

Compte-Rendu

Objet Projet Infrastructure et SI dépot git: https://github.com/Akram-abdl/b2_ProjetInfra	Date 02/04/2021	Référence B2A
	Rédacteur Ianis BORDREZ Laurent MARECHAL Akram ABDOUL-WAHAB	
Destinataire Léo GODEFROY		

I. Présentation

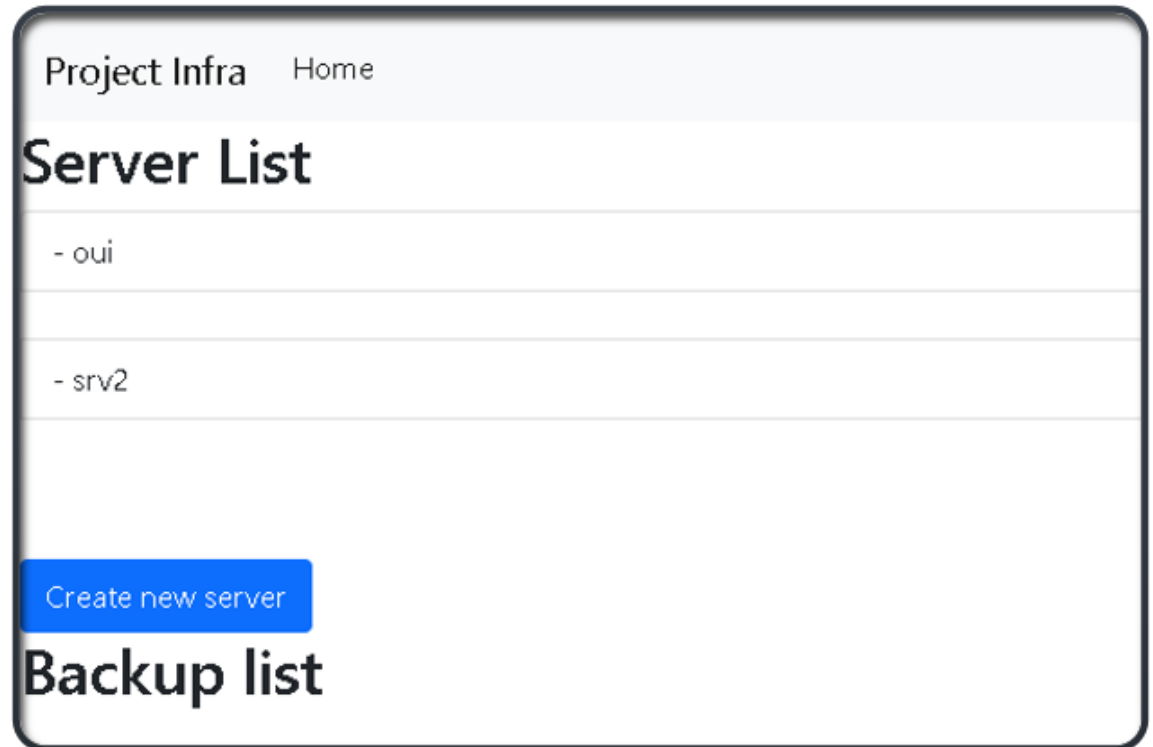
A. Choix du projet

Nous avons longuement hésité sur le thème général du projet. En effet, plusieurs thèmes ont été proposés au sein du groupe. Nous avons finalement opté pour une interface web d'administration de serveur Metin 2.

Ce projet serait intéressant pour la simple raison que la totalité du groupe joue à ce jeu. De ce fait, nous voulions avoir une interface web disponible lors de la création d'un futur serveur pour ce jeu mais aussi pour le fait que très peu d'outils sont disponibles pour ce genre d'interface sur les forums.

B. Utilisation

1. Page d'accueil:



- a. Liste des serveurs créés
- b. Liste Backups
- c. Accès page création serveur
- d. Accès page gestion serveurs

2. Page création serveur:

Project Infra

Home

Create server

server_prefix

srv1

srv1

server_channelNumber

2

2

server_coreNumber

3

3

server_port

30000

30000

server_mapAllow

core1

1 2 3 4

1 2 3 4

core2

5 6 7

5 6 7

core3

8

8

core4

9

9

core5

10

10

core99

72

72

mysql_user

mt2

mt2

mysql_password

mt2

mt2

mysql_bases

account

account

account

common

common

common

hotbackup

hotbackup

hotbackup

log

log

log

player

player

player

submit

- Formulaire configuration serveur
- Génère un fichier config spécifique au serveur

3. Page gestion serveur:

Project Infra

Server srv2 2 5

Start Server

Close Server

Process Status

@Status

Status

```
./srv2-ch1-core1 Process is not Running on this server
./srv2-ch1-core2 Process is not Running on this server
./srv2-ch1-core3 Process is not Running on this server
./srv2-ch1-core4 Process is not Running on this server
./srv2-ch1-core5 Process is not Running on this server
./srv2-ch2-core1 Process is not Running on this server
./srv2-ch2-core2 Process is not Running on this server
./srv2-ch2-core3 Process is not Running on this server
./srv2-ch2-core4 Process is not Running on this server
./srv2-ch2-core5 Process is not Running on this server
```

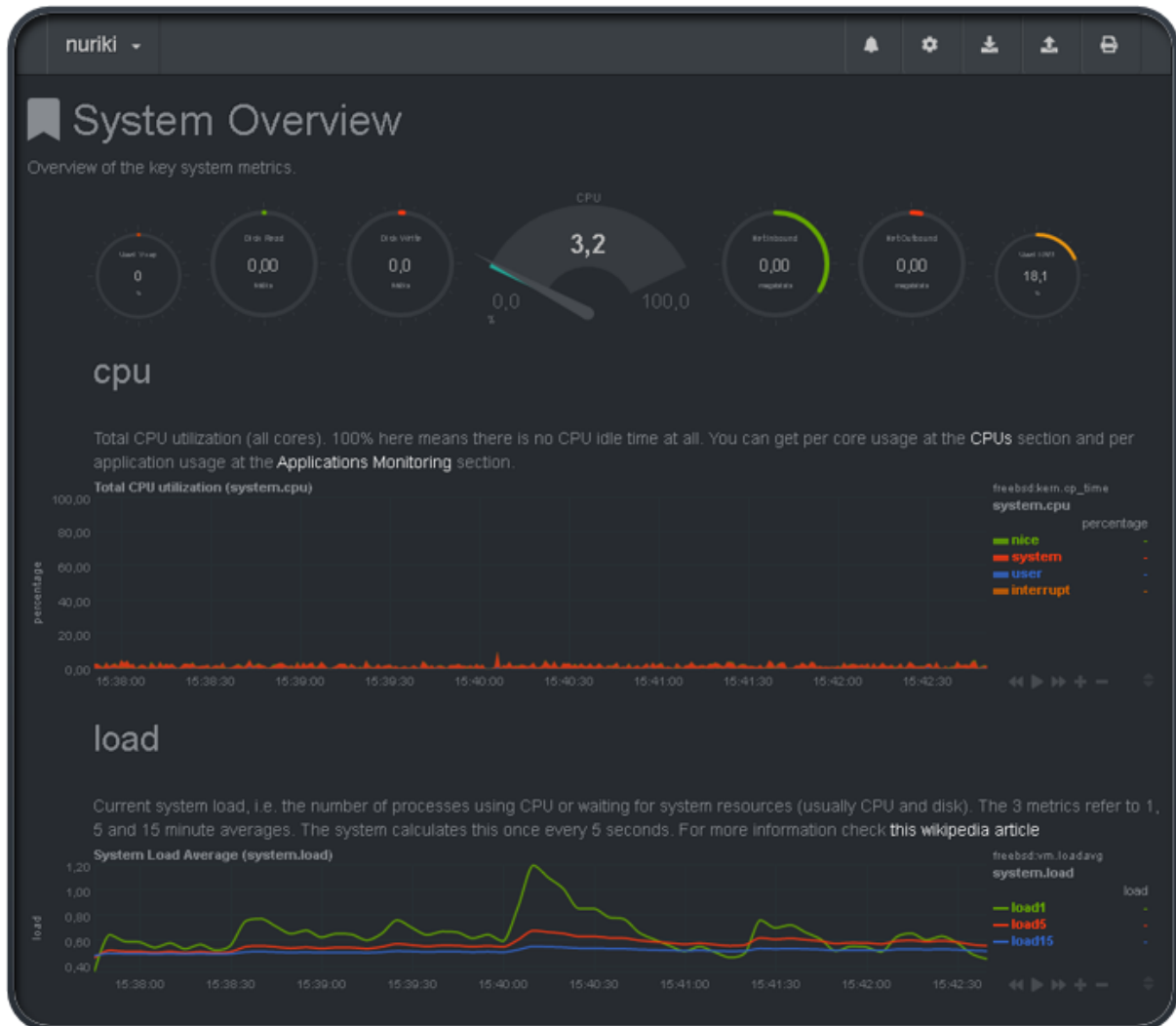
Delete Server

Backup list

Name	Action
<div>Create backup</div>	

- Boutons Lancer / Arrêter / Supprimer
- Créer Backup
- Statuts des processus

4. Page Netdata:



- CPU / Disques / Mémoire etc...
- Lier à discord

C. Choix des outils

1. Software

- NGINX: Système est un logiciel libre de serveur Web. C'est un outil simple à mettre en place et d'utilisation mais aussi assez puissant pour l'ampleur de notre projet. C'est donc le choix optimal dans notre situation ce qui nous a permis de servir notre solution via une page web.
- Netdata: Netdata est un outil open source qui permet de récolter des données en temps réel sur une machine. (CPU / RAM / DISQUE...)

2. Environnement:

- Hyperviseur : Oracle VM VirtualBox (anciennement VirtualBox) est un logiciel libre de virtualisation. Nous l'utilisons pour nos machines virtuelles.
- VM:
 - Serveur Jeu: FreeBSD est un système d'exploitation UNIX libre. C'est sur cette architecture que Metin 2 a été développé, il n'y a donc pas le choix pour héberger un serveur.
 - Serveur Backup : CentOS (Community enterprise Operating System) est une distribution GNU/Linux destinée aux serveurs (et aux postes de travail). Nous voulions un système léger pour le backup, c'est pour cette raison que nous avons choisi CentOS 8 .

3. Langage

- Bash: Le Bash est un shell, c'est-à-dire un interpréteur de commandes. Nous avons pu automatiser certaines de nos actions aux travers de script grâce à celui-ci.
- PHP: Nous a permis d'établir la communication entre les VM et l'interface web. De même pour les données envoyées entre les pages.

II. Guide d'installation

Vous devez posséder une machine sous FreeBSD 11.4 pour ce projet.

1. Clonage du projet GIT:
 - a. dans un terminal : git clone [git@github.com:Akram-abdl/b2_ProjetInfra.git](https://github.com/Akram-abdl/b2_ProjetInfra.git)
 - b. lancer le fichier init.sh
2. Installer Netdata:
 - a. pkg install -y netdata
 - b. sysrc netdata_enable="YES"
 - c. service netdata start
 - d. dans le fichier /usr/local/etc/netdata/go.d/nginx.conf ajouter les lignes suivantes :


```
jobs:
- name: local
  url : http://127.0.0.1/stub_status
```
3. Installer et le configurer NGINX:
 - a. pkg install -y nginx
 - b. sysrc nginx_enable="YES"

c. dans la configuration nginx situé à /usr/local/etc/nginx/nginx.conf copié la configuration suivante :

```
user www;
worker_processes 1;

pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    upstream netdata {
        server 127.0.0.1:19999;
        keepalive 64;
    }
    server {
        access_log /var/log/nginx/projet_netdata.com.access.log;
        error_log /var/log/nginx/projet_netdata.com.error.log;
        listen 8080;
        server_name localhost;
        auth_basic "Protected";
        auth_basic_user_file passwords;
        location / {
            return 301 /netdata/;
        }
        location /netdata {
            return 301 /netdata/;
        }
        location ~ /netdata/(?<ndpath>.*) {
            proxy_redirect off;
            proxy_set_header Host $host;
            proxy_set_header X-Forwarded-Host $host;
            proxy_set_header X-Forwarded-Server $host;
            proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
            proxy_http_version 1.1;
            proxy_pass_request_headers on;
            proxy_set_header Connection "keep-alive";
            proxy_store off;
            proxy_pass http://netdata/$ndpath$is_args$args;
            gzip on;
            gzip_proxied any;
            gzip_types *;
        }
    }
    server {
        access_log /var/log/nginx/projet.com.access.log;
        error_log /var/log/nginx/projet.com.error.log;

        listen 443 default_server ssl;
        listen 80;
        server_name localhost;
        index index.php index.html;

        ssl_certificate /etc/ssl/localhost.crt;
        ssl_certificate_key /etc/ssl/localhost.key;

        ssl_session_cache shared:SSL:1m;
        ssl_session_timeout 5m;

        ssl_ciphers HIGH:!aNULL:!MD5;
        ssl_prefer_server_ciphers on;

        location / {
            root /usr/local/www/b2_ProjetInfra/projet;
```

```

}
location /stub_status {
    stub_status;
    allow 127.0.0.1;
    # Deny anyone else
    deny all;
}
location ~ [^/]\.php(/|$) {
    fastcgi_split_path_info ^(.+?\.php)(/.*)$;
    if (!-f $document_root$fastcgi_script_name) {
        return 404;
    }

    root /usr/local/www/b2_ProjetInfra/projet;

    fastcgi_param HTTP_PROXY "";

    fastcgi_pass unix:/var/run/php74-fpm.sock;
    fastcgi_index index.php;

    include fastcgi_params;

    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
}
}
}

```

4. Installer et configurer PHP :

- Installation de php

`<https://www.cyberciti.biz/faq/freebsd-install-php-7-2-with-fpm-for-nginx/>`

- `pkg install php74 php74-mysqli php74-json`

- `cp /usr/local/etc/php-fpm.d/www.conf{,.backup}`

- modifier dans `/usr/local/etc/php-fpm.d/www.conf`

```

----
listen = /var/run/php74-fpm.sock
listen.owner = www
listen.group = www
listen.mode = 0660
----

```

- `ln -s /usr/local/etc/php.ini-production /usr/local/etc/php.ini`

- `sysrc php_fpm_enable=yes`

- `chown -R www:www /usr/local/www/b2_ProjetInfra/`

- edit in `/usr/local/etc/nginx/nginx.conf`

```

----
location ~ [^/]\.php(/|$) {
    fastcgi_split_path_info ^(.+?\.php)(/.*)$;
    if (!-f $document_root$fastcgi_script_name) {
        return 404;
    }

    root /usr/local/www;

```



```
fastcgi_param HTTP_PROXY "";
```

```
fastcgi_pass unix:/var/run/php74-fpm.sock;
```

```
fastcgi_index index.php;
```

```
include fastcgi_params;
```

```
fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
```

```
}
```

```
----
```

5. Relancer votre machine
6. Aller sur le lien suivant pour avoir accès à l'interface web : <http://localhost>

