2016

Go-Live Self Assessment

BUILDING THE EXECUTABLE

INMAN, JOSIAH

outofc0ntr0l@outlook.com

Contents

Prerequisites

Nexpose

This is the Rapid7 Guide to getting ready with Ruby and the Gem.

If you have Ubuntu

https://github.com/rapid7/nexpose-client/wiki/Getting-started-on-Ubuntu

If you have Windows

https://github.com/rapid7/nexpose-client/wiki/Getting-started-on-Windows

If you follow the guides above then you will be okay except you will need to install ocra

If you don't follow the guides above here is some info. You have to run ocra from a windows box.

Ruby

Ruby Gems

https://rubygems.org/pages/download

Nexpose-client

https://rubygems.org/gems/nexpose

```
gem install nexpose
```

Highline

https://rubygems.org/gems/highline

```
gem install highline
```

Ocra (only runs on windows)

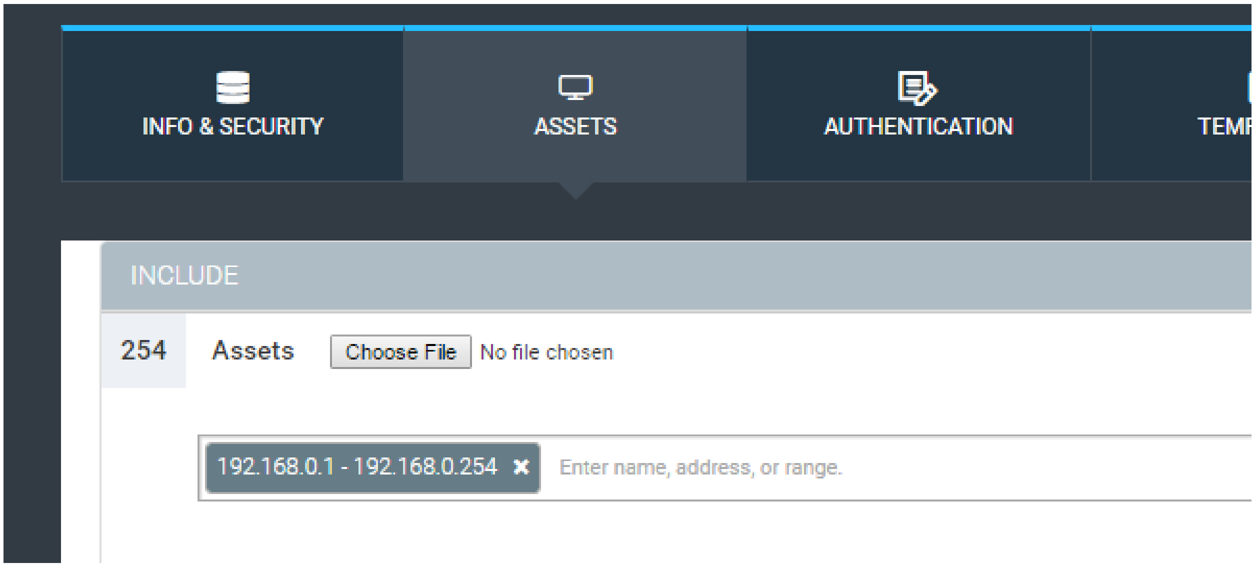
https://rubygems.org/gems/ocra

```
gem install ocra
```

Create a Go-Live Site

Create a site to hold the scan results (We do this so we can specify ahead of time the range to scan)

This site will hold all the go-live data. If you have just 1 site already and want to use that it is fine too just need to make sure you have the ranges that you will allow users to scan loaded



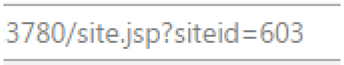
192.168.0.1 - 192.168.0.254

Assign an engine

Assign any required creds

Assign a default template

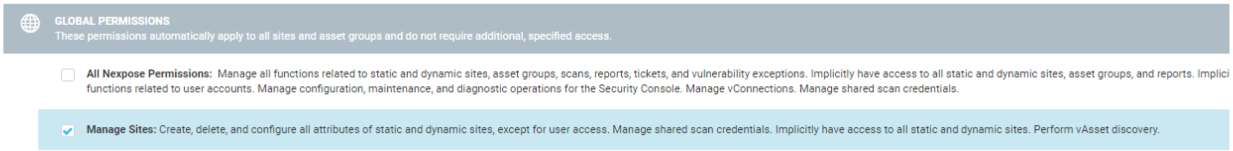
Once you save the group get the site ID _____________________ (to do this look in the address bar after clicking on the site to manage it)



3780/site.jsp?siteid=603

Create an Account - Go-Live Account

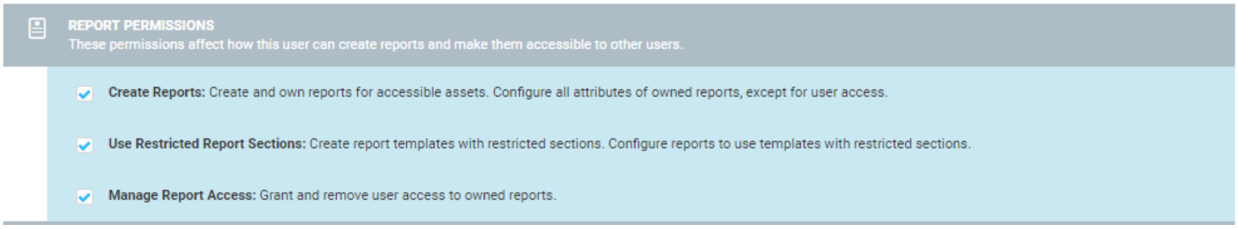
And give the following permissions. The user account must have manage Sites due to the need to access shared creds. If you are not using shared creds then these can be removed from the script and more granular controls emplaced. The API controls are not super ganular so you have to st this up with these permissions. :(



GLOBAL PERMISSIONS
These permissions automatically apply to all sites and asset groups and do not require additional, specified access.

☐ **All Nexpose Permissions:** Manage all functions related to static and dynamic sites, asset groups, scans, reports, tickets, and vulnerability exceptions. Implicitly have access to all static and dynamic sites, asset groups, and reports. Implici functions related to user accounts. Manage configuration, maintenance, and diagnostic operations for the Security Console. Manage vConnections. Manage shared scan credentials.

☑ **Manage Sites:** Create, delete, and configure all attributes of static and dynamic sites, except for user access. Manage shared scan credentials. Implicitly have access to all static and dynamic sites. Perform vAsset discovery.



REPORT PERMISSIONS
These permissions affect how this user can create reports and make them accessible to other users.

☑ **Create Reports:** Create and own reports for accessible assets. Configure all attributes of owned reports, except for user access.

☑ **Use Restricted Report Sections:** Create report templates with restricted sections. Configure reports to use templates with restricted sections.

☑ **Manage Report Access:** Grant and remove user access to owned reports.

Give permission to just the 1 site you created.

Edit the script for your environment

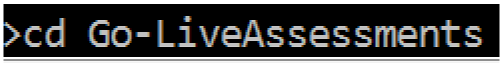
Replace everything in red. We hard code the username in the tool as it allows for a bit of obfuscation but also a username specific to the tool. (We use a new one each time we compile the tool to ensure the users are using the latest version.

```
####################################################################
# script version
@version = "6"
# Name of team to reach out to if there is an issue
@team_name = <YOUR TEAM NAME HERE>
# Email of the team
@team_email = <YOUR TEAM EMAIL HERE>
#
# Load Site - This is the site we use to scan all assets from.
# We are not using cross site correlation. If you are using it then this could be
# adjusted to pick the site the asset belongs to to scan it.
# if you are setting up like we have then you want to set the site ID of the site
# to the below.
@siteIDNum = <YOUR SITE ID OF THE SITE CREATED>
# template to use for the scan
@template = 'full-audit-without-web-spider' ←Template we use you can use whatever
# Report ID Name
@report_ID = 'audit-report-TEMPLATE' <- Report Template we use you can use whatever
#
# get the Username
# I prefer to keep the username in the code as not to give it to
# everyone so they arent trying to login to the console. I understand that
# someone could get it anyway this is merely out of sight out of mind.
# we have this as acceptable risk. Your org may be different
#
# I also use the version number with the username so they arent using old versions
#
#uncomment this line and comment out the @account below it to ask the user for the username
#leave as is for hard coded username
#@account = get_user
@account = "gl" + "ac" + "co" + "un" + "t" + @version <- this is what we use (glaccount6) you can use whatever
#
#IP of console
@IP = <YOUR CONSOLE IP>
#####################################################################
```

Compile the script

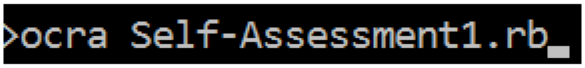
Open ruby command prompt

Cd to the directory with the Self assessment script (or wherever you put the script)



```
>cd Go-LiveAssessments
```

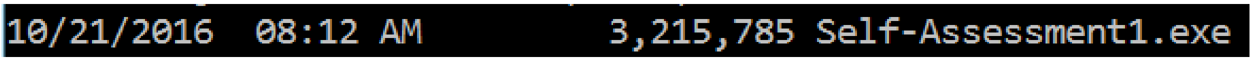
Next run the compilation. Ocra has many options if you want to use them. At the most basic run the following:



```
>ocra Self-Assessment1.rb
```

It will run you through the script. Even if it errors out it will finish the compile.

Once ran you will see a complied version of the script



```
10/21/2016  08:12 AM         3,215,785 Self-Assessment1.exe
```

Once compiled you can run the executable anywhere you need that has access to the console.