HomeKit Accessory Tester (HAT)

User Manual

Revision 10

Contents

NOTICE OF PROPRIETARY PROPERTY: THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS THE PROPRIETARY PROPERTY OF APPLE INC. THE POSSESSOR AGREES TO THE FOLLOWING: (I) TO MAINTAIN THIS DOCUMENT IN CONFIDENCE, (II) NOT TO REPRODUCE OR COPY IT, (III) NOT TO REVEAL OR PUBLISH IT IN WHOLE OR IN PART, (IV) ALL RIGHTS RESERVED.

ACCESS TO THIS DOCUMENT AND THE INFORMATION CONTAINED HEREIN IS GOVERNED BY THE TERMS OF THE MFI LICENSE AGREEMENT. ALL OTHER USE SHALL BE AT APPLEâĂŹS SOLE DISCRETION.

Chapter 1

Overview

This document describes the configuration and use of Apple's HomeKit Accessory Tester (HAT). HAT is a Mac application which acts as multiple HomeKit Accessory Protocol (HAP) clients for the purpose of developing, debugging, and testing accessory functionality.

> **Note**
>
> Please report all issues to bugreport.apple.com using the template below as well as "MFi HAT" in the title of the bug report.
>
> SUMMARY
> Recap the problem and include more descriptive summary information.
>
> IMPACT
> Explain what impact this issue has on your development. eg. inconvenience, blocked, etc.
>
> STEPS TO REPRODUCE
> 1. Setup or prep work
> 2. Explicit and accurate steps to reproduce
>
> RESULTS
> Describe your results and how they differed from what you expected.
>
> NOTES
> Document any additional information, configuration, spec language, workarounds, etc.
>
> ENVIRONMENT
> Setup version numbers Eg. HAT, HAS, OS X, Accessory Firmware, Router, etc.
>
> ACCESSORY DETAIL
> Please include developer company and accessory name as well as chipset manufacturer.
>
> ATTACHMENTS
> A trace or screenshot with annotations is required to quickly and efficiently understand your issue. Additional logs and/or videos will help expedite a fix.

> **Note**
>
> Known Issue:
>
> BLE rediscovery inconsistency and occasional failure. Mac BLE hardware sometimes gets in a bad state causing pairing and discovery failures, this seems to be a hardware issue. If you experience repeated problems with BLE discovery, consider resetting your NVRAM.
>
> Instructions can be found here: https://support.apple.com/en-us/HT204063

1.1 Equipment List

HAT requires a Mac capable of running macOS 10.14 (Mojave) or greater, with Wi-Fi connectivity and capable of communicating over BLE. No additional hardware is required to run HAT.

1.2 Application Modes

The HAT application has two primary modes: a Manual Mode and a Certification Assistant mode. The application currently only contains the Manual Mode.

1.3 Event Trace

HAT's event trace windows represent streams of events which occurred during usage of an accessory. This includes high level descriptions such as "Start Discovery", but extends to include decrypted HTTP and BLE payloads. HAT will also capture low level packets across each networking interface where HAP is being used. Event traces can then be exported through the File > Save menu item or by pressing Command-S.

The following HTTP payload types have additional display properties

- JSON Payloads

- TLV8 Payloads

- Pairings Lists

Each event includes a set of details, which may be expanded either in the main trace view, by double clicking the event, or by clicking "Details" in the toolbar. You may also copy these details to the pasteboard, resulting in text which is similarly formatted to what is displayed.

The event trace includes a filter, which limits what is shown to events containing the supplied subtext.

Figure 1.1: Event Trace - Basic View

By either holding down "option" or turning on the setting in Preferences, the time stamp delta of the selected row will will be shown relative to the others.

Figure 1.2: Event Trace - Time Stamp Delta

Upon double clicking on an event in traffic view, the details view pops up on the right hand side of the window with tabs for additional details.

Figure 1.3: Event Trace - Details View

1.4 Manual Mode

HAT's Manual Mode is for development and debugging an accessory. It allows the accessory developer to initiate client actions as they see fit in a lower level way than possible with an actual public facing client.

1.4.1 Opening the Trace

The Manual Mode window will keep a trace of all events which occur while it is open. This trace can be opened via the "Trace" button in the bottom right corner of the Manual Mode window. The Event Trace chapter describes in detail what this window shows. The trace records all events lower level networking data. This trace view will also include error information in future releases. The trace may be saved to your computer and re-opened at a later date if needed.

Figure 1.4: Manual Mode - Trace Button
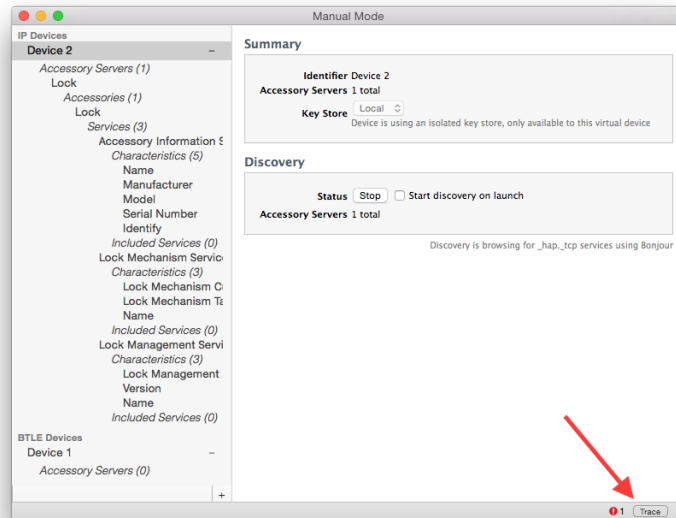
1.4.2   Creating a Virtual Device

Upon launching the application, you should be presented with the Manual Mode window. This window allows you to create and manage IP and BLE clients, which act as virtual iOS devices. By default there will be one IP and BLE device created for you. If you choose to delete them you may have to manually re-create them using the "+" button near the bottom left of the window. It may be beneficial to test your accessory with multiple virtual clients at once while emulating keychain sharing.

Figure 1.5: Manual Mode - Initial Window
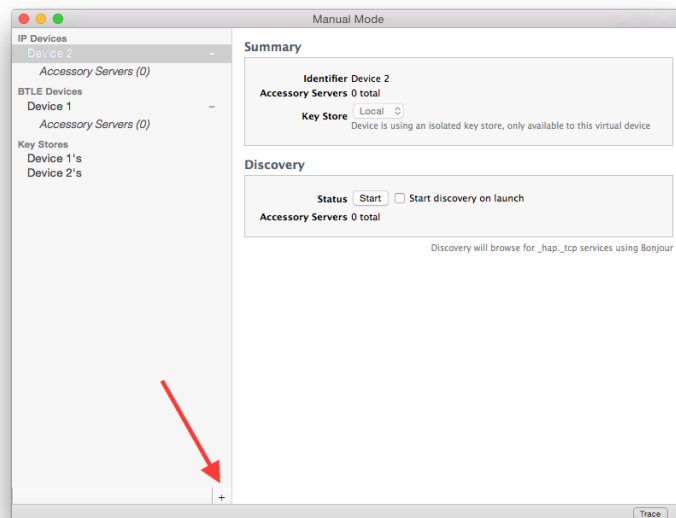
1.4.3 Device Setup

Once you have created a device, you will be presented with a screen to configure its initial properties. You may choose to use the device's own key store, or, if you have created more than 1 device, you may share another device's key store to simulate iCloud keychain sharing. Explicitly selecting a key store is optional. The button's choice will be applied as soon as the device is used, at which point other devices may share that key store.

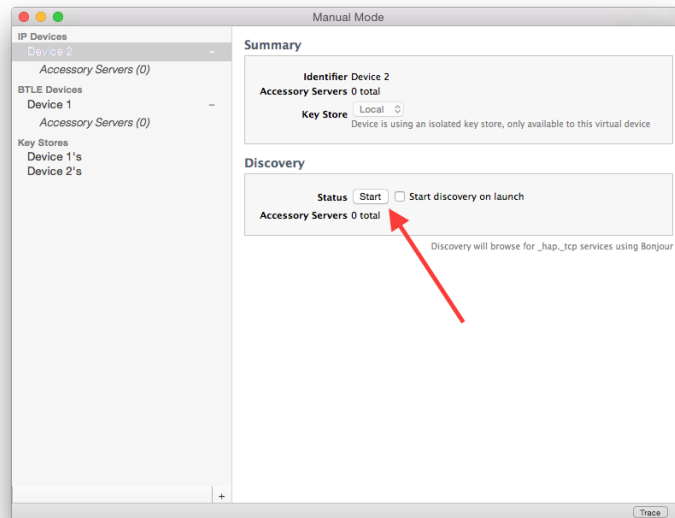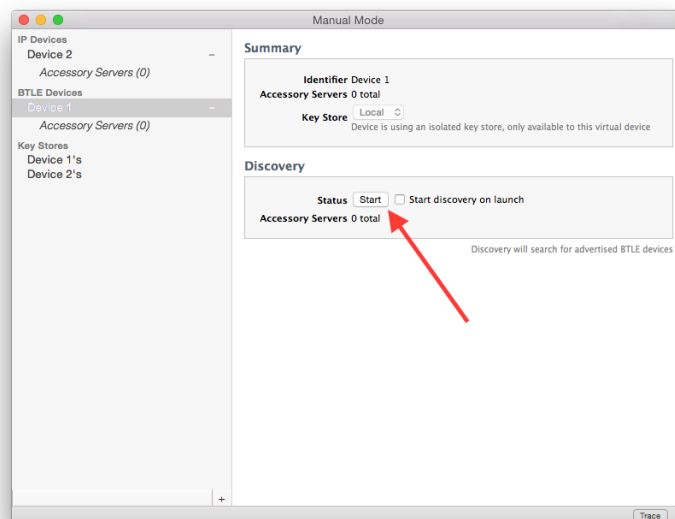Figure 1.6: Manual Mode - Initial IP Device Setup

Figure 1.7: Manual Mode - Initial BLE Device Setup

1.4.4 Accessory Discovery

After starting discovery your Bonjour-enabled or BLE accessories will show up under the device. Selecting the accessory will display more detailed information.

Figure 1.8: Manual Mode - IP Accessory Summary
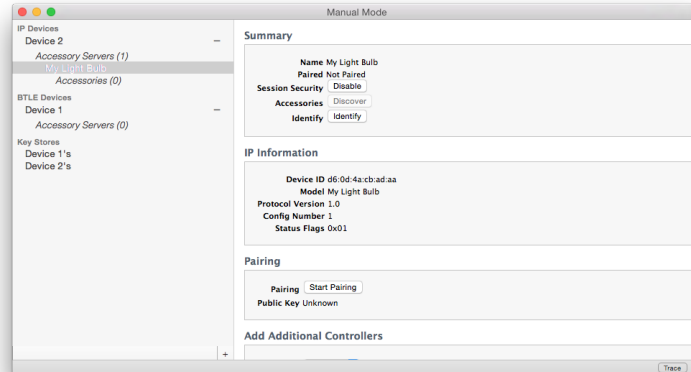
Figure 1.9: Manual Mode - BLE Accessory Summary
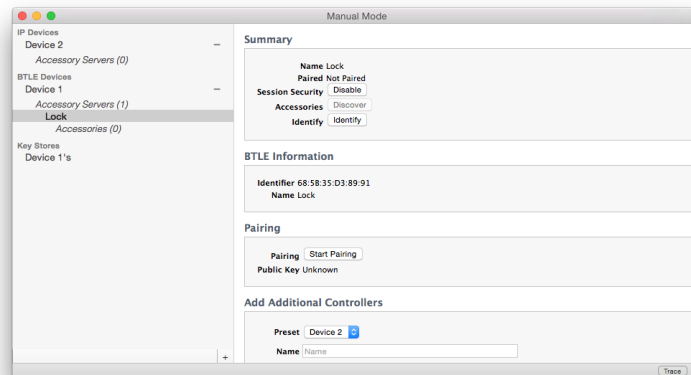
1.4.5 Pairing

Once you see the device, you can start pairing. During the pairing process you will be prompted for the accessory's passcode. HAT also provides the ability to disable session security, disabling encryption and making pairing optional. This feature may be beneficial to allow development of higher level application functionality before pairing has been implemented on the accessory.

> **Note**
>
> The ability to disable session security exists for development and debugging purposes only, and an accessory must **not** ship with the ability to function without session security.

Figure 1.10: Manual Mode - Pairing and Session Security
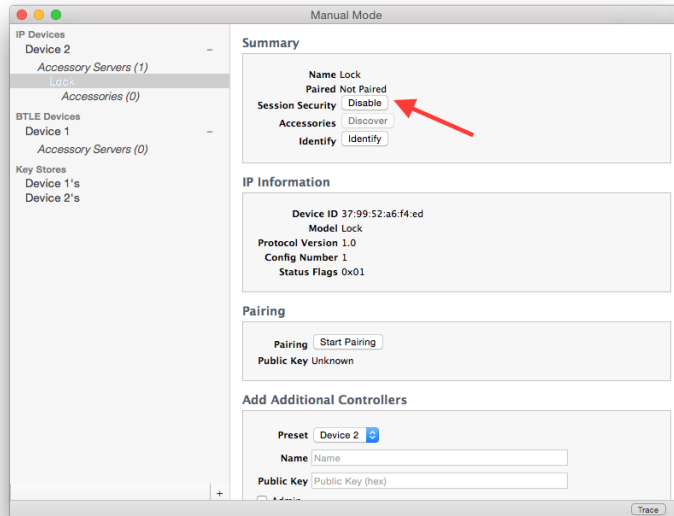
1.4.6   Accessory/Service Discovery

Once paired, you can initiate service discovery with the accessory by clicking the "Discover" button next to "Accessories" in the accessory server's summary box. Doing so will discover all accessories presented by the accessory server, and allow you to access all the included characteristics.
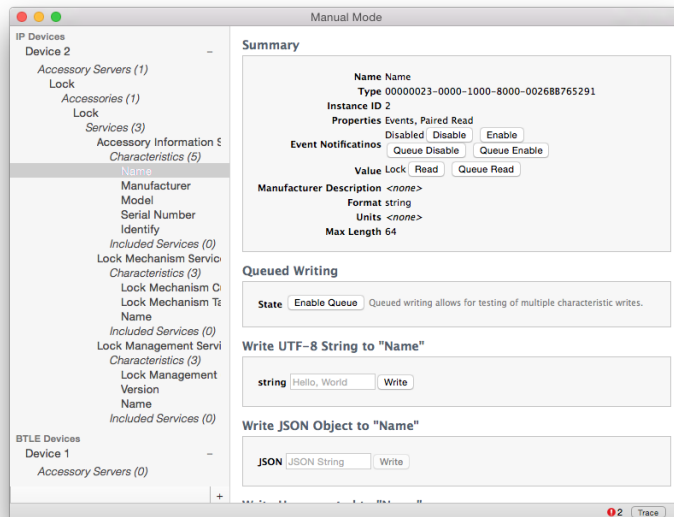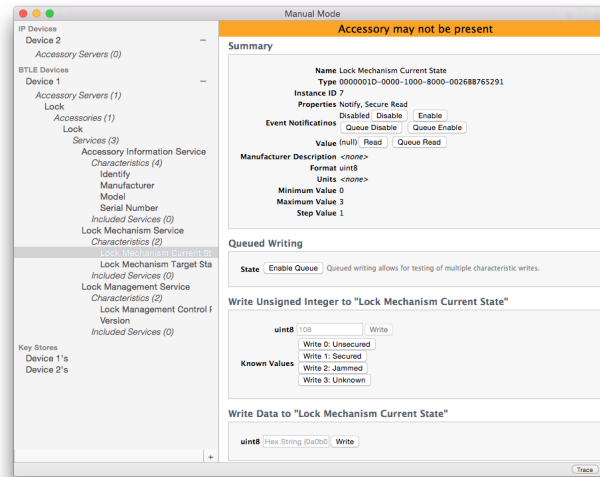
Figure 1.11: Manual Mode - Discovery

1.4.7   Read/Write Characteristics

When viewing a characteristic, you may read its value and also write values of various types to it.

Figure 1.12: Manual Mode - Read/Write Characteristics

1.5   Tunneling

1.5.1   Setting up Tunnel

First create a shared key store, IP Device, and a BLE Device. The IP Device will be the tunnel and BLE Device the tunneled accessory. Upon adding the devices, select Key Store for both to use the Shared Key Store just created.

> **Note**
>
> Known Issue: Due to an UI bug, the selection may need to be toggled between local and Shared Key Store. Ensure the text below displays âĂŸDevice will use Shared Key Store x, which may be used by other virtual devicesâĂŹ.

Upon setting up Key Stores, pair with your BLE Tunneled Accessory or HAS (HomeKit Accessory Simulator) and discover services. You can then pair your IP Tunnel and after you discover services, the tunneled accessory will be displayed HAT will display the tunneled accessory which is indicated in parenthesis as a second Accessory. The tunneled accessory can then be interacted with such as read/write. Keep in mind added controllers must use the same same key store as tunnel and tunneled accessory.

Chapter 2

Revision History

Date	Rev	Notes
5/3/16	10	New template for filing bugs
4/5/16	9	Tunneling Setup for HAT 3.2
1/7/16	8	Support for HAP Specification R6 in HAT 3.0
4/17/15	7	Added rules to validate pairing lists and updated BLE functionality for adding and removing controllers in HAT 1.0
1/30/15	6	Updated HTTP validation and added UI enhancements for writing to custom characteristics in HAT 1.0 Beta 6
1/16/15	5	Support for custom services and characteristics added to HAT 1.0 Beta 5
11/7/14	4	Update images and notes for HAT 1.0 Beta 4
7/30/14	3	IP and BLE validation rules associated with HAT 1.0 Beta 3
6/23/14	2	BLE support associated with HAT 1.0 Beta 2a
4/7/14	1	Quick start guide associated with HAT 1.0 Beta 1