Building a Security Dashboard Using PHP and MySQL

In the previous tutorial, you used SQL queries to analyse imported JSON data. In this tutorial, you will build a dashboard that displays security statistics directly from the database.

The dashboard will display:

- Total Devices
- Total Events
- Critical Events
- High Events
- Recent Security Events

Create the Dashboard File

Create a new file called:

```
dashboard.php
```

Add the database connection:

```
<?php

$conn = new mysqli(
    "localhost",
    "root",
    "",
    "project_db"
);
```

```
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

?>
```

Get the Total Number of Events

Add:

```
$totalEventsQuery = "
    SELECT COUNT(*) AS total_events
    FROM security_events
";

$totalEventsResult =
    $conn->query($totalEventsQuery);

$totalEvents =
    $totalEventsResult->fetch_assoc()["total_events"];
```

Get the Number of Devices

Add:

```
$totalDevicesQuery = "
    SELECT COUNT(DISTINCT device_id)
    AS total_devices
    FROM security_events
";

$totalDevicesResult =
    $conn->query($totalDevicesQuery);

$totalDevices =
```

```
    $totalDevicesResult->fetch_assoc()["total_devices"];
```

Get Critical Events

Add:

```
$criticalEventsQuery = "
    SELECT COUNT(*) AS critical_events
    FROM security_events
    WHERE severity = 'critical'
";

$criticalEventsResult =
    $conn->query($criticalEventsQuery);

$criticalEvents =
    $criticalEventsResult->fetch_assoc()["critical_events"];
```

Get High Events

Add:

```
$highEventsQuery = "
    SELECT COUNT(*) AS high_events
    FROM security_events
    WHERE severity = 'high'
";

$highEventsResult =
    $conn->query($highEventsQuery);

$highEvents =
    $highEventsResult->fetch_assoc()["high_events"];
```

Create the Dashboard Page

Add the following HTML underneath the PHP code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Security Dashboard</title>
</head>
<body>

<h1>Security Dashboard</h1>

<div>
    <h2>Total Devices</h2>
    <p><?php echo $totalDevices; ?></p>
</div>

<div>
    <h2>Total Events</h2>
    <p><?php echo $totalEvents; ?></p>
</div>

<div>
    <h2>Critical Events</h2>
    <p><?php echo $criticalEvents; ?></p>
</div>

<div>
    <h2>High Events</h2>
    <p><?php echo $highEvents; ?></p>
</div>

</body>
</html>
```

Test the Dashboard

Open:

```
http://localhost/json-import/dashboard.php
```

Using the sample data, you should see something similar to:

```
Security Dashboard

Total Devices: 2
Total Events: 3
Critical Events: 1
High Events: 1
```

Display Recent Events

Add the following query before the HTML:

```
$recentEventsQuery = "
    SELECT *
    FROM security_events
    ORDER BY event_timestamp DESC
";

$recentEvents =
    $conn->query($recentEventsQuery);
```

Create the Events Table

Add the following underneath the dashboard cards:

```
<h2>Recent Events</h2>

<table border="1">
```

```
<tr>
    <th>Device</th>
    <th>Event Type</th>
    <th>Severity</th>
    <th>Timestamp</th>
</tr>

<?php

while (
    $row =
    $recentEvents->fetch_assoc()
) {

?>

<tr>
    <td><?php echo $row["device_id"]; ?></td>
    <td><?php echo $row["event_type"]; ?></td>
    <td><?php echo $row["severity"]; ?></td>
    <td><?php echo $row["event_timestamp"]; ?></td>
</tr>

<?php

}

?>

</table>
```

Refresh the page.

You should now see all imported events underneath the dashboard.

Add Basic Styling

Inside the `<head>` section add:

```
<style>

body {
    font-family: Arial, sans-serif;
}

.card {
    border: 1px solid #cccccc;
    border-radius: 8px;
    padding: 15px;
    margin-bottom: 10px;
}

table {
    border-collapse: collapse;
    width: 100%;
}

th,
td {
    border: 1px solid #cccccc;
    padding: 10px;
}

th {
    background-color: #f2f2f2;
}

</style>
```

Then update each dashboard `<div>`:

```
<div class="card">
```

The dashboard should now look much more professional.

Complete Dashboard Features

Your dashboard now:

- Connects to MySQL
- Runs SQL queries
- Displays summary statistics
- Displays recent events
- Updates automatically whenever new JSON data is imported

This is one of the major advantages of importing JSON into a database rather than displaying the JSON directly.

Next tutorial: **Creating Filters and Search Options for the Dashboard**.

Revision #1
Created 2026-06-10 01:58:21 UTC by Mr Napper
Updated 2026-06-10 01:58:26 UTC by Mr Napper