

Admins - Demande #4373

Mettre en place un accès web pour le serveur mumble de l'April

26/03/2020 17:34 - Frédéric Couchet

Statut:	Fermé	Début:	26/03/2020
Priorité:	Normale	Echéance:	
Assigné à:	Frédéric Couchet	% réalisé:	0%
Catégorie:		Temps estimé:	0.00 heure
Version cible:	Backlog	Temps passé:	0.00 heure
Difficulté:	2 Facile		
Description			
Ce serait bien de pouvoir permettre un accès web au serveur mumble de l'April.			
Demandes liées:			
Lié à Admins - Demande #4446: Normaliser mumble-web		Fermé	30/04/2020

Historique

#1 - 26/03/2020 21:04 - Christian P. Momon

- Version cible changé de Backlog à Avril 2020

#2 - 27/03/2020 08:31 - Frédéric Couchet

L'outil utilisé par Framasoft pour son mumble : <https://github.com/johni0702/mumble-web>

#3 - 27/03/2020 08:40 - Frédéric Couchet

Installation sur la VM mumble :

```
apt install npm
cd /var/www
git clone https://github.com/johni0702/mumble-web
cd mumble-web
npm install
npm run build
apt install websockify
```

Un peu de modification de config pour supprimer les champs password, token dans la fenêtre de connexion :

```
cd /var/www/mumble-web/dist
vi config.local.js
config.connectDialog.password = false;
config.connectDialog.token = false;
config.settings.voiceMode = 'ptt';
config.settings.joinDialog = true;
```

Pour franciser un peu :

```
vi index.html
```

Ajout dans systemd

```
Création de /etc/systemd/system/mumble-web.service
systemctl daemon-reload
```

```
systemctl start mumble-web
```

Sur bastion configuration par François de la conf de mumble.april.org :

```
vi /etc/nginx/sites-available/mumble.april.org
```

```
systemctl restart nginx
```

#4 - 27/03/2020 14:59 - Frédéric Couchet

- Fichier *config.local.js* ajouté

- Fichier *index.html* ajouté

Ci-joint les fichiers modifiés.

#5 - 27/03/2020 17:07 - Quentin Gibeaux

J'ai déployé un certificat LE pour ce démon :

Installation de nginx-light + dehydrated + conf :

```
(April) root@mumble:~# cat /etc/nginx/sites-enabled/letsencrypt
server {
    listen 8080;
    listen [::]:8080;

    server_name mumble.april.org;

    location /.well-known/acme-challenge {
        alias /var/lib/dehydrated/acme-challenges;
    }
}
(April) root@mumble:~# cat /etc/dehydrated/domains.txt
mumble.april.org
```

Conf bastion :

```
April) root@bastion:/etc/nginx/sites-enabled[master$]# cat mumble.april.org
server {
    listen 80;

    server_name mumble.april.org;

    access_log /var/log/nginx/mumble.april.org/mumble.april.org.access_log;
    error_log /var/log/nginx/mumble.april.org/mumble.april.org.error_log;

    include /etc/nginx/custom_50x;
    include /etc/nginx/deny_.git;

    # challenge sur la vm avec nginx-light sur 8080
    location ^~ /.well-known/acme-challenge/ {
        proxy_pass http://172.16.0.8:8080;
    }

    location / {
        proxy_pass https://172.16.0.8:8443;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $connection_upgrade;
    }
}

server {
    listen 443;
    ssl on;

    server_name mumble.april.org;

    access_log /var/log/nginx/mumble.april.org/mumble.april.org.access_log;
    error_log /var/log/nginx/mumble.april.org/mumble.april.org.error_log;

    include /etc/nginx/acme-challenge;
```

```

include /etc/nginx/custom_50x;
include /etc/nginx/deny_.git;
include /etc/nginx/hsts.conf;

location / {
    proxy_pass https://172.16.0.8:8443;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection $connection_upgrade;
}
}

map $http_upgrade $connection_upgrade {
    default upgrade;
    '' close;
}

```

Génération du certificat :

```

(April) root@mumble:~# /usr/bin/dehydrated --register --accept-terms
(April) root@mumble:~# /usr/bin/dehydrated -c
(April) root@mumble:/var/lib/dehydrated# chmod o+rx /var/lib/dehydrated/certs/
(April) root@mumble:/var/lib/dehydrated# chown -R mumble-server /var/lib/dehydrated/certs/mumble.april.org/

```

Conf du démon :

```

(April) root@mumble:/etc/systemd/system[master*]# cat mumble-web.service
[Unit]
Description=Mumble web interface
Documentation=https://github.com/johni0702/mumble-web
Requires=network.target mumble-server.service
After=network.target mumble-server.service

[Service]
Type=simple
User=mumble-server
ExecStart=websockify -v --cert=/var/lib/dehydrated/certs/mumble.april.org/fullchain.pem --key=/var/lib/dehydrated/certs/mumble.april.org/privkey.pem --ssl-only --ssl-target --web=/var/www/mumble-web/dist/8443 localhost:64738

[Install]
WantedBy=multi-user.target

```

Ajout du cron :

```

(April) root@mumble:~# cat /etc/cron.weekly/dehydrated
#!/bin/bash

date >> /var/log/dehydrated.log 2>&1
/usr/bin/dehydrated -c >>/var/log/dehydrated.log 2>&1

find /var/lib/dehydrated/certs/ -name 'fullchain-*.pem' -mmin -500 -exec systemctl restart mumble-web >>/var/log/dehydrated.log 2>&1 \;

(April) root@mumble:~# chmod +x /etc/cron.weekly/dehydrated

```

#6 - 28/03/2020 09:45 - Frédéric Couchet

j'ai ajouté une mini doc du mumble-web https://wiki.april.org/w/Mumble#Acc.C3.A8s_avec_un_navigateur_web

j'ai ajouté le lien et quelques dans le message d'accueil de mumble.april.org, fichier /var/www/mumble-web/dist/index.html

#7 - 28/03/2020 17:37 - Frédéric Couchet

- Fichier index.html supprimé

#8 - 28/03/2020 17:37 - Frédéric Couchet

- Fichier index.html ajouté

#9 - 30/03/2020 01:19 - Christian P. Momon

- Statut changé de Nouveau à En cours de traitement

#10 - 30/03/2020 01:24 - Christian P. Momon

Dans le README.md du projet (<https://github.com/johni0702/mumble-web>), on peut lire que Mumble-web est utilisable soit en mode *standalone*, soit en mode *proxied*.

Le mode *standalone* (méthode actuelle sur SI April) consiste à confier à websockify la gestion du flux vers mumble et aussi celle des pages web statiques.

Au premier abord, ça semble plus simple mais en fait :

- nécessite de gérer un certificat sur la vm mumble et donc :
 - déplace la gestion du certificat de la vm bastion vers la vm mumble,
 - nécessite le déploiement d'un nginx sur la vm mumble uniquement pour gérer la génération letsencrypt,
 - impose un aiguillage via un `try_files` dans nginx sur la vm bastion,
 - introduit un cas particulier par rapport aux autres vm (en plus de l'usage de `dehydrated` à la place de `certbot`),
- empêche d'avoir des logs http de fréquentation de la page d'accueil.

Le mode *proxied* (méthode sur SI Chapril), consiste à confier au nginx de la vm allo la gestion des pages statiques et à lui demander de relayer le flux `webrtc` vers `websockify`. Les avantages :

- `websockify` n'a pas besoin de certificat sur la vm allo donc :
 - on peut garder la gestion de certificat sur la vm bastion de façon homogène à tout le SI April,
- le nginx de la vm mumble permet de générer localement des logs http,
- évitement de `try_files` en configurant l'URL `foo.bar.org/ws` comme aiguillage du flux `webrtc`.

Vu la simplification et l'homogénéisation, préconisation de migrer la conf en mode *proxied*.

Comme actuellement, ça fonctionne, pas d'urgence, attendre le moment où ça gêne le moins.

#11 - 29/04/2020 21:17 - Frédéric Couchet

- Assigné à Frédéric Couchet supprimé

- Version cible changé de Avril 2020 à Backlog

#12 - 29/04/2020 21:57 - Quentin Gibeaux

- Statut changé de En cours de traitement à Fermé

ça fonctionne

#13 - 30/04/2020 01:18 - Christian P. Momon

- Lié à Demande #4446: Normaliser mumble-web ajouté

#14 - 04/12/2020 21:44 - Christian P. Momon

- Assigné à mis à Frédéric Couchet

Fichiers

<code>config.local.js</code>	855 octets	27/03/2020	Frédéric Couchet
<code>index.html</code>	27,6 ko	28/03/2020	Frédéric Couchet