



**Qualys Auth using Puppet**

Project Title: Qualys auth using puppet

Project ID:

Authors: vishal gori,

Date: 07/21/2016

**Table of Contents**

Executive Summary 3

Project Name: 3

Basic Solution Description 3

Request Information 3

Platforms Impacted 3

Platform Interdependencies 3

Subsequent Projects 3

Project Requirements 4

Project Requirements 4

Technical Stack 5

High Level Technical Stack Description 5

Data Overview 5

High Level Technical Stack Diagram 5

Present State (Slide 1 - Today) 5

Target State (Slide 2 - Tomorrow) 5

Project Technical Information 6

Integrations 6

Web Services & Endpoints 6

High Level Testing Scope Expected 6

Additional Materials 6

Project Delivery 7

Project Roadmap & Schedule 7

Project Dependencies 7

Product Dependencies 7

Project Staffing 8

Project Staffing Requirements 8

Key roles 8

Estimates / Effort 8

Appendix 9

Appendix A: Architecture Checklist 9

# Executive Summary

## Project Name: Qualys Auth using Puppet

## Basic Solution Description

Vulnerability management of existing AWS infrastructure using Puppet for automation and Qualys for vulnerability assessment.

## Request Information

**Date Requested: 07/21/2016**

**Anticipated Kickoff Date: 08/01/2016**

**Anticipated Delivery Date: 09/30/2016**

**Knowledge Center/Platform Owner *(if applicable)*: Alex Lapitski**

## Platforms Impacted

Authenticated Qualys scan requires a new user specific to Qualys; be created with administrator privileges on each host which needs to be scanned.

## Platform Interdependencies

Current scope of the project is limited to all hosts in DEV/DEV2/QA/QA2/PROD AWS accounts. These are all Linux based hosts. Puppet client already exists in each of these hosts.

## Subsequent Projects

* Continuous vulnerability assessment and remediation of vulnerabilities in DEV/DEV2/QA/QA2/PROD AWS accounts.
* Periodic risk assessment and controls.
* Security baselining.

# Project Requirements

## Project Requirements

* Development of Puppet module to create Qualys user.
* Scheduling of key rotation in co-relation with existing Qualys scan schedule.

A similar solution already exists which creates and deletes the user created for Qualys auth scan. This approach needs to be modified slightly so that Puppet only deletes authentication keys instead of deleting the user itself. This way we only need to push new keys and not worry about re-creating the Qualys user each time. If a new set of keys are to be used, they need to be pushed to Qualys as well. The schedule exists already for instances not built using AppBuilder. Exact same schedule needs to be followed.

Puppet master can rotate keys and upload the newly created ones to CES service location below:

<https://ces-aws.cengage.info/services/dataservice/qualys/qualys_auth_scan/id_rsa.pub>

The command to replace existing keys is below:

curl -H "Content-Type:text/plain" -X "POST" -d @id\_rsa.pub <https://ces-aws.cengage.info/services/dataservice/qualys/qualys_auth_scan/id_rsa.pub>

Currently, no authentication is needed to access this service.

# Technical Stack

## High Level Technical Stack Description

* Technical inventory for existing implementation is maintained at below locations:

<http://wiki.cengage.com/display/SEC/Qualys_auth_scan_AB>

and

<http://stash.corp.web:7990/projects/SEC/repos/qualys_auth_scan/browse>

* New project’s inventory can be maintained at:

<http://wiki.cengage.com/display/SEC/Qualys_auth_scan_puppet>

and

<http://stash.corp.web:7990/projects/SEC>

## Data Overview

<< Includes types of content in scope for this project, volume of content, delivery format, schema to be used etc. (envision this as a chart) >>

## High Level Technical Stack Diagram

### Present State (Slide 1 - Today)



Target State (Slide 2 - Tomorrow)



# Project Technical Information

## High Level Testing Scope Expected

## Additional Materials

Additional Materials and Diagrams that will be used in the creation of the project. Additional examples include Story Boards, User Interaction Diagrams etc.

# Project Delivery

## Project Roadmap & Schedule

JIRA ticket below shows estimate project roadmap.

<https://jira.cengage.com/browse/SEC-19>

Below table maintains progress:

|  |  |  |  |
| --- | --- | --- | --- |
| Date | To Do | In Progress | Done |
| 07/21/2016 | Risk Analysis | Planning | [Present State](#_Present_State_(Slide) |
|  |  |  |  |
|  |  |  |  |

## 

## Project Dependencies

* Linux VM in AWS SecLabs account with developer libraries installed
* Puppet
* Qualys API

## External Dependencies

SmartShift

[CloudSec](https://jira.cengage.com/secure/RapidBoard.jspa?rapidView=1664&projectKey=SEC&view=detail&selectedIssue=SEC-19&quickFilter=9260)

# Project Staffing

## Project Staffing Requirements

### Key roles

* Vishal Gori - Security Architect
* Aaron Weaver - System / Business Analyst
* Chetan Krishna, Vishal Gori - Engineers

### Estimates / Effort

TBD

# Appendix

Additional Materials and Details required for the project

## 

## Appendix A: Architecture Checklist

The checklist is not to drive behavior but to create the discussion. Deviations from company standards is accepted and expected in many cases.

New Services? If yes, which platform(s) and how big?

Technologies used are core to Cengage or reviewed by architecture?

Extensions / plug-ins are using best practices? How large of extensions? Extended platform aware?

Transient data storage (CES, caches, queues) identified and reviewed?

Long lived data is persisted using content services (LCS et. Al.)?

Schema or data model changes are identified in the build document and have been reviewed by IA domain?

Content pipeline tools to support scaling for use in products considered?

Platform Overview updated with this projects impacts?

Components are deployed using App Builder?

Systems are designed for the cloud (auto healing, horizontal and dynamic scaling)?

SSO is used for authentication, CMP is used for identity, and OLR is used for entitlements?

Magma and React for the UI? If not, why?

Transaction monitoring built in and designed for exceptions?

Security concerns (e.g. FERPA, COPPA, PCI, SOX)?