Helper script for acme.sh

Helper script for acme.sh

acme.sh is a client for generating Let's Encrypt certificates.

Certbot is a great alternative, but I've just stuck to what I know (hence why this exists).

This wiki page will just be backup for a GitHub Gist I've made:

https://gist.github.com/Decicus/93d698347cf600af5ea822870eeef54c

GitHub Gist is also mirrored to my personal Gitea instance.

Script by default attempts to install certs into `/srv/ssl` using Let's Encrypt.

You can pass `--buypass` when executing the script to use BuyPass Go SSL instead.

Script was last updated April 30th, 2021. A newer version might be available via GitHub Gist.

```
#!/bin/bash

# Make sure to load environment variables.
. ~/.bashrc

ACME_DIR="/root/.acme.sh"
ACME="${ACME_DIR}/acme.sh --force"
BASE="/srv/ssl"
ECHO_PREFIX="[acme.sh Helper Script]"

CMD_PARAMS="$@";

# Check if we should use BuyPass instead of Let's Encrypt
# as the certificate authority for this certificate.
BUYPASS=0;
if [[ "${CMD_PARAMS}" =~ "--buypass" ]]; then
    BUYPASS=1;
    echo "${ECHO_PREFIX} '--buypass' specified - Using BuyPass CA (Go SSL)."
fi

# BuyPass requires a valid email to be registered
# before we issue certificates.
if [[ $BUYPASS -eq 1 ]]; then
    CA_DIR="${ACME_DIR}/ca/api.buypass.com";

    if [[ ! -d "${CA_DIR}" ]]; then
        echo "${ECHO_PREFIX} Account email for BuyPass CA (required)?"
        read ACCOUNT_EMAIL
```

```
        eval "${ACME} --server https://api.buypass.com/acme/directory --register-account --
accountemail '${ACCOUNT_EMAIL}'"
    fi
fi

# Create directory if it exists, make sure permissions are as strict as possible.
echo "${ECHO_PREFIX} Creating base certificate directory: ${BASE}"
mkdir -p $BASE
chmod -R 600 $BASE
chown -R root:root $BASE

echo "${ECHO_PREFIX} Name of folder containing certificates? (Will be created under ${BASE})"
read FOLDERNAME

echo "${ECHO_PREFIX} Creating folder if it doesn't exist: ${BASE}/${FOLDERNAME}"
mkdir -p "${BASE}/${FOLDERNAME}"

# ¯\_(?)_/¯ - https://timmurphy.org/2012/03/09/convert-a-delimited-string-into-an-array-in-
bash/
OIFS=$IFS
IFS=' '

echo "${ECHO_PREFIX} Space-separated list of domains to generate a certificate for?"
read DOMAIN_LIST

DOMAINS=($DOMAIN_LIST)
IFS=$OIFS
DOMAIN_PARAMS=""
ACME_PARAMS=""

for (( i = 0; i < ${#DOMAINS[@]}; i++ )); do
    DOMAIN_PARAMS+=" -d ${DOMAINS[$i]}"
done

echo "${ECHO_PREFIX} DNS? [y/N]"
read IS_DNS

IS_DNS=${IS_DNS,,}
if [[ $IS_DNS == *"y"* ]]; then
    echo "${ECHO_PREFIX} DNS provider? For example: Cloudflare = dns_cf."
    echo "${ECHO_PREFIX} Provider also assumes the proper environment variables are set. Read:
https://github.com/Neilpang/acme.sh/tree/master/dnsapi#how-to-use-dns-api"
    read DNS_PROVIDER

    ACME_PARAMS+="--dns ${DNS_PROVIDER}"
else
    echo "${ECHO_PREFIX} Webroot? For example: /var/www/html"
    read WEBROOT_DIR

    ACME_PARAMS+="-w ${WEBROOT_DIR}"
fi

# Make sure we point to the right CA.
if [[ $BUYPASS -eq 1 ]]; then
    ACME_PARAMS+=" --server https://api.buypass.com/acme/directory"
fi

echo "${ECHO_PREFIX} Reload command? For example: nginx -s reload"
read RELOADCMD
```

```
echo "${ECHO_PREFIX} Requesting certificate using the chosen methods:"
eval "${ACME} ${DOMAIN_PARAMS} ${ACME_PARAMS} --issue"

SSL_PATH="$BASE/$FOLDERNAME"
if [[ "$?" == "0" ]]; then
    echo "${ECHO_PREFIX} Certificate request completed. Installing certificate with reload
command."
    eval "${ACME} ${DOMAIN_PARAMS} --key-file '${SSL_PATH}/key.pem' --fullchain-file
'${SSL_PATH}/fullchain.pem' --cert-file '${SSL_PATH}/cert.pem' --reloadcmd '${RELOADCMD}' --
install-cert"
else
    echo "${ECHO_PREFIX} An error occurred during certificate request. Aborting."
fi
```

Created 2019-01-13 07:09:24 UTC by Alex Thomassen
Updated 2021-04-30 06:31:20 UTC by Alex Thomassen