

Document préparé par Dr Hamid MADANI

mailto: drmdh@msn.com

Date 08 mars 2024 Mostaganem

Objectif : collecter les adresses IP de divers serveurs qui ne sont pas équipés d'une adresse IP fixe.

Goal: Retrieve the various IP addresses of servers that do not have a fixed IP address.

## 1) Préparation SQL :

```
hmd@amia26:~$ sudo mysql -u root -p
```

```
mysql> create database ip_db;
```

```
mysql> use ip_db;
```

```
mysql>
```

```
CREATE TABLE table_ip (  
id INT AUTO_INCREMENT PRIMARY KEY,  
ip_address VARCHAR(15) NOT NULL,  
timestamp DATETIME NOT NULL,  
servername VARCHAR(255)  
);
```

```
mysql> select * from table_ip;
```

```
Empty set (0,01 sec)
```

```
mysql> select * from table_ip;
```

```
+-----+-----+-----+-----+  
| id | ip_address | timestamp          | servername |  
+-----+-----+-----+-----+  
| 115 | 41.97.83.38 | 2024-03-09 18:59:02 | NULL      |  
| 116 | 41.97.83.38 | 2024-03-09 19:59:01 | NULL      |  
| 117 | 41.97.83.38 | 2024-03-09 20:26:02 | amia26    |  
| 118 | 41.97.83.38 | 2024-03-09 20:27:02 | amia26    |
```

119	41.97.83.38	2024-03-09 20:28:01	amia26
120	41.97.83.38	2024-03-09 20:29:02	amia26
121	41.97.83.38	2024-03-09 20:30:02	amia26

+-----+-----+-----+-----+

7 rows in set (0,00 sec)

## 2) PHP sender

```
<?php
```

```
// Utilisation de l'API d'un service tiers pour obtenir l'adresse IP publique
```

```
$api_url = 'https://api64.ipify.org?format=json';
```

```
// Obtenez le nom d'hôte
```

```
$hostname = gethostname();
```

```
// Effectuer une requête HTTP pour récupérer les données
```

```
$response = file_get_contents($api_url);
```

```
$remote_url = 'https://ip.amia.fr/receivip.php';
```

```
// $remote_url = 'http://localhost/ip/receivip.php';
```

```
// Vérifier si la requête a réussi
```

```
if ($response !== false) {
```

```
// Convertir la réponse JSON en tableau associatif
```

```
// Décoder la réponse JSON en un tableau associatif
```

```
$data = json_decode($response, true);
```

```
// Ajouter une clé et une valeur au tableau associatif
```

```
$data['servername'] = $hostname;
```

```
// Re-encoder le tableau associatif en JSON
```

```
$new_response = json_encode($data);
```

```
// Vérifier si la conversion a réussi
```

```
if ($data !== null) {  
    // Récupérer l'adresse IP publique  
    $public_ip = $data['ip'];  
  
    $options = [  
        'http' => [  
            'method' => 'POST',  
            'header' => 'Content-type: application/json',  
            'content' => $new_response,  
        ],  
    ];  
  
    $context = stream_context_create($options);  
    $result = file_get_contents($remote_url, false, $context);  
  
    echo 'result : '.$result;  
  
    echo 'http://' . $public_ip."  
    // Afficher l'adresse IP publique  
    /*  
    echo '<a href="https://' . $public_ip . ':443">';  
    echo 'https://' . $public_ip."  
    echo '</a>';  
    echo '<br/>';  
  
    echo '<a href="http://' . $public_ip . ':80">';  
    echo 'http://' . $public_ip."  
    echo '</a>';  
    echo '<br/>';  
  
    echo 'http://' . $public_ip."  
    */
```

```
} else {  
    echo 'Erreur lors de la conversion de la réponse JSON.';  
}  
} else {  
    echo 'Erreur lors de la requête HTTP vers le service.';  
}  
?>
```

---

### 3) PHP Receiver

```
<?php
```

```
// Connexion à la base de données (à personnaliser avec vos propres informations de connexion)
```

```
$servername = "localhost";
```

```
$username = "htmc";
```

```
$password = "gjetfm26";
```

```
$dbname = "ip_db";
```

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Vérifier la connexion
```

```
if ($conn->connect_error) {
```

```
    die("Échec de la connexion à la base de données : " . $conn->connect_error);
```

```
}
```

```
// Recevoir les données JSON
```

```
$json_data = file_get_contents('php://input');
```

```
$data = json_decode($json_data, true);
```

```
if ($data !== null && isset($data['ip'])) {
```

```
    // Préparer la requête SQL d'insertion
```

```
    $ip_address = $conn->real_escape_string($data['ip']);
```

```

$servername = $conn->real_escape_string($data['servername']);

$timestamp = date('Y-m-d H:i:s'); // Obtenez le timestamp actuel

$sql = "";

if ($servername !== null ){

    $sql = "INSERT INTO table_ip (ip_address, timestamp, servername) VALUES ('$ip_address',
    '$timestamp','$servername')";

} else {

    $sql = "INSERT INTO table_ip (ip_address, timestamp) VALUES ('$ip_address', '$timestamp')";

}

// Exécuter la requête SQL

if ($conn->query($sql) === TRUE) {

    // Répondre avec succès

    $response = ['status' => 'success', 'message' => 'IP enregistrée avec succès'];

} else {

    // Gérer les erreurs d'insertion

    $response = ['status' => 'error', 'message' => 'Erreur lors de l'enregistrement de l'IP : ' . $conn->error];

}

} else {

    // Gérer les erreurs de décodage JSON ou de données manquantes

    $response = ['status' => 'error', 'message' => 'Données JSON non valides ou adresse IP manquante'];

}

// Fermer la connexion à la base de données

$conn->close();

// Retourner la réponse en format JSON

header('Content-Type: application/json');

echo json_encode($response);

```

?>

---

#### 4)Affichage de la table Table

Index.php ^ showip.php

<?php

// Connexion à la base de données (à personnaliser avec vos propres informations de connexion)

\$servername = "localhost";

\$username = "htmc";

\$password = "gjetfm26";

\$dbname = "ip\_db";

\$conn = new mysqli(\$servername, \$username, \$password, \$dbname);

// Vérifier la connexion

if (\$conn->connect\_error) {

die("Échec de la connexion à la base de données : " . \$conn->connect\_error);

}

// Supprimer les enregistrements obsolètes (par exemple, supprimer tous les enregistrements plus anciens que 24 heures)

\$expiry\_time = date('Y-m-d H:i:s', strtotime('-24 hours'));

\$sql\_delete = "DELETE FROM table\_ip WHERE timestamp < '\$expiry\_time'";

\$conn->query(\$sql\_delete);

// Traitement de la suppression si un identifiant est fourni dans l'URL

if (isset(\$\_GET['delete\_id'])) {

\$delete\_id = \$\_GET['delete\_id'];

\$sql\_delete = "DELETE FROM table\_ip WHERE id = \$delete\_id";

\$conn->query(\$sql\_delete);

}

// Requête SQL SELECT pour récupérer les adresses IP enregistrées

```

$sql = "SELECT id, ip_address, timestamp, servername FROM table_ip";

$result = $conn->query($sql);

if ($result->num_rows > 0) {
    // Afficher les résultats dans un tableau HTML
    echo "<table><tr><th>Server Name </th> <th>Adresse IP</th><th>Timestamp</th></tr>";

    while($row = $result->fetch_assoc()) {
        echo "<tr>" ;

        echo "<td>" . $row["servername"] . "</td>";

        echo "<td>" ;
        echo '<a href="https://" . $row["ip_address"] . ":443">';
        echo 'https://' . $row["ip_address"] . ":443";
        echo '</a>';
        echo "</td>";

        echo "<td>" ;
        echo '<a href="https://" . $row["ip_address"] . ":443">';
        echo 'http://' . $row["ip_address"] . ""';
        echo '</a>';
        echo "</td>";

        echo "<td>" . $row["timestamp"] . "</td>";

        echo "<td>";
        echo "<a href='?delete_id=" . $row["id"] . "'>X</a>";
        echo "</td>";

        echo "</tr>";

    }

    echo "</table>";
} else {

```

```
echo "Aucune adresse IP enregistrée dans la base de données.";
```

```
}
```

```
// Fermer la connexion à la base de données
```

```
$conn->close();
```

```
?>
```

-----

5) Pour automatiser l'exécution d'un script PHP à intervalles réguliers avec Cron, suivez ces étapes :

Ouvrez la crontab avec la commande suivante :

```
$ crontab -e
```

Ajoutez une ligne à votre crontab pour définir le moment où vous souhaitez que le script s'exécute. Par exemple, pour exécuter le script toutes les heures, ajoutez la ligne suivante :

```
$ which php
```

```
$ /usr/bin/php
```

```
MAILTO=drmdh@msn.com
```

```
59 * * * * /usr/bin/php /home/hmd/public_html/ip/sendip.php >> /home/hmd/public_html/ip/info.log  
2>&1
```

```
cat /home/hmd/public_html/ip/info.log
```

0 \* \* \* \* : Cela signifie que le script s'exécute toutes les heures (minute la 59 emme minute de chaque heure, tous les jours du mois, tous les mois, tous les jours de la semaine).

Si vous n'êtes pas sûr du chemin de votre interpréteur PHP, vous pouvez le trouver en exécutant la commande suivante dans votre terminal :