FAMP

```
┌──────────────────────────────────────────────┐
│                                                │
│         Welcome to FreeBSD!                    │
│                                                │
│                                                │
│   1. Boot FreeBSD [default]                    │
│   2. Boot FreeBSD with ACPI disabled           │
│   3. Boot FreeBSD in Safe Mode                 │
│   4. Boot FreeBSD in single user mode          │
│   5. Boot FreeBSD with verbose logging         │
│   6. Escape to loader prompt                   │
│   7. Reboot                                    │
│                                                │
│                                                │
│                                                │
│   Select option, [Enter] for default          │
│   or [Space] to pause timer  8                 │
│                                                │
└──────────────────────────────────────────────┘
```

```
  _____                     
 |  ___| __ ___   ___       
 | |_ | '__/ _ \ / _ \      
 |  _|| | |  __/|  __/      
 |_|  |_|  \___| \___|      
  ____   ____  ____         
 | __ ) / ___||  _ \        
 |  _ \ \___ \| | | |       
 | |_) | ___) | |_| |       
 |____/ |____/|____/        
```

Apache2.4, MySQL 8.0, PHP 8.1 on FreeBSD 12.3, 12.4, 13.0 and 13.1 (FAMP)

December 2, 2022 femtopc.com

Synopsys: osTicket is a widely used and trusted open source support ticketing system. Easily scale and streamline your customer service and drastically improve your customer's experience. This document is a supplement to FemtoPC's "FAMP" installation script file for FreeBSD 12.3, 12.4, 13.0 and 13.1. It is customized to install MySQL 8.0 and PHP 8.1 and will grab the latest osTicket version from GitHub.

https://www.femtopc.com/shell_scripts/FAMP/osTicket/

Requirements:

1. FreeBSD server, on discrete hardware or running in a virtual machine.

2. Script designed for server with clean FreeBSD install, no other software installed.

3. Disk requirement: After installation, only 3.1GB are used, including the vanilla FreeBSD installation.

4. Must be run with 'root' access, or as a user that su to root.

5. Server must have Internet access.

6. Script file must run in /root directory.

7. Depending on Internet speed and server processing power, installation is complete in under 10 minutes.

Step-by-step:

1. Login to FreeBSD server as root, or su to root.

2. Fetch the FAMP package

```
fetch https://www.femtopc.com/shell_scripts/FAMP/osTicket/install.gz
```

3. Unzip the download

```
Tar -xvf install.gz
```

4. Run the shell program and answer when prompted.

```
./ost.sh
```

5. Script will check to make sure you are running as root, in the /root directory and have Internet access. It will then check for any Apache, MySQL and PHP packages and exit if found. The install versions of MySQL and PHP are fixed. After packages are installed, the configuration will be automatically done. The php.ini can choose either the 'production' source or the 'developer' source before the file is modified. A password will be requested for MySQL root user. Please make it secure. View the logfile.log to troubleshoot any errors.

Here is a breakdown of the comments and commands that are run in the script.

install Apache2.4

pkg install -y apache24

install MySQL80-Server

pkg install -y mysql80-server

install PHP81

pkg install -y php81

install PHP81-extensions

pkg install -y php81-extensions

Extra extensions for osTicket will be installed

pkg install -y php81-gd >> logfile.log

pkg install -y php81-fileinfo >> logfile.log

pkg install -y php81-intl >> logfile.log

pkg install -y php81-mbstring >> logfile.log

pkg install -y php81-mysqli >> logfile.log

pkg install -y php81-imap >> logfile.log

pkg install -y php81-zip >> logfile.log

pkg install -y php81-pecl-apcu >> logfile.log

install Apache mod_php81

pkg install -y mod_php81

These are what the configuration routines done

modify the /usr/local/etc/apache24/httpd.conf file

make a backup and then edit

cd /usr/local/etc/apache24

cp httpd.conf httpd.conf.original

ee httpd.conf

set the ServerAdmin, in the script it is set to root. You may also set a webmaster email here.

set the ServerName to the IP address of server. A domain name could be set here if available.

set the second time AllowOverride None, where None is capital "N" to All. This will allow .htaccess files to work.

set DirectoryIndex from just 'index.html' to 'index.php index.htm index.html' to capture PHP files

setup Apache to run PHP by creating an Includes/php.conf file

cd /usr/local/etc/apache24/Includes

ee php.conf

```
<FilesMatch "\.php$">
    SetHandler application/x-httpd-php
</FilesMatch>
<FilesMatch "\.phps$">
    SetHandler application/x-httpd-php-source
</FilesMatch># return to our root directory
```

cd /root

Make Apache start automatically when FreeBSD boots

sysrc apache24_enable="yes"

Startup the Apache Server

service apache24 start

```
# Setup to run MySQL server

# Make Server start automatically when FreeBSD boots.

sysrc mysql_enable="yes"

# start the mysql server for the first time

service mysql-server start

# MySQL81 has no password set for root

# Set your new password; it MUST BE in single quotes

/usr/local/bin/mysqladmin -u root password 'your new password'

# For MySQL 81, just press Enter when prompted for a password, as none was assigned


# Now configure PHP

cd /usr/local/etc

# choose a development or a production environment

# development (displays errors in web pages)

cp php.ini-development php.ini

# or production (will not display web page errors)

cp php.ini-production php.ini

# edit the php.ini file

ee php.ini

# find where memory_limit = 128M and increase to 1024M

# find where post_max_size = 8M and set to 32M (or set your limit here)

# find where upload_max_filesize = 2M and set to 1G (or your limit here)

# find where max_file_uploads = 20 and increase to 80

# lastly, find ;date.timezone and uncomment (delete ;) and enter timezone (IMPORTANT!!)

# hint: find your server's setting by doing "cat /var/db/zoneinfo"

# after changes to php.ini, Apache must be restarted

service apache24 restart

# at this point you have a working FAMP server

cd /root
```

```
# This is the specific installation and configuration for osTicket

# Install the git package

pkg install -y git

# Get the latest osTicket files from GitHub (they will be placed in the /root/osticket directory)

git clone https://github.com/osticket/osticket

# Get our Apache webroot ready by creating an 'osticket' directory

cd /usr/local/www/apache24/data

mkdir osticket

chown -R www:www osticket

cd /root/osticket

# make our source files owned by the web server

chown -R www:www *

# Copy all files to the ~webroot/osticket directory and copy a config-ost.php

cp -R * /usr/local/www/apache24/data/osticket/

cd /usr/local/www/apache24/data/osticket/include/

mv ost-sampleconfig.php ost-config.php

cd /root

# Restart the Apache Server

service apache24 restart

# Create a database for OST

/usr/local/bin/mysql -uroot -p -e "create database osticket_db character set utf8mb4 collate utf8mb4_general_ci"

# Use the MySQL root user password that was assigned

# Open a browser and connect to the web server's IP address

# A working server will reply with "It works!"

# Start the OST Web setup by opening http://{ip of server}/osticket/index.php

# Remember the following and input as required:

# DBName: osticket_db

# BDUser: root

# DBPass: your assigned password for MySQL root

# Continue with the osTicket web configuration

<end>
```