Splunk for DevOps? Absolutely!
Using Splunk across Docker, Bitbucket, Jenkins, Boomerang.js & Splunk-JS

Tom Martin | Staff ITOA Practitioner
Domnick Eger | Global DevOps Practitioner

August 8, 2017  Version 1.0
Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

Splunk, Splunk>, Listen to Your Data, The Engine for Machine Data, Splunk Cloud, Splunk Light and SPL are trademarks and registered trademarks of Splunk Inc. in the United States and other countries. All other brand names, product names, or trademarks belong to their respective owners. © 2017 Splunk Inc. All rights reserved.
Today’s Agenda

- Introduction
- Buttercup-Go!
- Modern Application Architecture
- DevOps Tooling & Monitoring
- Splunking the Modern App Dev Ecosystem
Technology Level Set

- Docker & docker-compose
- Buttercup-Go: a Node.js application
- Login: Auth0
- DevOps: CI/CD
  - Source Code Control: Git & Bitbucket
  - Build Automation: Jenkins
- Application Monitoring
  - Logging: Splunk JavaScript Logging + HTTP Event Collector (HEC) … or Mint.js!
  - End User Monitoring: Boomerang.js + HEC
  - APM Tools: New Relic
Buttercup-Go

A quick introduction to our application
Buttercup Go!

Give it a try!

https://buttercup.rocks/

How did you do?
Modern Architecture

The Technology behind the application
Buttercup Go Original Technology Layout

Source Code Management

Build Toolset

Jenkins Build Slaves

NGINX

APP SERVER

APP SERVER

JENKINS

BITBUCKET

>
‘Docker-ized’ Environment

Mac OS X or Linux Versions:
Mac OS X 10.11 or Higher
CentOS 6x or 7x
Ubuntu 14.04 or Higher

Reference Link: https://store.docker.com/editions/community/docker-ce-ce-desktop-mac
...and Docker Swarm Configured
What about Authentication?

Using Auth0 to track users and device information
How It Works From The User’s Perspective
How Auth0 Really Works

1. Initiates the authentication specifying the connection to use (e.g. Twitter)
2. User gets authenticated
3. Redirect to https://yourapp#id_token...
4. Validate id_token and get user profile
5. Call API sending JWT in Authorization header
6. Validate token and extract profile from the token payload
What Can I See In Splunk?

- User Behaviors
  - Number of logins
  - Authentications by source
  - Errors for a given user
  - Platforms
  - Trends
Modern Application Development Tools

What will we be using today?
Bitbucket (Source Code Management)

- Bitbucket (or GitHub) enables:
  - Simultaneous development for a single project
  - Different developers code different features
  - Tracking of Changes in Near-Realtime
  - Merge & Conflict Management
  - Issue Tracking and Jira Integrations
  - Automation with API's
Source Code Management (Bitbucket/GitHub)

- Devs work on ‘Projects’
- Projects have ‘repos’
- ‘git clone’ your repo
- Source is now local
Making Changes...There's a Process

**Actions**
- Create a new "branch"
- Switch context to your new branch
- Modify the code
- Verify the changes you made
- Add your changes to be submitted
- Commit your changes
- Push your changes to Bit Bucket
- Bitbucket UI
  - Create Pull Request
  - Merge

**Command Line**
- `git branch <branch name>`
- `git checkout <branch name>`
- `git status`
- `git add *`
- `git commit -m "your comments here"`
- `git push origin module1`
Automated Deployment - CI/CD Using Jenkins

- 1 Build Plan per Developer
- 1 Branch per Build Plan
- Jenkins will automatically build your project and deploy it when you merge your code

Build Plan

```bash
npm install
npm install newrelic
npm build
npm start
```
Collecting End User Data in Splunk

Splunking Application Data too!
Splunk HTTP Event Collector (HEC)

- A highly scalable endpoint to send data into Splunk over http(s)
- Settings → Data Input → HTTP Event Collector

```
curl -k https://<host>:8088/services/collector \
-H 'Authorization: Splunk <token>' \ 
-d '{"sourcetype": "mysourcetype", "event":"Hello, World!"}'
```
Capture Business Metrics From The App Via HEC

- Create HEC Token
- Install Splunk Logging
  ```
  npm install --save splunk-logging
  ```
- Add logging code
- Monitor your app

```javascript
var SplunkLogger;

var config = {
  token: "your-token-here",
  url: "https://splunk.local:8088"
};

var Logger = new SplunkLogger(config);

var payload = {
  message: {
    temperature: "70F",
    chickenCount: 500
  }
};

Logger.send(payload,
  function(err, resp, body) {
    // If successful, body will be {
    //   text: 'Success',
    //   code: 0
    // }
    console.log("Response from Splunk", body);
  }
);`
End User Experience - Boomerang.js & HEC

- Create HEC Token
- Add JavaScript to your page
- Monitor Real Users

```html
<!-- Boomerang.js Browser Code -->
<script src="https://s3.amazonaws.com/boomerang-splunk/boomerang.js"></script>
<script src="https://s3.amazonaws.com/boomerang-splunk/plugins/rt.js"></script>

BOOMR.init({
  beacon_type: "POST",
  beacon_auth_token: "Splunk <HEC Token>",
  site_domain: "What's up Buttercup?"
});

BOOMR.addVar({
  "ua_raw": navigator.userAgent,
  "appName": "Buttercup-Go-Student"
});

<!-- END Boomerang.js Browser Code -->
```
Add Mint.js to your app (mint.min.js) [ /app/utils/splunkhttp.js ]

Initialize Mint.js & start sending data!

Monitor Real Users

```
// Adding Mint.js
const mint = require('assets/scripts/mint/mint.min.js');

// Initialize Mint.js
Mint.initAndStartSession({ apiKey: 'abcdefgh', collectLogs: true, collectNetworkCalls: false });
Mint.setPackageName('Buttercup-Go');
```

```
// Send data to Mint!
Mint.setUserIdentifier('abc123');
Mint.logEvent('Buttercup-Go', 'info', <JSON - whatever data you want to send>);
```
Instrument the app
Add the JavaScript
APM Add-On in Splunk
View APM Data!
How Does This All Fit Together?

Understanding the pieces...
Run & Monitor

Node.js → User Interactions → Browser/Mobile → CDN

Auth0

Application Metrics → APM → Metrics & Alerts

Application Logs → APM

APM

CDN → Static Content

Auth0

Cache Metrics

Metrics & Alerts

End User Metrics → Auth0

Auth

Splunk
Build Chain Automation

- Developer
- Bitbucket
- Jenkins
- Node.js

Commit ➔ Build ➔ Deploy

Metrics

Splunk
Bringing It Together for DevOps
Bringing It Together for Business & Users
Realtime Gaming Metrics
Splunk with Modern Application Development

- Dev (Commit)
- Build
- Test
- Deploy
- Monitor
- End User
- APM
- Business

```bash
$ git clone https://Splunk@bitbucket.buttercuplabs.win/scm/aw/buttercupgo-student02.git .
Cloning into '.'...  
remote: Counting objects: 320, done. 
remote: Compressing objects: 100% (297/297), done. 
remote: Total 320 (delta 79), reused 204 (delta 9) 
Receiving objects: 100% (320/320), 3.27 MiB | 1.02 MiB/s, done. 
Resolving deltas: 100% (79/79), done. 
$ git branch images 
$ git checkout images 
Switched to branch 'images' 
$ git add * 
$ git commit -m "making stuff up"
```
Splunk with Modern Application Development

- Dev (Commit)
- Build
- Test
- Deploy
- Monitor
- End User
- APM
- Business

Visualize, Correlate, Search & Discover
Bringing It Together for Business & Users

Test drive the Splunk App for AWS: [https://splunkbase.splunk.com/app/1274/](https://splunkbase.splunk.com/app/1274/)

Tryout the New Relic Integration:
   Add-On: [https://splunkbase.splunk.com/app/3465/](https://splunkbase.splunk.com/app/3465/)
   App: [https://splunkbase.splunk.com/app/3465/](https://splunkbase.splunk.com/app/3465/)

Tryout the Jenkins App: [https://splunkbase.splunk.com/app/3332/](https://splunkbase.splunk.com/app/3332/)

Test Drive the Akamai App: [https://splunkbase.splunk.com/app/3026/](https://splunkbase.splunk.com/app/3026/)
Thank You

Don't forget to rate this session in the .conf2017 mobile app