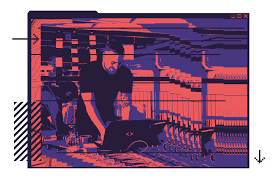
Puesta en producción segura

## **Instalación de Apache con una imagen en Docker hub**



**U1\_A5**

**Francisco Iván Ramírez Ortega**

Curso 2023 - 2024

**IES El Rincón**

Curso de especialización en Ciberseguridad en entornos de las tecnologías de la información

# Índice

| **Portada****1**  [DockerHub](#_ivl6i7zhmykx) 2  [Aspectos relevantes del DockerFile](#_iotqrze56y8g) 4  [SO de la imagen httpd:alpine3.15 (27.11.2023)](#_f2d3j05biso) 4  [Comprueba si existe el usuario www-data y si no lo crea](#_f2d3j05biso) 4  [Ruta expuesta](#_43vlek8z7rwa) 4  [Configuración del Entorno:](#_6h8l7dx02q2r) 4  [Instalación de Dependencias y Herramientas:](#_y58lpgbh3d5q) 4  [Configuración y compilación de apache HTTPD](#_dor8oilo7lug) 4  [Creación y lanzamiento del container](#_t2old04rlo9q) 5  [Dockerfile httpd](#_7140mnevlaso) 5 |
| --- |
| DockerHub |
| Navegamos hacia Docker Hub en busca de imágenes de Apache, preferiblemente oficiales |
| https://hub.docker.com/ |
|  |
| Se trata de una máquina Alpine. |
|  |
| Aspectos relevantes del DockerFile |
| SO de la imagen httpd:alpine3.15 (27.11.2023) |
| FROM alpine:3.18 |
| Comprueba si existe el usuario www-data y si no lo crea |
| RUN set -x \  && adduser -u 82 -D -S -G www-data www-data |
| Ruta expuesta |
| EXPOSE 80 |
| Configuración del Entorno: |
| ENV HTTPD\_PREFIX /usr/local/apache2  ENV PATH $HTTPD\_PREFIX/bin:$PATH |
| **Instalación de Dependencias y Herramientas:** |
| **RUN set -eux; \**  **apk add --no-cache \**  **apr \**  **apr-util \**  **apr-util-ldap \**  **ca-certificates \**  **perl \**  **;** |
| **Configuración y compilación de apache HTTPD** |
| ENV HTTPD\_VERSION 2.4.58  ENV HTTPD\_SHA256 fa16d72a078210a54c47dd5bef2f8b9b8a01d94909a51453956b3ec6442ea4c5  ENV HTTPD\_PATCHES="" |
| Creación y lanzamiento del container |
| **sudo docker run -d -p 69:80 --name apachehub httpd:2.4-alpine3.15**  **curl localhost -P 69** |
|  |

| Dockerfile httpd |
| --- |
| FROM alpine:3.18  # ensure www-data user exists  RUN set -x \  && adduser -u 82 -D -S -G www-data www-data  # 82 is the standard uid/gid for "www-data" in Alpine  # https://git.alpinelinux.org/cgit/aports/tree/main/apache2/apache2.pre-install?h=v3.14.0  # https://git.alpinelinux.org/cgit/aports/tree/main/lighttpd/lighttpd.pre-install?h=v3.14.0  # https://git.alpinelinux.org/cgit/aports/tree/main/nginx/nginx.pre-install?h=v3.14.0  ENV HTTPD\_PREFIX /usr/local/apache2  ENV PATH $HTTPD\_PREFIX/bin:$PATH  RUN mkdir -p "$HTTPD\_PREFIX" \  && chown www-data:www-data "$HTTPD\_PREFIX"  WORKDIR $HTTPD\_PREFIX  # install httpd runtime dependencies  # https://httpd.apache.org/docs/2.4/install.html#requirements  RUN set -eux; \  apk add --no-cache \  apr \  apr-util \  apr-util-ldap \  # https://github.com/docker-library/httpd/issues/214  ca-certificates \  perl \  ;  ENV HTTPD\_VERSION 2.4.58  ENV HTTPD\_SHA256 fa16d72a078210a54c47dd5bef2f8b9b8a01d94909a51453956b3ec6442ea4c5  # https://httpd.apache.org/security/vulnerabilities\_24.html  ENV HTTPD\_PATCHES=""  # see https://httpd.apache.org/docs/2.4/install.html#requirements  RUN set -eux; \  \  apk add --no-cache --virtual .build-deps \  apr-dev \  apr-util-dev \  coreutils \  dpkg-dev dpkg \  gcc \  gnupg \  libc-dev \  patch \  # mod\_md  curl-dev \  jansson-dev \  # mod\_proxy\_html mod\_xml2enc  libxml2-dev \  # mod\_lua  lua-dev \  make \  # mod\_http2  nghttp2-dev \  # mod\_session\_crypto  openssl \  openssl-dev \  pcre-dev \  tar \  # mod\_deflate  zlib-dev \  # mod\_brotli  brotli-dev \  ; \  \  ddist() { \  local f="$1"; shift; \  local distFile="$1"; shift; \  local success=; \  local distUrl=; \  for distUrl in \  # https://issues.apache.org/jira/browse/INFRA-8753?focusedCommentId=14735394#comment-14735394  'https://www.apache.org/dyn/closer.cgi?action=download&filename=' \  # if the version is outdated (or we're grabbing the .asc file), we might have to pull from the dist/archive :/  https://downloads.apache.org/ \  https://www-us.apache.org/dist/ \  https://www.apache.org/dist/ \  https://archive.apache.org/dist/ \  ; do \  if wget -O "$f" "$distUrl$distFile" && [ -s "$f" ]; then \  success=1; \  break; \  fi; \  done; \  [ -n "$success" ]; \  }; \  \  ddist 'httpd.tar.bz2' "httpd/httpd-$HTTPD\_VERSION.tar.bz2"; \  echo "$HTTPD\_SHA256 \*httpd.tar.bz2" | sha256sum -c -; \  \  # see https://httpd.apache.org/download.cgi#verify  ddist 'httpd.tar.bz2.asc' "httpd/httpd-$HTTPD\_VERSION.tar.bz2.asc"; \  export GNUPGHOME="$(mktemp -d)"; \  # $ docker run --rm buildpack-deps:bookworm-curl bash -c 'wget -qO- https://downloads.apache.org/httpd/KEYS | gpg --batch --import &> /dev/null && gpg --batch --list-keys --with-fingerprint --with-colons' | awk -F: '$1 == "pub" && $2 == "-" { pub = 1 } pub && $1 == "fpr" { fpr = $10 } $1 == "sub" { pub = 0 } pub && fpr && $1 == "uid" && $2 == "-" { print "#", $10; print "\t\t" fpr " \\"; pub = 0 }'  for key in \  # Rodent of Unusual Size (DSA) <coar@ACM.Org>  DE29FB3971E71543FD2DC049508EAEC5302DA568 \  # Rodent of Unusual Size <coar@ACM.Org>  13155B0E9E634F42BF6C163FDDBA64BA2C312D2F \  # Jim Jagielski <jim@apache.org>  8B39757B1D8A994DF2433ED58B3A601F08C975E5 \  # Dean Gaudet <dgaudet@apache.org>  31EE1A81B8D066548156D37B7D6DBFD1F08E012A \  # Cliff Woolley <jwoolley@apache.org>  A10208FEC3152DD7C0C9B59B361522D782AB7BD1 \  # Cliff Woolley <jwoolley@virginia.edu>  3DE024AFDA7A4B15CB6C14410F81AA8AB0D5F771 \  # Graham Leggett <minfrin@apache.org>  EB138C6AF0FC691001B16D93344A844D751D7F27 \  # Roy T. Fielding <fielding@gbiv.com>  CBA5A7C21EC143314C41393E5B968010E04F9A89 \  # Justin R. Erenkrantz <jerenkrantz@apache.org>  3C016F2B764621BB549C66B516A96495E2226795 \  # Aaron Bannert <abannert@kuci.org>  937FB3994A242BA9BF49E93021454AF0CC8B0F7E \  # Brad Nicholes <bnicholes@novell.com>  EAD1359A4C0F2D37472AAF28F55DF0293A4E7AC9 \  # Sander Striker <striker@apache.org>  4C1EADADB4EF5007579C919C6635B6C0DE885DD3 \  # Greg Stein <gstein@lyra.org>  01E475360FCCF1D0F24B9D145D414AE1E005C9CB \  # Andre Malo <nd@apache.org>  92CCEF0AA7DD46AC3A0F498BCA6939748103A37E \  # Erik Abele <erik@codefaktor.de>  D395C7573A68B9796D38C258153FA0CD75A67692 \  # Astrid Kessler (Kess) <kess@kess-net.de>  FA39B617B61493FD283503E7EED1EA392261D073 \  # Joe Schaefer <joe@sunstarsys.com>  984FB3350C1D5C7A3282255BB31B213D208F5064 \  # Stas Bekman <stas@stason.org>  FE7A49DAA875E890B4167F76CCB2EB46E76CF6D0 \  # Paul Querna <chip@force-elite.com>  39F6691A0ECF0C50E8BB849CF78875F642721F00 \  # Colm MacCarthaigh <colm.maccarthaigh@heanet.ie>  29A2BA848177B73878277FA475CAA2A3F39B3750 \  # Ruediger Pluem <rpluem@apache.org>  120A8667241AEDD4A78B46104C042818311A3DE5 \  # Nick Kew <nick@webthing.com>  453510BDA6C5855624E009236D0BC73A40581837 \  # Philip M. Gollucci <pgollucci@p6m7g8.com>  0DE5C55C6BF3B2352DABB89E13249B4FEC88A0BF \  # Bojan Smojver <bojan@rexursive.com>  7CDBED100806552182F98844E8E7E00B4DAA1988 \  # Issac Goldstand <margol@beamartyr.net>  A8BA9617EF3BCCAC3B29B869EDB105896F9522D8 \  # "Guenter Knauf" ("CODE SIGNING KEY") <fuankg@apache.org>  3E6AC004854F3A7F03566B592FF06894E55B0D0E \  # Jeff Trawick (CODE SIGNING KEY) <trawick@apache.org>  5B5181C2C0AB13E59DA3F7A3EC582EB639FF092C \  # Jim Jagielski (Release Signing Key) <jim@apache.org>  A93D62ECC3C8EA12DB220EC934EA76E6791485A8 \  # Eric Covener <covener@apache.org>  65B2D44FE74BD5E3DE3AC3F082781DE46D5954FA \  # Yann Ylavic <ylavic@apache.org>  8935926745E1CE7E3ED748F6EC99EE267EB5F61A \  # Daniel Ruggeri (http\x3a//home.apache.org/~druggeri/) <druggeri@apache.org>  E3480043595621FE56105F112AB12A7ADC55C003 \  # Joe Orton (Release Signing Key) <jorton@apache.org>  93525CFCF6FDFFB3FD9700DD5A4B10AE43B56A27 \  # Christophe JAILLET <christophe.jaillet@wanadoo.fr>  C55AB7B9139EB2263CD1AABC19B033D1760C227B \  # Stefan Eissing (icing) <stefan@eissing.org>  26F51EF9A82F4ACB43F1903ED377C9E7D1944C66 \  ; do \  gpg --batch --keyserver keyserver.ubuntu.com --recv-keys "$key"; \  done; \  gpg --batch --verify httpd.tar.bz2.asc httpd.tar.bz2; \  command -v gpgconf && gpgconf --kill all || :; \  rm -rf "$GNUPGHOME" httpd.tar.bz2.asc; \  \  mkdir -p src; \  tar -xf httpd.tar.bz2 -C src --strip-components=1; \  rm httpd.tar.bz2; \  cd src; \  \  patches() { \  while [ "$#" -gt 0 ]; do \  local patchFile="$1"; shift; \  local patchSha256="$1"; shift; \  ddist "$patchFile" "httpd/patches/apply\_to\_$HTTPD\_VERSION/$patchFile"; \  echo "$patchSha256 \*$patchFile" | sha256sum -c -; \  patch -p0 < "$patchFile"; \  rm -f "$patchFile"; \  done; \  }; \  patches $HTTPD\_PATCHES; \  \  gnuArch="$(dpkg-architecture --query DEB\_BUILD\_GNU\_TYPE)"; \  ./configure \  --build="$gnuArch" \  --prefix="$HTTPD\_PREFIX" \  --enable-mods-shared=reallyall \  --enable-mpms-shared=all \  # PIE and hardening flags are unnecessary as Alpine enables them automatically (https://alpinelinux.org/about/)  ; \  make -j "$(nproc)"; \  make install; \  \  cd ..; \  rm -r src man manual; \  \  sed -ri \  -e 's!^(\s\*CustomLog)\s+\S+!\1 /proc/self/fd/1!g' \  -e 's!^(\s\*ErrorLog)\s+\S+!\1 /proc/self/fd/2!g' \  -e 's!^(\s\*TransferLog)\s+\S+!\1 /proc/self/fd/1!g' \  -e 's!^(\s\*User)\s+daemon\s\*$!\1 www-data!g' \  -e 's!^(\s\*Group)\s+daemon\s\*$!\1 www-data!g' \  "$HTTPD\_PREFIX/conf/httpd.conf" \  "$HTTPD\_PREFIX/conf/extra/httpd-ssl.conf" \  ; \  grep -E '^\s\*User www-data$' "$HTTPD\_PREFIX/conf/httpd.conf"; \  grep -E '^\s\*Group www-data$' "$HTTPD\_PREFIX/conf/httpd.conf"; \  \  deps="$( \  scanelf --needed --nobanner --format '%n#p' --recursive /usr/local \  | tr ',' '\n' \  | sort -u \  | awk 'system("[ -e /usr/local/lib/" $1 " ]") == 0 { next } { print "so:" $1 }' \  )"; \  apk add --no-network --virtual .httpd-so-deps $deps; \  apk del --no-network .build-deps; \  \  # smoke test  httpd -v  # https://httpd.apache.org/docs/2.4/stopping.html#gracefulstop  STOPSIGNAL SIGWINCH  COPY httpd-foreground /usr/local/bin/  EXPOSE 80  CMD ["httpd-foreground"] |