Heartland Advanced Training

Dealer Facing Logs

Agenda

1. Introduction
2. Accessing Logs via Admin Portal
3. Types of Device Logs
4. How to troubleshoot Common Errors
5. Level 2 and Dev Teams Use Logs to Work Escalations
6. Keys Terms and Phrases
7. Q&A

Today's Presenters

Mathew Turner

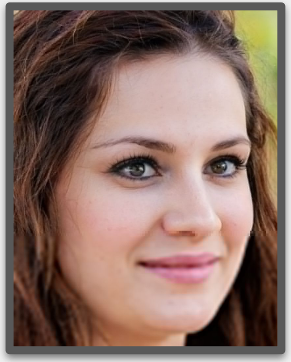
Sales Engineer

William Anderson

Sales Engineer

Jared Hughes

Senior Product Analyst

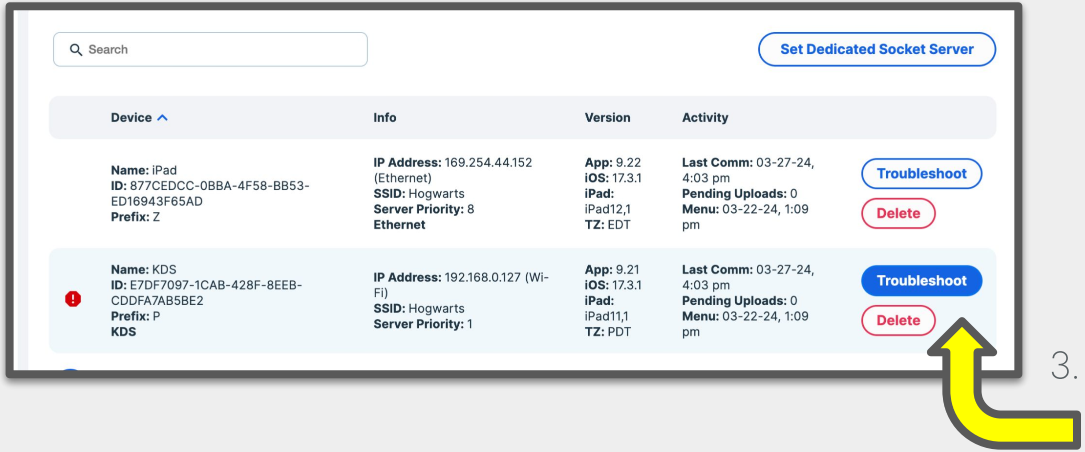


Courtney Gayetty

Implementation Support Specialist

Heartland

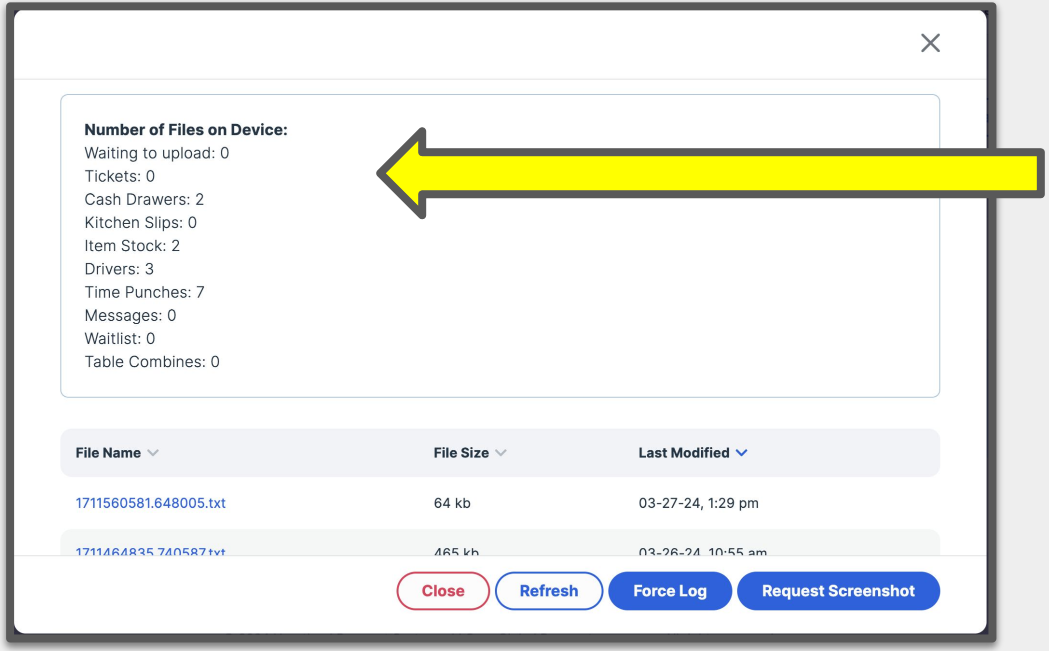
Accessing Logs Through The Admin Portal For Authorized Tablets



	Device	Info	Version	Activity	
	Name: iPad **ID:** 877CEDCC-0BBA-4F58-BB53-ED16943F65AD **Prefix:** Z	**IP Address:** 169.254.44.152 (Ethernet) **SSID:** Hogwarts **Server Priority:** 8 **Ethernet**	**App:** 9.22 **iOS:** 17.3.1 **iPad:** iPad12,1 **TZ:** EDT	**Last Comm:** 03-27-24, 4:03 pm **Pending Uploads:** 0 **Menu:** 03-22-24, 1:09 pm	Troubleshoot Delete
❗	**Name:** KDS **ID:** E7DF7097-1CAB-428F-8EEB-CDDFA7AB5BE2 **Prefix:** P **KDS**	**IP Address:** 192.168.0.127 (Wi-Fi) **SSID:** Hogwarts **Server Priority:** 1	**App:** 9.21 **iOS:** 17.3.1 **iPad:** iPad11,1 **TZ:** PDT	**Last Comm:** 03-27-24, 4:03 pm **Pending Uploads:** 0 **Menu:** 03-22-24, 1:09 pm	Troubleshoot Delete

1. Access the **Admin Portal**
2. Select Location **Setup > Authorized Tablets**:
   a. Displays all Tablets
   b. Management options for Troubleshoot and Delete
3. Choose a Tablet and Select **Troubleshoot** to open the Troubleshoot Window

Troubleshoot Window and Tablet Information



Number of Files on Device:
Waiting to upload: 0
Tickets: 0
Cash Drawers: 2
Kitchen Slips: 0
Item Stock: 2
Drivers: 3
Time Punches: 7
Messages: 0
Waitlist: 0
Table Combines: 0

File Name	File Size	Last Modified
1711560581.648005.txt	64 kb	03-27-24, 1:29 pm
1711464835.740587.txt	465 kb	03-26-24, 10:55 am

The **top half** of the window shows an overview of the current status of the selected tablet.

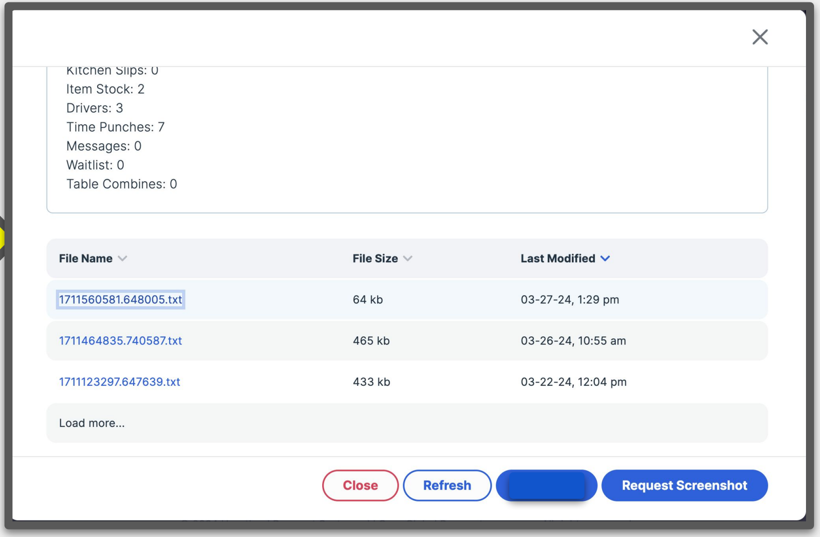
This provides a quick overview of the point of sale software state.

How To Sort Device Logs

Device logs are located in **bottom half** of troubleshooting window and are sorted by their Unix timestamp name.

View and sort files by:

- File Name
- File Size
- Last Modified



Kitchen Slips: 0
Item Stock: 2
Drivers: 3
Time Punches: 7
Messages: 0
Waitlist: 0
Table Combines: 0

File Name	File Size	Last Modified
1711560581.648005.txt	64 kb	03-27-24, 1:29 pm
1711464835.740587.txt	465 kb	03-26-24, 10:55 am
1711123297.647639.txt	433 kb	03-22-24, 12:04 pm

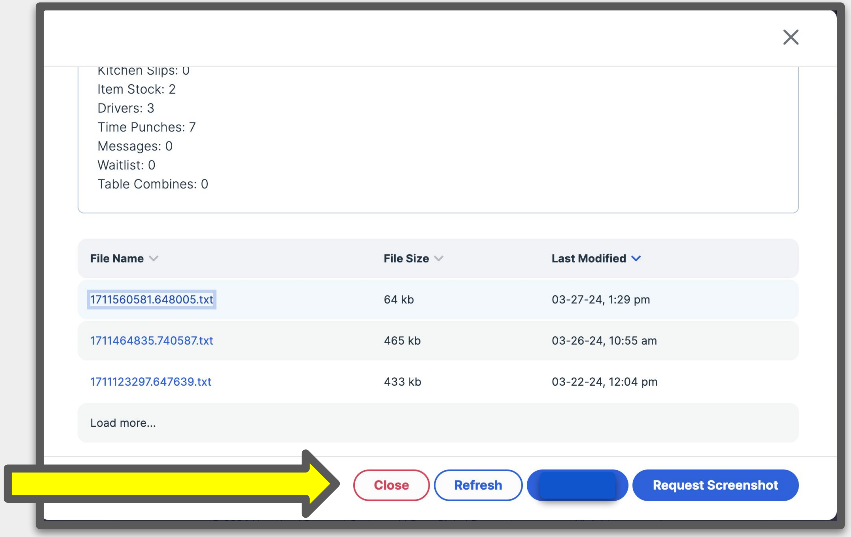
Load more...

Generating Device Logs and Screenshots

- **Request Screenshot** - This will take a screenshot of the selected tablet and upload that to the troubleshooting screen. Will show as a .jpg

- **Refresh** - Refresh will pull new files into the troubleshooting panel. Use this if you do not see a screenshot or forced log appear in the list

- **Close** - This will Close the Troubleshooting panel out and bring you back to the authorized tablets page.



Kitchen Slips: 0
Item Stock: 2
Drivers: 3
Time Punches: 7
Messages: 0
Waitlist: 0
Table Combines: 0

File Name	File Size	Last Modified
1711560581.648005.txt	64 kb	03-27-24, 1:29 pm
1711464835.740587.txt	465 kb	03-26-24, 10:55 am
1711123297.647639.txt	433 kb	03-22-24, 12:04 pm

Load more...

Types of Device Logs

We break the logs into four main types of logs.

- System Logs
- Operational Logs
- Ticket Logs
- Transaction Logs

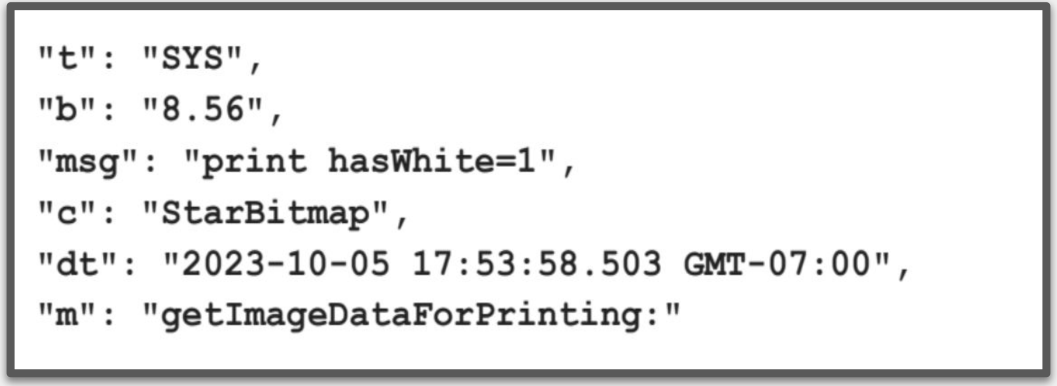
Device Logs provide information about various actions and events related to and occurring in the application.

Types of Device Logs

● System Logs	“t” : “SYS”
● Operational Logs	“t”: “OP”
● Ticket Logs	“t”: “TICKET”
● Transaction Logs	“t”: “TRANS”

System Logs

Capture events and activities related to the internal workings of the application or device.

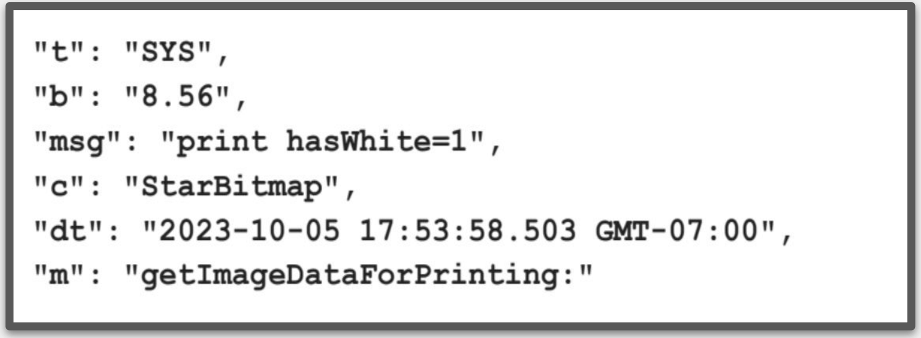


```
"t": "SYS",
"b": "8.56",
"msg": "print hasWhite=1",
"c": "StarBitmap",
"dt": "2023-10-05 17:53:58.503 GMT-07:00",
"m": "getImageDataForPrinting:"
```

System logs provide:

- Component Activities
- Identify Method Calls
- Indicate Addresses Internal ID's
- Display Messages Being Sent

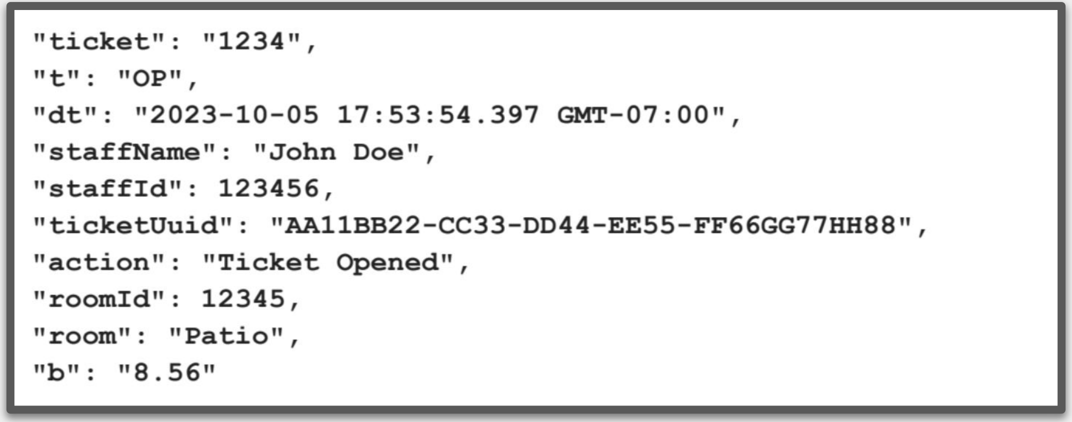
Examples of System Logs



```
"t": "SYS",
"b": "8.56",
"msg": "print hasWhite=1",
"c": "StarBitmap",
"dt": "2023-10-05 17:53:58.503 GMT-07:00",
"m": "getImageDataForPrinting:"
```

- **“t”** - Indicates the type
- **“b”** - a value that indicates what version “build” the application was using during the date the log was captured
- **“msg”** - the logged message being sent to clients, captures inter-component communication
- **“c”** - indicates the component the system event relates to
- **“dt”** - the timestamp of the event being captured
- **“m”** - indicates the method being called during this event

Operational Logs

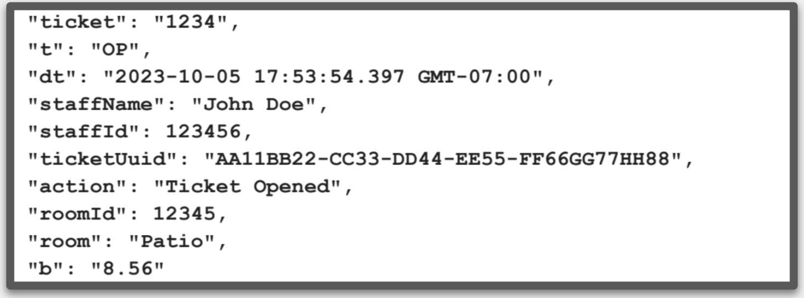


```
"ticket": "1234",
"t": "OP",
"dt": "2023-10-05 17:53:54.397 GMT-07:00",
"staffName": "John Doe",
"staffId": 123456,
"ticketUuid": "AA11BB22-CC33-DD44-EE55-FF66GG77HH88",
"action": "Ticket Opened",
"roomId": 12345,
"room": "Patio",
"b": "8.56"
```

Operation logs capture **actions and events** related to the operational aspects of the application. These logs help understand the flow of operations within the application.

They offer insights into how the application operates **in real time**, which allows for better understanding of user behavior, diagnose problems, and optimize system performance.

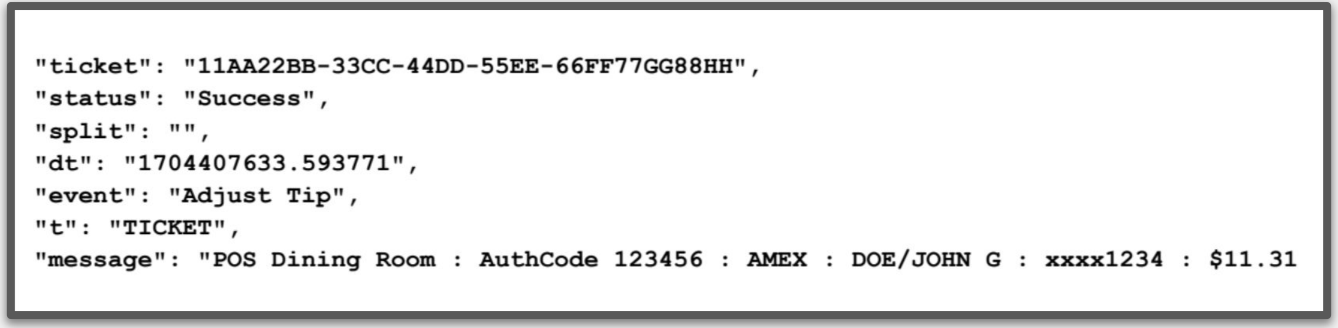
Examples of Operational Logs



```
"ticket": "1234",
"t": "OP",
"dt": "2023-10-05 17:53:54.397 GMT-07:00",
"staffName": "John Doe",
"staffId": 123456,
"ticketUuid": "AA11BB22-CC33-DD44-EE55-FF66GG77HH88",
"action": "Ticket Opened",
"roomId": 12345,
"room": "Patio",
"b": "8.56"
```

- **“ticket”** indicates the ticket number that event relates to
- **“t”** indicates the type of log this is.
- **“dt”** is the timestamp of the event being captured
- **“staffName”** indicates the name of the staff member operating the ticket
- **“staffId”** is a unique database identifier for that staff membertion was using during the event
- **“ticketUuid”** is the ticket universal unique identifier.
- **“action”** is the captured action occurring within the event
- **“roomId”** Unique database identifier for the room
- **“room”** the room where the ticket is assigned
- **“b”** the value that indicates what version of the application was using during the event

Ticket Logs



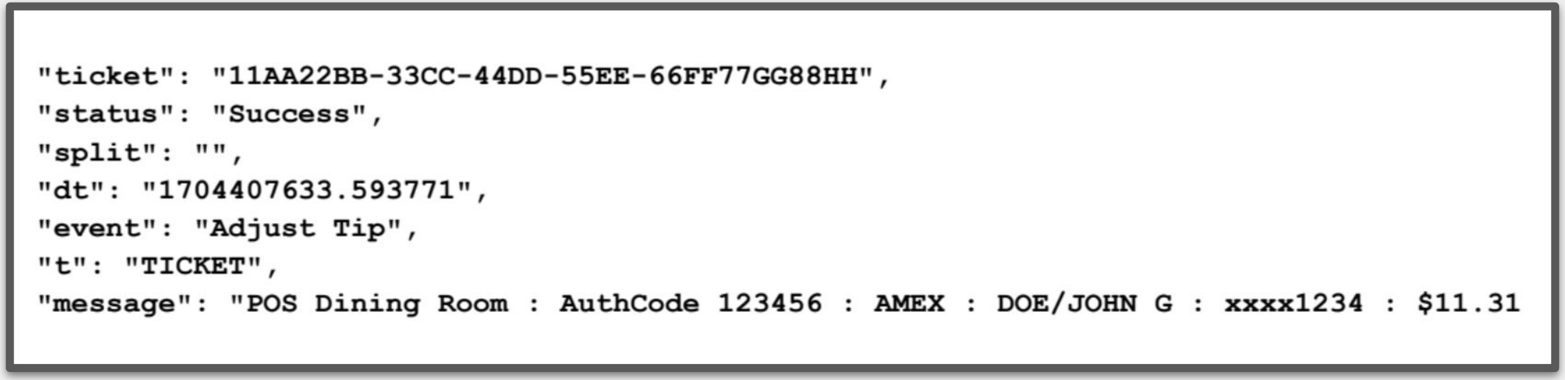
```
"ticket": "11AA22BB-33CC-44DD-55EE-66FF77GG88HH",
"status": "Success",
"split": "",
"dt": "1704407633.593771",
"event": "Adjust Tip",
"t": "TICKET",
"message": "POS Dining Room : AuthCode 123456 : AMEX : DOE/JOHN G : xxxx1234 : $11.31
```

Ticket logs serve as a **comprehensive transactional history**, recording every significant event related to a particular ticket.

Ticket logs provide **detailed information about payment processing**, including credit card transactions, authorization codes, and amounts.

These logs can be used as an **audit trail for investigating** discrepancies or issues.

Examples of Ticket Logs



```
"ticket": "11AA22BB-33CC-44DD-55EE-66FF77GG88HH",
"status": "Success",
"split": "",
"dt": "1704407633.593771",
"event": "Adjust Tip",
"t": "TICKET",
"message": "POS Dining Room : AuthCode 123456 : AMEX : DOE/JOHN G : xxxx1234 : $11.31
```

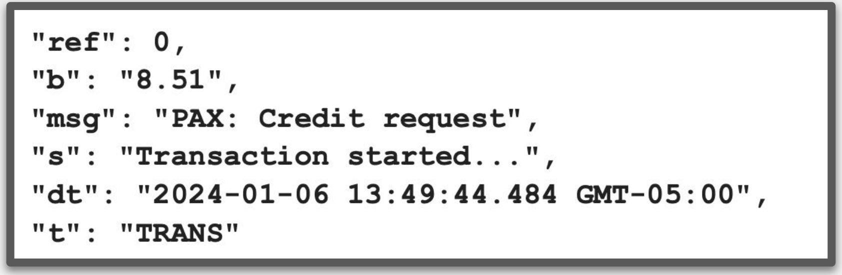
- **“Ticket”** - indicates the type of log this is
- **“Status”** - indicates the status of log this ticket
- **“Split”** - indicates the split status of the ticket
- **“Event”** - indicates the type of event captured
- **“t”** - indicates the type of log this is.
- **“Message”** - shows the ticket information

Transactional Logs

Detail record for transactions including:

- EMV/PAX credit requests
- payments
- tip capture

Logs will create audit trail to allow or accountability and ability to trace events.

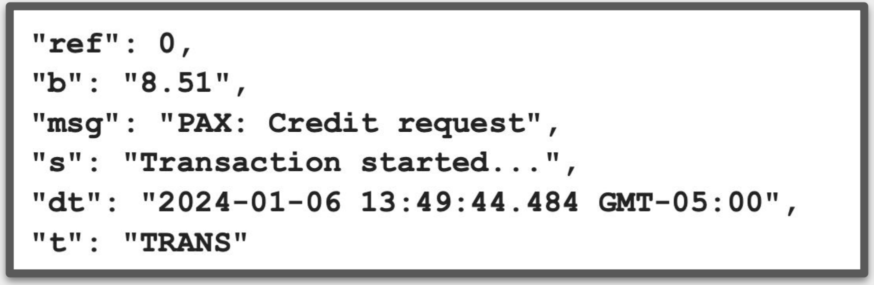


```
"ref": 0,
"b": "8.51",
"msg": "PAX: Credit request",
"s": "Transaction started...",
"dt": "2024-01-06 13:49:44.484 GMT-05:00",
"t": "TRANS"
```

Transaction logs record and track various transactional activities and detailed account transactions, statuses, and time stamps to help:

- audit
- track errors
- monitor performance

Examples of Transactional Logs



```
"ref": 0,
"b": "8.51",
"msg": "PAX: Credit request",
"s": "Transaction started...",
"dt": "2024-01-06 13:49:44.484 GMT-05:00",
"t": "TRANS"
```

- **“ref”** - Reference or identifier associated with transaction
- **“b”** - Value that indicates what version of the app the tablet was using
- **“msg”** - Describes transaction type
- **“s”** - Transaction status
- **“dt”** - Timestamp of Transaction
- **“t”** - Indicted the type of log this is

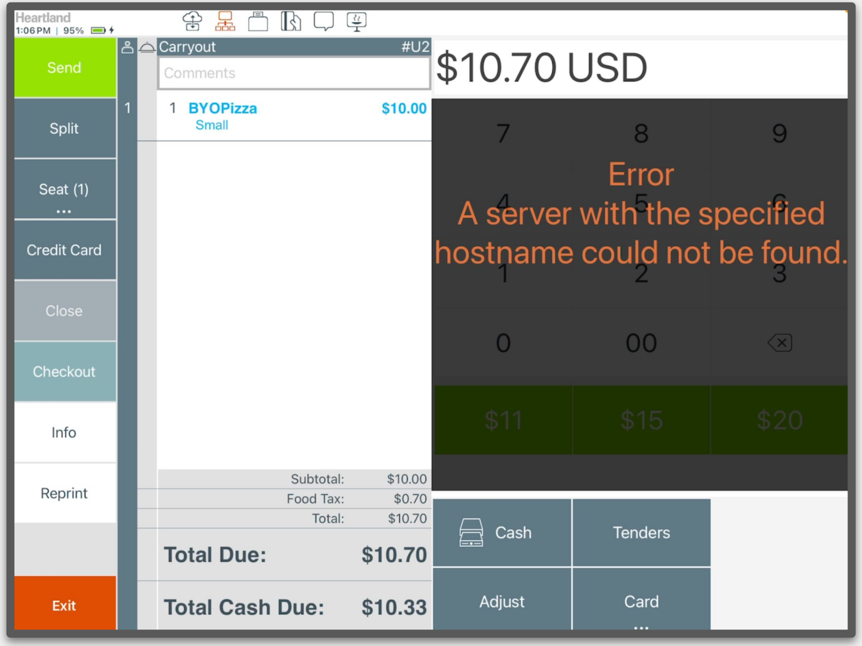
Understanding Log Entries

The best way to find errors in a log.

Troubleshooting each log type:

- Payment Error
- Identifying Crashes
- Online Orders
- Print Errors

Troubleshooting Payment errors



Carryout #U2

1 BYOPizza $10.00
Small

Subtotal:	$10.00
Food Tax:	$0.70
Total:	$10.70
Total Due:	**$10.70**
Total Cash Due:	**$10.33**

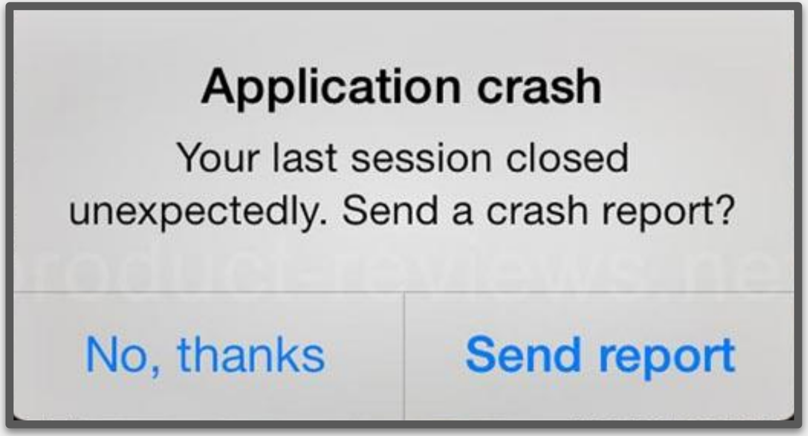
$10.70 USD

Error
A server with the specified hostname could not be found.

How to Troubleshoot:

- Review the tablet logs
- Search logs (ctrl+F) for "Authentication Error"
- If this error is found, contact SIP Support to verify credentials.

How to Identify App Crashes



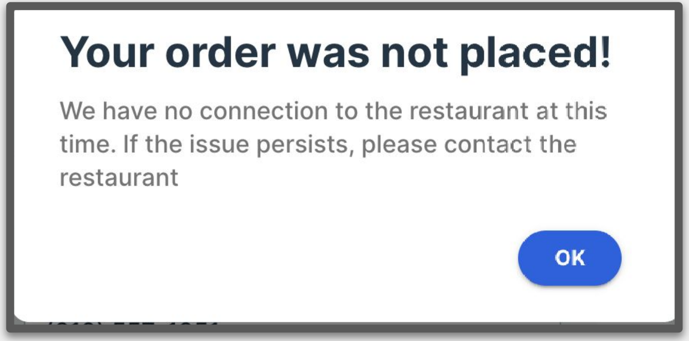
Application crash

Your last session closed unexpectedly. Send a crash report?

How to Identify

- First thing to realize is log size will be abnormal
- The log will end without an app *entered background* or *terminal app message*
- Crashes usually will require assistance from our Development team as it’s usually a bug causing the crash

Troubleshooting Online Ordering Errors



Your order was not placed!

We have no connection to the restaurant at this time. If the issue persists, please contact the restaurant

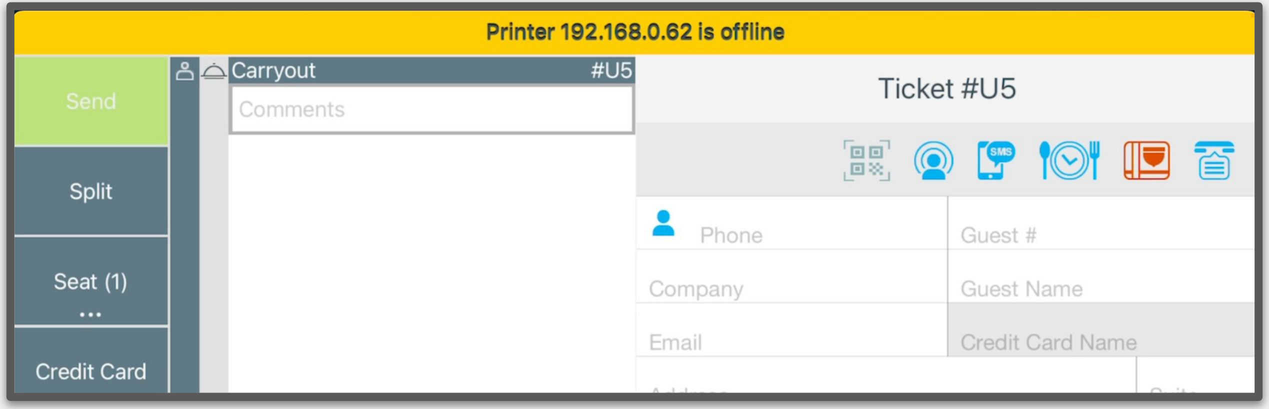
How to Troubleshoot Related to the Tablet

- Related to tablet being offline or signed out of the app / Troubleshoot: confirm tablet is online and on the correct network
- Also could be due to not having an assigned tablet for the Online Ordering / Troubleshoot: confirm there's an Assigned tablet under OLO settings / confirm it's online

How to Troubleshoot Related to the not Printing

- getPort: networking error that leads to issues with printing and potentially cash drawer usage

Troubleshooting Printing Errors

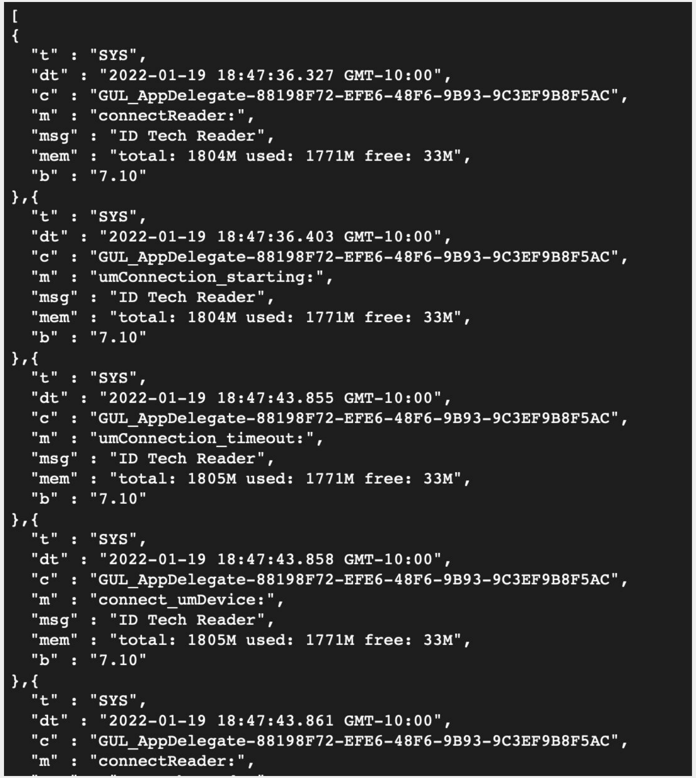


Printer 192.168.0.62 is offline

How to Troubleshoot

- Check Socket Server is Online/App is Running
- Power Cycle the Printer
- Verifying the IP Address of th Printer

Beyond Basic Logs



```
[
{
  "t" : "SYS",
  "dt" : "2022-01-19 18:47:36.327 GMT-10:00",
  "c" : "GUL_AppDelegate-88198F72-EFE6-48F6-9B93-9C3EF9B8F5AC",
  "m" : "connectReader:",
  "msg" : "ID Tech Reader",
  "mem" : "total: 1804M used: 1771M free: 33M",
  "b" : "7.10"
},{
  "t" : "SYS",
  "dt" : "2022-01-19 18:47:36.403 GMT-10:00",
  "c" : "GUL_AppDelegate-88198F72-EFE6-48F6-9B93-9C3EF9B8F5AC",
  "m" : "umConnection_starting:",
  "msg" : "ID Tech Reader",
  "mem" : "total: 1804M used: 1771M free: 33M",
  "b" : "7.10"
},{
  "t" : "SYS",
  "dt" : "2022-01-19 18:47:43.855 GMT-10:00",
  "c" : "GUL_AppDelegate-88198F72-EFE6-48F6-9B93-9C3EF9B8F5AC",
  "m" : "umConnection_timeout:",
  "msg" : "ID Tech Reader",
  "mem" : "total: 1805M used: 1771M free: 33M",
  "b" : "7.10"
},{
  "t" : "SYS",
  "dt" : "2022-01-19 18:47:43.858 GMT-10:00",
  "c" : "GUL_AppDelegate-88198F72-EFE6-48F6-9B93-9C3EF9B8F5AC",
  "m" : "connect_umDevice:",
  "msg" : "ID Tech Reader",
  "mem" : "total: 1805M used: 1771M free: 33M",
  "b" : "7.10"
},{
  "t" : "SYS",
  "dt" : "2022-01-19 18:47:43.861 GMT-10:00",
  "c" : "GUL_AppDelegate-88198F72-EFE6-48F6-9B93-9C3EF9B8F5AC",
  "m" : "connectReader:",
```

Support Escalations

- SalesForce to JIRA
- From Triage to Release

Resources Needed for Development

- Setups
- Occurrences & Use Cases
- Logging

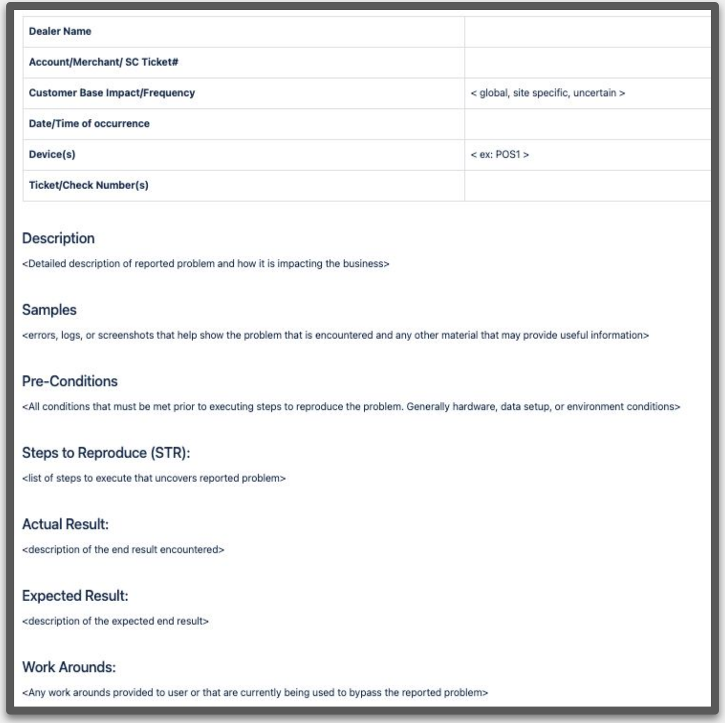
Bug Fixes & Resolutions

- Prioritization
- Improving the Velocity

Beyond Basic Logs

Development request

- **Description**
- **Pre-conditions**
- **Samples**
  - Ticket Numbers
  - Timestamps
- **Steps to Reproduce**
- **Expected vs Actual Results**
- **Additional Information**
  - Frequency
  - Devices
    - Models
    - Software and/or hardware versions
  - Screenshots/Videos



Dealer Name	
Account/Merchant/ SC Ticket#	
Customer Base Impact/Frequency	< global, site specific, uncertain >
Date/Time of occurrence	
Device(s)	< ex: POS1 >
Ticket/Check Number(s)	

Description

<Detailed description of reported problem and how it is impacting the business>

Samples

<errors, logs, or screenshots that help show the problem that is encountered and any other material that may provide useful information>

Pre-Conditions

<All conditions that must be met prior to executing steps to reproduce the problem. Generally hardware, data setup, or environment conditions>

Steps to Reproduce (STR):

<list of steps to execute that uncovers reported problem>

Actual Result:

<description of the end result encountered>

Expected Result:

<description of the expected end result>

Work Arounds:

<Any work arounds provided to user or that are currently being used to bypass the reported problem>

Regular review and maintenance of logs can prevent and resolve issues.

Conclusion

➔ Leverage device logs for a deeper understanding of system and operational health.

➔ Build confidence in supporting common errors in system.

➔ Best escalation path to follow to help get bugs reported and give our development teams work follow to fix bugs as fast as possible.

Terms To Know

- ➔ **applicationDidEnterBackground**: This is an event or method call in iOS development. It is part of the application lifecycle and is called when the app enters the background. This is typically associated with other services closing or stopping.

- ➔ **stopWebSockets**: This method call is associated with closing a web socket connection. When the application enters the background or is about to terminate, it often clears resources, and stopping web sockets is one such action.

- ➔ **stopSocketService**: Similar to **'stopWebSockets,'** this method stops the socket service

- ➔ **applicationWillTerminate**: This is another part of the application lifecycle in iOS. This method is called when the app is about to terminate.

- ➔ **'applicationDidEnterBackground'** it is a point where cleanup tasks are often performed.

- ➔ **AppDelegate**: This is a central component in an iOS app that receives events related to the app's lifecycle. Typically, after this component is triggered in the logging, a follow-up might look like.

- ➔ **mbWebSocketEventText**: This is one of the indicators for MBLogger when it is used to log WebSocket events, typically related to token and order information.

- ➔ "**GUL_AppDelegate-1A2B3C4D....**". This UUID is given to the App Delegate to help track the current logging session. Matching UUIDs in logs would indicate that the logs are all part of the same application session. A UUID change here would indicate that the logging session was interrupted at some point, likely due to a closing of the application or some related event.

- ➔ **applicationDidBecomeActive**: This method call in iOS is triggered when the app becomes active. It often involves actions needed to resume or initialize processes after the app returns from the background.

- ➔ **FlatButton**: This references a UI component, a button in the flat style. It is usually partnered with a message indicating what button was pressed, such as a number or the button's label.

- ➔ **MBLogger**: This component is primarily associated with logging WebSocket events, handling notification messages, and processing data from a server over a socket connection. Its details may reveal insights into the application's communication with external services, especially regarding orders and WebSocket-related activities.