, your environment will be set in read-only mode for the duration of the database sync (which needs to be done from scratch).

This can take from a few minutes to an hour depending on your database size.

We highly suggest you delete data from the following tables if you don't need it:

- `radius_audit_log`: contains the data in *Auditing→RADIUS Audit Logs*
- `ip4log_history`: Archiving data for the IPv4 history
- `ip4log_archive`: Archiving data for the IPv4 history
- `locationlog_history`: Archiving data for the node location history

You can safely delete the data from all of these tables without affecting the functionnalities as they are used for reporting and archiving purposes. Deleting the data from these tables can make the sync process considerably faster.

In order to truncate a table:

```
mysql -u root -p pf
MariaDB> truncate TABLE_NAME;
```

Elect node C as database master

In order for node C to be able to elect itself as database master, we must tell it there are other members in its cluster by re-enabling nodes A and B

```
/usr/local/pf/bin/cluster/node node-A-hostname enable
/usr/local/pf/bin/cluster/node node-B-hostname enable
```

Next, enable node C on nodes A and B by executing the following command on the two servers:

```
systemctl start packetfence-config
/usr/local/pf/bin/cluster/node node-C-hostname enable
```

Now, stop `packetfence-mariadb` on node C, regenerate the MariaDB configuration and start it as a new master:

NOTE

Before starting this step, be sure that the galera_replication_username has grant permission PROCESS

```
mysql -u root -p
select * from information_schema.user_privileges where PRIVILEGE_TYPE=
"PROCESS";
# If it's not the case
GRANT PROCESS ON *.* TO ''`galera_replication_username`''@localhost;
```

```
systemctl stop packetfence-mariadb
/usr/local/pf/bin/pfcmd generatemariadbconfig
/usr/local/pf/sbin/pf-mariadb --force-new-cluster
```

You should validate that you are able to connect to the MariaDB database even though it is in read-only mode using the MariaDB command line:

```
mysql -u root -p pf -h localhost
```

If its not, make sure you check the MariaDB log (`/usr/local/pf/logs/mariadb_error.log`)

Sync nodes A and B

On each of the servers you want to discard the data from, stop `packetfence-mariadb`, you must destroy all the data in `/var/lib/mysql` and start `packetfence-mariadb` so it resyncs its data from scratch.

```
systemctl stop packetfence-mariadb
rm -fr /var/lib/mysql/*
/usr/local/pf/bin/pfcmd generatemariadbconfig
```

```
systemctl start packetfence-mariadb
```

Should there be any issues during the sync, make sure you look into the MariaDB log (`/usr/local/pf/logs/mariadb_error.log`)

Once both nodes have completely synced (try connecting to it using the MariaDB command line), then you can break the cluster election command you have running on node C and start node C normally (using `systemctl start packetfence-mariadb`).

Upgrading nodes A and B

Next, you can upgrade your operating system and/or PacketFence on nodes A and B by following instructions of Packages upgrades section.

WARNING	You only need to merge changes of new configuration files that will not be synced by `/usr/local/pf/bin/cluster/sync` command described below.
IMPORTANT	If you are on a RHEL/CentOS based systems, the command to install `packetfence-release` released with 10.3.0 version will be:

```
https://www.packetfence.org/downloads/PacketFence/RHEL7/packetfence-release-
7.stable.noarch.rpm
```

Maintenance patches (on nodes A and B)

In order to have latest bug fixes on your PacketFence version, you can apply maintenance patches by running:

```
/usr/local/pf/addons/pf-maint.pl
```

Configuration synchronisation

You do not need to follow the upgrade procedure when upgrading these nodes. You should instead do a sync from node C on nodes A and B:

```
/usr/local/pf/bin/cluster/sync --from=192.168.1.5 --api-user=packet --api
-password=anotherMoreSecurePassword
/usr/local/pf/bin/pfcmd configreload hard
```

Where:

- `192.168.1.5` is the management IP of node C
- `packet` is the webservices username (*Configuration→Webservices*)
- `fence` is the webservices password (*Configuration→Webservices*)

Start nodes A and B

Before starting PacketFence services on nodes A and B, `packetfence-mariadb` need to be

restarted again to take into account changes introduced by packages upgrades:

```
systemctl restart packetfence-mariadb
```

You can now safely start PacketFence on nodes A and B using following instructions:

```
/usr/local/pf/bin/pfcmd fixpermissions
/usr/local/pf/bin/pfcmd pfconfig clear_backend
systemctl restart packetfence-config
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd service pf restart
```

Restart node C

Now, you should restart PacketFence on node C using following instructions:

```
/usr/local/pf/bin/pfcmd fixpermissions
/usr/local/pf/bin/pfcmd pfconfig clear_backend
systemctl restart packetfence-config
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd service pf restart
```

So it becomes aware of its peers again.

You should now have full service on all 3 nodes using the latest version of PacketFence.

Reactivate the configuration conflict handling

Now that your cluster is back to a healthy state, you should reactivate the configuration conflict resolution.

In order to do, so go in *Configuration→System Configuration→Maintenance* and re-enable the *Cluster Check* task.

Once this is done, restart `pfcron` on all nodes using:

```
/usr/local/pf/bin/pfcmd service pfcron restart
```

Reactivate galera-autofix

You now need to reactivate and restart the `galera-autofix` service so that it's aware that all the members of the cluster are online again.

In order to do so, go in *Configuration→System Configuration→Services* and re-enable the `galera-autofix` service.

Once this is done, restart `galera-autofix` service on **all** nodes using:

```
/usr/local/pf/bin/pfcmd service galera-autofix updatesystemd
/usr/local/pf/bin/pfcmd service galera-autofix restart
```

17.2. Rename win_agent_download_uri → windows_agent_download_uri

```
/usr/local/pf/addons/upgrade/to-10.3-provisioners-windows_agent_download_uri.pl
```

17.3. LDAP port per server has been deprecated

The ability to define a specific port per host in the list of the LDAP servers of a single authentication source has been deprecated. If you have such entries, adjust them accordingly. If you have been using the same LDAP port for all the hosts in an authentication source, then this will not apply to you.

17.4. Removal of `inline_accounting` table

`inline_accounting` table will be removed by upgrade of database schema (see below) because it has been replaced by `bandwidth_accounting` table since v10.

You are only concern by this item if you extract data from `inline_accounting` table before v10 for external usage.

17.5. `pfdhcplistener` is now tenant aware

To add the default `tenant_id` (1) to all network configurations run:

```
/usr/local/pf/addons/upgrade/to-10.3-network-conf.pl
```

17.6. Regenerate domain(s) configuration

Active Directory fail-over have been improved. To benefit from such improvements, you need to regenerate domain(s) configuration with following commands:

```
/usr/local/pf/bin/pfcmd generatedomainconfig
```

IMPORTANT | During a cluster upgrade, you need to run these commands on each cluster member.

17.7. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL

upgrade script has been provided to upgrade the database from the 10.2.0 schema to 10.3.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-10.2.0-10.3.0.sql
```

17.8. Restart of `packetfence-mariadb` service (standalone installations only)

To be sure you are running latest MariaDB configuration provided by PacketFence packages, you need to restart `packetfence-mariadb`:

```
systemctl restart packetfence-mariadb
```

18. Upgrading from a version prior to 11.0.0

Starting from PacketFence 11.0.0, Debian 9 and CentOS 7 support are dropped in benefit of Debian 11 and RHEL 8. In place upgrades are not supported. You will have to provision new operating system(s) in order to migrate.

To simplify upgrade process to PacketFence 11.0.0 and future versions, we now rely on an export/import mechanism.

Before doing anything else, be sure to read assumptions and limitations of this mechanism.

18.1. Export (on current installation)

18.1.1. If you are running a PacketFence version before 10.3.0

1. Follow upgrade path to PacketFence 10.3.0
2. Go to next section

18.1.2. If you are running PacketFence version 10.3.0 or later

Follow instructions related to export process.

18.2. Import (on new installation)

Follow instructions related to import process.

18.3. Instructions for upgrades without import

If you don't use import mechanism to upgrade your previous PacketFence installation, you will need to follow the instructions in this section to upgrade the configuration and database schema.

18.3.1. Configuration upgrade

```
# Only run this if you don't import your previous configuration
/usr/local/pf/addons/upgrade/to-11.0-firewall_sso-conf.pl
/usr/local/pf/addons/upgrade/to-11.0-no-slash-32-switches.pl
/usr/local/pf/addons/upgrade/to-11.0-openid-username_attribute.pl
```

18.3.2. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 10.3 schema to 11.0.

To upgrade the database schema, run the following command:

```
# Only run this if you don't import your previous configuration
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-10.3-11.0.sql
```

18.4. NTLM cache background job deprecated in Active Directory Domains

The option `NTLM cache background job` and its associated parameters have been deprecated. If you previously used this option on one of your domains, it will now automatically use the `NTLM cache on connection` method.

18.5. pf-maint.pl script deprecated

The `pf-maint.pl` script used to get maintenance patches has been deprecated. You can now get maintenance patches using your package manager, see Apply maintenance patches section.

18.6. WMI scan engine deprecated

The WMI scan engines have been deprecated. If you previously used these scan engines, you should migrate to other scan engines.

18.7. TLS 1.0 and 1.1 are disabled by default in FreeRADIUS

TLS 1.0 and TLS 1.1 are now disabled by default. If you still have supplicants using theses protocols, you should move to TLS 1.2. If it's not possible, you can adjust `TLS Minimum version` in *Configuration → System configuration → RADIUS → TLS profiles*.

19. Upgrading from a version prior to 11.1.0

19.1. Automation of upgrades for standalone servers

Upgrades are now automated for standalone servers starting from PacketFence 11.0.0. Follow instructions related to automation of upgrades.

19.2. Support of custom rules in iptables.conf

PacketFence now provides a way to add custom rules in `/usr/local/pf/conf/iptables.conf` using two files:

- `/usr/local/pf/conf/iptables-input.conf.inc` for all input traffic
- `/usr/local/pf/conf/iptables-input-management.conf.inc` for all input traffic related to management interface

If you previously added custom rules in `iptables.conf`, we recommend you to move these rules into these files.

19.3. Support of local authentication for 802.1X in web admin

PacketFence now allow to enable or disable local authentication for 802.1X directly in web admin.

If you previously enabled `packetfence-local-auth` feature in `/usr/local/pf/conf/radiusd/packetfence-tunnel`, we recommend you to enable this feature in PacketFence web admin (see EAP local user authentication).

19.4. Support of Monit configuration in pf.conf

Monit configuration is now managed directly in `/usr/local/pf/conf/pf.conf`. An upgrade script will be used during upgrade process to automatically migrate existing Monit configuration into `/usr/local/pf/conf/pf.conf`.

19.5. Note for cluster upgrades

If you use a cluster, their upgrade isn't yet automated so you will need to follow the instructions in this section to upgrade the configuration and database schema.

19.5.1. Configuration upgrade

```
# Only run this for cluster upgrades
/usr/local/pf/addons/upgrade/to-11.1-cleanup-ntlm-cache-batch-fields.pl
/usr/local/pf/addons/upgrade/to-11.1-migrate-monit-configuration-to-pf-conf.pl
```

```
/usr/local/pf/addons/upgrade/to-11.1-remove-unused-sources.pl
/usr/local/pf/addons/upgrade/to-11.1-remove-wmi-scan.pl
/usr/local/pf/addons/upgrade/to-11.1-update-reports.pl
```

19.5.2. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 11.0 schema to 11.1.

To upgrade the database schema, run the following command:

```
# Only run this for cluster upgrades
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-11.0-11.1.sql
```

20. Upgrading from a version prior to 11.2.0

20.1. Automation of upgrades for standalone servers

Upgrades are now automated for standalone servers starting from PacketFence 11.0.0. Follow instructions related to automation of upgrades.

20.2. Note for cluster upgrades

If you use a cluster, their upgrade isn't yet automated so you will need to follow the instructions in this section to upgrade the configuration and database schema.

20.2.1. Configuration upgrade

```
/usr/local/pf/addons/upgrade/to-11.2-pfcron.pl
/usr/local/pf/addons/upgrade/to-11.2-pfcron-populate_ntlm_redis_cache.pl
/usr/local/pf/addons/upgrade/to-11.2-upgrade-pf-privileges.sh
```

20.2.2. Database schema

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 11.1 schema to 11.2.

To upgrade the database schema, run the following command:

```
# Only run this for cluster upgrades
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-11.1-11.2.sql
```

20.3. Change of behavior for filter engines not_equals operator

If any condition for filters (VLAN, RADIUS, Switch, DNS, DHCP, and Profile) uses a `not equals` operator. Check if the logic is still ok if the value is null/undef.

If the filter needs to ensure that the value is defined you would need to add an additional defined condition to that filter.

20.4. Notification on certificates expiration in pfpki

If you use `pfpki` and you created PKI templates without email attribute, we recommend you to set a value for this attribute.

By doing this, `pfpki` will use email addresses defined in PKI templates to notify about next certificates expirations for certificates without emails.

21. Upgrading from a version prior to 12.0.0

21.1. Tenant code deprecated

The code used to manage tenants in PacketFence has been removed. If you previously used tenants in PacketFence, you should consider staying on a release prior to v12.

21.2. Clusters now use ProxySQL to load balance the DB connections

PacketFence previously used haproxy (via the haproxy-db service) to load balance and failover database connections from the PacketFence services to the database servers. This is now performed by ProxySQL which allows for splitting reads and writes to different members which offers greater performance and scalability.

If you suspect that using ProxySQL causes issues in your deployment, you can revert back to using haproxy-db by following these instructions

21.3. Bandwidth accounting is now disabled by default.

Tracking the bandwidth accounting information is now disabled by default. If you rely on bandwidth reports, security events or online/offline then enable it by doing the following. Go to *Configuration → System Configuration → RADIUS → General* Then enable 'Process Bandwidth Accounting'. `pfacct` service needs to be restarted to apply changes.

21.4. Fix permissions and checkups deprecated

API calls used to fix permissions and to perform checkups from web admin have been deprecated. With the containerization of several services, it didn't make sense to keep them available.

However, it's still possible to perform these commands on a PacketFence server using `pfcmd fixpermissions` and `pfcmd checkup`.

21.5. Change of behavior for the RADIUS source NAS-IP-Address

NOTE

This applies to administrators that have a RADIUS authentication source configured in PacketFence. If you are using PacketFence as a RADIUS server but do not have any RADIUS authentication source configured, this section does not apply to you.

RADIUS authentication sources previously used the source IP of the packet in the NAS-IP-Address field when communicating with the RADIUS server. This behavior has been deprecated

in favor of using the management IP address (or VIP in a cluster) in the NAS-IP-Address. If you do need to use another value in the NAS-IP-Address attribute, it is configurable in the RADIUS authentication source directly.

21.6. Log files names updated

The name of some log files have changed. You can find a list below:

Table 1. Mapping between old and new log files

Service	Old log file(s)	New log file(s)
MariaDB	mariadb_error.log	mariadb.log
httpd.aaa (Apache requests)	httpd.aaa.access and httpd.aaa.error	httpd.apache
httpd.collector (Apache requests)	httpd.collector.log and httpd.collector.error	httpd.apache
httpd.portal (Apache requests)	httpd.portal.access, httpd.portal.error, httpd.portal.catalyst	httpd.apache
httpd.proxy (Apache requests)	httpd.proxy.error and httpd.proxy.access	httpd.apache
httpd.webservices (Apache requests)	httpd.webservices.error and httpd.webservices.access	httpd.apache
api-frontend (Apache requests)	httpd.api-frontend.access	httpd.apache
HAProxy (all services)	/var/log/syslog or /var/log/messages	haproxy.log

21.7. Remote database backups

The ability to backup a remote database configured in PacketFence has been deprecated. From now on, a dedicated tool on the database server itself must be used to backup the external database. If your database is hosted on the PacketFence server (default behavior), then no adjustment is required for this.

22. Upgrading from a version prior to 12.1.0

22.1. configreload deprecated on pfcmd service pf restart

configreload call has been deprecated on pfcmd service pf restart due to a file synchronisation issue on each restart. If you modify a config file directly from the filesystem then you have to do the configreload manually.

```
/usr/local/pf/bin/pfcmd configreload hard
```

23. Archived upgrade notes

23.1. Upgrading from a version prior to 4.0.0

Upgrading an old version of PacketFence to v4 will be quite an endeavor. While it's entirely possible if done meticulously, we suggest you start from scratch and move your customizations and nodes information over to your new installation.

23.1.1. Database schema update

The temporary password table has been extended to include roles information. Moreover, an "admin" user is now automatically created. The default password is also "admin". Finally, a new table has been added for saved searches in the new Web administrative interface.

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-3.6.1-4.0.0.sql
```

23.1.2. Other important changes

PacketFence v4 received a major overhaul, especially regarding the authentication sources. Authentication modules found in `conf/authentication/` are no longer being used and have been replaced by the `conf/authentication.conf` file. While this file can be hand-edited, you should create your authentication sources and perform roles-mapping using the Configuation > Users > Sources page from PacketFence's Web administrative interface.

Also, in PacketFence v4, the VLANs can be assigned in `conf/switches.conf` by constructing the parameter names from the VLAN names and the `Vlan` suffix. The VLAN names must match one of the default names (registration, isolation, macDetection, inline, and voice) or one of the defined roles. If you were using custom VLANs, you must create a new role per VLAN and assign them accordingly.

Other key changes were done, such as:

- moved remediation templates in `html/captive-portal/templates/violations` and converted them to Template Toolkit
- dropped guests_admin_registration.category
- dropped guests_self_registration.access_duration
- dropped guests_self_registration.category
- dropped guests_self_registration.sponsor_authentication
- dropped guests_self_registration.sponsors_only_from_localdomain
- dropped ports.listeners
- dropped registration.auth and registration.default_auth
- dropped registration.maxnodes

- dropped registration.expire_* and registration.skip_*
- dropped trapping.blacklist
- dropped support for resetVlanAllPort in `bin/pfcmd_vlan`
- dropped `sbin/pfredirect` binary
- splitted the httpd services in three: httpd.admin, httpd.portal and httpd.webservices
- domain-name is no longer required in each section of networks.conf

For all parameters related to authentication (categories, access duration, sponsor authentication, etc.), you should now set proper actions in the `conf/authentication.conf` file.

Finally, the `pf` must be sudoer access to the `/sbin/ip` (and others) binary. As root, please do:

```
echo "pf ALL=NOPASSWD: /sbin/iptables, /usr/sbin/ipset, /sbin/ip,
/sbin/vconfig, /sbin/route, /sbin/service, /usr/bin/tee,
/usr/local/pf/sbin/pfdhcplistener, /bin/kill, /usr/sbin/dhcpd,
/usr/sbin/radiusd" >> /etc/sudoers
```

23.2. Upgrading from a version prior to 4.0.1

This release only fixes various bugs and doesn't need the database schema to be modified. Simply update the file /usr/local/pf/conf/currently-at to match the new release number. === Upgrading from a version prior to 4.0.2

This release only fixes various bugs and doesn't need the database schema to be modified. Simply update the file /usr/local/pf/conf/currently-at to match the new release number.

LDAP SSL and STARTTLS is now correctly implemented. Make sure the server you specify in authentication.conf supports the encryption type requested on the port configured. Failure to do so will break LDAP and Active Directory authentication.

23.3. Upgrading from a version prior to 4.0.3

You need to downgrade the version of perl-Net-DNS and perl-Net-DNS-Nameserver to version 0.65-4 in order to fix the issue with pfdns crashing.

23.4. Upgrading from a version prior to 4.0.4

The parameter guest_self_reg in the profiles.conf file is no longer necessary. The self-registration is now automatically enabled if at least one external authentication source is selected (Email, SMS, SponsorEmail, or Oauth2).

23.5. Upgrading from a version prior to 4.0.5

This release adds a new dependency on the Perl module Apache::SSLLookup. Once installed, update the file /usr/local/pf/conf/currently-at to match the new release number.

23.6. Upgrading from a version prior to 4.0.6

23.6.1. Changes to authentication API

The method pf::authentication::authenticate now expects an array of pf::authentication::Source objects instead of an array of source IDs.

The methods getSourceByType, getInternalSources, and getExternalSources of the module pf::Portal::Profile now return pf::authentication::Source objects instead of source IDs.

23.7. Upgrading from a version prior to 4.1.0

23.7.1. Database schema update

The category column in the temporary_password should not be mandatory.

Also, the access_level of the temporary_password table is now a string instead of a bit string.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.0.0-4.1.0.sql
```

23.7.2. Configuration changes

The parameters `trapping.redirecturl` and `trapping.always_use_redirecturl` from `pf.conf` (or `pf.conf.defaults`) were moved to the default portal profile in `profiles.conf`.

The parameter `registration.range` has been deprecated. Make sure you remove it from your configuration file.

The action `set_access_level` of authentication sources in `authentication.conf` must now match one of the admin roles defined in `adminroles.conf`. The previous level `4294967295` must be replaced by **ALL** and the level `0` by **NONE**.

Adjust your configuration files accordingly.

Once the configuration completed, update the file /usr/local/pf/conf/currently-at to match the new release number.

23.8. Upgrading from a version prior to 4.2.0

23.8.1. Database schema update

The person table has many new columns that can be used for registration.

The node table has new columns to store the time and bandwidth balances of a node.

The node table has also a new column to keep the audit-session-id from the RADIUS request to use with the CoA.

Added a new column config_timestamp in radius_nas table.

The locationlog table has new columns to store the switch IP and MAC when using dynamic controllers.

New table for inline (layer 3) accounting.

New table for WRIX data.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.1.0-4.2.0.sql
```

23.8.2. Configuration changes

The parameter `guests_self_registration.mandatory_fields` from `pf.conf` (or `pf.conf.defaults`) was moved to the default portal profile in `profiles.conf`.

The parameters `registration.gaming_devices_registration` and `registration.gaming_devices_registration_role` are replaced with `registration.device_registration` and `registration.device_registration_role`.

Adjust your configuration files accordingly.

The captive portal has been rewritten using the Catalyst MVC framework. Any customization to the previous CGI scripts will need to be ported to the new architecture.

Once the configuration completed, update the file /usr/local/pf/conf/currently-at to match the new release number.

23.9. Upgrading from a version prior to 4.3.0

23.9.1. Database schema update

The person table has 2 new column to keep the portal and the source used to authenticate.

The tables email_activation and sms_activation have been merged in a table named `activation`. It has an additional column to keep the portal used to register.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.2.0-4.3.0.sql
```

23.9.2. Configuration changes

The parameters `VlanMap` and `RoleMap` have been added in `switches.conf`; be sure to add them in the [default] switch section.

The OAuth passthroughs will not be activated unless `trapping.passthrough` in `pf.conf` is enabled. Make sure you enable it if you have OAuth authentication sources (Google, Facebook, Github, LinkedIn and Windows Live).

Once the configuration is completed, update the file /usr/local/pf/conf/currently-at to match the new release number.

23.10. Upgrading from a version prior to 4.4.0

23.10.1. Database schema update

Introduced the 'iplog_history' table for easier cleanup of the existing 'iplog' table.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.3.0-4.4.0.sql
```

23.10.2. Cache serialization

The serialization of the objects in the cache changed, making all the previous cached objects invalid. With PacketFence completely stopped do :

```
rm -fr /usr/local/pf/var/cache/*
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 4.4.0).

23.11. Upgrading from a version prior to 4.5.0

23.11.1. Database schema update

The class table has a new column delay_by.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.4.0-4.5.0.sql
```

23.11.2. Violation configuration

A new parameter 'delay_by' has been introduced in the violation configuration. Make sure to add the following to the 'defaults' section of 'conf/violations.conf' to avoid any problem.

delay_by=0s

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 4.5.0).

23.12. Upgrading from a version prior to 4.6.0

23.12.1. Database schema update

The locationlog and locationlog_history table have 2 new columns stripped_user_name and realm. We added new INDEX on iplog, violation and locationlog tables.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.5.0-4.6.0.sql
```

23.12.2. Violation template pages language handling

Code to match violation template pages have been reworked. Make sure to lowercase FR to fr in french template files name.

23.12.3. Realm configuration

Realm are now managed by Freeradius server so if your users authenticate with a username like username@acme.com then add the realm acme.com in the Radius Realms configuration menu and in your Active Directory source select 'Use stripped username'.

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 4.6.0).

23.13. Upgrading from a version prior to 4.7.0

23.13.1. Database schema update

The 'node' table has a new column (machine_account).

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.6.0-4.7.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 4.7.0).

23.14. Upgrading from a version prior to 5.0.0

Upgrading a version of PacketFence older than 4.1 to v5 will be a complex undertaking. While it's entirely possible if done meticulously, we suggest you start from scratch and move your customizations and nodes information over to your new installation.

Please note that the sections below are cumulative. That is to say, if you are upgrading from version 4.3 to version 5.0 you must apply in order all changes in between the two versions, including database schema changes.

As always, taking a complete backup of your current installation is strongly recommended. A backup should contain a copy of all PacketFence files as well as a copy of the database. You can take a backup of the pf directory with the following command:

```
tar -C /usr/local -czf /root/packetfence.tar.gz pf
```

A backup of the database can be taken using the procedure described in the next section.

23.14.1. Database schema update

Before making any changes to your database, ensure that you have a backup. A complete database backup can be taken using this command:

```
mysqldump --opt --routines -u root -p pf | gzip > /root/packetfence_db.sql.gz
```

If your database is more than a few hundred megabytes, you may also want to consider using a tool such as Percona XtraBackup which makes for much faster restores than mysqldump.

Multiple changes have been made to the database schema. You will need to update it accordingly. Since we will be dropping and recreating the 'iplog' table it is essential that you have a backup if you need the data it contains.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-4.7.0-5.0.0.sql
```

23.14.2. Configuration changes

You must manually enter the MySQL password of the pf user in the conf/pfconfig.conf file. The MySQL password is saved in the conf/pf.conf file under the [database] section. Copy the following from conf/pf.conf to conf/pfconfig.conf:

```
pass=$YOURPASSWORDHERE
```

23.14.3. Violations configuration

The violation triggers have been reworked for the new Fingerbank integration. We highly suggest you copy `conf/violations.conf.example` over `conf/violations.conf` and then reconfigure any violations you had before.

Also, make sure you adjust the following triggers to their new ID (Can be found under 'Configuration→Fingerbank'):

- `USERAGENT` becomes `user_agent`
- `VENDORMAC` becomes `mac_vendor`

The `OS` trigger has been deprecated over the new `dhcp_fingerprint` trigger. You will need to adjust these triggers to the new ids as well as renaming them.

23.14.4. iptables changes

The iptables configuration file doesn't use the generated rules '%%input_mgmt_guest_rules%%' anymore. Make sure you remove this line from conf/iptables.conf.

Also a lot of additions were made to the iptables configuration file. Make sure you add the new rules in conf/iptables.conf.example to your existing iptables file or execute the following command to replace the whole file.

```
cp /usr/local/pf/conf/iptables.conf.example /usr/local/pf/conf/iptables.conf
```

23.14.5. Using EAP local authentication

If you are using EAP MS-CHAP local authentication, meaning your 802.1x connections authenticate against your local database, you will need to make sure you deactivate password encryption in the database. In the administration interface, go in 'Configuration → Advanced' and set 'Database passwords hashing method' to `plaintext`

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.0.0).

23.15. Upgrading from a version prior to 5.1.0

23.15.1. Database schema update

Multiple changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.0.0-5.1.0.sql
```

23.15.2. pfsetvlan and snmptrapd

These two services have been disabled by default. If you are using SNMP traps enforcement on your switches (like port-security), make sure you re-enable them in 'Configuration→Services'.

23.15.3. Active Directory domain join

The Microsoft Active Directory domain join configuration is now part of PacketFence. A migration script has been made so you can migrate an existing domain join into this configuration. Note that this step is not mandatory, as the old join method is still supported. But if you do not perform this step, you will not see its configuration from the PacketFence web administrative interface.

Simply execute the following script and follow its instructions `/usr/local/pf/addons/AD/migrate.pl`

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.1.0).

23.16. Upgrading from a version prior to 5.2.0

23.16.1. Database schema update

Multiple changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.1.0-5.2.0.sql
```

23.16.2. Database monitoring host

If you are using an Active/Active cluster, you will need to adjust the monitoring database host to point to your database as it is not forced anymore.

In `conf/pf.conf` :

```
[monitoring]
db_host=127.0.0.1
```

23.16.3. New 'portal' interface type

If you are using email registration, web-auth enforcement (external captive-portal), device registration feature, or anything that would require to access the captive portal from outside the registration/isolation VLANs, you might want (actually, you need otherwise it will no longer works!) to add the 'portal' type to the existing 'management' interface.

In `conf/pf.conf` :

```
[interface eth42]
type=management,portal
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.2.0).

23.17. Upgrading from a version prior to 5.3.0

23.17.1. Database schema update

Changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.2.0-5.3.0.sql
```

23.17.2. Debian and Ubuntu

A downgrade in a package version may cause an error when trying to upgrade.

If you receive this error:

```
The following packages have unmet dependencies:
 packetfence : Depends: libhtml-formhandler-perl (= 0.40013-2) but 0.40050-2 is
to be installed
E: Unable to correct problems, you have held broken packages.
```

Run the following commands:

```
# dpkg -r --ignore-depends=packetfence   libhtml-formhandler-perl
# apt-get install  libhtml-formhandler-perl  libtemplate-autofilter-perl
libmoo-perl
# apt-get install packetfence packetfence-config packetfence-pfcmd-suid
libdist-checkconflicts-perl libimport-into-perl
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.3.0).

23.18. Upgrading from a version prior to 5.4.0

23.18.1. Database schema update

Changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.3.0-5.4.0.sql
```

23.18.2. Authentication sources rules rework

Authentication sources rules have been reworked in a way to differentiate an 'authentication' rule and an 'administration' rule. Codewise, that means that codeflow will look into specific types of rules depending of the use case.

Please take a minute or two to go through the existing rules for each of the authentication sources and make sure there is no 'administration' class actions into an 'authentication' class rule and vice versa, otherwise the "invalid" action will be ignored.

Authentication sources rules structure is as follow:

- 'authentication' rule class available actions:
  - Set role (set_role)
  - Set access duration (set_access_duration)

    - Set unregistration date (set_unreg_date)
- 'administration' rule class available actions:
    - Set access level of Web admin (set_access_level)
    - Mark as sponsor (mark_as_sponsor)

For example, if an existing rule is as follow:

- Name: AllAdmins
- Class: No class defined since the class attribute is new
- Conditions: ...
- Actions:
    - Set access level of Web admin → ALL
    - Set role → default
    - Set access duration → 24H

That existing rule will default to the 'authentication' class if none is being set. If that's the case, the first action "Set access level of Web admin" will then be ignored.

To replicate that existing rule with the new classes, you would have to create two separate rules, as follow:

Rule for 'administration' purposes

- Name: AllAdmins_admin
- Class: administration
- Conditions: ...
- Actions:
    - Set access level of Web admin → ALL

Rule for 'authentication' purposes

- Name: AllAdmins_auth
- Class: authentication
- Conditions: ...
- Actions:
    - Set role → default
    - Set access duration → 24H

Configuration will be validated on every start / restart so that "bogus" authentication sources / rules can be identified.

23.18.3. OAuth2 authentication sources changes

The Facebook API now requires to specify the fields to be defined in the query. In all your facebook sources, change the parameter protected_resource_url to https://graph.facebook.com/me?fields=id,name,email,first_name,last_name

Change the parameter scope to user,user:email in all your Github sources as PacketFence is now fetching the email address of the user when registering with Github.

23.18.4. StatsD configuration changes

monitoring.statsd_host and monitoring.statsd_port have been removed from pf.conf. If you have specified a specific host or port, remove them from your configuration and change them in /usr/local/pf/lib/pf/StatsD.pm

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.4.0).

23.19. Upgrading from a version prior to 5.5.0

23.19.1. Database schema update

Changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.4.0-5.5.0.sql
```

23.19.2. VLAN Filter configuration changes

The VLAN filter has been reworked to use a more generalized syntax to allow more complex filters to be created.

This mean nested conditions no longer need to specify the attribute in the condition.

So the following attribute

```
[condition]
filter=node_info
attribute=category
operator=is
value=default
```

Should be rewritten as

```
[condition]
filter=node_info.category
operator=is
value=default
```

The older syntax is still supported but will be deprecated in a future release.

The operators match and match_not has changed their behavior. They will match (or not match) the exact string given in the condition. The following condition

```
[condition]
filter=node_info.computername
operator=match
value=^Bob
```

Will match node_info.computername only if it contians '^Bob'. It will not match if node_info.computername start with 'Bob'

If you need to use a regex then use the regex/regex_not operator. So the following condition should be changed from

```
[condition]
filter=node_info.mac
operator=match
value=^00:
```

To the following

```
[condition]
filter=node_info.mac
operator=regex
value=^00:
```

23.19.3. pf.conf configuration file changes

The following parameters have been removed from pf.conf. Make sure to remove them from your file if configured.

- alerting.wins_server
- alerting.admin_netbiosname

23.19.4. violations.conf configuration file changes

Violations have been reworked and configuration changes are necessary in order to maintain functionnality.

In violations.conf the following actions have been renamed, please update them accordingly.

- trap → reevaluate_access
- email → email_admin

The following action have been removed from the violations :

- popup

Also in violations.conf, the parameter whitelisted_categories has been renamed into whitelisted_roles

23.19.5. Billing configuration change

The parameter `billing_engine` of the Portal Profiles has been deprecated. Remove it from all your profiles configuration in `/usr/local/pf/conf/profiles.conf`.

The billing engine of PacketFence has been reworked completely.

It will require to reconfigure existing billing providers from scratch as there is no retro-compatibility with the previous configuration.

Please see the Administration Guide for details on how to configure the billing engine.

23.19.6. Mod_qos configuration changes

Mod_qos configuration has been moved from "services" to "captive_portal" section. Make sure to apply the appropriate changes if needed.

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.5.0).

23.20. Upgrading from a version prior to 5.6.0

23.20.1. Database schema update

Changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.5.0-5.6.0.sql
```

23.20.2. Extension points changes

The file lib/pf/vlan/custom.pm has now been renamed to lib/pf/role/custom.pm. Most of the customizations that used to be made in vlan/custom.pm can now be handled by configuring a vlan filter. You should take a good look at your existing vlan/custom.pm and consider porting the changes to conf/vlan_filters.conf.

23.20.3. VLAN filters changes

The scopes for the VLAN filters have changed. The following have been renamed according to these rules:

NormalVlan → RegisteredRole RegistrationVlan → RegistrationRole ViolationVlan → ViolationRole InlineVlan → InlineRole

If you have defined any filters in /usr/local/pf/conf/vlan_filters.conf, make sure to rename all references to the left hand side with the new names on the right hand side.

23.20.4. Default type for the switches

The default type for the switches now needs to be set explicitly. Add the following line in the

default section of `/usr/local/pf/conf/switches.conf`

```
type=Generic
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.6.0).

23.21. Upgrading from a version prior to 5.7.0

23.21.1. Suricata violation trigger renaming

With the introduction of the ability to trigger a violation based on a MD5 hash detected by Suricata, a new trigger type has been introduced, requiring the modification of the actual 'suricata' trigger. Make sure to go through your violations configuration and change any 'suricata' trigger name for 'suricata_event'.

23.21.2. Database schema update

Changes have been made to the database schema. You will need to update it accordingly.

Make sure you run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.6.0-5.7.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 5.7.0).

23.22. Upgrading from a version prior to 6.0.0

Upgrading PacketFence from a version older than v6.0.0 will be a complex undertaking. While it's entirely possible if done meticulously, we suggest you start from scratch and move your customizations and nodes information over to your new installation.

23.22.1. Devices parking

The new registration devices parking requires that you add the following violation in `/usr/local/pf/conf/violations.conf`

```
[1300003]
priority=1
desc=Parking violation
max_enable=3
grace=10m
actions=log,reevaluate_access
enabled=Y
auto_enable=Y
vlan=registration
trigger=Internal::parking_detected
```

23.22.2. Chained authentication

The chained source has been deprecated in favor of a fully customizable flow in the captive portal.

Make sure you delete the source **BEFORE** upgrading your installation.

Once you upgrade, configure a portal module for each of your sources and a chained one that contains both. Refer to the administration guide for a detailed example.

23.22.3. Redesigned captive portal

The parameter `mandatory_fields` of the Portal Profiles has been deprecated. Remove it from all the profiles in `profiles.conf`

To configure mandatatory fields in the portal, refer to the 'Portal Modules' section of the Administration guide

You need to add the `root_module` parameter to your default portal profile. In `profiles.conf` add `root_module=default_policy` to the default portal profile

23.22.4. Changes to OAuth2 sources callback URL

All the OAuth2 sources you have configured (Facebook, Github, Google, LinkedIn ,Twitter, Windows Live) need to be adjusted as the redirect URL is now the same for all the types.

In the admin interface change `Portal URL` from `https://YOUR_HOSTNAME/oauth2/SOURCE_TYPE` to `https://YOUR_HOSTNAME/oauth/callback` (where `SOURCE_TYPE` would be the lower case name of the source type). Note that this parameter is named `redirect_url` in the configuration file.

23.22.5. Changes to Cisco Web auth

Use the Cisco::Catalyst_2960 switch module instead of the Cisco::Catalyst_2960_http as switch type.

Use the Cisco::WLC switch module instead of the Cisco::WLC_http as switch type.

The portalURL configuration parameter is now configured per-role so make sure you have `http://ip_portal/$session_id` assigned to the registration role in the `Role by Web Auth URL` section of the switch configuration.

See the Network Device configuration guide for additional details.

23.22.6. SMS carrier database table

Google Project Fi have been added as a supported carrier. Since an ID is hardcoded on creation of a new entry in the 'sms_carrier' database table, a manual intervention may be required in the case the database schema update fails.

23.22.7. pf.conf configuration parameters

'expire' and 'maintenance' section have been reworked and 'expire' section is no longer a thing. Make sure to adjust configuration parameter accordingly if needed;

- expire.node is now maintenance.node_cleanup_window
- expire.iplog is now maintenance.iplog_cleanup_window
- expire.locationlog is now maintenance.locationlog_cleanup_window
- expire.radius_audit_log is now maintenance.radius_audit_log_cleanup_window
- expire.traplog is now maintenance.traplog_cleanup_window

23.22.8. node category / role

The 'REJECT' role is now a default standard role. If you already have such role, make sure no conflict exists.

Also, add the following line to the default section of `switches.conf` :

```
REJECTVlan = -1
```

23.22.9. Changes to the generated smb.conf

If you have a domain configured directly in PacketFence (in 'Configuration→Domains'), you need to re-generate the associated configuration files as changes have been made to the samba configuration.

Using the CLI `/usr/local/pf/bin/pfcmd generatedomainconfig` or in the admin interface in 'Configuration→Domains', click 'Refresh domain configuration'

23.22.10. Upgrade from FreeRADIUS 2 to FreeRADIUS 3

PacketFence 6 relies on FreeRADIUS 3 rather that FreeRADIUS 2 as provided in PacketFence 5. The configuration files, directory layout and "unlang" directives have changed significantly. The packaging will automatically rename the existing raddb directory to raddb-pre6. All your existing configuration and certificates (if stored under raddb/certs) should be preserved but may need to be merged with the new raddb directory layout if you customized them.

The configuration files under conf/radiusd/**.example have also changed. Make sure to compare them to your conf/radiusd/** files if you have any customizations, and merge any *.rpmnew files that may have been created by the packaging.

The default location for the FreeRADIUS server certificates has changed from conf/ssl/ to raddb/certs/. The configuration of the certificates location is in conf/radiusd/eap.conf. You may point it to any valid certificate and key by setting the value of `certificate_file` and `private_key_file` respectively. It is not recommended to use the same server certificate for the HTTP services and the RADIUS server as the requirements for each are different. Reusing the same certificate will work, but you would be well advised to consider separate certificates.

Finally, the database schema for the RADIUS accounting tables and stored procedures have changed. Make sure to apply the database changes as described in the following section.

23.22.11. Database schema update

Significant changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 5.7 schema to 6.0.

Since the schema of the radacct table has been reworked, the script will rename the existing table to radacct2 and insert it's content into the new radacct table. If your existing radacct table is large (as is sometimes the case), the operation may take a long time and consume a significant amount of disk space. Make sure to have plenty of both before running the upgrade script.

You can estimate the size of the existing radacct table by running the following command:

```
mysql> SELECT table_name AS "Table", round(((data_length + index_length) / 1024
/ 1024), 2) "Size in MB" FROM information_schema.TABLES WHERE table_schema =
"pf" AND table_name = "radacct";
```

You should have at least twice as much space as that table uses in the filesystem on which the MySQL data directory is mounted (usually /var/lib/mysql).

If you do not have enough space or time, you may consider truncating the radacct table (or simply deleting some of the rows) before running the upgrade script.

When ready, run the following to update your schema:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-5.7.0-6.0.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.0.0).

You will also want to drop the radacct2 table from the database as it will no longer be needed.

23.23. Upgrading from a version prior to 6.1.0

Significant changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.0 schema to 6.1.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.0.0-6.1.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.1.0).

23.23.1. Dynamically created local secret

The management IP(s) of PacketFence are now defined as switches with a forced RADIUS secret defined in /usr/local/pf/conf/local_secret. Make sure you reconfigure the secret in the file if necessary and that this file is synchronized on all your cluster members if that applies. Note that this doesn't affect the RADIUS secret you have configured for wireless controllers and switches. It only affects RADIUS requests that originate from the management IP(s)

23.23.2. Changes to LinkedIn source

A change to the authorize URL of LinkedIn was made. Make sure to change the 'API Authorize Path' in all your LinkedIn source to `/uas/oauth2/authorization`.

23.24. Upgrading from a version prior to 6.2.0

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.1 schema to 6.2.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.1.0-6.2.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.2.0).

23.25. Upgrading from a version prior to 6.2.1

Changes have been made to the httpd.admin configuration. Make sure you copy the conf/httpd.conf.d/httpd.admin.tt.example file over conf/httpd.conf.d/httpd.admin.tt. If you customized that file in any way, you will have to merge the changes.

Restart the httpd.admin process once that is done by running /usr/local/pf/bin/pfcmd service httpd.admin restart

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.2.1).

23.26. Upgrading from a version prior to 6.3.0

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.2 schema to 6.3.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.2.0-6.3.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.3.0).

23.26.1. RADIUS configuration file changes

The following file: /usr/local/pf/conf/radiusd/eap.conf was modified to use TemplateToolkit, you will need to replace it by /usr/local/pf/conf/radiusd/eap.conf.example, make sure to re-edit the new file and add your certificate if needed.

23.26.2. Samba cache directory changed

Rejoining the domains from PacketFence GUI is required.

Go under `Configuration RADIUS Domains` and click `Rejoin` for each domain configured.

23.26.3. Configuration changes to the Provisioning and Scaning

The configuration of the Scan engines and the Provisioners has been reworked to use the Fingerbank device IDs in the OS matching. `scan.conf` and `provisioning.conf` need to be migrated to use the new values. A migration script should be run `# /usr/local/pf/addons/upgrade/to-6.3-os-rewrite.pl` to migrate the configuration. This will output the migrated configuration in `/usr/local/pf/conf/provisioning.conf.new` and `/usr/local/pf/conf/scan.conf.new`. First run the script and then validate that their content is fine. Once that is done, copy the files over the original ones using :

```
# cp /usr/local/pf/conf/provisioning.conf.new
/usr/local/pf/conf/provisioning.conf
# cp /usr/local/pf/conf/scan.conf.new /usr/local/pf/conf/scan.conf
# /usr/local/pf/bin/pfcmd configreload hard
```

23.26.4. Fingerbank database moving to MySQL (optionnal but highly suggested)

The Fingerbank database can now be hosted in the same MySQL database PacketFence uses.

In order to do so, you need to collect the database credentials from the PacketFence configuration:

```
# /usr/local/pf/bin/pfcmd pfconfig show resource::Database
$VAR1 = {
          'pass' => 'myPassword',
          'db' => 'pf',
          'user' => 'pf',
          'port' => '3306',
          'host' => 'localhost'
        };
```

Now, you need to create the database and assign the proper rights to the user by executing the following commands:

```
# mysql -u root -p -e "CREATE DATABASE pf_fingerbank"
# mysql -u root -p -e "GRANT ALL PRIVILEGES ON pf_fingerbank.* TO 'pf'@'%'
IDENTIFIED BY 'myPassword'"
# mysql -u root -p -e "GRANT ALL PRIVILEGES ON pf_fingerbank.* TO
'pf'@'localhost' IDENTIFIED BY 'myPassword'"
```

Replace `myPassword` by the password displayed (`pass`) when running the first command.

Next, head to 'Configuration→Fingerbank Settings' in the web administration interface and configure the following parameters:

- MySQL host : set this to the value of `host` you got from running the command above.
- MySQL port : set this to the value of `port` you got from running the command above.
- MySQL username : set this to the value of `user` you got from running the command above.
- MySQL password : set this to the value of `pass` you got from running the command above.
- MySQL database : set this to `pf_fingerbank`.

After saving those new parameters, at the top of the same page, click 'Initialize MySQL database' to start the import process. Once that is completed, you will receive an e-mail to the one configured for alerting and PacketFence will start using the MySQL backend for the Fingerbank database.

23.27. Upgrading from a version prior to 6.4.0

23.27.1. Database schema updates

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.3 schema to 6.4.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.3.0-6.4.0.sql
```

23.27.2. Changes to web authentication configuration

Rework of the external captive portal capabilities involves some significant changes in the switch modules configuration. Some switch modules have been moved to other ones and some others have been removed. Please adjust the configuration (type) accordingly within switches.conf.

- AeroHIVE::AP_http → AeroHIVE::AP
- Meraki::AP_http → Meraki::MR
- Meraki::AP_http_V2 → Meraki::MR_v2
- Xirrus:AP_http → Xirrus

To instruct a switch module to perform external captive portal enforcement, a new switch configuration parameter have been added. Make sure to adjust the following parameter to your needs in switches.conf

```
ExternalPortalEnforcement = Y
```

External captive portal URLs have also changed. Change them accordingly depending on the type of equipment you use:

- AeroHIVE: http://portal_IP/AeroHIVE::AP
- Aruba: http://portal_IP/Aruba

- Cisco Catalyst 2960: http://portal_IP/Cisco::Catalyst_2960
- Cisco WLC: http://portal_IP/Cisco::WLC
- CoovaChilli: http://portal_IP/CoovaChilli
- Meraki: http://portal_IP/Meraki::MR
- Ruckus: http://portal_IP/Ruckus
- Xirrus: http://portal_IP/Xirrus

Where portal_ip is the IP Address (or DNS name) of your captive portal as it was configured before

23.27.3. Changes to WMI

If you use WMI, you must modify conf/wmi.conf in order to make sure that a namespace parameter is defined for each rule. For example, you could have:

```
[SCCM]
request=select * from Win32_Process where name='CcmExec.exe'
action=[sccm]
namespace=ROOT\cimv2
on_tab=1
```

23.27.4. Changes to default cronjob

Upon PacketFence installation, a default cronjob will be in /etc/cron.d/. You should make sure you do not invoke the /usr/local/pf/addons/backup-and-maintenance.sh script from any other cronjob.

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.4.0).

23.28. Upgrading from a version prior to 6.5.0

23.28.1. Database schema updates

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.4 schema to 6.5.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.4.0-6.5.0.sql
```

23.28.2. Custom code warning

The method signature of pf::node::node_register has been modified. Make sure you adjust any custom code / external scripts to handle the new returned values.

23.28.3. Switches Configuration

You must rename "controllerPort" to "disconnectPort" in your switches.conf configuration file. You can automate this using:

```
cd /usr/local/pf
find . -name "switches.conf" -exec sed -i "s/controllerPort/disconnectPort/g"
'{}' \;
```

23.28.4. Eduroam

Eduroam authentication source is now an "exclusive" authentication source rather than an "external" one. That being said, make sure to adjust portal profile accordingly (an "exclusive" authentication source can be the only one configured in a portal profile).

23.28.5. Improved Logging

In order to be sure all your logging facilities use the new logging backend which ensures the processes will not die in case of a logging failure, you must execute the following command:

```
cd /usr/local/pf
find conf/log.conf.d/ -type f -exec sed -i.bak
"s/Log::Log4perl::Appender::File/pf::log::FileAppender/g" {} \; ; find
conf/log.conf.d/ -name '*.bak' -delete
```

23.28.6. Email templates

The email templates have been moved from /usr/local/pf/conf/emails/ to /usr/local/pf/html/captive-portal/templates/emails/ as they are now configurable by portal profile. Also you can configure the language in which PacketFence should send emails to the administrator in the Advanced section of the configuration.

Make sure you run the following command after upgrading:

```
/usr/local/pf/bin/pfcmd cache configfiles clear
```

23.28.7. Violations

When whitelisting roles in a violation, the registration role will now match unregistered devices where before it would never match. Make sure to go through violations that may include this role to make sure it is relevant.

23.28.8. Database schema updates

The "configfile" and "traplog" database tables are now deprecated. If you wish to reclaim the disk space used by those two database tables, they should be manually removed.

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 6.5.0).

23.28.9. Default RoleMap for the switches

If you were using the default 'RoleMap = Y' in the conf/switches.conf it's disabled by default now. You will need to put 'RoleMap = Y' under your switches or switch group configuration.

23.29. Upgrading from a version prior to 7.0.0

NOTE | You cannot upgrade from CentOS 6 or Debian Wheezy to PacketFence 7.0 and above

23.29.1. Debian upgrade

The requirement for MariaDB 10.1 means that a simple "apt upgrade" will not be enough. You will need to help apt through the upgrade by manually removing some packages and installing some others. The need to ensure you have backups cannot be overstated.

Make sure the apt database is up to date

```
apt update
```

Remove the MySQL 5.5 packages (do not purge them, as that would delete the database)

```
dpkg -r --force-all mysql-client-5.5 mysql-common mysql-server mysql-server-5.5
mysql-server-core-5.5 libmysqlclient18
```

Install the newer Mariadb-10.1 packages

```
apt install libmariadbclient18 libmysqlclient18 mariadb-common mariadb-server-
10.1 galera-3 gawk mariadb-client-10.1 mariadb-server-core-10.1 rsync socat
libmpfr4 mariadb-client-core-10.1 mysql-common
```

Finally, upgrade the rest of the packages

```
apt full-upgrade
```

Note that "full-upgrade" may also affect other packages you might have installed on the system if you had other software than PacketFence on it.

23.29.2. MariaDB upgrade (CentOS + RHEL only)

Upgrading to PacketFence 7+ will install a more recent version of MariaDB than the one that is shipped with CentOS.

In order to upgrade the MariaDB metadata files and tables, first stop any started process.

```
systemctl stop mariadb
systemctl stop packetfence-mariadb
```

Then start a mysqld_safe process manually (this will start a background process)

```
mkdir -p /var/run/mariadb
chown mysql: /var/run/mariadb
mysqld_safe --basedir=/usr &
```

Then, execute the upgrade script and enter the root password when prompted

```
mysql_upgrade -u root -p
```

When done, kill the mysqld_safe process we started before the update, reattach to it and wait for it to exit

```
kill %1 && fg
```

Note that it might take up to a few minutes for the process to exit depending on the size of your database.

Once done, restart the MariaDB service (managed by PacketFence)

```
systemctl start packetfence-mariadb
```

23.29.3. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 6.5 schema to 7.0.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-6.5.0-7.0.0.sql
```

23.29.4. Systemd integration

All PacketFence services are managed individually via systemd unit files instead of one unit file (`packetfence.service`). When you updated the PacketFence package, it already set the system target to `packetfence.target`.

If you are hosting the MySQL/MariaDB service on your PacketFence servers (it is by default), you should now manage the service via `packetfence-mariadb.service` instead of `mariadb.service`.

The changes in the server startup will be done automatically by the packaging.

23.29.5. Admin roles configuration

Given the portal profiles have now been renamed to connection profiles, you need to make sure any admin role that allowed portal profile Create/Read/Update/Delete operations is updated to be valid for connection profiles.

```
cd /usr/local/pf
sed -i "s/PORTAL_PROFILE/CONNECTION_PROFILE/g" conf/adminroles.conf
```

23.29.6. PacketFence configuration

Multiple parameters inside `pf.conf` have been renamed for better clarity. Execute the following in order to migrate the parameters.

```
/usr/local/pf/addons/upgrade/to-7.0-pf-conf-changes.pl
```

23.29.7. Maintenance configuration

Maintenance related configuration for pfmon has been moved to a dedicated configuration file (`/usr/local/pf/conf/pfmon.conf`).

In order to migrate your settings from `pf.conf` to `pfmon.conf`, run the following script:

```
/usr/local/pf/addons/upgrade/to-7.0-pf.conf-to-pfmon.conf.pl
```

23.29.8. DHCP filters configuration

Minor changes were made to the DHCP filters configuration (`/usr/local/pf/conf/dhcp_filters.conf`).

First, the `computer_name` attribute was renamed to `computername` to be consistent with the rest of the application. Then, the `DhcpFingerbank` scope was changed to `Fingerbank`

In order to rename those in an automated way:

```
cd /usr/local/pf
sed -i "s/computer_name/computername/g" conf/dhcp_filters.conf
sed -i "s/DhcpFingerbank/Fingerbank/g" conf/dhcp_filters.conf
```

23.29.9. Roles configuration

The source of truth for roles is now in a configuration file (`/usr/local/pf/conf/roles.conf`) instead of being in the database. In order to pull the existing roles from your database into the configuration file, execute the following command:

```
/usr/local/pf/addons/upgrade/to-7.0-roles-conf.pl
```

NOTE

The roles still exist in the database like before (node_category table), but their source of truth is now in the configuration file. Should you remove a role manually from `roles.conf`, it will **not** be removed from the database unless you manually go delete it from the database.

23.29.10. pfdetect configuration

New parameters have been introduced in `conf/pfdetect.conf`. Run the following script to migrate your configuration.

```
/usr/local/pf/addons/upgrade/to-7.0-pfdetect-conf.pl
```

23.29.11. LinkedIn Source changes

If you are using the LinkedIn OAuth2 source, a change has been made on their API, thus you will need to do the following:

```
cd /usr/local/pf
sed -i "s/uas\/oauth2/oauth\/v2/g" conf/authentication.conf
```

23.29.12. Logging service

Since all logging now goes through rsyslog, if you had edited the logging configuration (e.g. to forward logs to a centralized syslog server) make sure that the new logging rules in `/etc/rsyslog.d/packetfence.conf` do not conflict with your changes.

Take a look at `/usr/local/pf/conf/log.conf` and `/usr/local/pf/conf/log.conf.d/*` for the detailed configuration of the PacketFence services.

23.29.13. Redis Queue

Clear the redis queue to avoid old stale jobs from being processes.

```
systemctl start packetfence-redis_queue
redis-cli -p 6380 FLUSHALL
systemctl stop packetfence-redis_queue
```

23.29.14. SSL certificates

Given that haproxy is now the termination point for the captive portal, any SSL configuration you have in `/usr/local/pf/conf/httpd.conf.d/ssl-certificates.conf` needs to be ported so that it works with haproxy.

Easiest solution is to bundle your server cert, your intermediates (if any) along with the key in the

default file used by the PacketFence haproxy process (/usr/local/pf/conf/ssl/server.pem)

In order to do so:

```
# cd /usr/local/pf/
# cat /path/to/your/server.crt /path/to/your/intermediates.crt
/path/to/your/server.key > /usr/local/pf/conf/ssl/server.pem
```

23.29.15. Running 7.0+ in a cluster

A complete re-visit of the database clustering stack was done in version 7.0. If you run your PacketFence installation in a cluster, make sure you read the following section.

23.29.16. Active/Active clusters with Active/Passive DB (default before 7.0)

We highly suggest you migrate your existing clustered installation using Corosync/Pacemaker to the new cluster stack of PacketFence that uses MariaDB Galera cluster. The easiest way to perform this is to build new servers and port your configuration (by copying the configuration files) and your database (using mysqldump). There are ways to migrate the 2 existing nodes to a 3 nodes cluster but this is not covered in this guide.

Corosync adjustment

Note that you can safely keep your existing 2-node cluster with Corosync/Pacemaker in place and things will work like before. You will simply have to adjust your Corosync configuration so that MariaDB points to the packetfence-mariadb file instead of the mariadb unit.

```
primitive MariaDB systemd:packetfence-mariadb \
        op start timeout=60s interval=0 \
        op stop timeout=60s interval=0 \
        op monitor interval=20s timeout=30s
```

Disabling Galera cluster

You must then disable the MariaDB Galera cluster as a replication mechanism as you will still be using DRBD. In order to do so, add the following in /usr/local/pf/conf/pf.conf

```
[active_active]
galera_replication=disabled
```

IP address bind

You must also instruct packetfence-mariadb to bind to the management IP address of the server manually.

In order to do so, replace the following section in /usr/local/pf/conf/mariadb/mariadb.conf.tt:

```
[% IF server_ip.length %]
bind-address=[% server_ip %]
[% ELSE %]
skip-networking
bind-address=
[% END %]
```

with: bind-address=1.2.3.4

Where 1.2.3.4 is the management IP address of the server.

Disable packetfence-mariadb on boot

Like in previous versions where mariadb shouldn't have been started on boot, now you must ensure its replacement (packetfence-mariadb) doesn't start on boot.

```
systemctl disable packetfence-mariadb
```

Enabling the packetfence-cluster target

Next, you must set the default target to packetfence-cluster:

```
systemctl set-default packetfence-cluster.target
```

23.29.17. Active/Active clusters with external DB

No changes to your clustering stack is required when using an external database.

23.29.18. Active/Passive clusters

CAUTION	You shouldn't be running active/passive clusters anymore. If you do, you're pretty much on your own for community support. Inverse provides professionnal services to help you maintain these clusters. If you intend to keep an active/passive cluster, we suggest you have deep knowledge of Corosync/Pacemaker and strong Linux skills.

First, no changes are required to your database stack as MariaDB supports being deployed in Active/Passive.

You will need to adjust the Corosync/Pacemaker configuration to take in consideration the changes made to systemd for PacketFence services. Before 7.0, PacketFence used to be controlled via a single systemd unit file while now it uses a multiple services grouped in targets. In order to mimic the single service behavior that was in previous versions, a unit file is provided here: https://github.com/inverse-inc/packetfence/blob/devel/packetfence-active-passive.service. You should install this file in `/etc/systemd/system/packetfence.service` and make sure there are no other leftovers of `packetfence.service` unit files on your system.

Then, you must adjust the systemd default target so PacketFence doesn't start on boot and note that this should be done on every future upgrade of your system.

```
# systemctl set-default multi-user.target
```

You should then change your Corosync configuration for MariaDB and PacketFence to the following:

```
primitive MariaDB systemd:packetfence-mariadb \
        op start timeout=60s interval=0 \
        op stop timeout=60s interval=0 \
        op monitor interval=20s timeout=30s
primitive PacketFence systemd:packetfence \
        op start timeout=300s interval=0 \
        op stop timeout=300s interval=0 \
        op monitor interval=300s timeout=300s
```

23.30. Upgrading from a version prior to 7.1.0

23.30.1. Multiple DNS servers per domain

The PacketFence Active Directory Domains integration now supports multiple DNS servers to be specified to find a DC. For this reason the parameter dns_server has been renamed to dns_servers in domain.conf. In order to automatically rename the parameters, run the following command:

```
sed -i.bak "s/^dns_server/dns_servers/g" /usr/local/pf/conf/domain.conf
```

23.30.2. Add default values to new auth source parameters

```
/usr/local/pf/addons/upgrade/to-7.1-authentication-conf.pl
```

23.30.3. Fix the Ubiquiti typo

In order to use the Ubiquiti switch module that has been renamed, run the following command:

```
sed -i.bak "s/Ubiquity/Ubiquiti/g" /usr/local/pf/conf/switches.conf
```

23.30.4. Instagram source changes

Due to a change in the API of Instagram please change the scope if you are using an Instagram OAuth2 source. Replace 'scope=email' by 'scope=basic' in conf/authentication.conf under the section '[Instagram Source]'.

23.30.5. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.0 schema to 7.1.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-7.0.0-7.1.0.sql
```

23.31. Upgrading from a version prior to 7.2.0

23.31.1. Ability to «pin» a domain DC

PacketFence is now able to instruct Samba to «pin» a DC for authentication or use all of them. You should instruct Samba to connect to all domain controllers by adding the following to each of your domains in domain.conf:

```
sticky_dc=*
```

And then regenerate the domain configuration:

```
/usr/local/pf/bin/pfcmd fixpermissions
/usr/local/pf/bin/pfcmd configreload hard
/usr/local/pf/bin/pfcmd generatedomainconfig
```

23.31.2. Change to sponsor CC address

The CC address for sponsors is now BCC. In order to adjust the configuration, execute the following:

```
cd /usr/local/pf
sed -i "s/sponsorship_cc/sponsorship_bcc/g" conf/authentication.conf
```

23.31.3. Changes to authentication sources codebase

Any custom authentication sources forms and templates would need to be copied to the new location.

Templates /usr/local/pf/html/pfappserver/root/authentication/source/type/ → /usr/local/pf/html/pfappserver/root/config/source/type/

Forms /usr/local/pf/html/pfappserver/lib/pfappserver/Form/Config/Authentication/Source → /usr/local/pf/html/pfappserver/lib/pfappserver/Form/Config/Source

23.31.4. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.1 schema to 7.2.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-7.1.0-7.2.0.sql
```

23.32. Upgrading from a version prior to 7.3.0

23.32.1. Device Registration

You will need to remove anything related to [device_registration] in the conf/pf.conf file. Once done, you will need to reconfigure any device registration policy using the following instructions: https://packetfence.org/doc/PacketFence_Installation_Guide.html#_devices_registration

23.32.2. Changes to `authentication.conf` and `domain.conf` regarding realms and source matching

You have to run the following script to change the configuration:

```
/usr/local/pf/addons/upgrade/to-7.3-authentication-conf.pl
```

23.32.3. MariaDB database read-only mode

There was, in some cases, an issue where the database cluster was put in a read-only mode which then prevent it to comes back gracefully.

A modification have been made to now use the wsrep_ready state of the DB as a read only indicator. Therefore, PacketFence will stop putting the DB in read only on quorum + primary loss of MariaDB and trust wsrep_ready instead

Ensure you merge changes in the galera section of `conf/mariadb/mariadb.conf.tt.rpmnew` into `conf/mariadb/mariadb.conf.tt`

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 7.3.0).

23.32.4. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.2 schema to 7.3.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-7.2.0-7.3.0.sql
```

23.33. Upgrading from a version prior to 7.4.0

23.33.1. New LinkedIn domain list

If you use social login with LinkedIn OAuth2, you will need to adjust the list of domains that are passthroughs in the LinkedIn source.

For all your LinkedIn sources, change the domains to:

```
www.linkedin.com,api.linkedin.com,*.licdn.comlatform.linkedin.com
```

23.33.2. Portal redirection timer

The redirection timer configuration (length of the timer bar at the end of the portal) has been moved from the fencing section to the captive_portal section. More precisely, it has moved from `fencing.redirtimer` to `captive_portal.network_redirect_delay`.

23.33.3. Database schema update (all Linux distributions)

Changes have been made to the database schema. You will need to update it accordingly. An SQL upgrade script has been provided to upgrade the database from the 7.3 schema to 7.4.

To upgrade the database schema, run the following command:

```
mysql -u root -p pf -v < /usr/local/pf/db/upgrade-7.3.0-7.4.0.sql
```

Once completed, update the file /usr/local/pf/conf/currently-at to match the new release number (PacketFence 7.4.0).

24. Additional Information

For more information, please consult the mailing archives or post your questions to it. For details, see:

- packetfence-announce@lists.sourceforge.net: Public announcements (new releases, security warnings etc.) regarding PacketFence
- packetfence-devel@lists.sourceforge.net: Discussion of PacketFence development
- packetfence-users@lists.sourceforge.net: User and usage discussions

25. Commercial Support and Contact Information

For any questions or comments, do not hesitate to contact us by writing an email to: support@inverse.ca.

Inverse (https://inverse.ca) offers professional services around PacketFence to help organizations deploy the solution, customize, migrate versions or from another system, performance tuning or aligning with best practices.

Hourly rates or support packages are offered to best suit your needs.

Please visit https://inverse.ca/ for details.

26. GNU Free Documentation License

Please refer to http://www.gnu.org/licenses/fdl-1.2.txt for the full license.