

<?php

```
// -----  
// VISITOR IP BLOCKER BY AARON PACKARD, PMP  
// -----  
// For use when blocking pesky users and subnets of IP networks. Originally built on  
// 30 June 2006 and refined on 08 July 2006. Ported and rewritten for PHP in  
// February 2012. (c)2006-2012 - Aaron Packard, PMP http://www.aaronpackard.net/  
// Weasel with Circle and Slash image is a registered trademark owned by Aaron Packard  
// -----  
// BACKGROUND  
// -----  
// The author of this script was unable find easily obtainable pre-written PHP  
// code to perform this function. The author created the necessary code and is  
// offering it to anyone at no cost. This code snippet may be freely distributed  
// and used by anyone. It should be noted that the author is not an expert PHP  
// programmer, and if another person can develop more efficient code - good for  
// them!  
// -----  
// GENERAL SETUP INSTRUCTIONS  
// -----  
// 1 Cut and paste this into a suitably named PHP file  
// 2 Ensure that a suitable redirect page has been created for banned visitors  
// 3 Calls to this script must be at the absolute top of every page that is intended  
//   to call this script, via the php require statement.  
// 4 Except copyright notice and credit, all other commenting within this script may  
//   be removed  
// 5 To improve efficiency, it is recommended that the script be compressed  
// -----  
// SUPPORT  
// -----  
// You may contact the author of this script with questions  
// apackard.at.aaronpackard.net  
// -----  
// DISCLAIMER  
// -----  
// NO WARRANTIES, EXPRESSED OR IMPLIED, ARE PROVIDED BY THE AUTHOR OF THIS SCRIPT  
// The author is under no obligation, legal or otherwise, to respond to inquiries  
// -----  
// Below is where IP ranges intended to be banned from your site should be indicated  
// The syntax is 'begin address octets:end address octets'  
// Each range must be separated with a semi-colon (;)  
// To block a single address, repeat all octets after the colon  
// - example: '192.168.1.128:192.168.1.128'  
// To block an entire Class-A block, use the range 0:255  
// - example: '192.0.0.0:192.255.255.255'  
// To block an entire Class-B block, use the range 0:255  
// - example: '192.168.0.0:192.168.255.255'  
// To block an entire Class-C block, use the range 0:255  
// - example: '192.168.1.0:192.168.1.255'  
// To block multiple blocks, use the range x:y  
// - example: '192.168.12.0:192.168.22.255'  
// This syntax applies to all other network classes  
  
// Note that the ranges below are for example purposes and must be replaced with the  
// actual ranges that are intended to be banned  
$ranges = array('192.168.2.0:192.168.2.255', '10.1.0.0:10.30.255.255', '192.168.14.32:192.168.14.32');  
  
// Below is where the redirect page for banned visitors should be specified  
$BootPage = 'http://www.<yourdomain>.com/<path to page>/<pagename>.php';  
  
// -----
```

```

// TO MAINTAIN INTENDED FUNCTIONALITY, ALL CODE BELOW SHOULD NOT BE MODIFIED
// -----
// Grab the visitor IP and split up into an array of octet chunks:
$socket = explode(".", $_SERVER['REMOTE_ADDR']);

// Run comparison for each of the ranges
foreach ($ranges as $range)
{
    $addresses = explode(":", $range);
    $begins = explode(".", $addresses[0]);
    $ends = explode(".", $addresses[1]);

    //if visitor IP falls in between a Class-A range
    if (intval($socket[0]) > intval($begins[0]))
    {
        if (intval($socket[0]) < intval($ends[0])) //Vistor restricted
        {
            header("Location: " . $BootPage); //Boot-em
            exit();
        }
    } //Move onto next comparison

    //if visitor IP matches a Class-A BEGIN point and falls in between Class-B ranges
    if (intval($socket[0]) == intval($begins[0]))
    {
        if (intval($socket[1]) > intval($begins[1]))
        {
            if (intval($socket[1]) < intval($ends[1])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    } //Move onto next comparison

    //if visitor IP matches a Class-A END point and falls in between Class-B ranges
    if (intval($socket[0]) == intval($ends[0]))
    {
        if (intval($socket[1]) > intval($begins[1]))
        {
            if (intval($socket[1]) < intval($ends[1])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    }

    //if visitor IP matches a Class-A BEGIN point, matches a Class-B BEGIN point, and falls in between Class-
    C range
    if (intval($socket[0]) == intval($begins[0]))
    {
        if (intval($socket[1]) == intval($begins[1]))
        {
            if (intval($socket[2]) < intval($begins[2]))
            {
                if (intval($socket[2]) > intval($ends[2])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
}

```

```

    }
} //Move onto next comparison

//if visitor IP matches a Class-A BEGIN point, matches a Class-B BEGIN point, matches a Class-C BEGIN
point, and falls within a range
if (intval($socket[0]) == intval($begins[0]))
{
    if (intval($socket[1]) == intval($begins[1]))
    {
        if (intval($socket[2]) == intval($begins[2]))
        {
            if (intval($socket[3]) >= intval($begins[3]))
            {
                if (intval($socket[3]) <= intval($ends[3])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //Move onto next comparison

//if visitor IP matches a Class-A BEGIN point, matches a Class-B BEGIN point, matches a Class-C END
point, and falls within a range
if (intval($socket[0]) == intval($begins[0]))
{
    if (intval($socket[1]) == intval($begins[1]))
    {
        if (intval($socket[2]) == intval($ends[2]))
        {
            if (intval($socket[3]) >= intval($begins[3]))
            {
                if (intval($socket[3]) <= intval($ends[3])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //Move onto next comparison

//if visitor IP matches a Class-A BEGIN point, matches a Class-B END point, and falls in between Class-C
range
if (intval($socket[0]) == intval($begins[0]))
{
    if (intval($socket[1]) == intval($ends[1]))
    {
        if (intval($socket[2]) >= intval($begins[2]))
        {
            if (intval($socket[2]) <= intval($ends[2])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    }
} //Move onto next comparison

```

//if visitor IP matches a Class-A BEGIN point, matches a Class-B END point, matches a Class-C BEGIN point, and falls within a range

```
if (intval ($socket[0]) == intval ($begins[0]))
{
    if (intval ($socket[1]) == intval ($ends[1]))
    {
        if (intval ($socket[2]) == intval ($begins[2]))
        {
            if (intval ($socket[3]) >= intval ($begins[3]))
            {
                if (intval ($socket[3]) <= intval ($ends[3])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //Move onto next comparison
```

//if visitor IP matches a Class-A BEGIN point, matches a Class-B END point, matches a Class-C END point, and falls within a range

```
if (intval ($socket[0]) == intval ($begins[0]))
{
    if (intval ($socket[1]) == intval ($ends[1]))
    {
        if (intval ($socket[2]) == intval ($ends[2]))
        {
            if (intval ($socket[3]) >= intval ($begins[3]))
            {
                if (intval ($socket[3]) <= intval ($ends[3])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //Move onto next comparison
```

//if visitor IP matches a Class-A END point, matches a Class-B BEGIN point, and falls in between Class-C range

```
if (intval ($socket[0]) == intval ($ends[0]))
{
    if (intval ($socket[1]) == intval ($begins[1]))
    {
        if (intval ($socket[2]) < intval ($begins[2]))
        {
            if (intval ($socket[2]) > intval ($ends[2])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    }
} //Move onto next comparison
```

//if visitor IP matches a Class-A END point, matches a Class-B BEGIN point, matches a Class-C BEGIN point, and falls within a range

```
if (intval ($socket[0]) == intval ($ends[0]))
{
```

```

if (intval ($socket[1]) == intval ($begins[1]))
{
    if (intval ($socket[2]) == intval ($begins[2]))
    {
        if (intval ($socket[3]) >= intval ($begins[3]))
        {
            if (intval ($socket[3]) <= intval ($ends[3])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    }
} //Move onto next comparison

```

//if visitor IP matches a Class-A END point, matches a Class-B BEGIN point, matches a Class-C END point, and falls within a range

```

if (intval ($socket[0]) == intval ($ends[0]))
{
    if (intval ($socket[1]) == intval ($begins[1]))
    {
        if (intval ($socket[2]) == intval ($ends[2]))
        {
            if (intval ($socket[3]) >= intval ($begins[3]))
            {
                if (intval ($socket[3]) <= intval ($ends[3])) //Vistor restricted
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //Move onto next comparison

```

//if visitor IP matches a Class-A END point, matches a Class-B END point, and falls in between Class-C range

```

if (intval ($socket[0]) == intval ($ends[0]))
{
    if (intval ($socket[1]) == intval ($ends[1]))
    {
        if (intval ($socket[2]) < intval ($begins[2]))
        {
            if (intval ($socket[2]) > intval ($ends[2])) //Vistor restricted
            {
                header("Location: " . $BootPage); //Boot-em
                exit();
            }
        }
    }
} //Move onto next comparison

```

//if visitor IP matches a Class-A END point, matches a Class-B END point, matches a Class-C BEGIN point, and falls within a range

```

if (intval ($socket[0]) == intval ($ends[0]))
{
    if (intval ($socket[1]) == intval ($ends[1]))
    {
        if (intval ($socket[2]) == intval ($begins[2]))
        {

```

```

    if (intval ($socket[3]) >= intval ($begins[3]))
    {
        if (intval ($socket[3]) <= intval ($ends[3])) //Visitor restricted
        {
            header("Location: " . $BootPage); //Boot-em
            exit();
        }
    }
}
} //Move onto next comparison

```

//if visitor IP matches a Class-A END point, matches a Class-B END point, matches a Class-C END point, and falls within a range:

```

if (intval ($socket[0]) == intval ($ends[0]))
{
    if (intval ($socket[1]) == intval ($ends[1]))
    {
        if (intval ($socket[2]) == intval ($ends[2]))
        {
            if (intval ($socket[3]) >= intval ($begins[3]))
            {
                if (intval ($socket[3]) <= intval ($ends[3])) //Visitor restricted!
                {
                    header("Location: " . $BootPage); //Boot-em
                    exit();
                }
            }
        }
    }
} //End of comparisons - if user made it this far, they arent on the ban list
}
?>

```