2016-2017 Work Notes

Contents

[Jun 2016 3](#_Toc516755991)

[Proxies: 3](#_Toc516755992)

[Change environment variables: 3](#_Toc516755993)

[July 4 – 10 , 2016 3](#_Toc516755994)

[Proxies for npm: 3](#_Toc516755995)

[Yeoman Gulp+Angular 3](#_Toc516755996)

[Proxies for Bower 3](#_Toc516755997)

[After Yeoman install 4](#_Toc516755998)

[gulp default browser to chrome 4](#_Toc516755999)

[July 11, 2016 4](#_Toc516756000)

[Install FontAwsome 4](#_Toc516756001)

[July 15, 2016 4](#_Toc516756002)

[Fix GIT SSL certificate problem: self signed certificate in certificate chain 4](#_Toc516756003)

[July 19, 2016 5](#_Toc516756004)

[Jetty install for angular web app 5](#_Toc516756005)

[To install jetty service 6](#_Toc516756006)

[To remove jetty service 6](#_Toc516756007)

[To start in command line 6](#_Toc516756008)

[To start jetty service in command line 6](#_Toc516756009)

[To stop jetty service in command line 6](#_Toc516756010)

[July 27, 2016 6](#_Toc516756011)

[A good Spark framework article 6](#_Toc516756012)

[A good article for Angular error handling 6](#_Toc516756013)

[AngularJS httpInterceptors and toastr -- circular dependency problem 6](#_Toc516756014)

[Nov 30, 2016 6](#_Toc516756015)

[Windows get groups 6](#_Toc516756016)

[Jan 10, 2017 7](#_Toc516756017)

[PowerShell: Get SID from AD User/Group 7](#_Toc516756018)

[PowerShell: Get AD User/Group from SID 7](#_Toc516756019)

[Jan 17, 2017 7](#_Toc516756020)

[Code not running in IE 11, works fine in Chrome 7](#_Toc516756021)

[Feb 7, 2017 7](#_Toc516756022)

[How to install Node.js & npm without running msi 7](#_Toc516756023)

[Feb 16, 2017 8](#_Toc516756024)

[How to add local jar files into maven project 8](#_Toc516756025)

[Jun 27, 2017 9](#_Toc516756026)

[Create local workspace for Sabre 9](#_Toc516756027)

# Jun 2016

## Proxies:

eclipse -- network

maven settings.xml -- look for M2\_HOME

## Change environment variables:

$ setx <var> “<value>”

Unset

$ setx <var> “”

The value will take effective when re-entering CMD

# July 4 – 10 , 2016

## Proxies for npm:

$ npm config get proxy

$ npm config get https-proxy

$npm config set registry http://registry.npmjs.org/

$ npm config set proxy <http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/>

$ npm config set https-proxy <http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/>

$ npm config set strict-ssl false

$ npm --proxy [http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080](http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/) --without-ssl --insecure -g install npm -g

\*\*\*$ npm --proxy [http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080](http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/) --without-ssl --insecure -g install npm –g

## Yeoman Gulp+Angular

$ npm --proxy [http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080](http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/) --without-ssl --insecure -g install -g yo gulp bower

$ npm --proxy [http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080](http://oak%5Chahe:Hewei123@oproxy.fg.rbc.com:8080/) --without-ssl --insecure -g install -g generator-gulp-angular

$ mkdir ProcessManagerUI

$ yo gulp-angular

## Proxies for Bower

In .bowerrc

{

"directory": "bower\_components",

"registry": "http://bower.herokuapp.com",

"proxy":"http://hahe:Hewei123@oproxy.fg.rbc.com:8080/",

"https-proxy":"http://hahe:Hewei123@oproxy.fg.rbc.com:8080/"

}

## After Yeoman install

It's time to use Gulp tasks:

- `$ gulp` to build an optimized version of your application in folder dist

- `$ gulp serve` to start BrowserSync server on your source files with live reload

- `$ gulp serve:dist` to start BrowserSync server on your optimized application without live reload

- `$ gulp test` to run your unit tests with Karma

- `$ gulp test:auto` to run your unit tests with Karma in watch mode

- `$ gulp protractor` to launch your e2e tests with Protractor

- `$ gulp protractor:dist` to launch your e2e tests with Protractor on the dist files

More details are available in docs and recipes

<https://github.com/Swiip/generator-gulp-angular/tree/master/docs>

## gulp default browser to chrome

In server.js

function browserSyncInit(baseDir, browser) {  
 //browser = browser === undefined ? 'default' : browser;  
 browser = browser === undefined ? 'chrome' : browser;

… …

# July 11, 2016

## Install FontAwsome

$ bower install components-font-awesome --save

# July 15, 2016

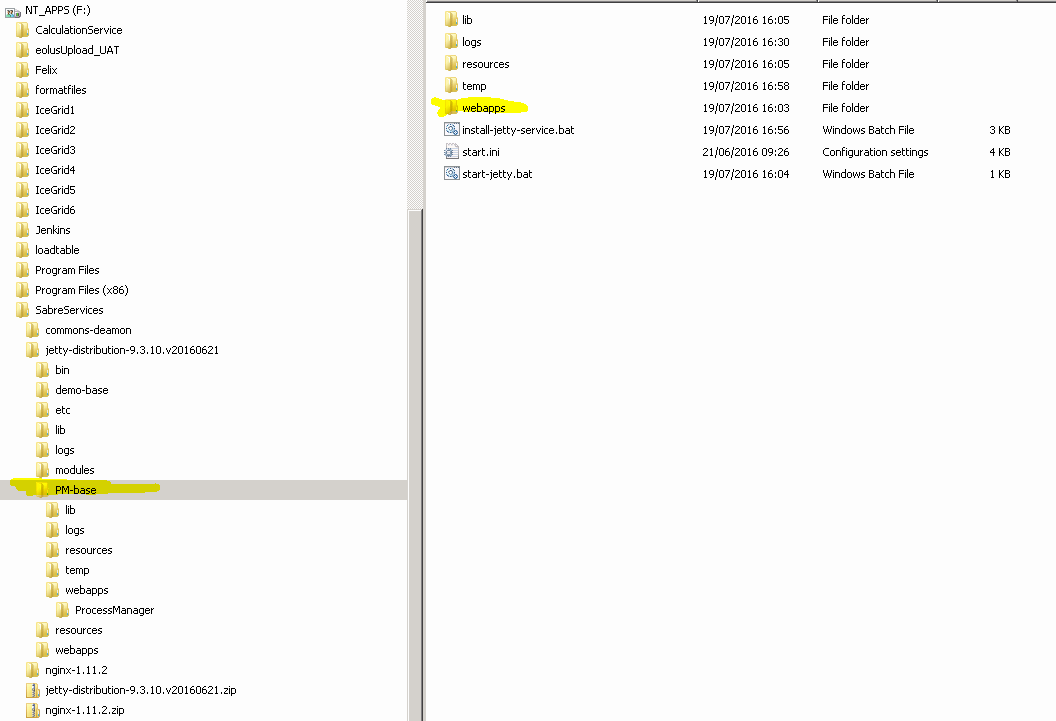
## Fix GIT SSL certificate problem: self signed certificate in certificate chain

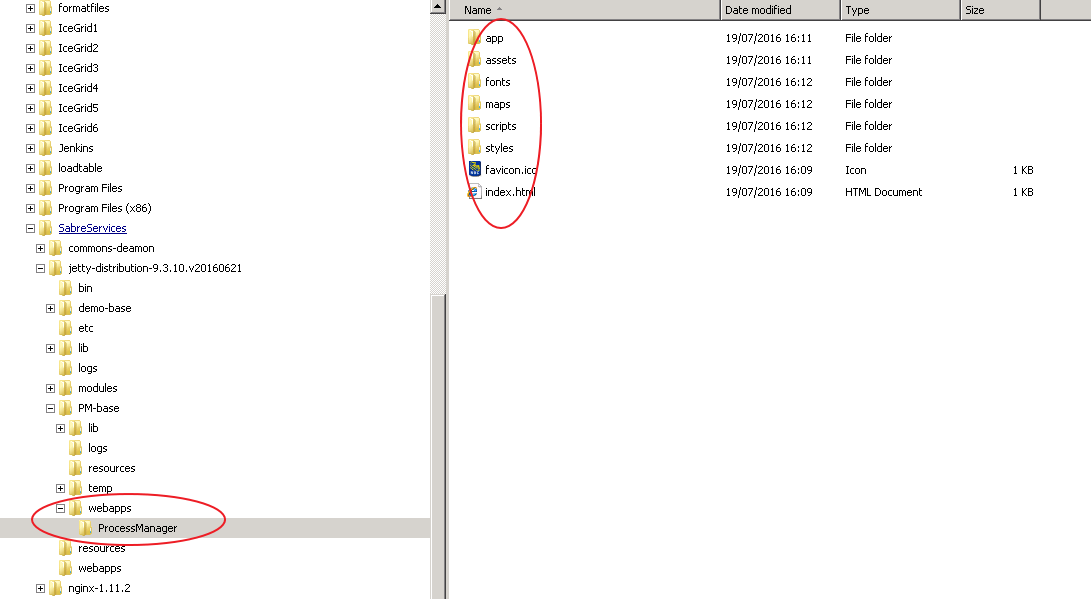
<http://stackoverflow.com/questions/11621768/how-can-i-make-git-accept-a-self-signed-certificate>

git -c http.sslVerify=false clone https://domain.com/path/to/git

# July 19, 2016

## Jetty install for angular web app





## To install jetty service

$ install.bat

## To remove jetty service

$ sc.exe delete JettyService

## To start in command line

$ start-jetty.bat

## To start jetty service in command line

$ net start JettyService

## To stop jetty service in command line

$ net stop JettyService

# July 27, 2016

## A good Spark framework article

<http://www.mscharhag.com/java/building-rest-api-with-spark>

## A good article for Angular error handling

<http://michalostruszka.pl/blog/2013/09/06/selective_http_errors_handling_in_angularjs/>

<https://gist.github.com/nadouani/7410645>

## [AngularJS httpInterceptors and toastr](http://stackoverflow.com/questions/31946379/angularjs-httpinterceptors-and-toastr) -- circular dependency problem

1. decouple the services by using events: when an error is detected in the interceptor, broadcast an event (with the error information) on the root scope. In the toastr service (or in a service that uses toastr), listen on these events on the root scope and display them.
2. instead of injecting toastr in the interceptor, inject the $injector service. And when you need a toastr instance, call $injector.get('toastr').

<http://brewhouse.io/blog/2014/12/09/authentication-made-simple-in-single-page-angularjs-applications.html>

<https://medium.com/opinionated-angularjs/techniques-for-authentication-in-angularjs-applications-7bbf0346acec#.x9m71b7rd>

# Nov 30, 2016

## Windows get groups

$ net user <user name> /domain

e.g. $ net user hahe /domain

# Jan 10, 2017

## PowerShell: Get SID from AD User/Group

<http://www.christiano.ch/wordpress/2009/08/26/powershell-get-sid-from-ad-active-directory-user-group-using-powershell/>

In windows PowerShell:

PS H:\> $AdObj = New-Object System.Security.Principal.NTAccount(**"*<OAK\DomainUser>***")

PS H:\> $strSID = $AdObj.Translate([System.Security.Principal.SecurityIdentifier])

PS H:\> $strSID.Value

S-1-5-21-1123561945-1364589140-839522115-513

## PowerShell: Get AD User/Group from SID

In windows PowerShell:

PS H:\> $objSID = New-Object System.Security.Principal.SecurityIdentifier("***S-1-5-21-1123561945-1364589140-839522115-513***")

PS H:\> $objUser = $objSID.Translate( [System.Security.Principal.NTAccount])

PS H:\> $objUser.Value

OAK\Domain Users

# Jan 17, 2017

## [Code not running in IE 11, works fine in Chrome](http://stackoverflow.com/questions/30867172/code-not-running-in-ie-11-works-fine-in-chrome)

IE 11 does not support string.startsWith, fix (angularJs):

***In index.run.js***

if (!String.prototype.startsWith) {

String.prototype.startsWith = function(searchString, position){

position = position || 0;

return this.substr(position, searchString.length) === searchString;

};

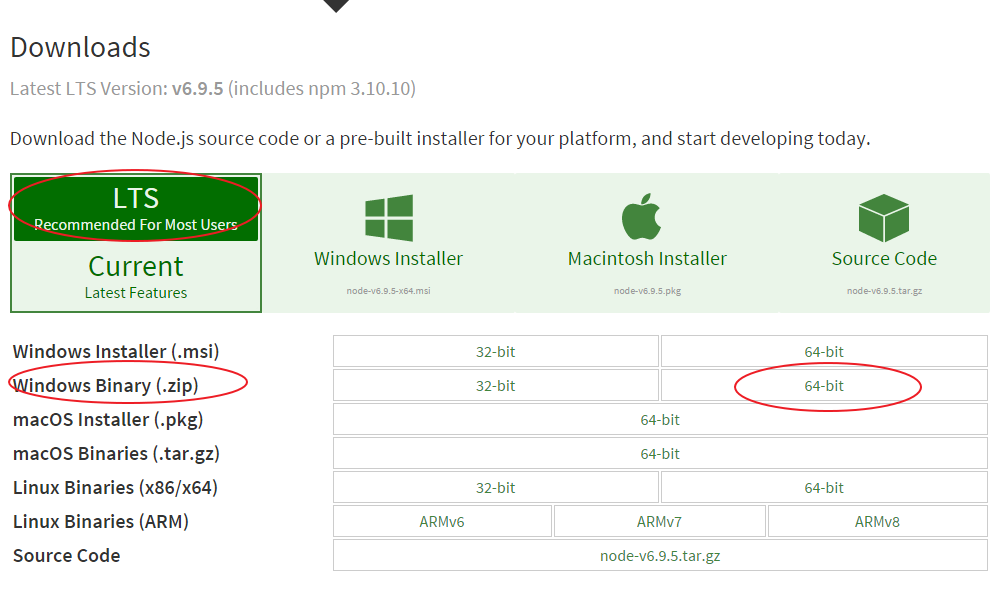
}

# Feb 7, 2017

## How to install Node.js & npm without running msi

1. Go to nodejs download <https://nodejs.org/en/download/>

Select proper version -- LTS, window Binary (.zip)



1. Unzip downloaded zip file
2. Add unzipped folder into %PATH%

# Feb 16, 2017

## How to add local jar files into maven project

*mvn install:install-file*

*-Dfile=<path-to-file>*

*-DgroupId=<group-id>*

*-DartifactId=<artifact-id>*

*-Dversion=<version>*

*-Dpackaging=<packaging>*

*-DgeneratePom=true*

*Where: <path-to-file> the path to the file to load*

*<group-id> the group that the file should be registered under*

*<artifact-id> the artifact name for the file*

*<version> the version of the file*

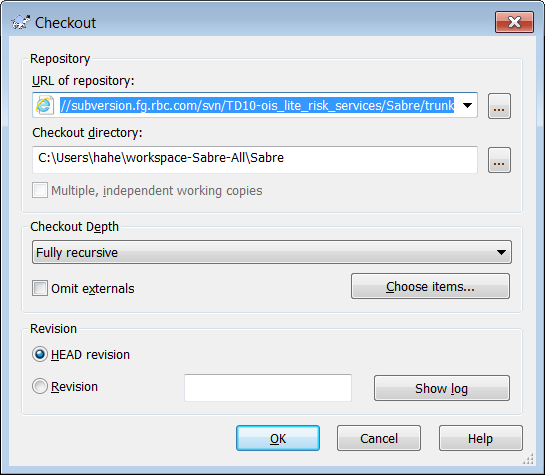
*<packaging> the packaging of the file e.g. jar*

**E.g.: $ mvn install:install-file -Dfile="C:\Users\hahe\Downloads\cml-logging-log4j1-2.0.0.jar" -DgroupId=com.rbc.rbccm.rik0 -DartifactId=cml-logging-log4j1 -Dversion=2.0.0 -Dpackaging=jar -DgeneratePom=true**

# Jun 27, 2017

## Create local workspace for Sabre

1. Create a directory for new workspace C:\Users\hahe\workspace-Sabre-All\
2. Using **TortoiseSVN**, checkout <http://subversion.fg.rbc.com/svn/TD10-ois_lite_risk_services/Sabre/trunk> to C:\Users\hahe\workspace-Sabre-All\Sabre



1. In Eclipse switch to workspace workspace-Sabre-All
2. Import existing maven project, select Root Directory C:\Users\hahe\workspace-Sabre-All\Sabre
3. In C:\Users\hahe\workspace-Sabre-All\Sabre, run

*$mvn –DskipTests clean install*