

Deploying Monowall with Riverbed RSP

Deployment Guide

INSTALLATION INSTRUCTION

DEPLOYING M0N0WALL ON RIVERBED SERVICE PLATFORM (RSP)

COMMUNITY.RIVERBED.COM TERMS AND CONDITIONS

PLEASE READ THESE TERMS AND CONDITIONS CAREFULLY. By using Riverbed Technology Inc.'s ("Riverbed") website located at <http://www.riverbed.com> (including, without limitation, any associated forums such as community.riverbed.com (the "Forum") (collectively, the "Website"), you agree to be bound by the following terms and conditions ("Terms and Conditions"). Riverbed reserves the right to modify these Terms and Conditions from time to time without notice. Please review these Terms and Conditions from time to time so that you will be apprised of any changes.

The terms and conditions are available at <http://community.riverbed.com/t5/Community-Packages/RSP-Community-Terms-and-Conditions/td-p/2188>

Introduction

The objective of this deployment guide is to show how to create your own RSP package and to choose the correct deployment type. After going through this deployment guide, you will see the possibility with RSP is endless

M0n0wall is a free firewall package that is based on BSD and it was chosen because of its small foot print in terms of memory and disk size.

More information on m0n0wall is available by going here: <http://m0n0.ch/wall/>. The m0n0wall package used in this deployment guide has been modified to include an extra NIC for management purpose.

Further, we will look into specific deployment types and how to configure the data flow rules to redirect the traffic to the package.

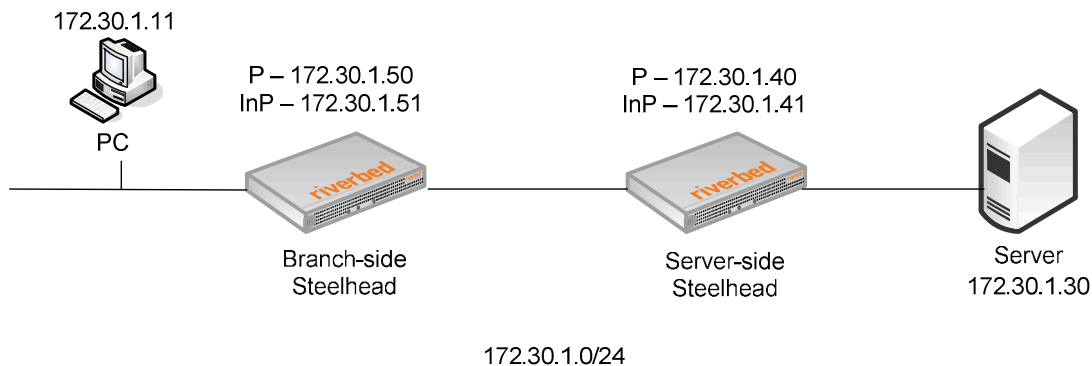
Software required for deployment of m0n0wall package

Unless otherwise noted, all the software can be downloaded from the Support website

- VMWare Workstation or VMWare Server
- m0n0wall package
- RiOS 6.0 and later
- RSP image for the Steelhead
- RSP license
- RSP Package Creator
- Putty SCP

Network Topology

Topology shown below is used as an example for particular deployment guide. Note that there is no Network Nightmare in this topology and that it is a bridged network.



Pre-requisite for deploying m0n0wall on Riverbed RSP

- Download and unzip the m0n0wall package to `c:\rsp\packages\m0n0wall` (note “zeros” and not alphabet “O’s” for m0n0wall)
- Place the Putty SCP in the `c:\rsp\packages` folder
- Place the RSP Package Creator in the `c:\rsp\packages` folder
- Install VMWare Workstation or VMWare Server on the client PC
- Install the RSP license on the Branch-side Steelhead

Part 1

Using the RSP Package Creator to create the package

Convert the growable VMDK to a pre-allocated VMDK

Creating VMWare images with growable VMDK is a common practice as it doesn't require the allocation of disk space at the time of creation. The current version of RSP 6.0 does not support growable VMDK, but it does allow ability to extend a VMDK. If there are VMWare images that use growable VMDK, it will need to be converted them to pre-allocated VMDK using the `vmware-vdiskmanager`.

`Vmware-vdiskmanager` is a tool that comes with VMWare Workstation/Server/ESX. This tool is not available with VMWare Player.

The author of m0n0wall created the package using a growable VMDK and therefore it needs to be converted to a pre-allocated VMDK.

1. Open a command prompt window
2. Navigate to the `c:\rsp\packages\m0n0wall` directory (note “zeros” and not alphabet “O’s” for m0n0wall)
3. Run the following command to convert the growable VMDK to pre-allocated VMDK:
`vmware-vdiskmanager -r m0n0wall.vmdk -t 2 m0n0wall2.vmdk`

`Vmware-vdiskmanager` should be in the search path. If that is not the case, specify the absolute path

4. Rename the m0n0wall2.vmdk to m0n0wall.vmdk:
`vmware-vdiskmanager -n m0n0wall2.vmdk m0n0wall.vmdk`

Running the renaming command will overwrite the original VMDK. This is intentional.

You should see, amongst other files, two files in the directory: m0n0wall.vmdk and m0n0wall-flat.vmdk. The m0n0wall.vmdk file should be < 1K, and the m0n0wall-flat.vmdk should be approximately 27MB.

Note: While converting the growable VMDK to a pre-allocated VMDK if follow error occurs: **Failed to rename: The file already exists (0x27000000c).**

Deploying M0n0wall with Riverbed RSP

Please follow below steps:

When you rename the VMDK, use the following command: `C:\rsp\packages\m0n0wall>vmware-vdiskmanager -n m0n0wall2.vmdk "...m0n0wall.vmdk"`

You should then see two files: **m0n0wall.vmdk** and **m0n0wall-flat.vmdk** in your `c:\rsp\packages` directory.

Now, go and delete the `m0n0wall.vmdk` in the `c:\rsp\packages\m0n0wall` directory (the original 9MB file) and then copy the `m0n0wall.vmdk` and `m0n0wall-flat.vmdk` from the `c:\rsp\packages\` into the `c:\rsp\packages\m0n0wall` directory.

Creating the package using the RSP Package Creator

While it's possible to create the RSP package manually, it is much easier to do so by using the RSP Package Creator. The RSP Package Creator creates the `rsp.conf` file and compresses the relevant VMDK into a single package. The `rsp.conf` file is based on the associated VMX file.

1. Launch the "RSP Package Creator" in `c:\rsp\packages`
2. In the "Virtual Machine Folder", navigate to `c:\rsp\packages\m0n0wall` and click Next
3. Under "General Preferences", enter the following information:
 - a. Name: **m0n0wall**
 - b. Version: **1.31**
 - c. Description: **m0n0wall**
4. Leave "Enable Watchdog" to "No" and click Next
5. Under "Management Interfaces", click Add
6. Enter the following information:
 - a. "Interface Name", type in **m0n0wall-mgmt**
 - b. "Virtual Interface", select **VM Network Adaptor 0** and click OK
7. Under "Optimization Interfaces", click Add
8. Enter the following information
 - a. "Interface Name", type in **m0n0wall-wan**
 - b. "Interface Type", select **wan**
 - c. "Virtual Interface", select **VM Network Adaptor 1**
9. Leave everything else in its default configuration and click OK
10. Repeat step 7 – 9, but enter the following information
 - a. "Interface Name", type in **m0n0wall-lan**
 - b. "Interface Type", type in **lan**
 - c. "Virtual Interface", select **VM Network Adaptor 2**
11. Click Next
12. Under "Package File Name", type in `c:\rsp\packages\m0n0wall.pkg` and click on "Create Package"
13. Once the package has been created, acknowledge the notification message and click on "Finish"

Installing the RSP image

RSP image can be downloaded from Riverbed support website for installing in to Branch Steelhead.

1. Login to the Branch-side Steelhead
2. Navigate to Configuration > Branch Servers > RSP Service page
3. Under "Install RSP", click on "From Local File" and click "Browse"
4. Locate RSP image then click "Install". Wait for the installation to complete.
5. Click "Start" to enable the RSP service

Uploading the package to the Steelhead

There are 2 ways to upload a package to the Steelhead: browser upload and URL download.

Deploying M0n0wall with Riverbed RSP

Browser upload is the simplest method of the three but there is also a size limitation of 2GB. If the package is > 2GB, package cannot be uploaded with the help of this method.

If the package is > 2GB, image can be placed on a HTTP server and configure the RSP to download the image instead. However, image should be placed on a HTTP server.

Slotting and enabling the package

1. Navigate to Configure > Branch Services > RSP Packages
2. Under "Slots", click on slot 5
3. (Optional) For "Slot Name", can be replaced by the name of the slot to be something more meaningful. e.g. **m0n0wall**.
4. In the pull-down list for "Package File Name", choose **m0n0wall.pkg**
5. Click on **Update Slot**
6. If successful, message would appear confirming that package has been deployed successfully on slot.

Binding the package's management interface to the primary/aux interface of the Steelhead

Not every package has a management interface, but if it does, it can be choose to bind either with the primary or auxiliary interface. For m0n0wall, we will bind the management interface to the Steelhead's primary interface.

1. Navigate to Configure > Branch Services > RSP Packages
2. Under "Slots", click on **5**, or if slot is renamed, click on **m0n0wall**
3. In the section "Management Virtual Network Interfaces:", select the **primary** radio button under "Physical Interface"
4. Click on **Update Slot**
5. The message, "Change will take effect when slot is next powered on" will appear.

Power on the package

Once package is deployed in slot, it will not be automatically powered on. To power on the package, perform the following steps.

1. Navigate to Configure > Branch Services > RSP Packages
2. Under "Slots", click on **5** or **m0n0wall**
3. Click on **Enable Slot**
4. The message, "Slot "5" is now enabled" or "Slot "m0n0wall" is now enabled"

Part 2

Accessing and configuring the package

There is only one way to access the packages with the VMWare web console itself.

Accessing the package through the console

1. Navigate to Configure > Branch Services > RSP Packages
2. Under "Slots", click on **5** or **m0n0wall**
3. Click on **Launch VM Console**
4. A new window/tab will open. Login as **admin** and enter the password
5. Click **Login**
6. Install the browser plug-in. If so, install the plug-in and log back into the Steelhead.
7. Click in the black window to open the virtual machine.

Deployment scenarios

The M0n0wall for the RSP is an "in-path" package. This means that the Virtual Network Interfaces (VNIs) for this package will be in-path of the data flow to and from the branch office.

In-path/In-band package

Configuring the m0n0wall package

As mentioned previously, m0n0wall is a stateful firewall based on BSD. M0n0wall also has the capability of inducing latency into the network – similar to the function of Network Nightmare.

Note: Here we have configured m0n0wall with 100ms of latency and set the network bandwidth to 512Kbps.

1. Access the m0n0wall package via VMWare web console.
2. Following screen should appear

```
*** This is M0n0wall, version 1.3b15
    built on Sat Oct 11 18:48:10 CEST 2008 for generic-pc
    Copyright (C) 2002-2008 by Manuel Kasper. All rights reserved.
    Visit http://M0n0.ch/wall for updates.

    LAN IP address: 192.168.1.1

    Port configuration:

    LAN    -> em0
    WAN    -> em1

M0n0wall console setup
*****
1) Interfaces: assign network ports
2) Set up LAN IP address
3) Reset webGUI password
4) Reset to factory defaults
5) Reboot system
6) Ping host

Enter a number: █
```

3. Type **"1"** to assign network ports
4. Answer **"n"** to the question, "Do you want to set up VLANs now? (y/n)"
5. For LAN interface name, type in **em0** (e, m, numeric zero)
6. For WAN interface name, type **em1** (e, m, numeric one)
7. For Optional 1 interface name, type **Inc0** (lower-case alphabet "l" for lima, n, c, numeric zero)
8. Press <Enter> when it asks for the Optional 2 interface name
9. Answer **"y"** to the question, "Do you want to proceed". The firewall will reboot.

A note regarding m0n0wall's interfaces: the default m0n0wall package does not include a dedicated management interface. The m0n0wall VM was modified to include a third interface so that a dedicated interface can be used for management. M0n0wall uses the LAN interface as its management interface. Therefore, the interface maps as follow:

m0n0wall LAN – management interface (em0)
m0n0wall WAN – VNI WAN (em1)
m0n0wall Opt 1 – VNI LAN (Inc0)

10. Once the firewall comes back up, enter **"2"** to assign an IP address to the LAN interface.
11. Configure the Branch Steelhead with m0n0wall, use the IP address 172.30.1.42
12. Enter **"24"** for the number of bits
13. Enter **"n"** for enabling DHCP on the LAN

Deploying M0n0wall with Riverbed RSP

14. Rest of the configuration can be done through the client's browser.
15. RDP into a client PC. Launch the browser and navigate to the IP address which has been allocated to m0n0wall.
16. Login as "admin", password "mono" (alphabet "o" and not zeroes "0")
17. On the left hand side under "Interfaces (assign)", click on "OPT1".
18. Click on the checkbox for "Enable Optional 1 interface"
19. In the pull-down for "Bridge with", select "WAN". Click "Save"
20. On the left hand side under "Interfaces (assign)", click on WAN. Scroll to the bottom and uncheck the box for "Block private networks". Click "Save"
21. On the left hand side under "Firewall", click on "Rules"
22. Click on "WAN"
23. Click on the "+" sign
24. Make sure the "Interfaces" says "**WAN**" and change the "Protocol" to "**Any**". Click "Save"
25. On the same page, click on "OPT1"
26. Click on the "+" sign
27. Make sure the "Interfaces" says "**OPT1**" and change the "Protocol" to "**Any**". Click "Save"
28. Click on "Apply changes"
29. On the left hand side under "Firewall", click on "Traffic Shaper".
30. Check the box "Enable traffic shaper". Click "Save"
31. On the same page, click on "Pipes", then click on the "+" sign
32. For "Bandwidth", enter **512**. For "Delay", enter **50**. Click "Save"
33. Click "Apply changes"
34. Now it would be diverted to "Firewall:Traffic shaper:Rules" page. Click on the "+" sign.
35. For "Interface", choose "**OPT1**". For "Protocol", choose "**any**". Click "Save"
36. Click on either one of the "+" sign.
37. For "Interface", choose "**WAN**". For "Protocol", choose "**any**". Click "Save"
38. There should now be two rules in the list. Click "Apply Changes"
39. m0n0wall is now configured to induce 100ms of latency (50ms in each direction).
40. From the client, initiate a ping to the server's IP (172.30.1.30).

Configuring data flow rules for in-path packages

For in-path or virtual in-path packages, slotting and turning on the package will not cause the traffic to be redirected to the package itself. This is by design to prevent users from disrupting any existing traffic. Traffic can be redirected by configuring the data flow rules.

1. Navigate to Configuration > Branch Services > RSP Data Flow
2. Click "Add a VNI"
3. Select "m0n0wall:LAN" and place that at the end
4. Click "Add a VNI"
5. Select "m0n0wall:WAN" and place that at the end
6. RSP data flow should now look like this:
LAN > RiOS > m0n0wall-LAN > m0n0wall-WAN > WAN
7. Now, ping from client to the server.

Even after adding the package's VNI to the dataflow, the default behavior is to pass-through the traffic. To start redirecting traffic to the package, any one of two things can be done: change the default IP and non-IP policy to redirect all the traffic to the package; or create specific data flow rules to redirect the traffic that you're interested in and bypass all other traffic.

Data flow rules are similar to in-path rules. Data flow governs what traffic should be redirected to the package. Over here, default IP and non-IP policy are changed to redirect all traffic to the package.

Dataflow rules are VNI specific. This means different rules need to be created for each VNI depending on the direction of the traffic.

1. Click on the "m0n0wall-LAN" VNI
2. Change the default IP and non-IP policy to "redirect traffic to slot"
3. Click on the "m0n0wall-WAN" VNI

Deploying M0n0wall with Riverbed RSP

4. Change the default IP and non-IP policy to “redirect traffic to slot”
5. Ping the server from the client and latency should be 100ms

Configuring Watchdog

When deploying the Steelheads in-path, there is an option of choosing fail-to-wire or fail-to-block. The same concept applies to RSP packages. For in-path and virtual in-path packages, any one can be choose either fail-to-wire or fail-to-block if the package fails.

In this section, example for fail-to-wire feature is demonstrated.

1. Ping the server from the client and ensure the latency is ~ 100ms
2. Navigate to Configure > Branch Services > RSP Packages
3. Under slots, click on **5** or **m0n0wall**
4. Complete the configuration per the settings below:
 - a. Watchdog: Bypass on failure
 - b. Watchdog IP: 172.30.1.42
 - c. Watchdog Frequency: 1
 - d. Watchdog Timeout: 3
5. Click on “Update Slot”.
6. Initiate a continuous ping from the client to the server.
7. Access the m0n0all console via the console or VMWare Infrastructure client.
8. Select option “5” to reboot m0n0wall
9. Watch what happens to the latency as the firewall reboots. The latency should decrease from 100ms to LAN speed for the duration of the reboot.
10. Repeat the above steps for fail-to-block.