The UserInsight Ingress Dashboard

Curran Kelleher
August 2013

This document summarizes my summer 2013 internship work at Rapid7.

I was part of a team developing a 1.0 product called UserInsight which is now in production. The UserInsight platform manages many kinds of live data feeds from corporate network assets and provides security analysts a Web-based interface to the data. The analysts can use the tool to closely monitor the overall security status of their corporate network. For more information on the product, check out the UserInsight page.

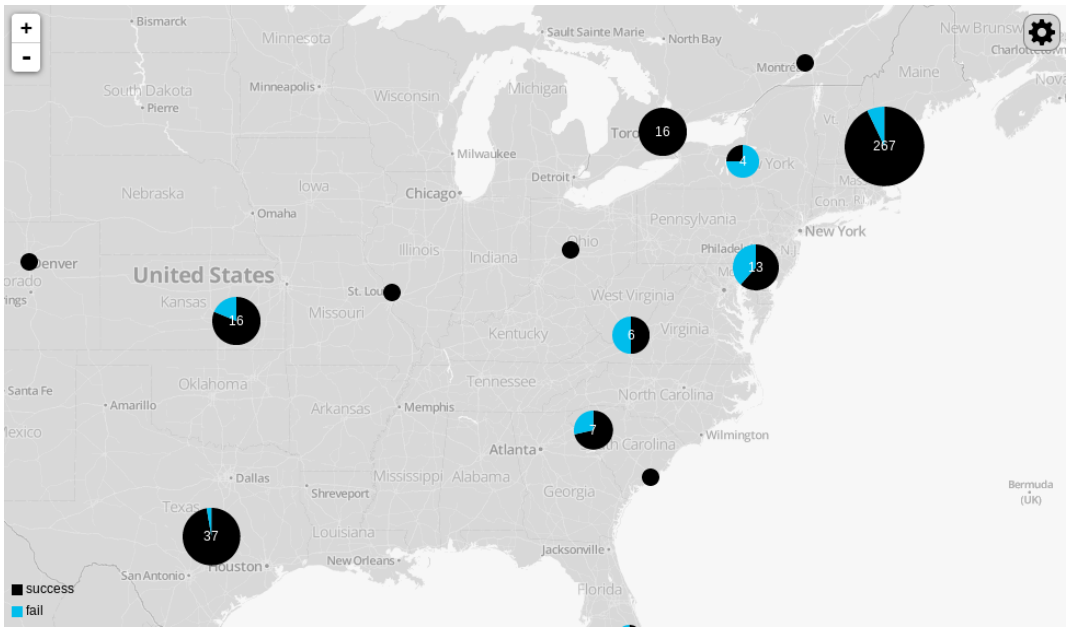
I was tasked with adding interactive visualizations to UserInsight. I engaged in an iterative visualization design and development process that involved roughly the following steps:

- Determine the structure of the data available (create data model diagrams).
- Determine the tasks that users must perform using the software (talk with the product managers).
- Based on the data and tasks, design a set of possible visualization approaches based on the visualization construction theory of Jaque Bertin and others in the field of data visualization.
- Prototype many visualizations and interactions that enable users to perform necessary tasks.
- Iterate the prototypes with the product team, throwing away most and enhancing others.
- Work with the engineering team to integrate the visualizations into the product through clean APIs.

During the course of my work at Rapid7, I authored the open source DashboardScaffold library. This library encapsulates generalized software patterns useful for creating interactive visualization dashboards with multiple linked views. The framework is oriented toward visualizations using D3.js, but is general enough to support other tools as well (such as Leaflet.js, which is used in the ingress dashboard for the map component).

The visualizations I created that ended up in UserInsight focus on logins to the corporate network. Login events are also called "ingresses". Each ingress event has many attributes including place, time, user, success status, and service used. After giving much thought to the visual representation of each of these attributes and working across many teams at Rapid7 to iterate the ideas (product managers, user experience designers, visual designers and developers), the Ingress Dashboard emerged. Below is a detailed description of the Ingress Dashboard and its features.

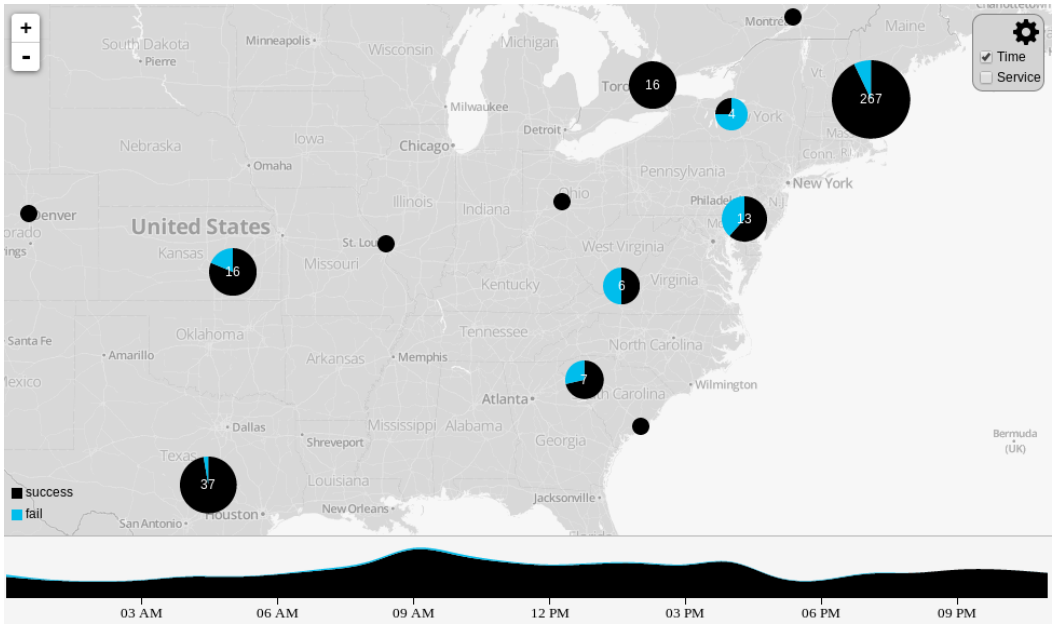
The map shows where users have logged into the network. Each pie chart on the map represents a group of ingresses clustered geographically. The number inside the pie chart is the number of ingresses in that cluster. The size of the pie charts corresponds to the number of ingresses using a log scale, so be aware that the area of the circles does not correspond linearly with the number of ingresses. Hovering over a pie chart shows the top five users in that cluster and their ingress count.



Zooming on the map can be done either by using the scroll wheel or clicking the + and - buttons in the top left corner. Panning can be done either by clicking and dragging on the map or using the arrow keys. Clicking on a pie chart causes the map to zoom into the area covered by that cluster. When zooming in, the

pie charts break apart to show a more detailed geographic distribution of the ingresses.

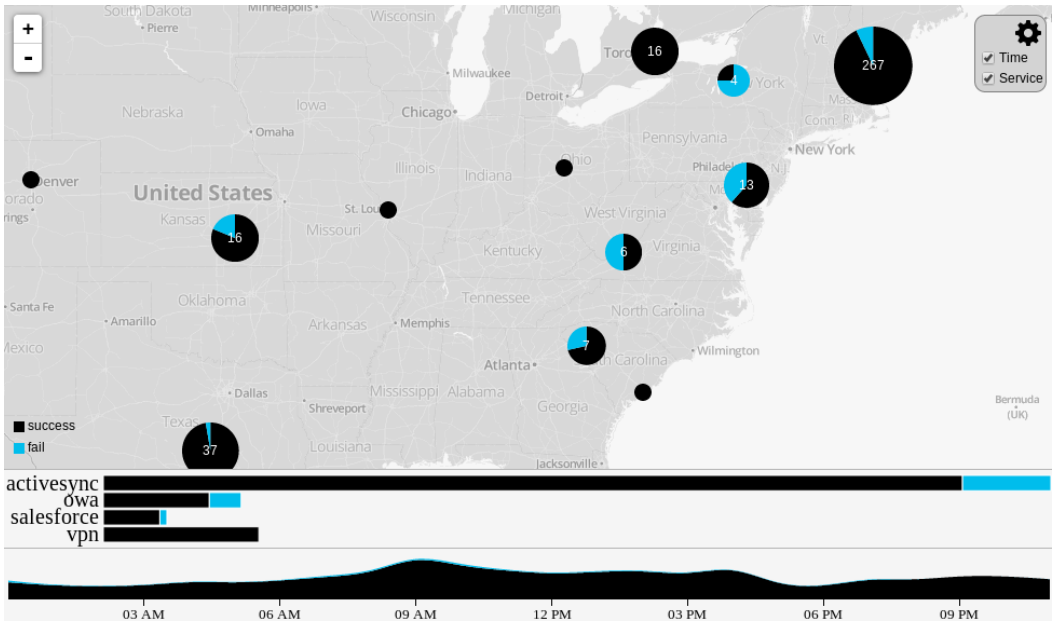
Clicking the gear in the upper right corner shows options for enabling two additional views of the data, the time view and the service view. Check the boxes to turn these views on and off. Click the gear again to hide the check boxes.



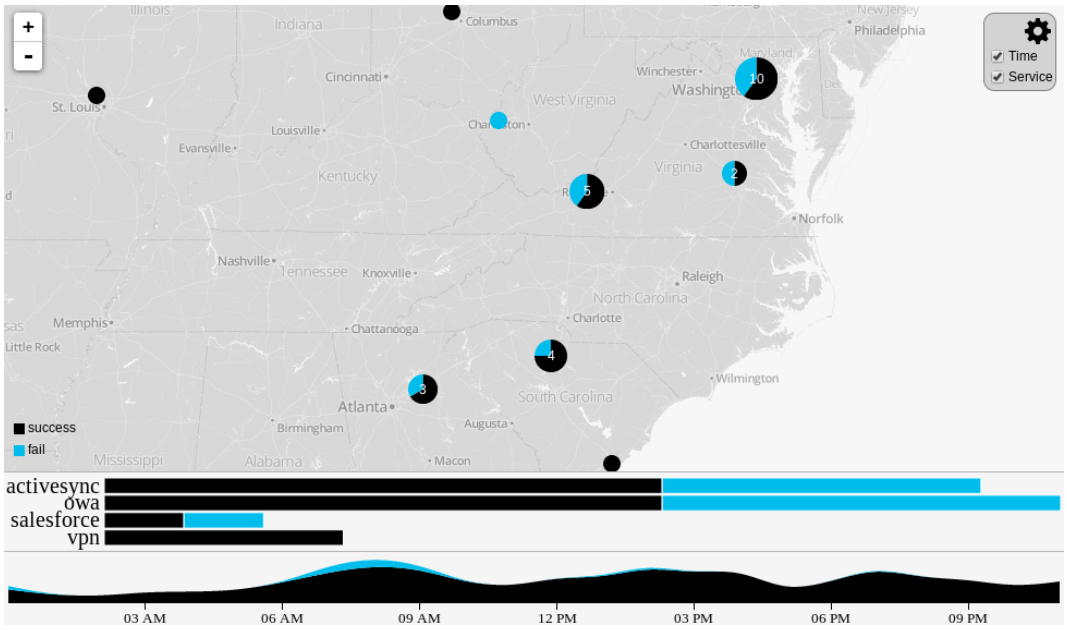
The time view uses a smoothed stacked area chart to show when ingresses occurred over time. Height represents the number of ingresses. On the time view, there is a black layer representing successful ingresses and a blue layer representing failed ingresses.

The service view shows the breakdown of services used to log in using a stacked bar chart. The size of each

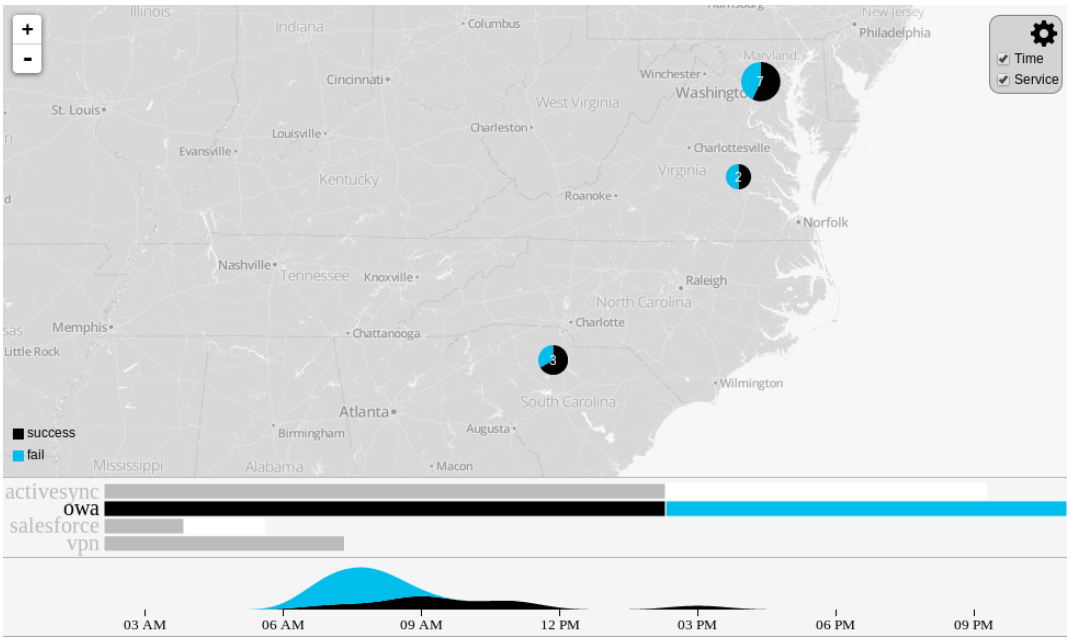
bar represents the number of ingresses. Here again, black represents successful ingresses and blue represents failed ingresses.



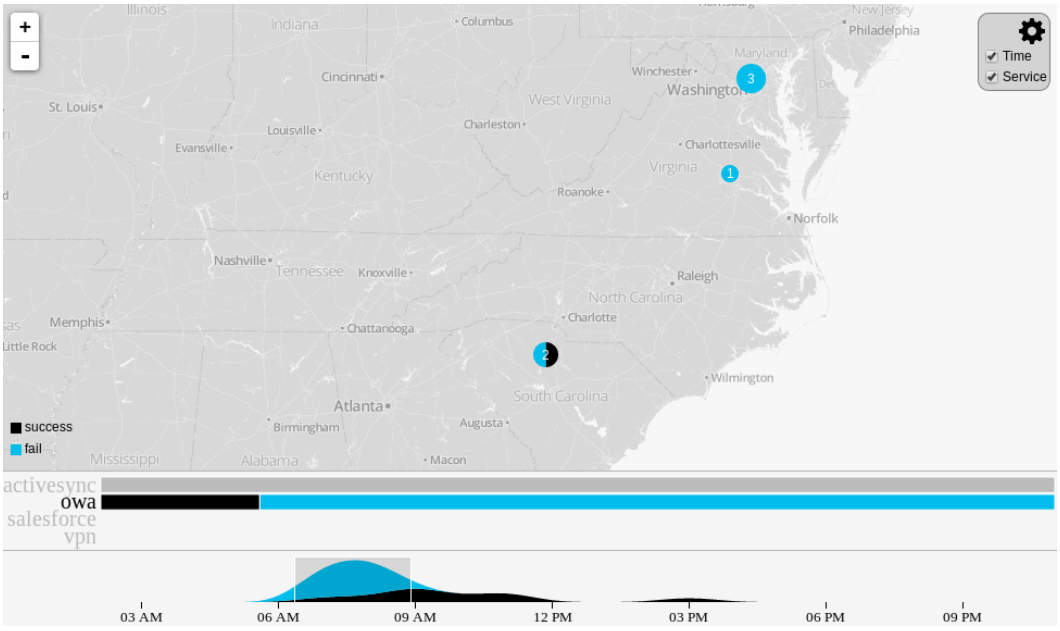
All views can be enabled simultaneously for a comprehensive overview of ingress activity.



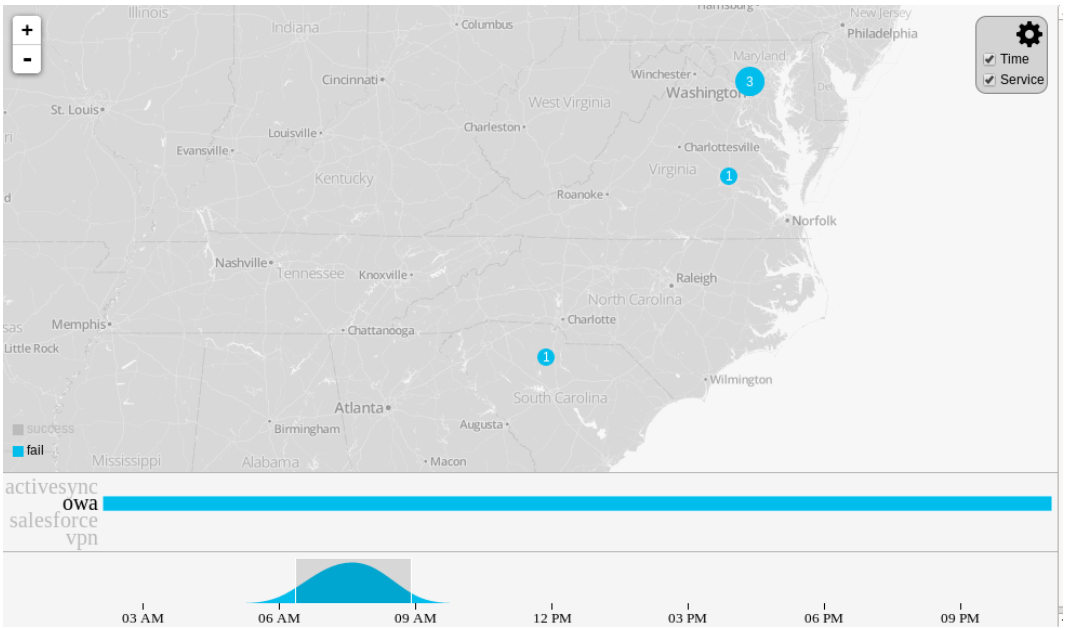
Zooming and panning in the map filters the ingresses shown in the other views. At any given moment, the data shown in the service and time views only shows ingress activity that happened within the region visible on the map.



Clicking on a bar in the service view filters the other views by service. At any given moment, the data shown in the map and time views only shows ingress activity that happened within the selected service. Clicking on the selected service again will turn off the service filter.



Clicking and dragging on the time view filters the other views by time. At any given moment, the data shown in the map and service views only shows ingress activity that happened within the selected time slice. Once a time slice is selected, the slice can be moved by dragging from the center and resized by dragging from the left or right edge. Clicking on an area outside the selected time slice turns off the time filter.



Clicking on legend elements causes all views to filter by success. If "success" is selected, only successful ingresses are shown. If "fail" is selected, only failed ingress attempts are shown. Clicking on the selected success status turns off the success filter.

Using this ingress visualization dashboard, multiple filters can be interactively manipulated to narrow down to ingress activity of interest.

Technologies used to create this dashboard include:

- HTML5 (CSS, JavaScript, SVG)
- D3.js
- Leaflet.js (using MapBox for the background map)
- Underscore.js
- Backbone.js
- Require.js