Lightweight Software Labs with EclipseRT in the Cloud

Johannes Eickhold, Markus Knauer, Florian Waibel

EclipseSource
Sandboxing
User Perspective

- register
- access
- upload
- manage
- experiment
- play
Goals

- reuse existing authentication infrastructure
- minimal / no manual preparation per lab
- one lightweight container per lab
User Management

**Homebrewn vs. OAuth (Google)**

- requires lots of pre-existing infrastructure
- password recovery
- encryption

- already provides full account management
- encrypted
- enables further social network integration
  => outreach
Heavy to Lightweight Isolation Level

- Lightweight
- Heavyweight
Serving HTTP

SSL for https?

Dynamic Rewriting?
Our Decision
The Backend Server

User

Trial Application User

User Portal

mongoDB
Using the Lab

1. register at website

2. receive mail with registration key

3. access user portal with registration key (OAuth)

4. upload app (Web Application)

5. URL under which app is available

6. access app
Google OAuth for RAP

- uses RAP Scripting
- JavaScript -> popup
  - retrieves token from Google
  - sends token to RAP server
- enables support for Google Client API in RAP code
- separate release as extension to RAP is planned
Dynamic Reverse Proxy++

http://tabristrial.eclipsesource.com/foo/examples

http://localhost:20002/myapp/examples
location / {

rewrite_by_lua '...

local firstSegment = string.match( uri, "^/(.-)/" )
local res = ngx.location.capture( "/_resolve_backend/" .. firstSegment )
targetUrl = res.body
local parsedUrl = url.parse( targetUrl )
targetUrlPath = string.gsub( parsedUrl[ "path" ], "/", "" )
targetUrlHost = parsedUrl[ "scheme" ] .. ":/" .. parsedUrl[ "authority" ]
ngx.var.targetUrlHost = targetUrlHost
local rewriteUri = string.gsub( uri, firstSegment, targetUrlPath )
ngx.req.set_uri(rewriteUri)
'

proxy_set_header Host $host;
proxy_pass $targetUrlHost;
Caching URL Mapping in nginx

# HTTP server
lua_shared_dict dockerContainerUrls 1m;

...

-- search entry in nginx dictionary
local targetUrl = dockerContainerUrls:get( firstSegment )
if targetUrl == nil then...

...

Docker Container

- Based on Ubuntu image
- Contains OpenJDK 7 + Virgo

Static and dynamic containers
FROM base:ubuntu-12.10
...
RUN apt-get install -y openjdk-7-jre-headless
...
CMD su virgo ${VIRGO_HOME}/bin/startup.sh
Docker Java

admin console: docker build
String **build**(String imageTag, String dockerFolder);

user portal: docker run
String **create**(String imageTag, String containerName, int containerPort);
void **start**(String containerId, int hostPort, int containerPort);
Inside Docker: Virgo

- Upload of app via HTTP
- Static Docker instance ("user portal") uses JMX bean of Virgo to deploy the app
Visit the Labs

DEMO
Outlook

- Release as well documented open source template project
- Use Virgo RAP Server?
- [Bug 428574 - Investigate deploying HIPP instances inside of Docker containers](#)
Questions?

DOCKER

ALL THE THINGS
Evaluate this session

1. Sign-in: [www.eclipsecon.org](http://www.eclipsecon.org)
2. Select session from schedule
3. Evaluate

---

San Francisco
March 17-20, 2014

EclipseCon 2014