# Apache mod_userdir Tweak

***For cPanel & WHM version 64***

(*Home >> Security Center >> Apache mod_userdir Tweak*)

## Overview

This interface allows you to disable the Apache `mod_userdir` module's functionality for your users.

> **Warnings:**
>
> - We **strongly** recommend that you restrict this access for most of your users. Before you use this interface, make certain that you read the Security Implications and Warnings sections below.
> - If you enable Apache's `ruby24-mod_passenger` module in WHM's *Apache mod_userdir Tweak* ( *Home >> Software >> EasyApache 4* ), the system disables Apache's `mod_userdir` module by default.

## The Apache `mod_userdir` module

The Apache `mod_userdir` module allows for visitors to access a user's website via a URL that contains that user's username. For example:

```
https://host.example.com/~username
https://example.net/~username
https://192.168.0.20/~username
```

Most servers use the Apache `mod_userdir` module as a temporary URL system that allows users to view their websites. This temporary URL system functions even if the system does not possess configured DNS or the domain does not yet point to the server.

> **Note:**
>
> When you enable the Apache `mod_userdir` module, any virtual host can access any website that uses the same IP address. It does **not** function only with the hostname.

## Enable `mod_userdir` access

To enable `mod_userdir` access, perform the following steps:

1. Select the *Enable mod_userdir Protection* checkbox.
2. To enable `mod_userdir` functionality for specific hosts, select the appropriate *Exclude Protection* checkboxes.

    > **Important:**
    >
    > This action allows all users to access content on the host via the Apache `mod_userdir` module. We recommend that you only enable `mod_userdir` functionality on the DefaultHost.

3. To only allow `mod_userdir` functionality for specific additional users to access these hosts, enter their usernames in the *Additional Users* text box.
    - Resellers can use this feature to allow their customers to access their own websites before DNS information propagates.
    - To enter multiple users, separate each account name with a space.
4. Click *Save*.

**Notes:**

To allow your users to access their **own** accounts through the `mod_userdir` module, but not circumvent any bandwidth limits, select the *Exclude Protection* checkbox for the *DefaultHost (nobody)* host.

**Warning:**

Do **not** select the *Exclude Protection* checkbox on a user's domain if you only wish to allow an individual user to access their site with a `mod_userdir` URL.

### Example

You own the following three cPanel accounts:

- Arthur's cPanel account (`arthur`) owns `arthurexample.com`
- Betty's cPanel account (`betty`) owns `bettyexample.com`
- Charles' cPanel account (`charles`) owns `charlesexample.com`

Arthur’s domain resolves, but Betty’s and Charles’ domains do not yet resolve.

To enable `mod_userdir` protection for the server to deny one user the ability to use another user's bandwidth, select the *Enable mod_userdir Protection* checkbox.

However, if you still want to allow Betty and Charles to use Arthur’s domain to see their sites, perform the following steps:

1. Do **not** select the checkbox next to `arthurexample.com` (Arthur)
2. Enter `betty charles` in the *Additional Users* text box.
3. Click *Save*.

Betty and Charles can browse their sites with the following URLs:

- `https://arthurexample.com/~betty`
- `https://arthurexample.com/~charles`

### Shared SSL Certificates

If a shared SSL certificate exists for a virtual host on a shared IP address, you can share that SSL certificate with users on the same IP address. This allows users to access their sites securely without a browser warning.

For example, if an SSL certificate exists on `host.example.com`, select the *Exclude Protection* checkbox for the *DefaultHost (nobody)* host. This allows the `username` cPanel user to access the `https://host.example.com/~username` url.

## Security Implications

We **strongly** recommend that you restrict `mod_userdir` functionality for most of your users. `mod_userdir` can expose potential security issues.

- Any user can access web content on domains that share an SSL certificate, which potentially allows one cPanel account to leech bandwidth from another cPanel account.
- The system accounts for bandwidth per-host rather than per-user. If a user access another user's content via `mod_userdir`, then the server will not record their bandwidth usage correctly. This can also potentially allow for one user to use the bandwidth of another.

When you disable `mod_userdir` protection for a host, we recommend that you do **not** exclude the entire host, but rather exclude only specific users via the "Additional Users" field.

## Warnings

### Enabled mod_userdir protection

Before you enable the Apache `mod_userdir` module, be aware of the following information:

- Java servlets do **not** work with `mod_userdir`-based URLs. This is because Tomcat requires you to add additional directives to the virtual host.

  **Important:**

  EasyApache 3 does **not** support Tomcat for new installations. EasyApache 4 does **not** support Tomcat and we do **not** plan to provide support in the future.

- The CGI, FastCGI, and PHP-FPM PHP handlers do **not** allow you to use the Apache `mod_userdir` module. Select the tabs below for directions on how to use this module:

  To use Apache's `mod_userdir` module, perform the following actions in WHM's *Apache mod_userdir Tweak* interface (*Home >> Security Center >> Configure PHP and suEXEC*):

  - Set the *PHP Handler 5* option to *suphp*.
  - Set the *Apache suEXEC* option to *off*.

- `open_basedir` protection restricts PHP's access to the home directory of the user who owns the base domain, **not** the home directory of the user account that a visitor accesses. If you enable `open_basedir` protection in WHM's *Apache mod_userdir Tweak* interface (*Home >> Security Center >> PHP open_basedir Tweak*), visitors **cannot** access some sites via the `mod_userdir` module.
- Websites that use the `mod_rewrite` or other directives in their `.htaccess` files will **not** function correctly when visitors view them through `mod_userdir` URLs.
- If you enable Apache's `mod_ruid2` module, then the `mod_userdir` module will **not** function correctly. For more information, read our Apache mod_userdir Tweak documentation.

  > **Warning:**
  >
  > Under certain conditions, a user can attack another user's account if they access a malicious script through a `mod_userdir` URL.

## The Symlink Race Condition Protection option

The following table describes when the *Symlink Race Condition Protection* option blocks `mod_userdir` access:

| Condition | `mod_userdiraccess` | Example URL |
|---|---|---|
| The requested URL includes a file and does not belong to the owner of the file. | Blocked. | `example.com/~username/file` |
| The requested URL includes a file and an IP address that belongs to another account. | Blocked. | `192.168.0.20/~username/file` |
| The requested URL contains a directory. | Not blocked. | `example.com/~username/dir` |
| You wish to access the server's hostname. | Not blocked. | `host.example.com/~username` |

## Disabled mod_userdir protection

Before you disable `mod_userdir` protection, be aware of the following information:

- While this WHM feature allows you to restrict `mod_userdir` functionality, it does **not** remove the module itself. Some PCI compliance scans may still detect it.
- This feature does **not** list IP addresses because the `mod_userdir` module uses virtual hosts.
  - You **cannot** use IP addresses to configure this feature.
  - If you do not protect the default host, you can access the server's main IP address through the `mod_userdir` module in **most** cases.
  - If you attempt to provide protection on a dedicated IP address, the site's contents will still display when protection is enabled. To disable this behavior, open the `/etc/apache2/conf.d/includes/post_virtualhost_global.conf` file with a text editor and add the following line:

    ```
    UserDir=disabled
    ```

# Additional documentation

- Apache mod_userdir Tweak
- SSL Storage Manager
- PHP open_basedir Tweak
- ModSecurity Vendors
- Apache Status