Evolving Binary Defense MDR

Project Team

Andrew Pucci (lead designer)
Jimmy Byrd (lead developer)
Development Team

Responsibilities

Generative User Research
Interaction & Visual Design
Front-end Development

Tools Used

Adobe XD
Bootstrap
axe DevTools extension
Tanaguru Contrast Finder
ColorBox

What is Binary Defense MDR?

Binary Defense MDR (once Vision) is an endpoint protection solution that provides detection of and defense against attacks on corporate networks.

Challenge

I joined the Binary Defense product team as the first designer. To bring the product to the next level, we built a better understanding of how people used it. We then used that feedback to make it faster and easier to use.

How Binary Defense MDR works

Event Collection

An agent collects event data from endpoints across customer networks.

Event Correlation

Custom algorithms correlate event data to surface potential threats.

Alarm Production

Alarms are produced for potential malicious events.

Investigation

Information from alarms is used to investigate suspicious network activity.

Escalation

Upon determining high likelihood of true malicious activity, alarms are escalated.

Remediation

Security experts take remediative action to secure network.

User research

When I joined Binary Defense as the first UX designer, there was a lot to figure out. There was no design direction or goal in place, I needed to work with company leadership, users, and the development team to build a plan.

Leadership had grand visions for new features and customer growth. The development team was small, but growing. They planned to improve the product to enable quicker iteration and prepare for scale. The users, well... no one knew much about their impressions of the product.

I spent my first week interviewing security analysts, sales, and customer support to get a basic understanding of who the users were. This high-level research led me to defining three major user types: In-House Security Operations Center (SOC) Analysts, Security Experts, and Managers. I decided to focus my efforts first on the In-House SOC Analysts.

While I was at the company headquarters, I used contextual inquiry and interviews to get a feel for how the In-House SOC Analysts went about their work. I watched as they investigated alarms, spoke with customers on the phone, and escalated tickets to customers. After talking through a few of these scenarios, it was easy to see that there were issues to be addressed.

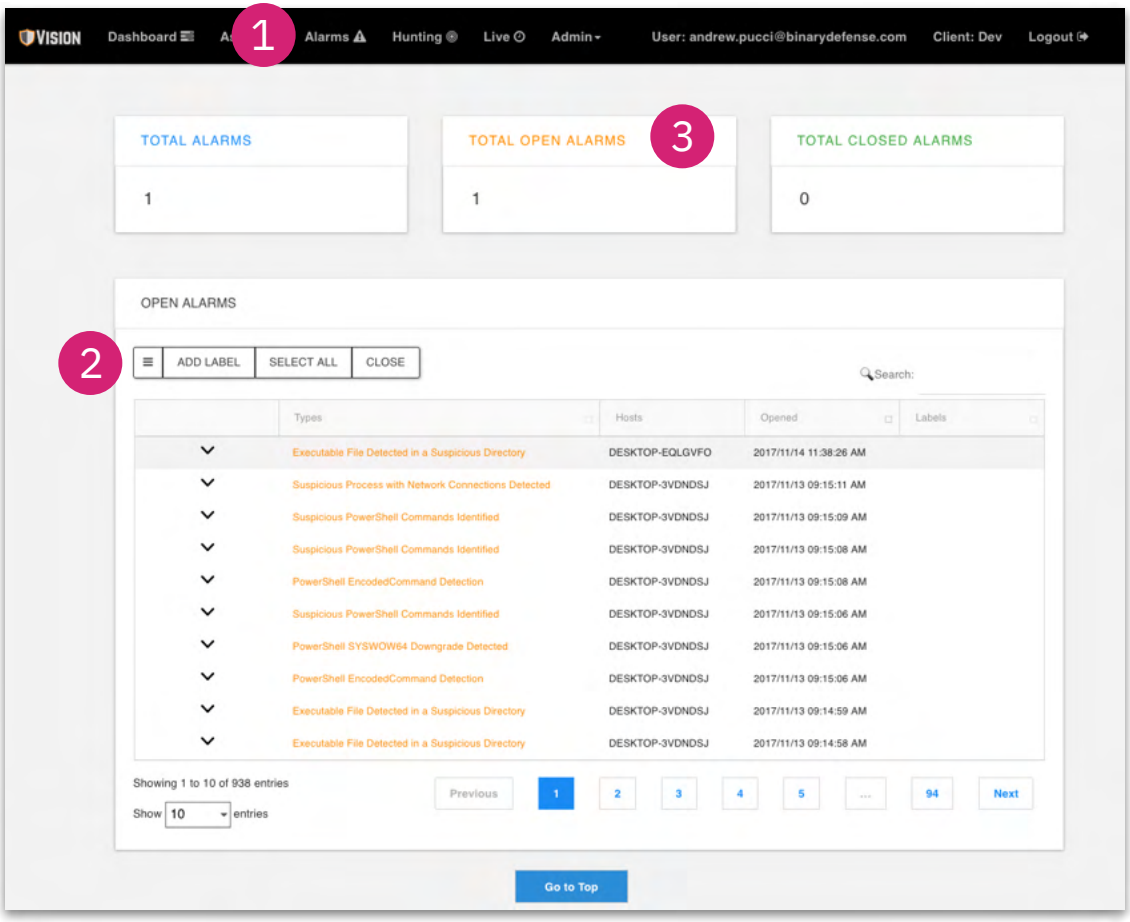
In this case study, I focus on the evolution of the Open Alarms interface, the most used feature by In-House SOC Analysts. This interface is used most often during the **Alarm Production** and **Investigation** phases.

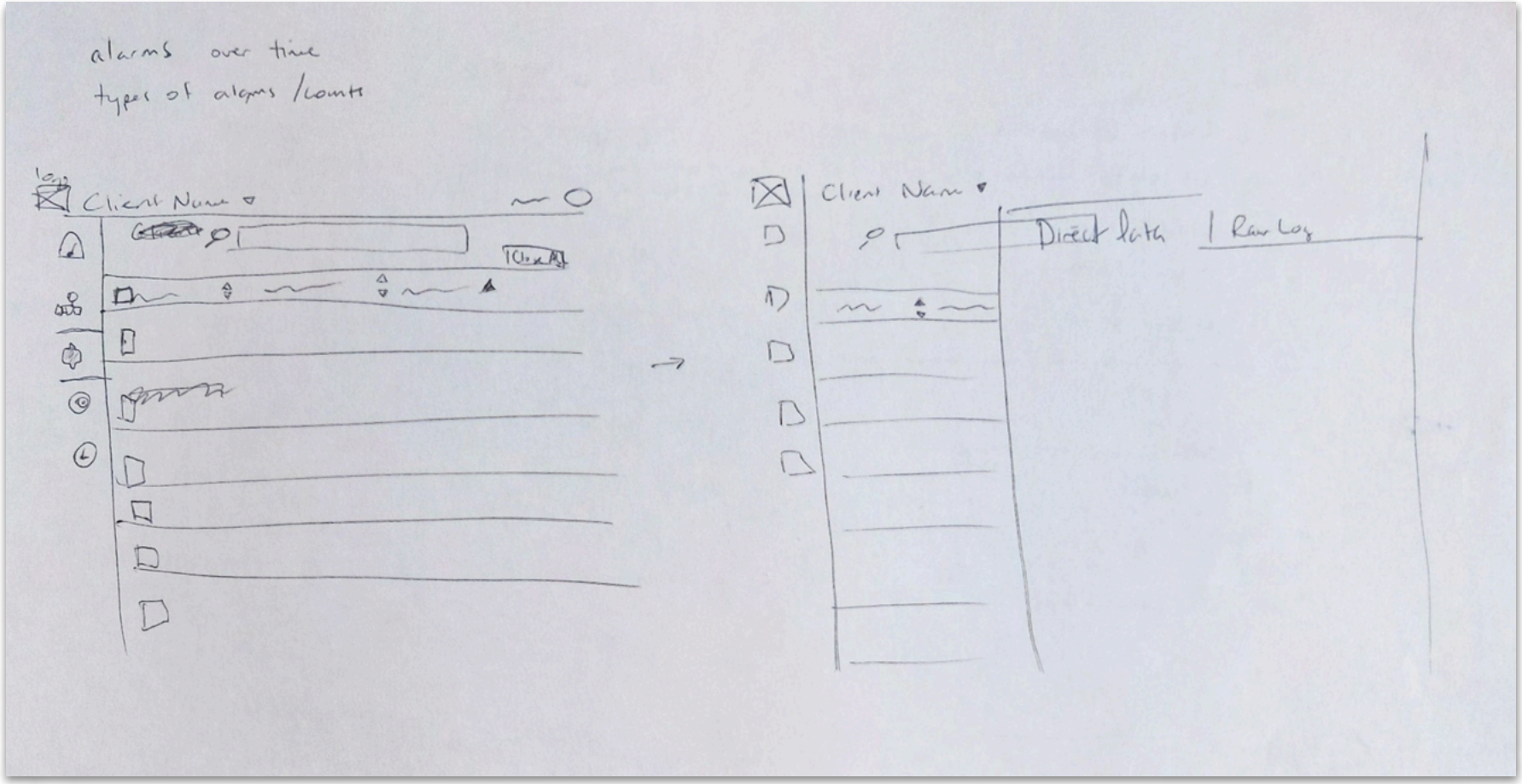
Key research findings

1. Users had trouble quickly understanding where they were in the application and which customer they were triaging.

2. A few key actions were hidden in this hamburger menu making them hard to find for new users and slow to get to for power users.

3. The matching text color here and in the table below caused confusion, only one was a link.





Ideation

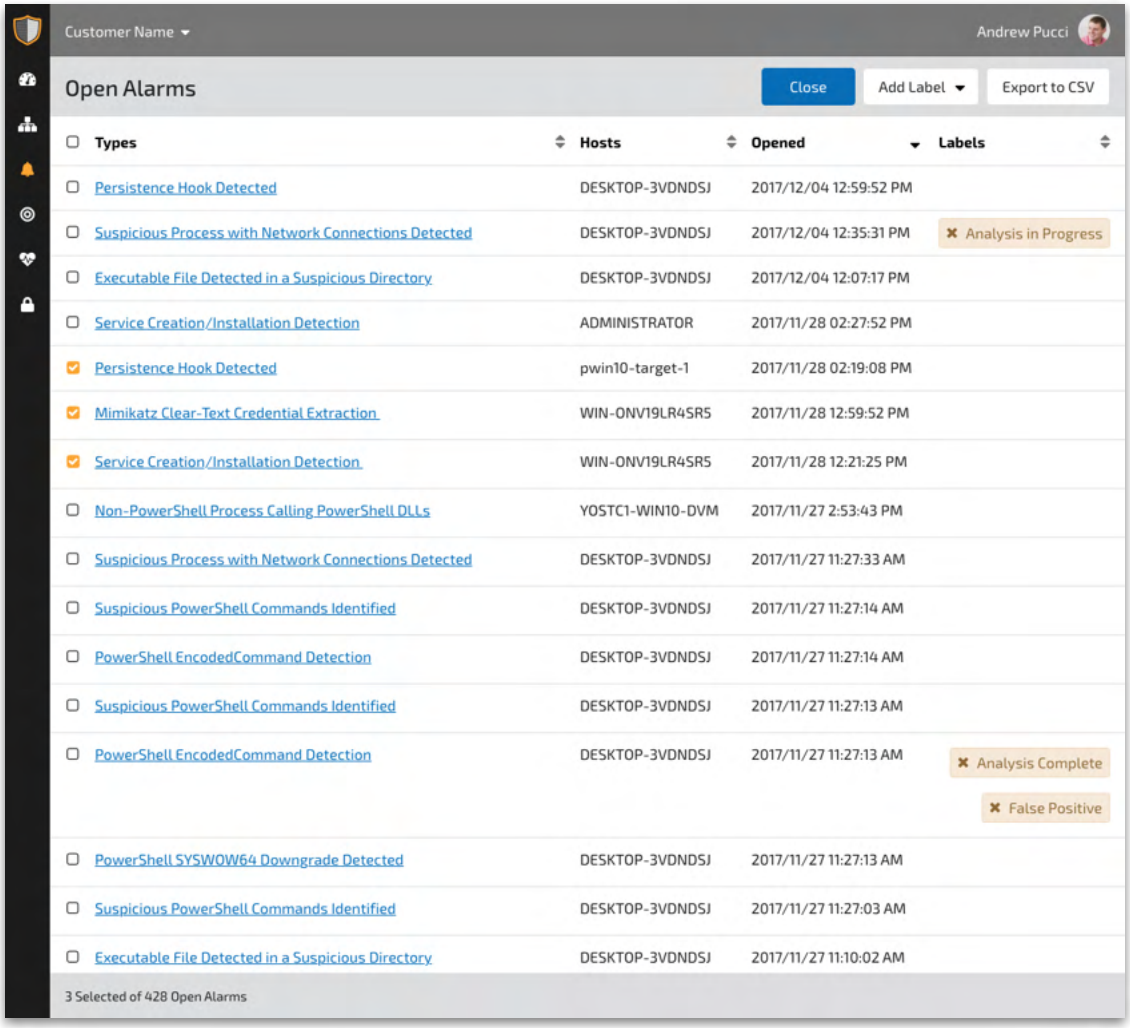
Once I identified the issues to address, I sketched out some ideas to help me decide on a design direction. I took some of these sketches, like the ones shown above, to both the development team and to a few In-House SOC Analysts to get feedback. This early feedback from developers and users was especially helpful, it prompted changes in the design even before I produced more high-fidelity mockups.

High-fidelity mockups

To help aid users in navigation, I moved the primary navigation to the left side of the screen and highlighted the current section. I also created a new contextual navigation along the top to help identify to which customer the data belonged.

The page name and available actions moved below the contextual navigation which gave us more room to expose buttons previously hidden in the hamburger menu. To take action on more than one alarm at once, we added checkboxes to the beginning of each row.

Once again, I brought these mockups to the development team and a few In-House SOC Analysts for feedback. After a few tweaks, we agreed to move forward.



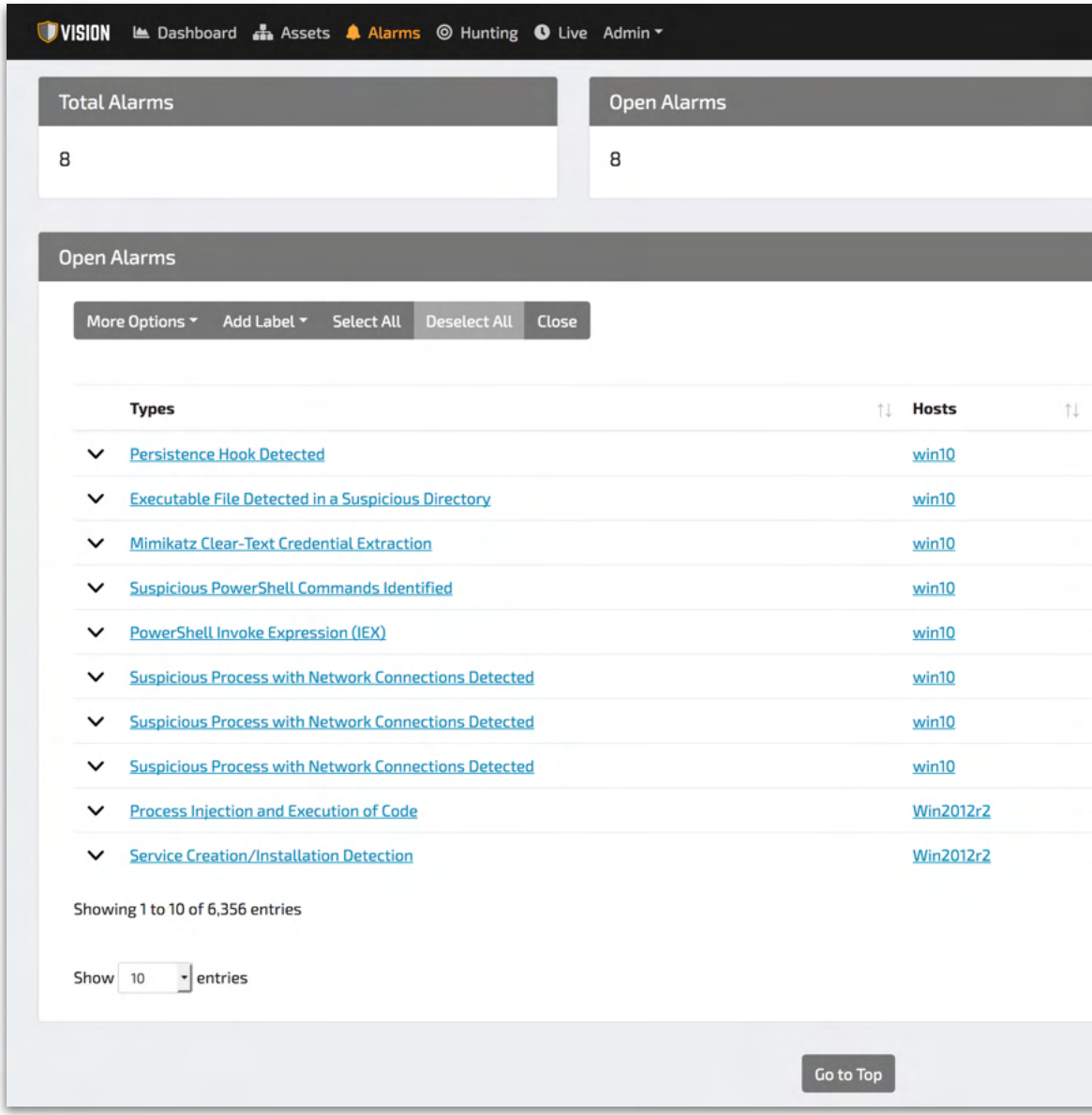
Incremental progress

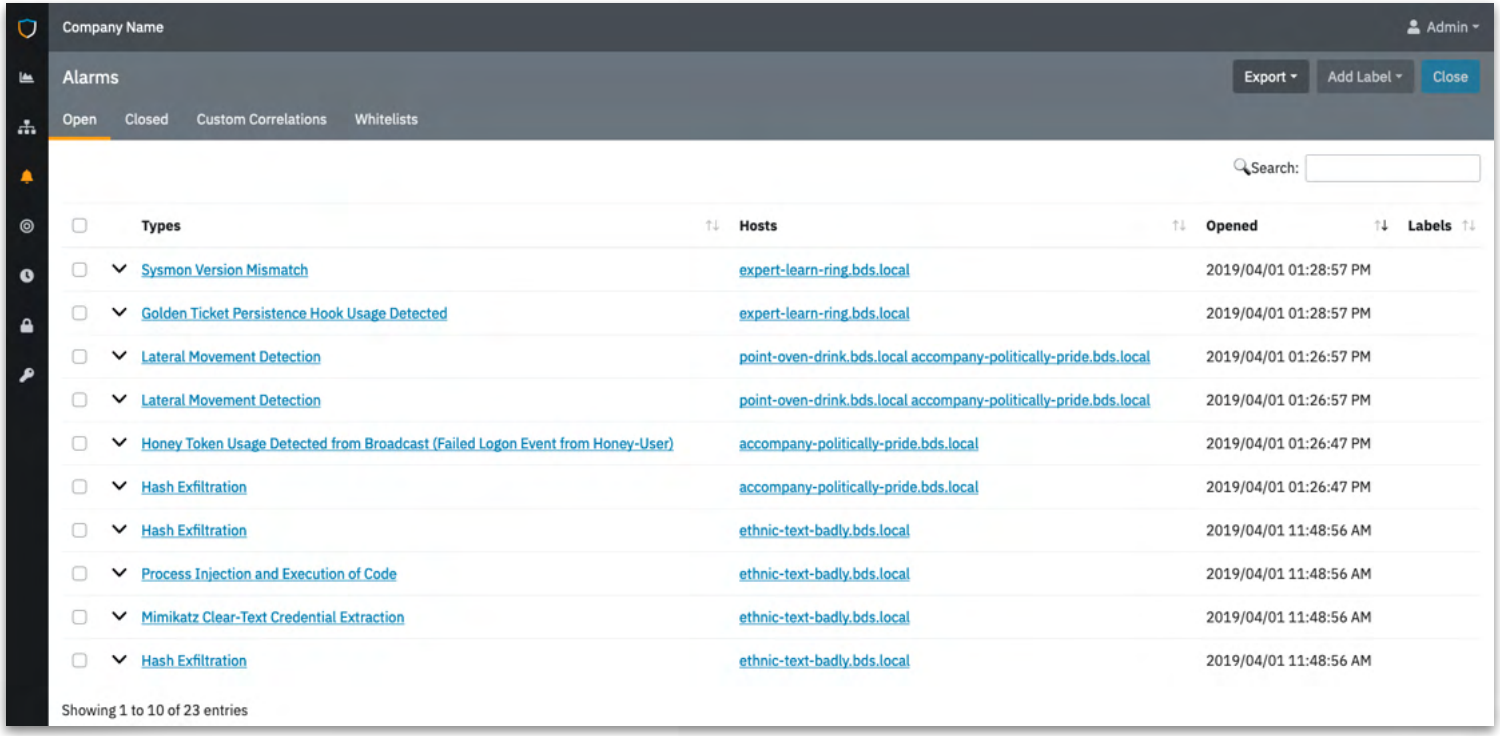
As the development team began to work on implementing the changes, we ran into a few snags. It turned out that the front-end was not in good shape. Based on an old version of Bootstrap and patched over with custom styles, it was hard to get the desired results.

At this point, we realized that this required a complete rework of the frontend. Since the developers were busy with feature work, I stepped up to build a custom Bootstrap 4 theme. I also made sure our color palette was accessible.

Before we could get to the interface changes I had mocked up, the theme needed applied throughout the product.

Shown here is the new theme applied to the Open Alarms page.





Realizing the design vision

A few months later, I worked with two front-end engineers to bring the design vision to realization. In the meantime, the company brand was also redesigned and the custom theme was updated to comply.

At this point, I performed usability studies with In-House SOC Analysts to determine whether the updated interface solved the issues surfaced in research. The analysts were extremely pleased with the update. And even though we weren't focusing much on our paid customer users at this point, we received glowing feedback from many of them, as well.

Results

In addition to overwhelming approval from In-House SOC Analysts, Binary Defense was recognized as a leader in Managed Detection and Response in the Forrester Wave and in the Gartner Market Guide for Managed Detection and Response. This recognition is a reflection of the work of the entire organization to improve the product experience.

This product evolution provided me with many opportunities for growth. For the first time, I designed an accessible color palette and selected a typeface to match both brand characteristics and usability standards. I also brushed off my development skills and learned a great deal about Bootstrap and modern JavaScript.

An area that I struggled with during this project was planning and scheduling the work. Since so many inputs were unknown, it made it hard to estimate the effort required. I worked to narrow scope to manageable chunks during future projects.

Some issues remained after this project. For instance, the pagination controls and row counts were not accessible without scrolling to the bottom of the screen. Also, it was hard for users to know if they had seen any alarms if they had to leave the interface and come back later. Fixes for these issues were planned for future releases.