Malwarebytes

ENDPOINT SECURITY

Mac Remediation Client Command Line Administrator Guide

Version 1.3.1

27 September 2017

Malwarebytes

Notices

Malwarebytes products and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. You may copy and use this document for your internal reference purposes only.

This document is provided “as-is.” The information contained in this document is subject to change without notice and is not warranted to be error-free. If you find any errors, we would appreciate your comments; please report them to us in writing.

The Malwarebytes logo is a trademark of Malwarebytes. Windows is a registered trademark of Microsoft Corporation. All other trademarks or registered trademarks listed belong to their respective owners.

Third Party Project Usage

Malwarebytes software is made possible thanks in part to many open source and third party projects. A requirement of many of these projects is that credit is given where credit is due. Information about each third party/open source project used in Malwarebytes software – as well as licenses for each – are available on the following page.

https://www.malwarebytes.com/support/thirdpartynotices/

Sample Code in Documentation

The sample code described herein is provided on an "as is" basis, without warranty of any kind, to the fullest extent permitted by law. Malwarebytes does not warrant or guarantee the individual success developers may have in implementing the sample code on their development platforms. You are solely responsible for testing and maintaining all scripts.

Malwarebytes does not warrant, guarantee or make any representations regarding the use, results of use, accuracy, timeliness or completeness of any data or information relating to the sample code. Malwarebytes disclaims all warranties, express or implied, and in particular, disclaims all warranties of merchantability, fitness for a particular purpose, and warranties related to the code, or any service or software related there to.

The Malwarebytes Protection Strategy

Malwarebytes’ products incorporate several prevention features which utilize a layered defense strategy to protect you against malware threats which you face daily. Each layer is designed to disrupt the attack chain at a different stage. While all Malwarebytes products are highly effective in dealing with attacks that are becoming all too commonplace, our protection capabilities are most effective when you take advantage of the full product suite, allowing each prevention layer to do the job they are best suited for.

It’s your data. Protect it wisely!

CMB20-1031a

Table of Contents

Introduction

The Mac remediation client is designed to allow business users to detect and remove malware from endpoints. It is built upon the power of our *Malwarebytes Anti-Malware* anti-malware client.

Implementation in a portable form provides increased flexibility for IT staff to quickly and easily deploy the client, detect and remediate threats, gather activity logs, and continue with their daily tasks – all with a minimal investment in time and resources.

What’s New

The following changes have been made in this version of the Mac remediation client.

- Added ability to handle larger threat signatures
- Fixed a bug that could cause the program to be unable to scan
- Added ability to control whether browser extensions are included in scans
- Improved quarantine functionality

Key Features

The Mac remediation client offers the following key features:

- Ability to utilize existing Mac protection updates, assuring that even the newest threats can be detected
- Ability to quarantine detected threats
- Ability to deploy client to endpoints using your preferred methods
- Command line capabilities allow IT staff to modify certain program configuration settings, execute scans, and gather logs through integration with customer-supplied scripts, batch files, and group policy updates
- Client leaves no lasting footprint on endpoint

System Requirements

Following are minimum requirements for an endpoint on which the Mac remediation client may be installed. Please note that these requirements do not include other functionality that the endpoint is responsible for.

- **Operating System:** macOS version 10.9.5 or later.
- **Security & Privacy**: Allow apps to be downloaded from Mac App Store and identified developers
- **Active Internet Connection**, for license validation and protection updates
- **USB 2.0 Port** (optional, depending on deployment method)

External Access Requirements

If your company’s Internet access is controlled by a firewall or other access-limiting device, you must grant access for the Mac remediation client to reach Malwarebytes services. These are:

https://data.service.malwarebytes.org	Port 443	outbound
https://data-cdn.mbamupdates.com	Port 443	outbound
https://*.mwbsys.com	Port 443	outbound

Remote Operations

The Mac remediation client can perform its role as a program which is locally installed and operated, or as a program which is remotely deployed and remotely executed. Many system administrators prefer to deploy and operate from a central location, so they can ensure a malware-free working environment and control the methods that are used. The two primary functions covered here are:

- **Deployment –** Installation, registration and updates of the program on a target endpoint
- **Execution & Remediation –** Scanning the target endpoint for malware threats

Please note that Malwarebytes cannot know which deployment tools that a customer currently uses (if any). For that reason, the required commands are listed here, with the expectation that the customer can supply the appropriate "wrapper" to allow these commands to work in conjunction with their deployment tools.

Please note that all examples and samples provided here are based on `mbbr` being installed into the endpoint's `/usr/local/bin/` directory. While `mbbr` can be installed and executed from any location in the file system, this specific location allows you to run it without providing a path to the executable file.

Manual deployment

Manual deployment requires having access to an admin user account on the target Mac endpoint, and having that account configured to allow remote login (in **System Preferences > Sharing**). In the instructions below, replace `[adminuser]` with the username of that admin user and `[dest_ip]` with the IP address of that target Mac endpoint.

Deployment

Open **Terminal** on your Mac, then run the following commands:

```
scp /path/to/mbbr-mac.pkg [adminuser]@[dest_ip]:~/
ssh [adminuser]@[dest_ip]
sudo installer -pkg mbbr-mac.pkg -target /
```

Be sure to provide the correct path to the mbbr-mac.pkg file on your endpoint in the first command.

Execution

To run Mac remediation client commands on the target Mac, establish a secure shell connection (if you do not still have one open):

```
ssh [adminuser]@[dest_ip]
```

Next, in the secure shell, enter commands like the following (replacing [`prodKey`] with a valid license key):

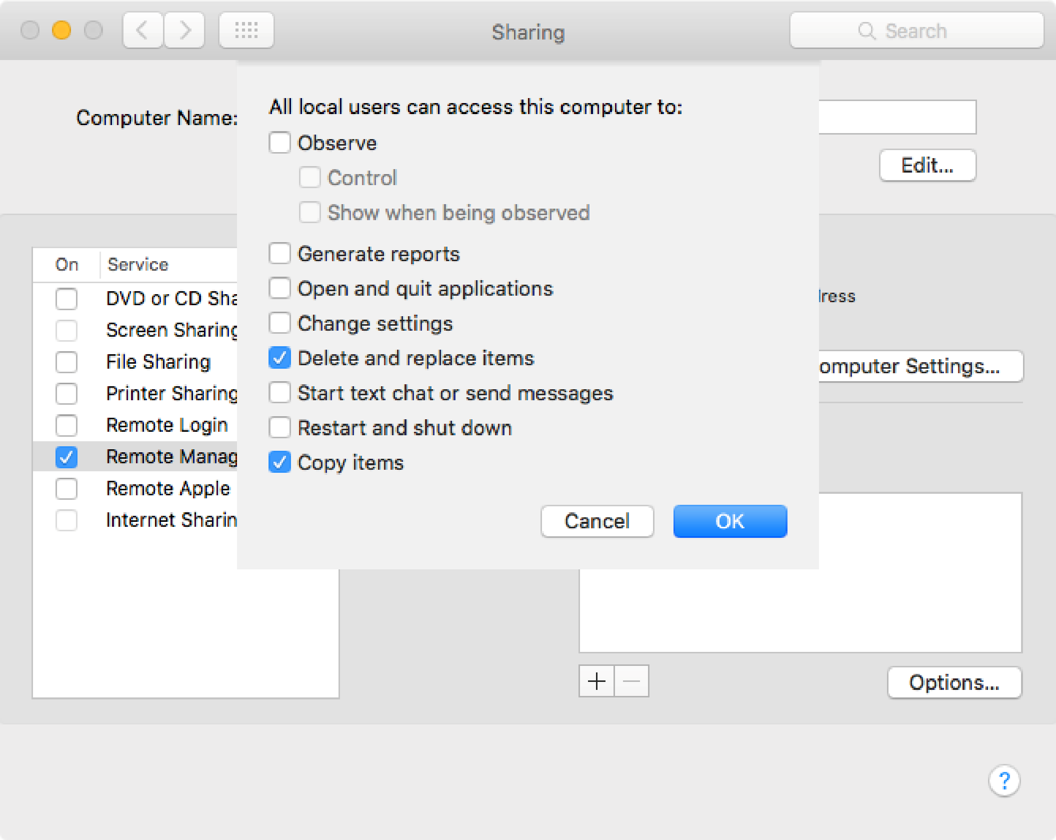
```
sudo mbbr register -key:[prodKey]
sudo mbbr update
sudo mbbr scan
```

Note the use of `sudo`; all remediation client commands should be run with root permissions.

Deployment using Apple Remote Desktop

Deployment via Apple Remote Desktop (ARD) can be done easily using either the **`mbbr-mac.pkg`** installer or the **`mbbr`** executable file. Which you use will depend on your preferences.

Installation on client machines requires that **Remote Management**, in **System Preferences -> Sharing** on the target machines, is allowed to Copy items and Delete and replace items. A screenshot is shown below.

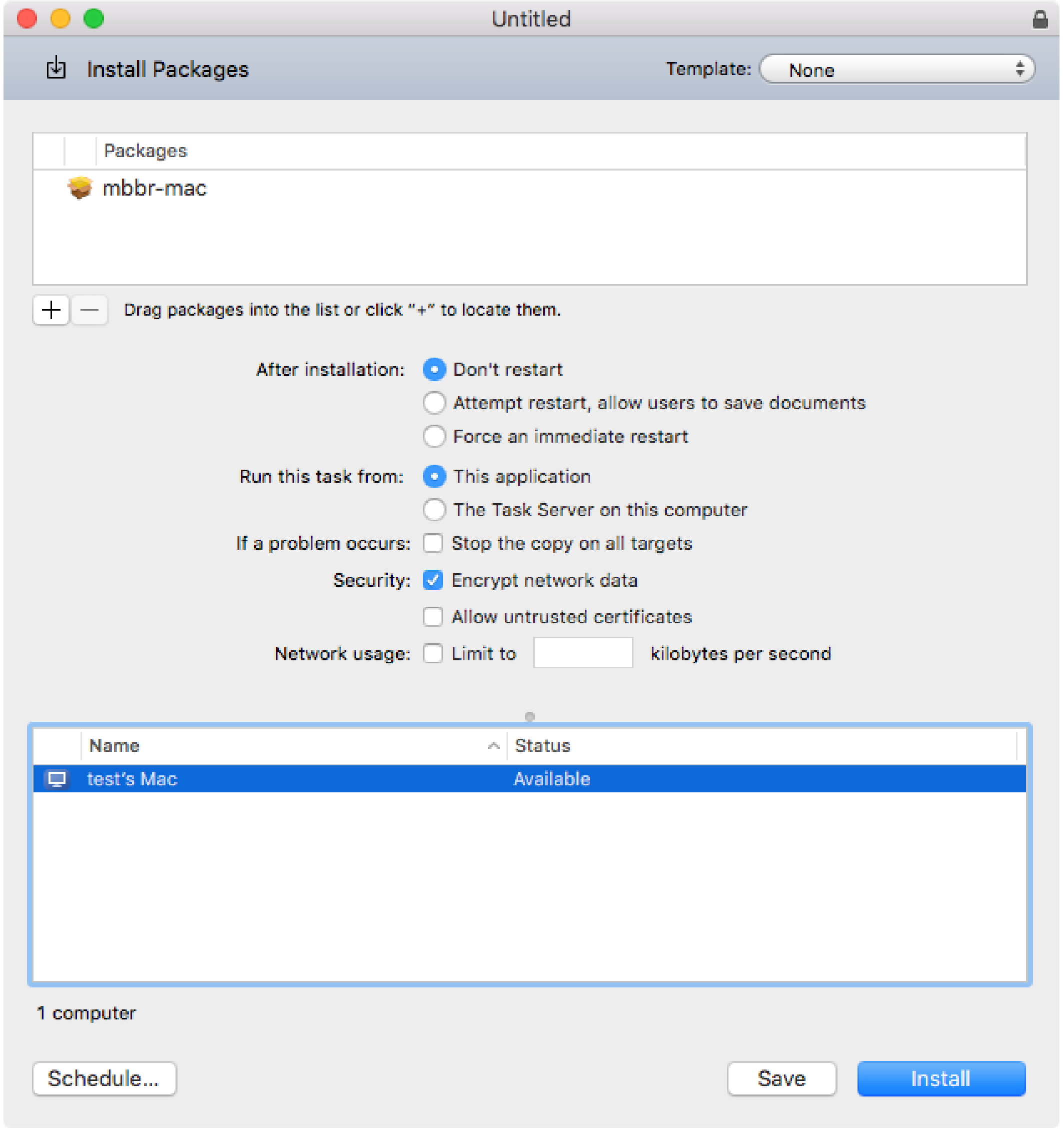


All local users can access this computer to:

- ☐ Observe
  - ☐ Control
  - ☐ Show when being observed
- ☐ Generate reports
- ☐ Open and quit applications
- ☐ Change settings
- ☑ Delete and replace items
- ☐ Start text chat or send messages
- ☐ Restart and shut down
- ☑ Copy items

Deployment

To deploy using the `mbbr-mac.pkg` installer, select a group of machines and use **Manage > Install Packages** to install `mbbr-mac.pkg` on those Macs. The Mac remediation client will be installed in `/usr/local/bin/`.



Drag packages into the list or click "+" to locate them.

After installation: ● Don't restart
○ Attempt restart, allow users to save documents
○ Force an immediate restart

Run this task from: ● This application
○ The Task Server on this computer

If a problem occurs: □ Stop the copy on all targets

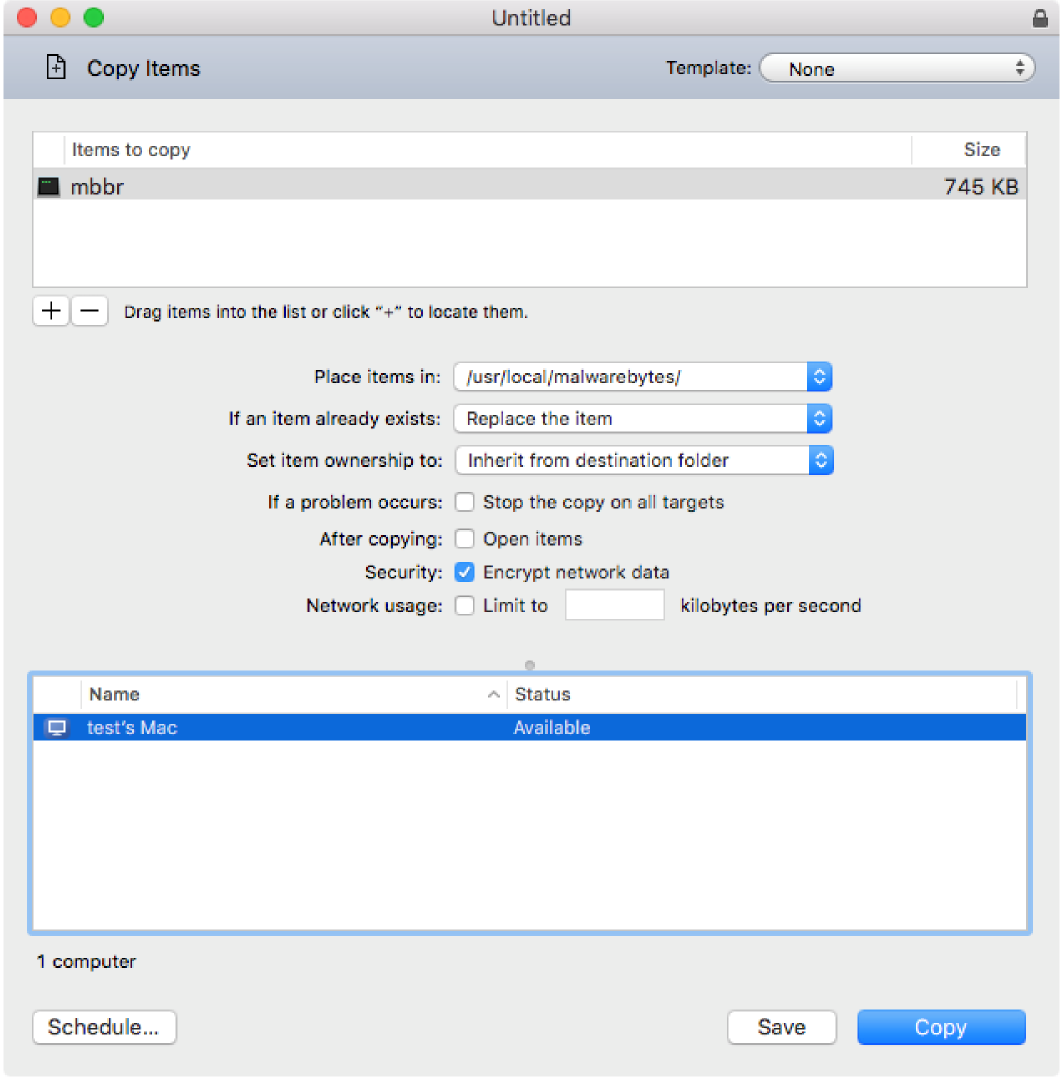
Security: ☑ Encrypt network data
□ Allow untrusted certificates

Network usage: □ Limit to ____ kilobytes per second

Name	Status
test's Mac	Available

1 computer

Alternately, because Apple Remote Desktop does not include `/usr/local/bin` in the PATH environment variable, and thus you will have to enter the path to the executable in each command if it's installed there, you can use **Manage -> Copy** Items and place the mbbr executable in any location you like.



Template: None

Items to copy	Size
mbbr	745 KB

Drag items into the list or click "+" to locate them.

Place items in: /usr/local/malwarebytes/

If an item already exists: Replace the item

Set item ownership to: Inherit from destination folder

If a problem occurs: □ Stop the copy on all targets

After copying: □ Open items

Security: ☑ Encrypt network data

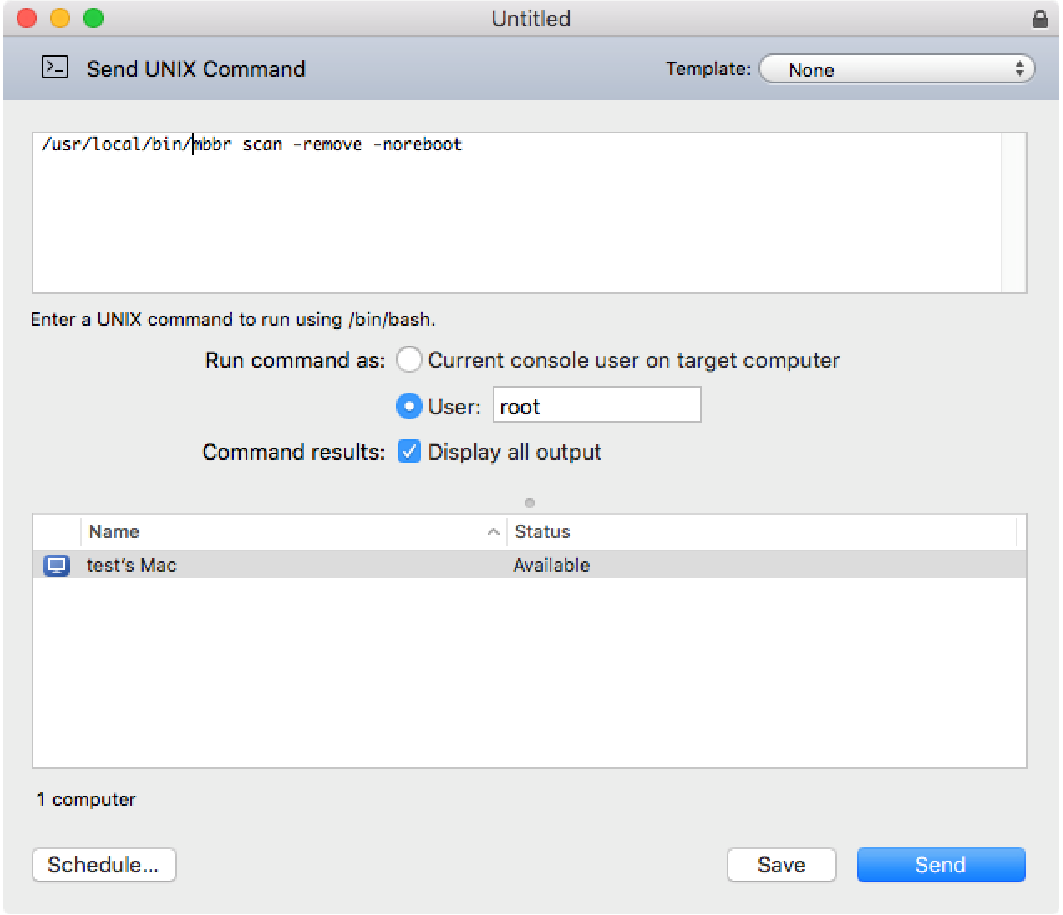
Network usage: □ Limit to kilobytes per second

Name	Status
test's Mac	Available

1 computer

Execution

Commands can be issued by selecting a group of machines and using **Manage -> Send UNIX Command**. Using this method, you must specify that the command is to run as the user “root”.



```
/usr/local/bin/mbbr scan -remove -noreboot
```

Enter a UNIX command to run using /bin/bash.

Run command as: ○ Current console user on target computer

● User: root

Command results: ☑ Display all output

Name	Status
test's Mac	Available

1 computer

Commands can be executed on-demand, or can be scheduled to run periodically or at a particular time via the same interface.

Because *Apple Remote Desktop* does not include **`/usr/local/bin`** in the default command search path, you will need to provide the full path to the executable each time you call it. For example:

```
sudo /usr/local/bin/mbbr register -key:prodKey
sudo /usr/local/bin/mbbr update
sudo /usr/local/bin/mbbr scan
```

Alternately, if you do not wish to include the full path in the command, you could place the executable in a location that *Apple Remote Desktop* does include, such as **`/bin`**, **`/sbin`**, **`/usr/bin`** or **`/usr/sbin`**, or could leave it in **`/usr/local/bin`** and modify the PATH variable.

Deployment using Munki

Deployment via *Munki* works best using the provided installer package (**`mbbr-mac.pkg`**). The following instructions assume an existing *Munki* deployment within your organization and basic familiarity with *Munki*.

A full description of many options and setup is outside the scope of this document. For more information on using *Munki*, see the *Munki* website:

https://www.munki.org/munki/

Deployment

On the server containing the Munki repository, run the following command in the Unix shell:

```
/usr/local/munki/munkiimport /path/to/mbbr-mac.pkg
```

(Be sure to enter the correct path to the mbbr-mac.pkg file.)

The **`munkiimport`** tool will prompt you for additional information. For the most part, this information is left to your discretion, but it is best to enter "true" for the "Unattended install" option. (The **`mbbr-mac.pkg`** file requires no user interaction and does not require a restart or any other post-install action.) Also be sure to enter a meaningful catalog.

Next, you need to add the imported **`mbbr-mac.pkg`** to a manifest. To do so in the command line, enter the following command:

```
/usr/local/munki/manifestutil
```

Within **`manifestutil`**, enter the following command:

```
add-pkg mbbr-mac --manifest some_manifest
```

Be sure to replace "**`mbbr-mac`**" with the name you gave the package when importing it, if you changed it from the default, and replace "***`some_manifest`***" with the name of a manifest in your *Munki* configuration.

This will schedule the Mac remediation client for automatic installation. Be aware that the installation will not happen until clients check in with the server and then do the installation, which may not happen for a few hours.

Execution

Munki does not provide any method for sending a shell command out to all client machines on demand. Execution of Mac remediation client commands will have to be scripted, via preflight or postflight scripts, or something that hooks into and extends those scripts, such as MunkiReport.

https://github.com/munkireport/munkireport-php

A full description of how this is done is outside the scope of this document.

Deployment using JAMF Casper Suite

Deployment via *Casper Suite* works best using the provided installer package (**`mbbr-mac.pkg`**). The following instructions assume an existing *Casper Suite* deployment within your organization and basic familiarity with *Casper Suite* and the *JAMF Software Server* (JSS).

A full description of many settings, hosting options and the like is outside the scope of this document. For more information on the features of *Casper Suite*, see the *Casper Suite Administrator's Guide*:

http://www.jamfsoftware.com/resources/casper-suite-administrators-guide/

Deployment

Log in to the JSS, then go to **Settings > Computer Management > Packages**. Add a new package. Configure this new package in a manner appropriate for your organization, upload the **`mbbr-mac.pkg`** file, then save the new package.

Once you have the package defined, you can either add it to an existing policy or create a new policy.

To create a new policy, go to **Computers > Policies** and add a new policy. Add the package to the policy. You do not need to change the Restart Options settings. The **`mbbr-mac.pkg`** installer will not trigger, and does not need, a restart. Set all other options as appropriate for your organization, then save the policy.

The package will now be set to be deployed according to the settings you chose.

Execution

You may use scripts to run the Mac remediation client tasks on a recurring basis. To set up a script, log in to the JSS, then go to **Settings > Computer Management > Scripts**. Add a new script. In the Script tab, enter the script you wish to run. For example, to run a diagnostic scan without removing anything, you could use the following shell script (replacing the **`prodKey`** value with a valid key):

```
#!/bin/bash
/usr/local/bin/mbbr register -key:prodKey
/usr/local/bin/mbbr update
/usr/local/bin/mbbr scan
```

Please note: It is not strictly necessary to use the **`register`** command every time, but doing so will not hurt anything and will ensure the script works in cases where Mac remediation client may never have been registered, or where the registration data has been removed.)

Save this script. Then, add this script to an existing policy, or create a new policy to run the script.

To create a new policy, go to **Computers > Policies** and add a new policy. Add the script to the policy. Set all other options as appropriate for your organization, then save the policy.

The script will now be set to run according to the settings you chose. The policy will run the script with root permissions, which is required by Mac remediation client.

Using the Mac Remediation Client

The Mac remediation client is designed specifically for use by IT staff. It may be deployed to an endpoint by local insertion of a USB drive which contains the program, or by pushing the program out to remote endpoints using preferred deployment methods. Once installed on the endpoint, the Mac remediation client quickly detects and remediates threats.

> **PLEASE NOTE:** Command line operation is not intended for use by anyone without root access, or is not fully familiar with operation of a UNIX operating system.

Installation

When used in command line mode, the Mac remediation client is fully dependent on deployment methods used by the customer. Admins may deploy the program to any location they wish on each managed system, or may instead use the included installer which installs the executable into **`/usr/local/bin`**. This allows ease of use in most configurations without specifying a path to the executable.

The Mac remediation client can also be used directly on a machine – even one that is offline – by running it from a USB flash drive or other similar media. In such cases, it must be registered and updated with the current signatures on that drive prior to scanning. Once that has been done, scans can be done entirely offline. With the exception of quarantined threats, all data will be stored on the drive, allowing for it to be moved from one machine to another without installation of any software on the target Mac.

License Key Status

The Mac remediation client uses a license key, provided to you when you purchased the client. Once registered, the license key is considered active for fourteen (14) days, unless a different time interval was specified at time of purchase. Each time the client is used on an endpoint, license status is checked.

If your license deactivates (times out), you cannot perform critical operations that the client is intended for. If this occurs, you must re-register the client (see the next section for further details). This is to prevent unauthorized use. There is no additional cost to re-register the client.

To ensure that your registration does not time out, you can use the *register* command, with or without a key, as a first step in any script designed to run scheduled scans. See the sample code included in the distribution for an example.

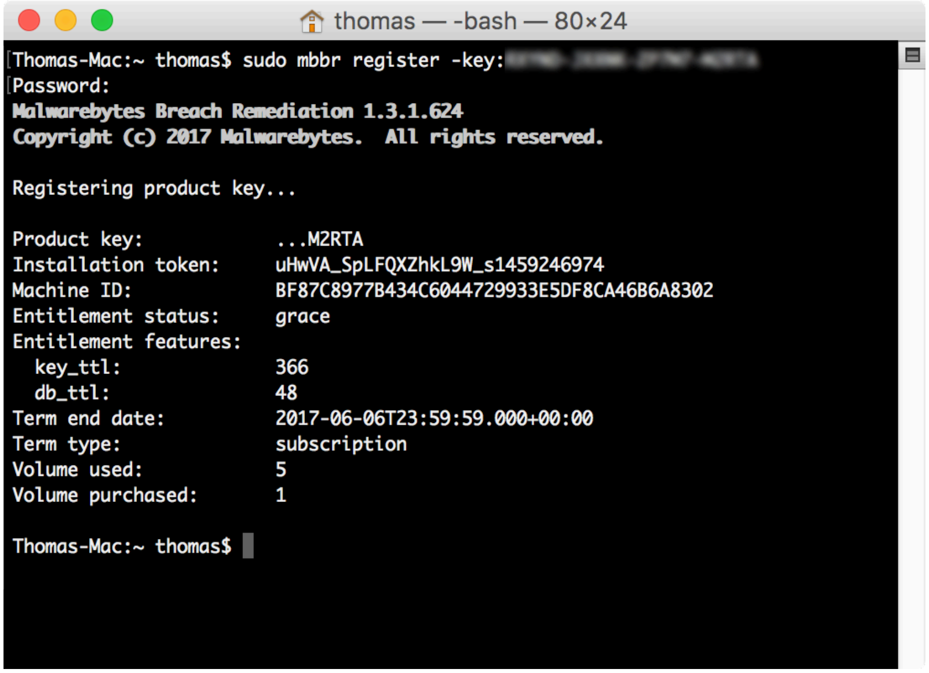
Getting Started

Getting started with the Mac remediation client is very simple. Using an endpoint with a live Internet connection, open the *Terminal* app and issue the following commands:

```
sudo <path>mbbr register -key:<prodkey>
sudo <path>mbbr update
```

The use of **`sudo`** is not required if your remote admin software gives you root privileges on the target Mac. It is included in this command as a reminder that root privileges are required. **`<path>`** indicates the file system location where the Mac remediation client is saved. You may also navigate to that location from the command line if you wish. If you have already changed to that directory, you must use the syntax "**`./mbbr`**". If the executable has been deployed to **`/usr/local/bin`**, no path is required on most OS X configurations and the program can be executed using the syntax "**`mbbr`**".

Please note: You must substitute your license key for ***`<prodkey>`*** in the above example.



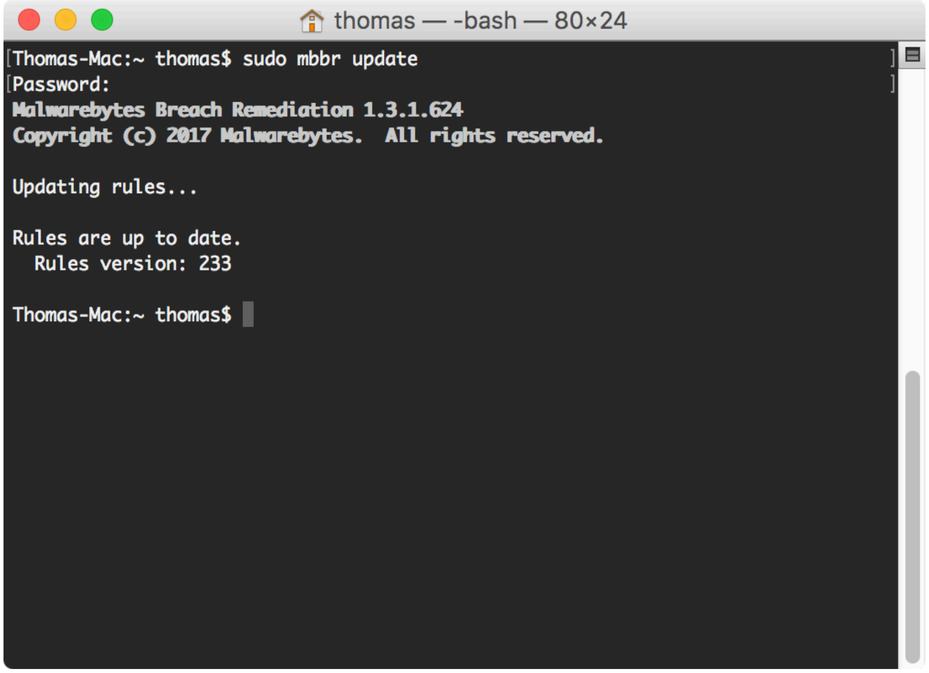
```
[Thomas-Mac:~ thomas$ sudo mbbr register -key:[redacted]
[Password:
Malwarebytes Breach Remediation 1.3.1.624
Copyright (c) 2017 Malwarebytes.  All rights reserved.

Registering product key...

Product key:            ...M2RTA
Installation token:     uHwVA_SpLFQXZhkL9W_s1459246974
Machine ID:             BF87C8977B434C6044729933E5DF8CA46B6A8302
Entitlement status:     grace
Entitlement features:
  key_ttl:              366
  db_ttl:               48
Term end date:          2017-06-06T23:59:59.000+00:00
Term type:              subscription
Volume used:            5
Volume purchased:       1

Thomas-Mac:~ thomas$ 
```

Once the program has been activated, it is necessary to load protection updates from Malwarebytes servers into the client. This enables the Mac remediation client to detect threats using the most current reference material available.



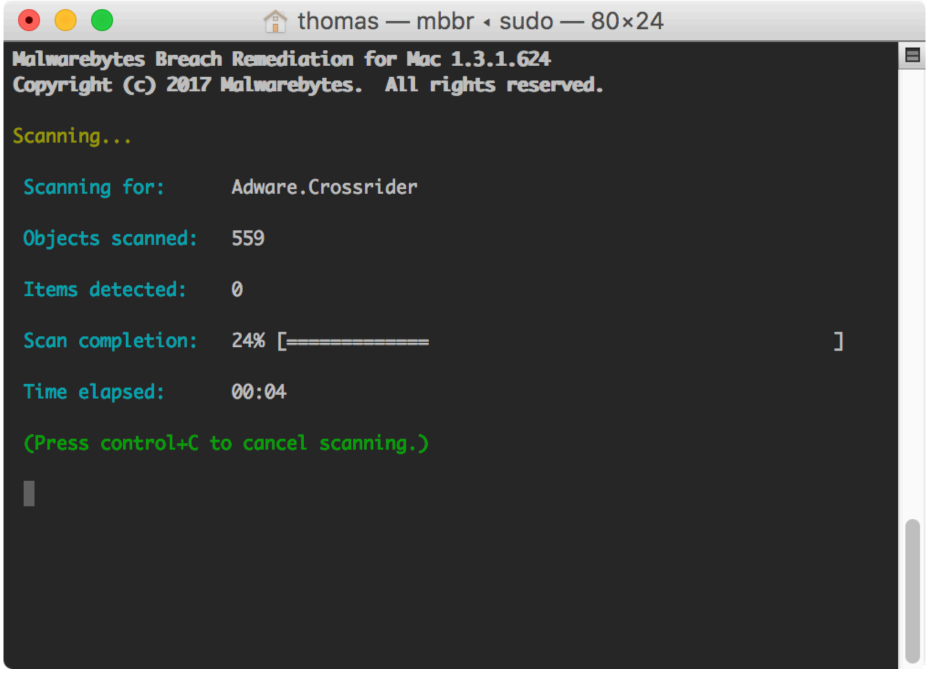
```
[Thomas-Mac:~ thomas$ sudo mbbr update
[Password:
Malwarebytes Breach Remediation 1.3.1.624
Copyright (c) 2017 Malwarebytes.  All rights reserved.

Updating rules...

Rules are up to date.
  Rules version: 233

Thomas-Mac:~ thomas$ 
```

Once protection updates have been downloaded in your local installation, you can use the Mac remediation client to detect and remove malware from your endpoint. Following is a screenshot of a scan in process.



```
Malwarebytes Breach Remediation for Mac 1.3.1.624
Copyright (c) 2017 Malwarebytes.  All rights reserved.

Scanning...

 Scanning for:       Adware.Crossrider

 Objects scanned:    559

 Items detected:     0

 Scan completion:    24% [=============                                     ]

 Time elapsed:       00:04

 (Press control+C to cancel scanning.)
```

Please note that some deployment utilities may not support color display as shown here.

Remediation Now or Later?

The Mac remediation client offers the capability to perform a scan only, or to scan and remove detected threats, and will log all detections and removals. This may be valuable in many circumstances, including:

- General assessment of an endpoint's health with regard to malware
- Ability to collect and analyze evidence of infections

Scans may be executed for the purpose of remediation, or for diagnostic discovery. A remediation scan combines a scan with a remediation method, so that detected threats may be immediately removed from the endpoint. A diagnostic scan omits the remediation method, so that a scan is executed and results are reported. The user may then determine how to proceed. This may be valuable if you wish to assess the general health of an endpoint, or if you wish to collect data about one or more endpoints without eliminating evidence that you may wish to retain.

These capabilities are listed below.

Diagnostic Scan

When executing a diagnostic scan, do not provide any specifications for remediation of threats detected during the scan. Detected threats will be logged to a file in the **Logs** folder found inside the folder from which the Mac remediation client was executed.

Remediation Scan

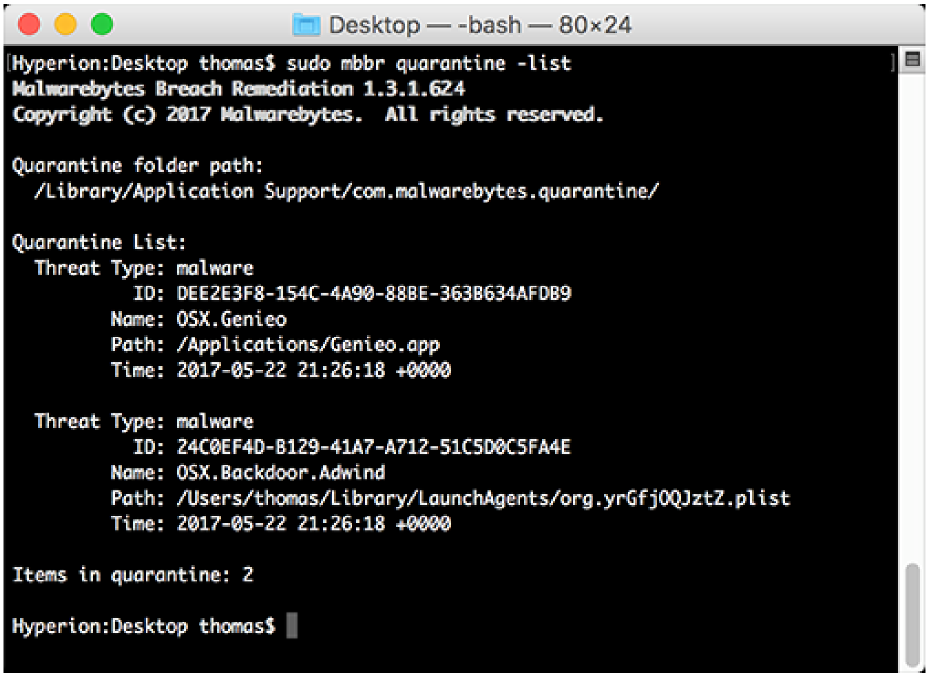
A *remediation scan* combines a scan with an automatic remediation method, so that detected threats may be immediately removed from the endpoint. No user intervention is required once the scan begins.

Restoring Items from Quarantine

The Mac remediation client offers several different methods of restoring items from quarantine. You may choose either of the following methods:

- **Restore all –** Restores all items currently stored in quarantine to their original locations
- **Restore by id –** This method utilizes quarantine item IDs to selectively restore items to their original locations. This is typically a manual operation, though it may also be performed using a script.

The following screenshot shows a list of items in Quarantine, and how they are represented. The ID field is used to specify items to be restored. Please consult page 16 in the Command Line Interface section of this guide for further information.



```
Hyperion:Desktop thomas$ sudo mbbr quarantine -list
Malwarebytes Breach Remediation 1.3.1.624
Copyright (c) 2017 Malwarebytes.  All rights reserved.

Quarantine folder path:
  /Library/Application Support/com.malwarebytes.quarantine/

Quarantine List:
  Threat Type: malware
           ID: DEE2E3F8-154C-4A90-88BE-363B634AFDB9
         Name: OSX.Genieo
         Path: /Applications/Genieo.app
         Time: 2017-05-22 21:26:18 +0000

  Threat Type: malware
           ID: 24C0EF4D-B129-41A7-A712-51C5D0C5FA4E
         Name: OSX.Backdoor.Adwind
         Path: /Users/thomas/Library/LaunchAgents/org.yrGfjOQJztZ.plist
         Time: 2017-05-22 21:26:18 +0000

Items in quarantine: 2

Hyperion:Desktop thomas$
```

If Threat Removal Fails

There are several potential reasons why a threat may not be able to be automatically removed, due to factors on the system beyond our control. If you do not find a solution in this section, please contact the Malwarebytes Customer Success team.

Use sudo

If you did not run the command using **sudo** (and did not use some other method of gaining root permissions), try that first. Many files require root permissions to remove. Be sure you are logged in to the target machine as an admin user, then perform the removal using the **sudo** command, as follows:

```
sudo mbbr scan –remove
```

Temporarily Disable SIP

In OS X 10.11 – also known as El Capitan – a feature was introduced called *System Integrity Protection* (SIP). This new feature prevents applications from making changes to certain protected locations in the file system.

When El Capitan is installed, it will clean out anything that should not be present in those locations and will turn on SIP by default. It is possible to turn SIP off, become infected by malware that writes to a protected location, then turn SIP on again later. In such a case, the Mac remediation client will be able to detect the threat, but will be unable to remove it completely.

If such an event occurs, the solution will be to turn off SIP again, remove the detected threat, and then re-enable SIP. Information on how to do this can be found here:

https://support.malwarebytes.com/customer/en/portal/articles/2144939-?b_id=9511

Restart in Safe Mode

It is possible that a third-party application could interfere with threat removal. If this happens, boot the target machine in safe mode:

http://support.apple.com/kb/HT1455

Attempt to perform the removal again while in safe mode. If the problem was caused by third-party software interference, this should fix it.

Command Line Parameters

The Mac remediation client supports a variety of command line parameters, which can be used from a command prompt, batch file or script. When used from a script, additional commands may be required to support the scripting model being used. **Please note that root privileges are required for all commands.** Examples will always use *sudo* as a reminder, but *sudo* itself is not required if your remote admin software provides you with the capability to run scripts with root privileges.

Conventions

The command line structure uses modifiers. These are shown as hyphens (-) immediately preceding parameters. Multiple modifiers may be combined with a parameter. When multiple parameters are used, they must be separated by spaces. In addition, the following conventions are used:

- **text without brackets or braces**
  Items you must type as shown
- **<text inside angle brackets>**
  Required information for which you must supply a value
  Example: **sudo mbbr <*parameter_1*>**
- **[text inside square brackets]**
  Optional items
  Example: **sudo mbbr [*parameter_1*]**
- **Grouping of dots (…)**
  A set of specifications
  Example: **sudo mbbr <*parameter_1*> [*parameter_2*] … [*parameter_n*]**
- **{text inside braces}**
  A set of required items; choose one from the list provided
  Example: **sudo mbbr {0 | 1 | 2 | 3}**
- **vertical bar (|)**
  Separator between mutually exclusive items; choose one
  Example: **sudo mbbr <0 | 1 | 2 | 3>**
- **Options**
  Options can be used with each parameter by using –optionname. Data can be passed along with an option, when appropriate, by separating it from the option with a space, colon (:) or equals sign (=)
  Examples: **sudo mbbr register -key:<prodkey>**
  sudo mbbr register -key <prodkey>
  sudo mbbr register -key=<prodkey>

Command Line Overview

Mac remediation client commands are specified in the following format:

mbbr { register | update | version | scan | quarantine | snapshot } [*options*]

Following is a list of high-level commands which may be executed. Each command is detailed in *Getting Started* (page 9).

register	Using your license key, this unlocks the features of the Mac remediation client, and shows license status.
update	Downloads the most recent protection updates.
version	Displays the program version number.
scan	Scans the endpoint for malware and optionally removes malware found during the scan.
quarantine	Controls program actions related to threat quarantine activities.
snapshot	Take a system snapshot and save it to a file.

In addition, you may type **mbbr** without any additional specifications to see a list of valid commands. This list will span multiple windows if the Terminal is launched to its default size, so you will achieve best results by stretching the window to show more command line dialog at one time.

To see usage information for one specific command, use the –help or -? option. For example, to get usage information for the scan command, the following command can be used:

```
sudo mbbr scan -help
```

Command Line Reference

Commands listed here are listed individually. Each command performs tasks according to parameters. These are primarily used by a system administrator via script, batch file, GPO update, or remote desktop. The admin may configure the Mac remediation client to operate as a remote task, invisible to the endpoint user.

register

Usage:

sudo mbbr register [-key:<*prodkey*>]

Purpose:

Specifies the unique license key assigned to the partner or customer. This will be passed to the licensing server for validation to ensure it is active (non-expired). **A live Internet connection is required.** If the key is valid and the license is active, it will also display status about the license, such as expiration date, volume purchased, volume used, etc.

If the key is active, the local installation will operate with this status for 14 days (or the time interval specified in your Malwarebytes license agreement). This "Last Known Good" status is persisted on the USB or wherever the binaries are stored. This allows the USB installation to work as if it were fully registered on offline endpoints or without needing the key.

If **-key** is not specified, license status and the expiration date/time will be displayed. **Please note** that if the key is not active, the user may not download protection updates, scan for malware, or obtain a diagnostic system snapshot.

Parameters:

-key:<*prodkey*>

Specification of <*prodkey*>, the license key assigned to the user.

update

Usage:

sudo mbbr update

Purpose:

Downloads protection updates. This command will result in an error condition if (a) the license is not active, or (b) if no active Internet connection is available. If protection updates have expired (timed out), this command must precede execution of a scan.

Parameters:

none

version

Usage:

sudo mbbr version

Purpose:

Displays the version number of the Mac remediation client installation.

Parameters:

none

scan

Usage:

```
sudo mbbr scan [-ignorepu]
               [-ignorebe]
               [-tag:<tagdata>]
               [-remove [-noreboot] ]
               [-stdout:{off | detail | summary}]
```

Purpose:

Executes a scan based on parameters specified. If the program license is inactive, attempts to perform a scan will result in an error. Current protection updates are also required. If this command is executed without a directive to remove detected threats, scan results are saved to a log file in the Logs folder, which is found in the same directory that the Mac remediation client was executed from.

Parameters:

-ignorepu

Instructs the scanner to ignore all Potentially Unwanted Programs (PUPs) that may be installed on the target endpoint.

-ignorebe

Instructs the scanner to ignore all web browser extensions that may be installed on the target endpoint. This is to speed up scans if they are taking too much time, as scanning for browser extensions usually takes the most time on most systems. Use of this flag is not recommended unless it improves performance of scans and you are able to scan for browser extensions at a different scan time or interval.

-tag:<*tagdata*>

This text string will be sent along with all usage data to the Malwarebytes billing system. It will help you to associate billing events with your billing system. Typical usage would be for you to add the Job ID or Store ID or Employee ID or all of these, so that you can see these on your invoice.

The data passed using this option must be no more than 100 characters, and can only contain alphanumeric characters and the space character. If the string includes spaces, it must be surrounded by double quotes (“).

-remove

Instructs the scanner to quarantine any malware, adware and PUPs found during the scan. By default, if any threats require a reboot for complete removal, the target machine will be restarted automatically. To change this behavior, use the -noreboot option.

-noreboot

Some malware executes in a manner that requires a reboot to complete the removal process. If this occurs, the scanner will automatically reboot the system. If an immediate reboot is not desired, use this option. Please note that certain malware may not be fully removed if this option is used. If **-noreboot** is specified in a scan command and the scan detected threats requiring a reboot, a warning message will be displayed after the scan has completed to notify the user that a reboot is required to remove the threat(s) from the endpoint.

-stdout:{off | detail | summary}

Controls the level of output to the console. Defaults to **summary** if not specified.

quarantine

Usage:

```
sudo mbbr quarantine [-list]
                     [-path:<path>]
                     [-resetpath]
                     [-restoreall]
                     [-restore:<UUID1>[,<UUID2>…,<UUIDn>]]
                     [-deleteall]
                     [-delete:<UUID1>[,<UUID2>…,<UUIDn>]]
```

Purpose:

Manage quarantine, including restore/delete from quarantine, set/reset location of quarantine and list quarantine contents. Use this command without any additional arguments to display the current quarantine location.

Parameters:

-path:<*path*>

Specifies the location to be used for quarantined content after this command has been executed. This replaces any previously-specified location. If <*path*> contains any embedded spaces, please enclose <*path*> in double quotes ("). **Please note** that quarantined content existing prior to execution of this command will not be moved to the new location.

-resetpath

Causes the quarantine folder to revert to the default folder specification. Files stored in quarantine prior to execution of this command will not be moved to the default folder. The default quarantine folder is:

/Library/Application Support/com.malwarebytes.antimalware/Quarantine/

-list

Shows the current quarantine location and lists contents of the quarantine to screen output.

-restoreall

Restores all quarantined items to their original locations.

-restore: <UUID1>[,<UUID2>…,<UUIDn>]

Restores one or more items from the list of quarantined items. Items are specified by their ID. When multiple items are to be restored via a single execution of this command, their IDs should be separated by commas without delimiting spaces. Please note that execution of this command will delete file RestoreList.xml, as that file will no longer have correct information.

-deleteall

Permanently deletes all quarantined items from the disk.

-delete: <UUID1>[,<UUID2>…,<UUIDn>]

Deletes one or more items from the list of quarantined items. Items are specified by their ID. When multiple items are to be restored via a single execution of this command, their IDs should be separated by commas without delimiting spaces. Please note that execution of this command will delete file RestoreList.xml, as that file will no longer have correct information.

snapshot

Usage:

```
sudo mbbr snapshot [-path:<file_path>]
```

Purpose:

Output a system snapshot, containing detailed information about the system, to the specified path. If the –path parameter is omitted, output the snapshot data to stdout.

Parameters:

-path:<*file_path*>

Specifies a path to which to write the system snapshot text. If <*path*> contains any embedded spaces, please enclose <*path*> in double quotes (").