

Situation Professionnelle I

Projet gsb.org

Table des matières

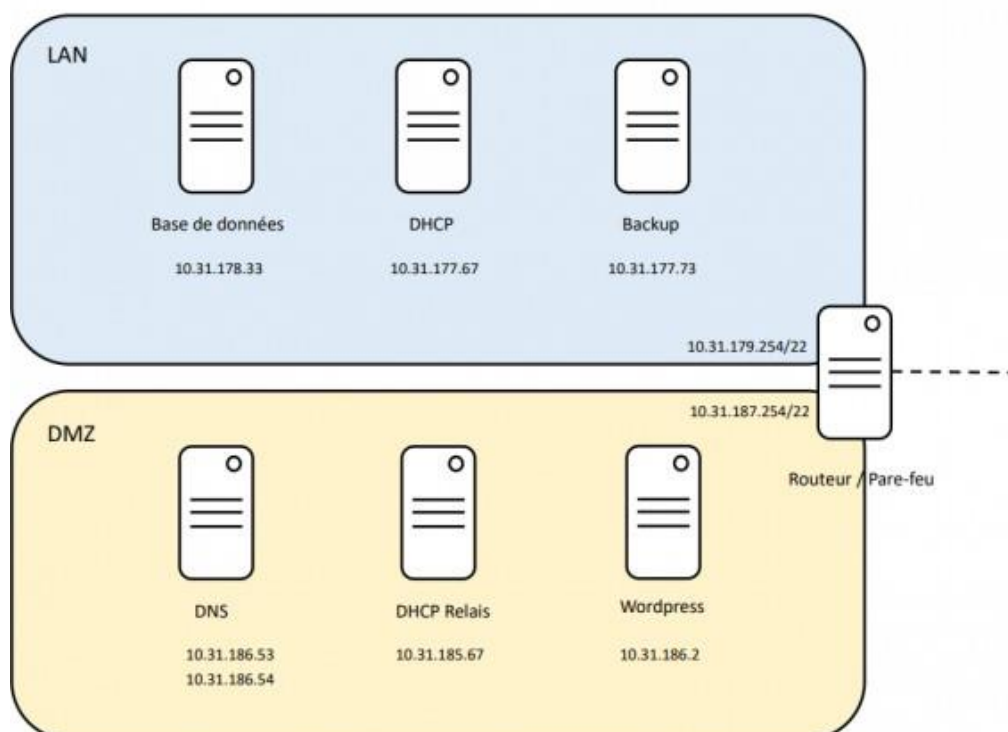
Introduction et contexte.....	3
Adaptation des services DNS, DHCP, BDD et pare-feu.....	4
Modification DNS.....	4
Modification DHCP.....	6
Modifications BDD.....	7
Modifications Pare-feu.....	8
Création et configuration du serveur web.....	9
Création du serveur web.....	9
Installation et configuration de Nginx.....	9
Sauvegarde du serveur web.....	15
Configuration de BackupPC.....	15
Restauration de fichiers.....	17



Introduction et contexte

La société GSB souhaite ouvrir une filiale au Japon. Pour ce faire, un nouveau site internet doit être créé. Nous devons donc effectuer toutes les modifications nécessaires sur les serveurs utilisés par la nouvelle machine et s'assurer de la sauvegarde de cette dernière.

Voici le schéma du réseau de chaque machine utilisée pour cette situation professionnelle :



Adaptation des services DNS, DHCP, BDD et pare-feu

Modification DNS

Dans un premier temps, nous devons modifier les DNS master et slave afin de créer une nouvelle zone "japon" et effectuer la redirection en conséquences. Nous commençons par modifier le DNS slave. Définissons la nouvelle zone dans le fichier named.conf.local :

```
zone "japon.gsb.org" IN {  
    # Définition de ce dns comme master  
    type master;  
    # Lien vers le fichier regroupant les informations nécessaires  
    file "/etc/bind/db.japon.gsb.org";  
    # Définition des DNS slaves  
    allow-transfer { localhost; 10.31.186.54; };  
};
```

```
zone "japon.gsb.org" IN {  
    # Définition de ce dns comme master  
    type master;  
    # Lien vers le fichier regroupant les informations nécessaires  
    file "/etc/bind/db.japon.gsb.org";  
    # Définition des DNS slaves  
    allow-transfer { localhost; 10.31.186.54; };  
};
```

Nous créons ensuite le fichier db.japon.gsb.org :

```
$TTL 604800;  
@ IN SOA ns2-1-pub.gsb.org. root.gsb.org. (  
    2023101701;  
    43200;  
    3600;  
    3600000;  
    172800 );  
  
; Adresse de la zone (le ping japon.gsb.org pinguera cette adresse)  
@ IN A 10.31.186.2;  
  
; DNS Server (on définit les DNS d'autorité pour cette zone)  
@ IN NS ns2-1-pub.gsb.org;  
@ IN NS ns2-2-pub.gsb.org;
```



; Association adresse IP des DNS avec leurs noms (si on ping ces noms on pinguera ces adresses)
ns2-1-pub IN A 10.31.186.53;
ns2-2-pub IN A 10.31.186.54;

; Association adresse IP du serveur web avec www (le ping www.japon.gsb.org pinguera ces adresses)
www IN A 10.31.186.2;

```
$TTL      604800;
@ IN SOA ns2-1-pub.gsb.org. root.gsb.org. (
    2823101701;
    43200;
    3600;
    3600000;
    172800 );

; Adresse de la zone (le ping japon.gsb.org pinguera cette adresse)
@ IN A 10.31.186.2;

; DNS Server (on définit les DNS d'autorité pour cette zone)
@ IN NS ns2-1-pub.gsb.org.;
@ IN NS ns2-2-pub.gsb.org.;

; Association adresse IP des DNS avec leurs noms (si on ping ces noms on pinguera ces adresses)
ns2-1-pub IN A 10.31.186.53;
ns2-2-pub IN A 10.31.186.54;

; Association adresse IP du serveur web avec www (le ping www.japon.gsb.org pinguera ces adresses)
www IN A 10.31.186.2;
```

Nous redémarrons le service et testons la nouvelle configuration :

dig a www.japon.gsb.org

```
$TTL      604800;
@ IN SOA ns2-1-pub.gsb.org. root.gsb.org. (
    2823101701;
    43200;
    3600;
    3600000;
    172800 );

; Adresse de la zone (le ping japon.gsb.org pinguera cette adresse)
@ IN A 10.31.186.2;

; DNS Server (on définit les DNS d'autorité pour cette zone)
@ IN NS ns2-1-pub.gsb.org.;
@ IN NS ns2-2-pub.gsb.org.;

; Association adresse IP des DNS avec leurs noms (si on ping ces noms on pinguera ces adresses)
ns2-1-pub IN A 10.31.186.53;
ns2-2-pub IN A 10.31.186.54;

; Association adresse IP du serveur web avec www (le ping www.japon.gsb.org pinguera ces adresses)
www IN A 10.31.186.2;
```

Nous pouvons maintenant modifier le DNS slave. Nous créons une nouvelle zone dans le fichier named.conf.local :



```
zone "japon.gsb.org" IN {
    # Définition de ce dns comme slave
    type slave;
    # Lien vers le fichier regroupant les informations nécessaires
    file "/etc/bind/db.japon.gsb.org";
    # Définition du DNS master
    masters { 10.31.186.53; };
};
```

```
zone "japon.gsb.org" IN {
    # Définition de ce dns comme slave
    type slave;
    # Lien vers le fichier regroupant les informations nécessaires
    file "/etc/bind/db.japon.gsb.org";
    # Définition du DNS master
    masters { 10.31.186.53; };
};
```

Nous redémarrons le service et testons la nouvelle configuration :

```
root@ns2-2-pub:~# dig a www.japon.gsb.org

;; <<>> DiG 9.18.19-1-deb12ul-Debian <<>> a www.japon.gsb.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52529
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; COOKIE: 764466923d5c45e90100000066179138f8b1f850f5f83a42 (good)
;; QUESTION SECTION:
;www.japon.gsb.org.          IN      A

;; ANSWER SECTION:
www.japon.gsb.org.         604800  IN      A      10.31.186.2

;; Query time: 0 msec
;; SERVER: 10.31.186.53#53(10.31.186.53) (UDP)
;; WHEN: Thu Apr 11 09:28:56 CEST 2024
;; MSG SIZE rcvd: 90
```

Modification DHCP

Nous devons maintenant modifier le DHCP afin d'inclure le serveur web dans les hôtes à IP fixes. Pour cela, nous devons modifier le fichier `/etc/dhcp/dhcpd.conf` :

```
host wordpress-sp1 {
    hardware ethernet 8E:B6:50:2D:5E:22;
    fixed-address 10.31.186.2;
    option domain-name-servers 10.31.186.53, 10.31.186.54;
}
```



```
# SP 1
host wordpress-sp1 {
    hardware ethernet 8E:86:50:2D:5E:22;
    fixed-address 10.31.186.2;
    option domain-name-servers 10.31.186.53, 10.31.186.54;
}
```

Nous redémarrons le service pour sauvegarder les changements.

Modifications BDD

Nous devons maintenant créer une nouvelle base de données qui accueillera les données de Wordpress. Pour ce faire, nous nous rendons sur l'interface CLI de MariaDB et entrons les commandes suivantes :

```
CREATE DATABASE wordpressjapon;
CREATE USER 'adminwpjapon'@'%' identified by 'password';
GRANT ALL PRIVILEGES ON wordpressjapon.* TO 'adminwpjapon'@'%';
FLUSH PRIVILEGES;
```

```
MariaDB [(none)]> CREATE DATABASE wordpressjapon;
Query OK, 1 row affected (0,000 sec)

MariaDB [(none)]> CREATE USER 'adminwpjapon'@'%' identified by 'password';
Query OK, 0 rows affected (0,001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON wordpressjapon.* TO 'adminwpjapon'@'%';
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,001 sec)
```

Nous vérifions que la base de données soit bien créée :









```
SHOW DATABASES;
```

```
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| glpi      |
| information_schema |
| mysql     |
| nextcloud |
| performance_schema |
| sys       |
| wordpressjapon |
| zabbix    |
+-----+
8 rows in set (0,000 sec)
```



Modifications Pare-feu

Pour que le serveur web puisse accéder à la base de données, nous devons rajouter une nouvelle règle dans notre pare-feu autorisant le serveur web à accéder au serveur de bases de données sur le port 3306 :

				IPv4 TCP	10.31.186.2	*	10.31.178.33	3306	*	*	(SP1) Allow BDD to Wordpress-sp1				
---	---	---	---	----------	-------------	---	--------------	------	---	---	----------------------------------	---	---	---	---



Création et configuration du serveur web

Création du serveur web

Dans un premier temps, nous commençons par cloner le template. Après avoir allumé la machine, nous vérifions que l'adresse IP donnée par le DHCP soit bien exacte (10.31.186.2):

```
ifconfig
```

```
root@template-VH:~# ifconfig
ens18: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.31.186.2 netmask 255.255.252.0 broadcast 10.31.187.255
    inet6 fe80::8cb6:50ff:fe2d:5e22 prefixlen 64 scopeid 0x20<link>
    ether 8e:b6:50:2d:5e:22 txqueuelen 1000 (Ethernet)
    RX packets 106514 bytes 145947100 (139.1 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 49157 bytes 3263757 (3.1 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Nous changeons le nom d'hôte de la machine :

```
# Change le nom d'hôte à l'aide d'une commande
hostnamectl set-hostname dhcp1-priv # Ou dhcp2-priv
# Change le nom d'hôte directement depuis le fichier de configuration
nano /etc/hosts
```

```
127.0.0.1      localhost
10.31.186.2    wordpress-sp1
```

Installation et configuration de Nginx

Avant de configurer Nginx, nous mettons à jour notre machine et téléchargeons les paquets nécessaires à la mise en place du serveur web utilisant le protocole HTTPS :

```
apt update && apt upgrade
apt install nginx php-mysqli ssl-cert
```



Nous créons ensuite le dossier qui accueillera notre site internet :

```
mkdir -p /home/htdocs/gsb/japon
```

Nous nous déplaçons dans ce dossier et téléchargeons la dernière version de Wordpress :

```
wget https://wordpress.org/latest.tar.gz
```

```
root@wordpress-spl:~# wget https://wordpress.org/latest.tar.gz
--2024-04-12 08:08:42-- https://wordpress.org/latest.tar.gz
Résolution de wordpress.org (wordpress.org): 198.143.164.252
Connexion à wordpress.org (wordpress.org)[198.143.164.252]:443_ connecté.
requête HTTP transmise, en attente de la réponse_ 200 OK
Taille : 24697732 (24M) [application/octet-stream]
Sauvegarde en : « latest.tar.gz »

latest.tar.gz      8%[>          ] 1,89M  515KB/s  tps 46s |
```

Nous extrayons l'archive Wordpress téléchargée précédemment :

```
tar -xvzf latest.tar.gz
```

Nous nous déplaçons dans le dossier créé nommé Wordpress et changeons le nom du fichier exemple de configuration :

```
cd wordpress
cp wp-config-sample.php wp-config.php
```

Nous devons maintenant modifier des éléments de notre fichier de configuration. Pour ce faire, nous modifions le fichier wp-config.php :

```
nano wp-config.php
```

Nous devons rajouter dans les endroits prévus à cet effet le nom de notre base de données créée précédemment, le nom de l'utilisateur de cette base de données, son mot de passe ainsi que l'adresse IP de la machine sur laquelle est stockée la base de données.



```

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpressjapon' );

/** Database username */
define( 'DB_USER', 'adminwpjapon' );

/** Database password */
define( 'DB_PASSWORD', 'password' );

/** Database hostname */
define( 'DB_HOST', '10.31.178.33' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

```

A présent, nous devons créer les fichiers de configuration Nginx et activer notre site internet. Commençons tout d'abord par créer le fichier `/etc/nginx/sites-available/www.japon.gsb.org`. Nous devons également adapter ce fichier pour activer le protocole HTTPS ainsi que la redirection :

```

server {
    listen 80;
    listen [::]:80;

    server_name www.japon.gsb.org;
    return 301 https://www.japon.gsb.org/;
}

server {
    # SSL configuration
    listen 443 ssl;
    listen [::]:443 ssl;

    include snippets/snakeoil.conf;

    root /home/htdocs/gsb.org/japon/wordpress;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.php index.nginx-debian.html;
    server_name www.asie.gsb.org;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
        try_files $uri $uri/ =404;
    }
}

```



```

location ~ ^/~(.+?)(/.*)?$ {
    alias /home/$1/public_html$2;
    index index.html index.htm;
    autoindex on;
    auth_basic "Zone sécurisée - Authentification requise";
    auth_basic_user_file /etc/nginx/.htpasswd;
}

# pass PHP scripts to FastCGI server
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    # With php-fpm (or other unix sockets):
    fastcgi_pass unix:/run/php/php8.2-fpm.sock;
}
}

```

Nous pouvons également vérifier que les clés et les certificats nécessaires soient installés au téléchargement du paquet `ssl-cert` en regardant le contenu du fichier `/etc/nginx/snippets/snakeoil.conf` :

```

# Self signed certificates generated by the ssl-cert package
# Don't use them in a production server!

ssl_certificate /etc/ssl/certs/ssl-cert-snakeoil.pem;
ssl_certificate_key /etc/ssl/private/ssl-cert-snakeoil.key;

```

Nous créons maintenant un lien symbolique entre le fichier de configuration `/etc/nginx/sites-available/www.japon.gsb.org` et le dossier `/etc/nginx/sites-enabled/` :

```
ln -s /etc/nginx/sites-available/www.japon.gsb.org /etc/nginx/sites-enabled/www.japon.gsb.org
```

Nous redémarrons notre service afin que les changements soient effectifs :

```
systemctl restart nginx
```

Nous pouvons maintenant accéder à l'interface graphique de Wordpress via notre adresse web `www.japon.gsb.org`. Nous configurons maintenant notre Wordpress en entrant le titre du site, l'identifiant de l'utilisateur, son mot de passe ainsi que son e-mail. Nous devons également cocher l'option "Confirmer l'utilisation du mot de passe faible" étant donné que notre mot de passe "password" n'est pas sécurisé.



Welcome

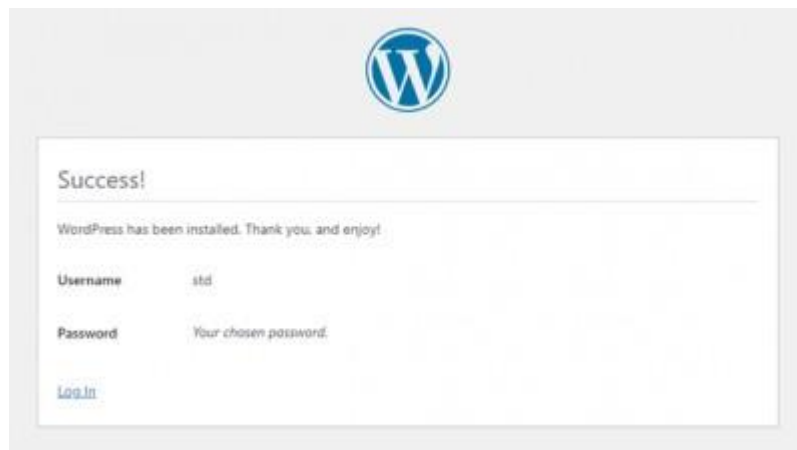
Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Do not worry, you can always change these settings later.

Site Title	<input type="text" value="japon"/>
Username	<input type="text" value="std"/> <small>Username can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.</small>
Password	<input type="password" value="password"/> Hide <div>Very weak</div> <p>Important: You will need this password to log in. Please store it in a secure location.</p>
Confirm Password	<input checked="" type="checkbox"/> Confirm use of weak password
Your Email	<input type="text" value="exemple@gmail.com"/> <small>Double-check your email address before continuing.</small>
Search engine visibility	<input type="checkbox"/> Discourage search engines from indexing this site <small>It is up to search engines to honor this request.</small>
Install WordPress	

Notre site Wordpress est maintenant configuré.





Sauvegarde du serveur web

Configuration de BackupPC

Nous devons maintenant sauvegarder les fichiers nécessaires dans le serveur de sauvegarde. Nous nous rendons sur le serveur de sauvegarde et entrons notre nouvel hôte :

```
ssh-copy-id backuppc@10.31.186.2
```

```
backuppc@backup-01:/root$ ssh-copy-id backuppc@10.31.186.2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/backuppc/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
backuppc@10.31.186.2's password:
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'backuppc@10.31.186.2'"
and check to make sure that only the key(s) you wanted were added.
```

Nous nous rendons maintenant sur le serveur web et entrons la commande suivante :

```
echo "backuppc ALL=NOPASSWD: /usr/bin/rsync" | sudo tee /etc/sudoers.d/backuppc
```

Nous faisons une première connexion SSH entre le compte backuppc du serveur de sauvegarde et le serveur web :

```
backuppc@backup-01:/root$ ssh backuppc@10.31.186.2
The authenticity of host '10.31.186.2 (10.31.186.2)' can't be established.
ED25519 key fingerprint is SHA256:05nXfzS0Y6cukYsHNLPUVW0AZRlMy0W6Fwa3SAN5qhhY.
This host key is known by the following other names/addresses:
-/.ssh/known_hosts:1: [hashed name]
-/.ssh/known_hosts:4: [hashed name]
-/.ssh/known_hosts:5: [hashed name]
-/.ssh/known_hosts:6: [hashed name]
-/.ssh/known_hosts:10: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.31.186.2' (ED25519) to the list of known hosts.
backuppc@10.31.186.2's password:
Linux wordpress-spl 6.1.0-13-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.55-1 (2023-09-29) x86_64

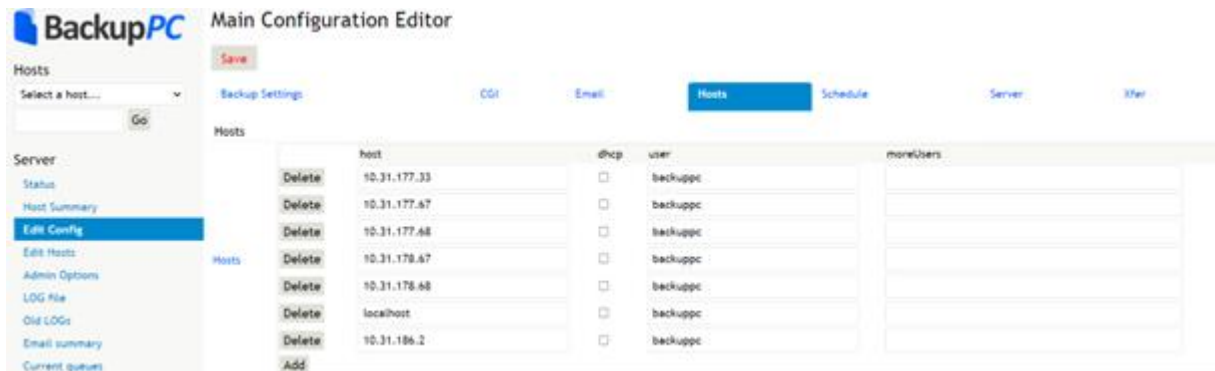
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Apr 12 08:32:39 2024 from 10.31.177.73
```

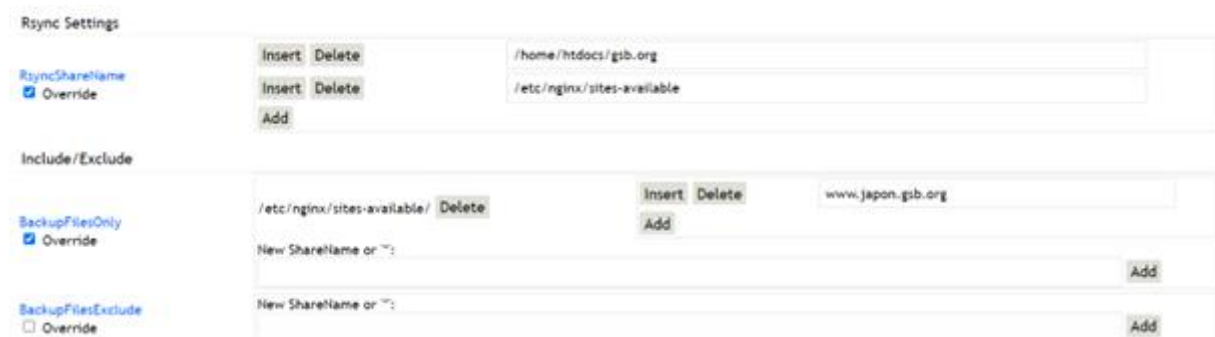
Attention : La clé backuppc du template est obsolète. Pour pouvoir faire marcher la connexion SSH, il faut supprimer l'ancienne clé dans le fichier `/home/backuppc/.ssh/authorized_keys`



Nous nous rendons ensuite sur l'interface graphique de backuppc (https://10.187.20.11/backuppc/) afin d'ajouter un nouvel hôte :



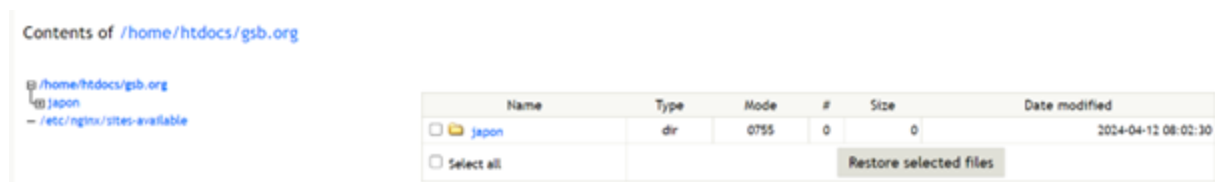
Nous entrons ensuite les fichiers et dossiers que nous souhaitons sauvegarder. Nous sauvegarderons le dossier /home/htdocs/gsb.org contenant le site Wordpress et le fichier de configuration /etc/nginx/sites-available/www.japon.gsb.org :



Nous commençons la sauvegarde :



Nous vérifions que les fichiers aient été correctement sauvegardés :



Restauration de fichiers

Nous allons maintenant tester la restauration de fichier. Pour ce faire, nous allons simuler une perte de données en supprimant le fichier `/etc/nginx/sites-available/www.japon.gsb.org` :

```
rm /etc/nginx/sites-available/www.japon.gsb.org
```

Nous vérifions que notre fichier soit bien supprimé :

```
ls /etc/nginx/sites-available/
```

```
root@wordpress-spl:/home/htdocs/gsb.org/japon/wordpress# rm /etc/nginx/sites-available/www.japon.gsb.org
root@wordpress-spl:/home/htdocs/gsb.org/japon/wordpress# ls /etc/nginx/sites-available/
default
```

Nous nous rendons dans l'onglet Browse backups de l'interface graphique de BackupPC puis cochons le fichier à restaurer.

Contents of `/etc/nginx/sites-available`

`-- /etc/nginx/sites-available`
`ls /home/htdocs/gsb.org`

	Name	Type	Mode	#	Size	Date modified
<input type="checkbox"/>	default	file	0644	0	2412	2023-03-14 16:19:32
<input checked="" type="checkbox"/>	www.japon.gsb.org	file	0644	0	1277	2024-04-12 08:08:45
<input type="checkbox"/>	Select all					

Restore selected files

Restore Options for 10.31.186.2

You have selected the following files/directories from share `/etc/nginx/sites-available`, backup number #0:

- `/www.japon.gsb.org`

You have three choices for restoring these files/directories. Please select one of the following options.

Option 1: Direct Restore

You can start a restore that will restore these files directly onto 10.31.186.2.

Warning: any existing files that match the ones you have selected will be overwritten!

Restore the files to host

10.31.186.2

Restore the files to share

`/etc/nginx/sites-available`

Restore the files below dir (relative to share)

`/`

Start Restore

Are you sure?

You are about to start a restore directly to the machine 10.31.186.2. The following files will be restored to share `/etc/nginx/sites-available`, from backup number 0:

Original file/dir	Will be restored to
<code>10.31.186.2:/etc/nginx/sites-available/www.japon.gsb.org</code>	<code>10.31.186.2:/etc/nginx/sites-available/www.japon.gsb.org</code>

Do you really want to do this?

Restore

No



Nous vérifions que le fichier soit bien restauré :

```
ls /etc/nginx/sites-available/
```

```
root@wordpress-sp1:/home/htdocs/gsb.org/japon/wordpress# ls /etc/nginx/sites-available/  
default  www.japon.gsb.org
```

Nous verrons dans la situation professionnelle suivante comment sécuriser et monitorer notre site web.

