OFFICE OF INSPECTOR GENERAL

CORPORATION FOR NATIONAL & COMMUNITY SERVICE

November 30, 2017

TO: Tom Hanley
Chief Information Officer

FROM: Stuart Axenfeld /s/
Assistant Inspector General for Audit

SUBJECT: Office of Inspector General's (OIG) Final Report: CNCS Web Application Vulnerability Assessment

Attached is the final OIG Report: CNCS Web Application Vulnerability Assessment. Under the Corporation's audit resolution policy, a final management decision on the findings and recommendations in this report is due by May 30, 2018. Notice of final action is due by November 30, 2018.

If you have questions or wish to discuss the report, please contact Guy Hadsall at (202) 606-9375, or S.Axenfeld@cncsoig.gov.

Attachment

cc: Lori Giblin, Chief Risk Officer
Monica Kitlas, Agency Audits & Investigations Coordinator
Sarah Mirzakhani, Principal, CliftonLarsonAllen LLP

CNCS Web Application Vulnerability Assessment

November 28, 2017

Contents

- Audit team
- Summary
- Objective
- Methodology
- Overview of Work Performed
- Results
- Findings categorized by Open Web Application Security Project Top 10
- Recommendations

Audit Team

Name	Phone Number	Email Address
Sarah Mirzakhani, CliftonLarsonAllen (CLA) IT Principal	571-227-9660	Sarah.Mirzakhani@claconnect.com
David Scaffido, CLA IT Manager	571-227-9668	David.Scaffido@claconnect.com
Chris Miller, CLA IT Senior Associate	571-227-9632	Chris.Miller@claconnect.com
Guy Hadsall, OIG, CTO	202-606-9375	G.Hadsall@cncsoig.gov

Summary

- **Findings**
  - Application vulnerabilities were more prevalent in production applications including SQL Injection and Cross-Site Scripting.
  - Missing patches
  - Unsupported software
  - Configuration weaknesses
  - Incorrect information in Interconnection Security Agreement
- **Recommendations**
  1. Improve patching effectiveness
  2. Update unsupported software
  3. Address configuration weaknesses to fully remediate vulnerabilities
  4. Practice secure coding to prevent SQL Injection and Cross-Site Scripting weaknesses
  5. Update Interconnection Security Agreement

Objective

- To perform web application vulnerability assessment and report any risks that could lead to IT security incidents, recommend improvements in the operations of the information systems, and identify gaps between the Corporation's current information security posture and industry best practices.

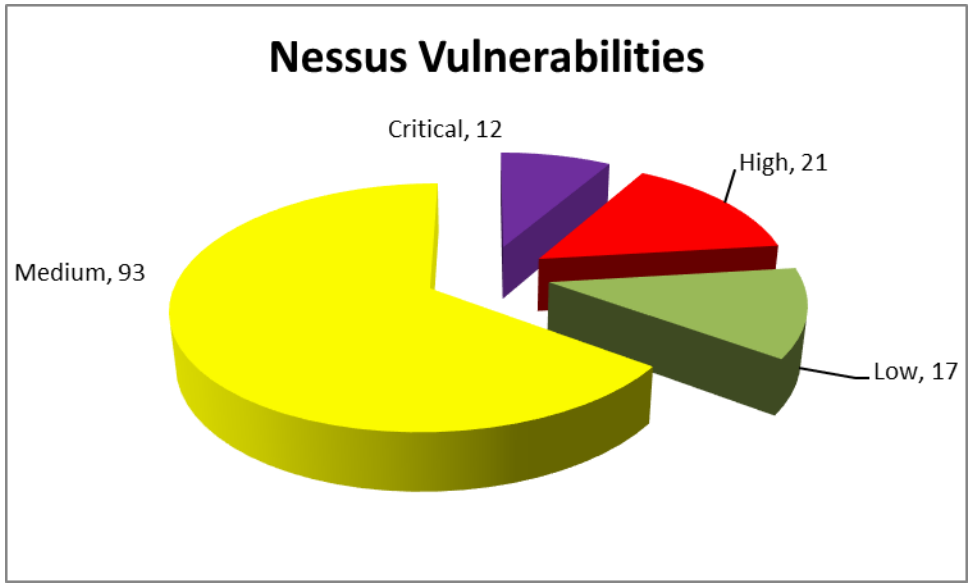
Methodology

- CLA's approach to performing the network and web application security testing is in accordance with NIST SP 800-115, *Technical Guide to Information Security Testing and Assessment*.
- CLA used the following tools: Nessus, NMAP, HP's WebInspect and BurpSuite.
- The assessment followed the following phases:
  - Discovery
  - Vulnerability Analysis
  - Credentialed Scanning
  - Monitoring and Reporting

Overview of Work Performed

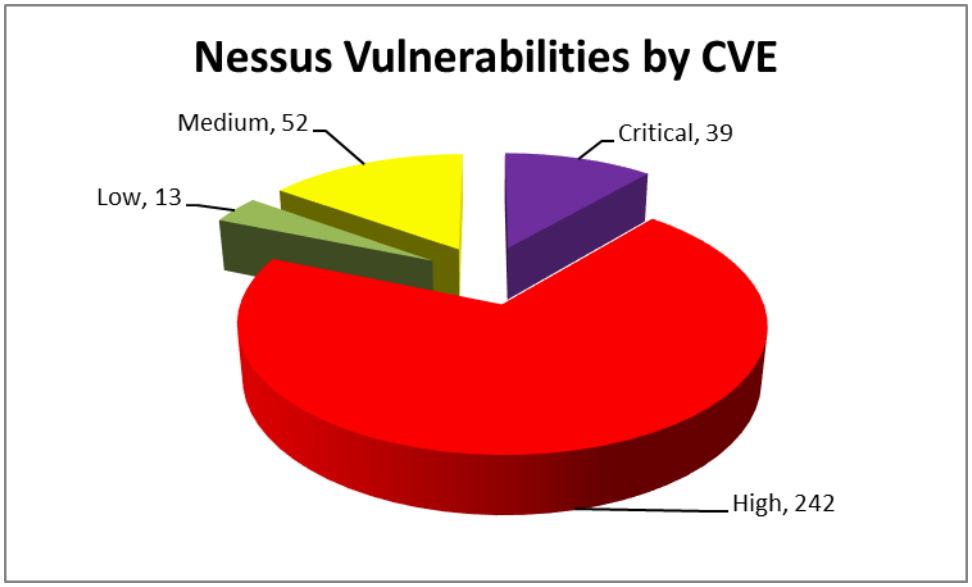
- Reviewed the following web applications and supporting infrastructure:
  - Employee On-boarding System (EOS)
    - ◊ Production, Development and Staging
  - Contractor On-boarding System (COS)
    - ◊ Production, Development and Staging
  - OnBoarding Admin and Development

Nessus Results



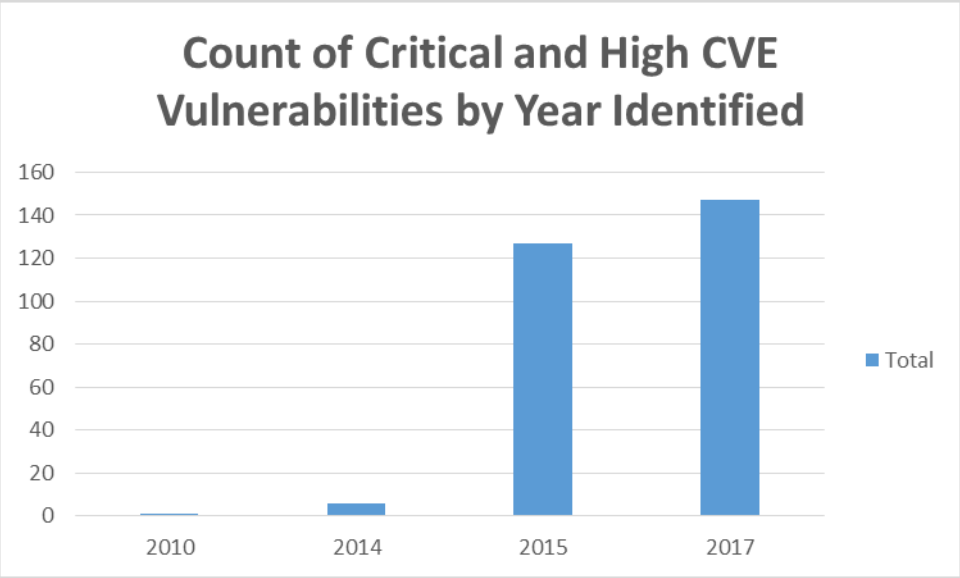
This chart describes the identified number of vulnerabilities by Nessus plugin identification number.

Nessus Results, continued



This chart describes the identified number of vulnerabilities by Common Vulnerability Exposure (CVE) Number. One Nessus plugin ID may have multiple known vulnerabilities as represented by the CVE number.

Nessus Results, continued

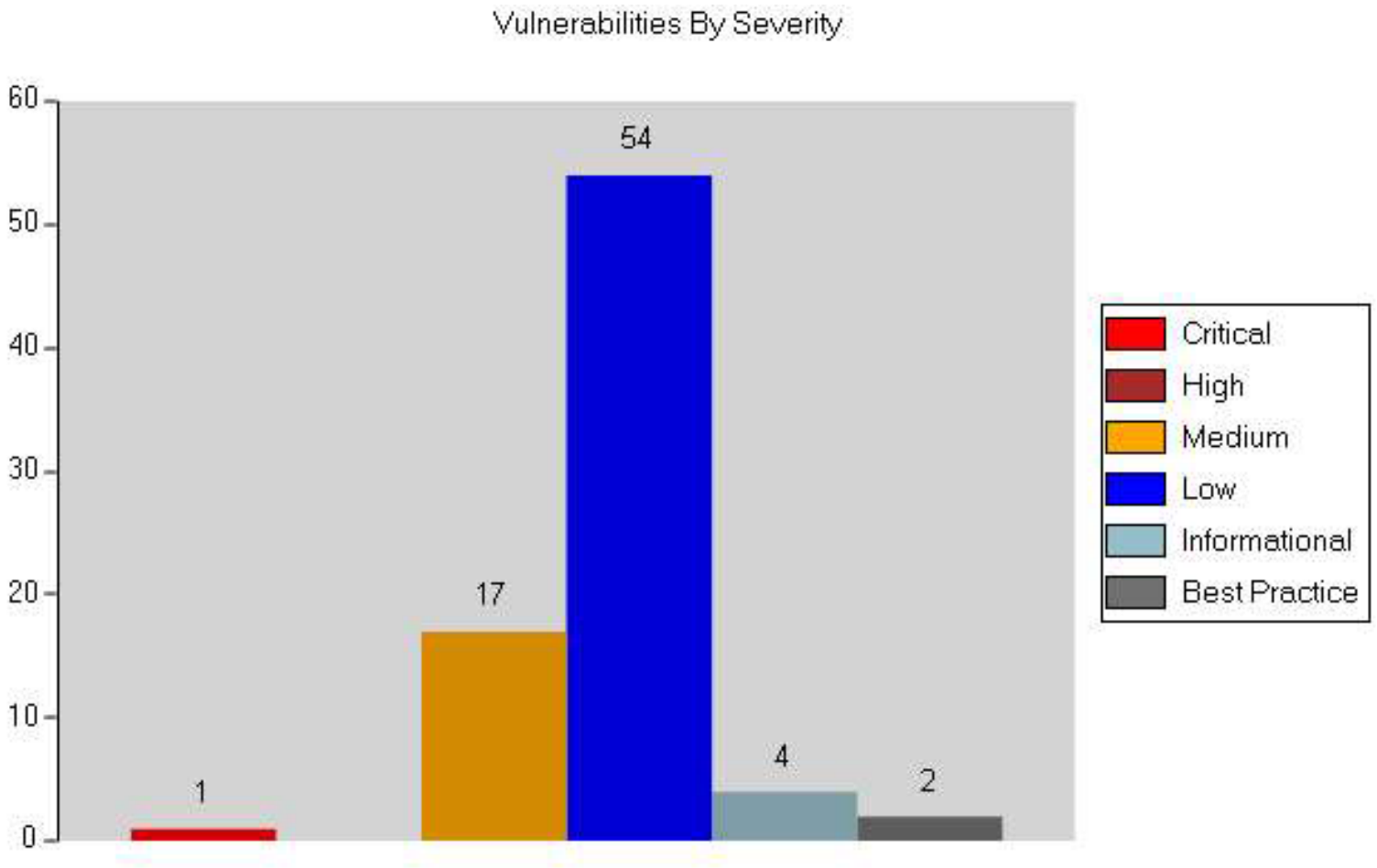


This chart describes the age of identified vulnerabilities by CVE number.

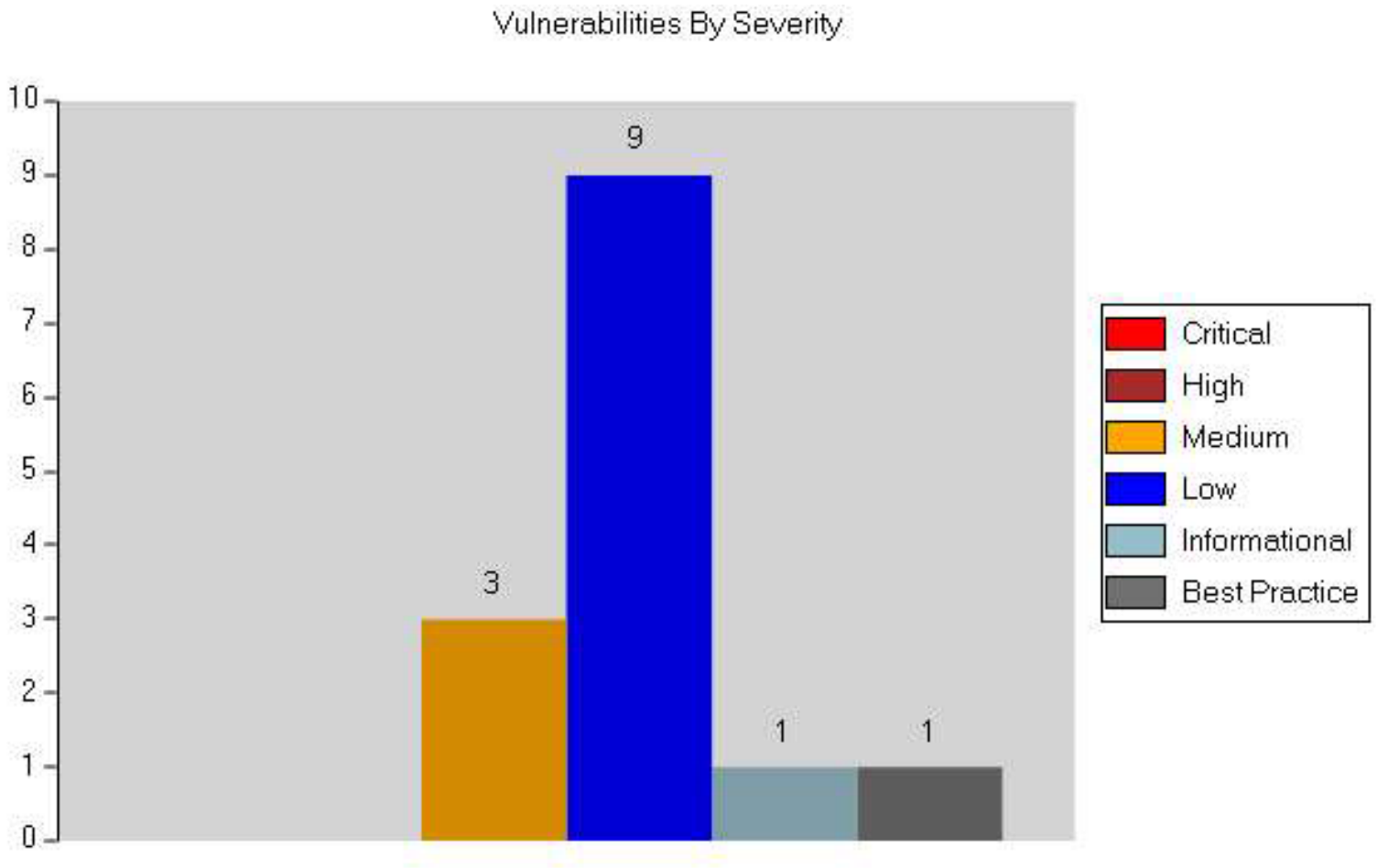
Webinspect Results

- The following slides highlight the number of vulnerabilities and configuration issues identified in the following systems:
  - COS, COS Development, and COS Staging
  - EOS, EOS Development, and EOS Staging
- We noted that the volume of identified vulnerabilities increased substantially in the production environment relative to the development and staging environments.

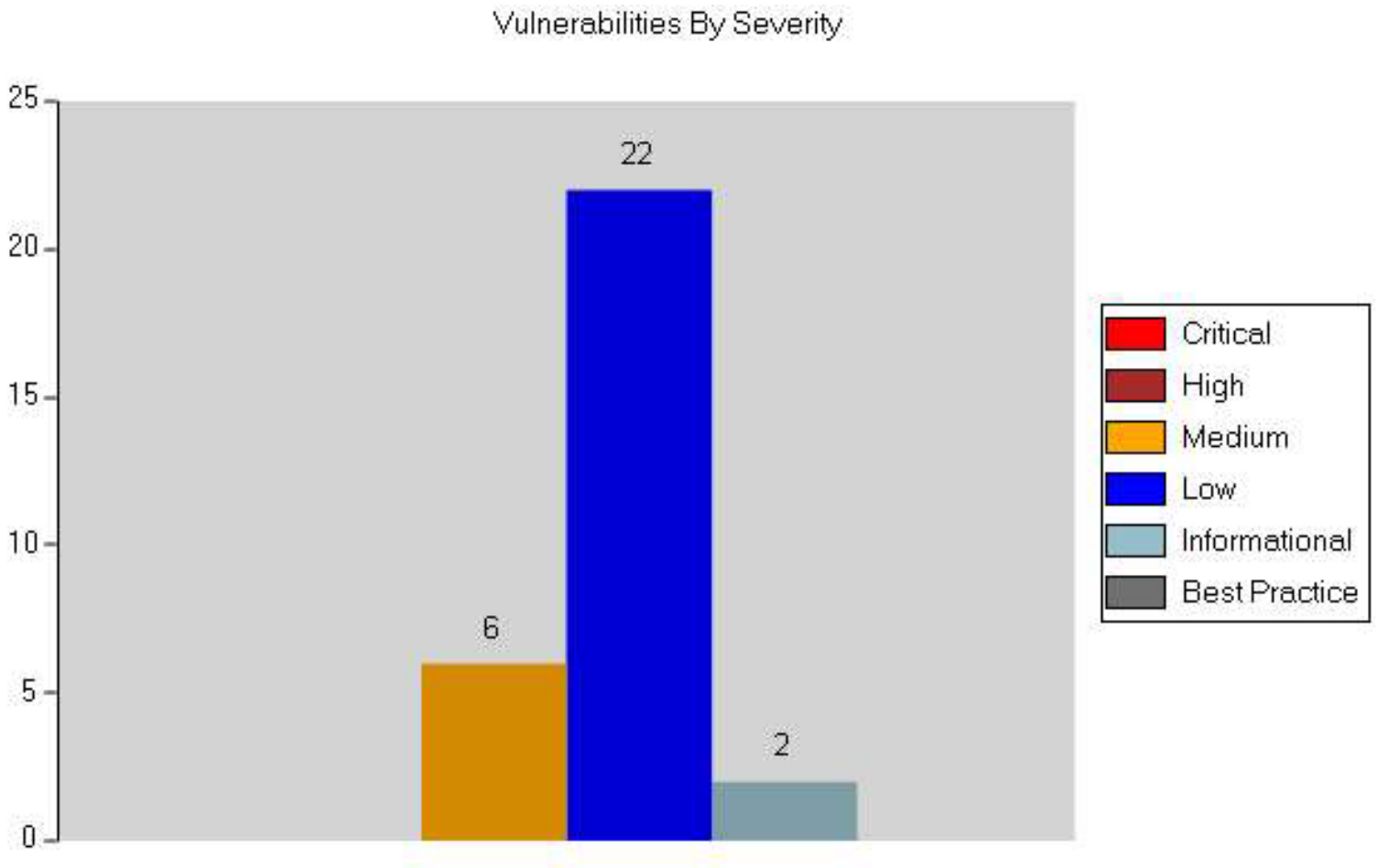
Webinspect COS Results



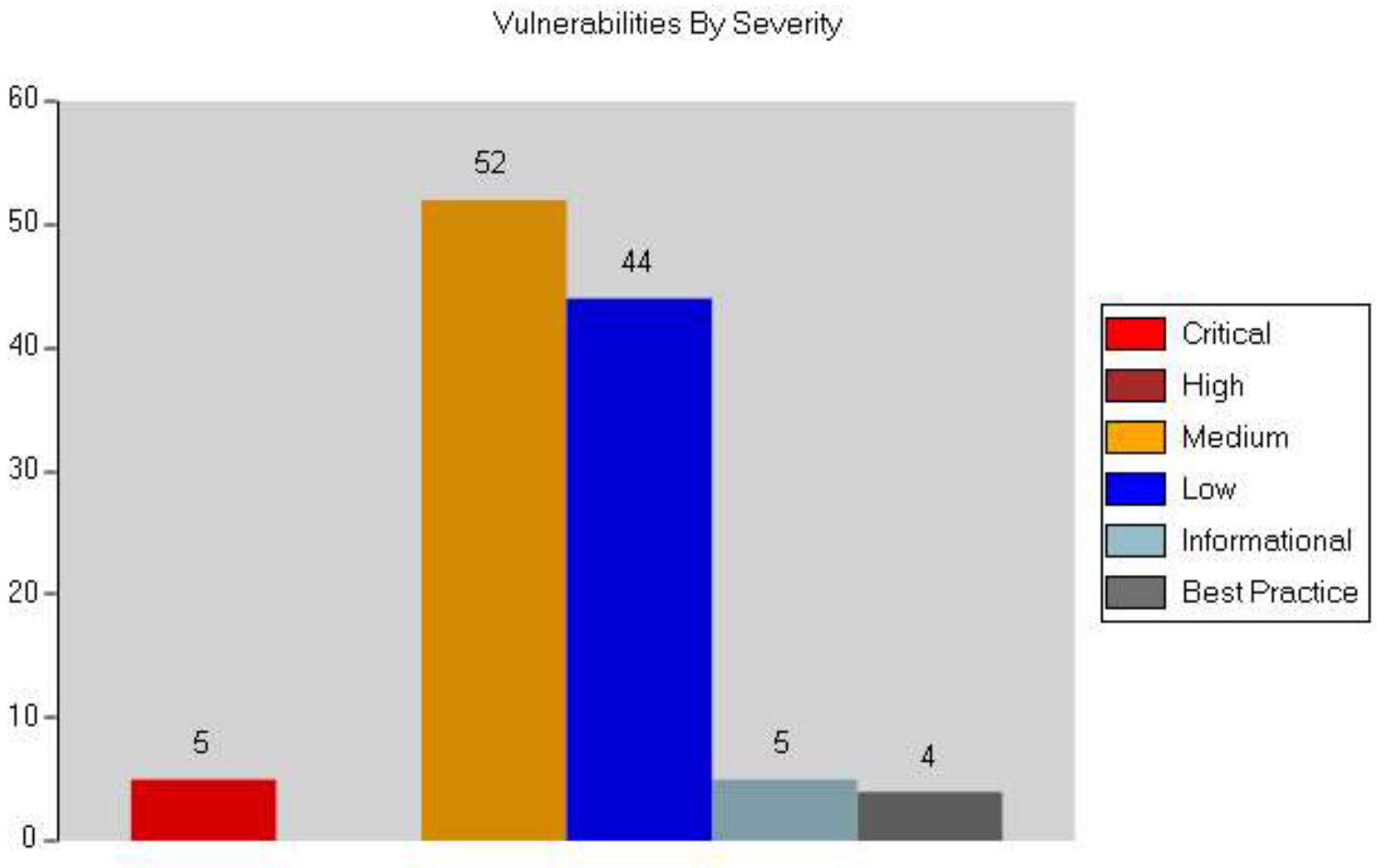
Webinspect COSstaging Results



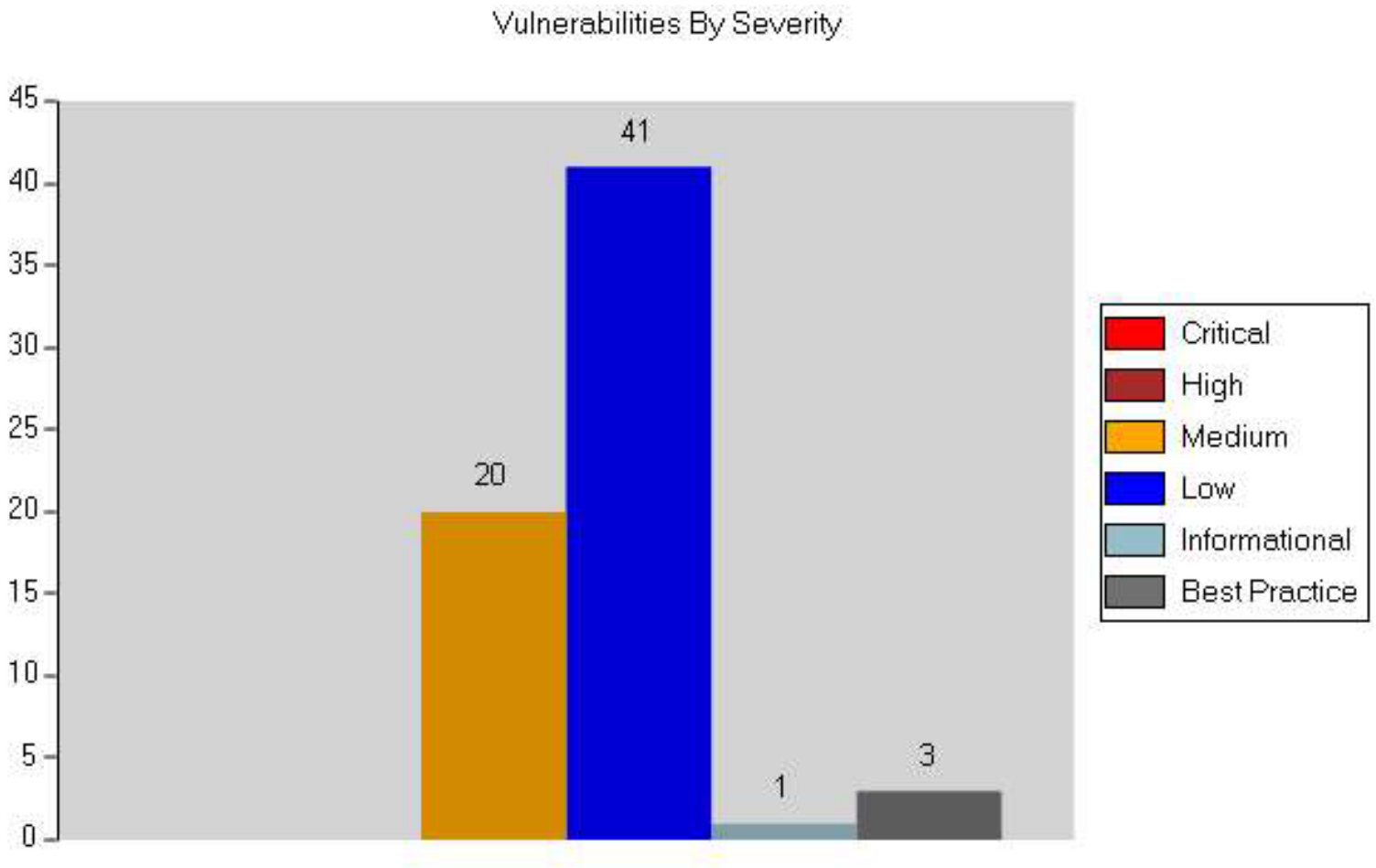
Webinspect COSdev Results



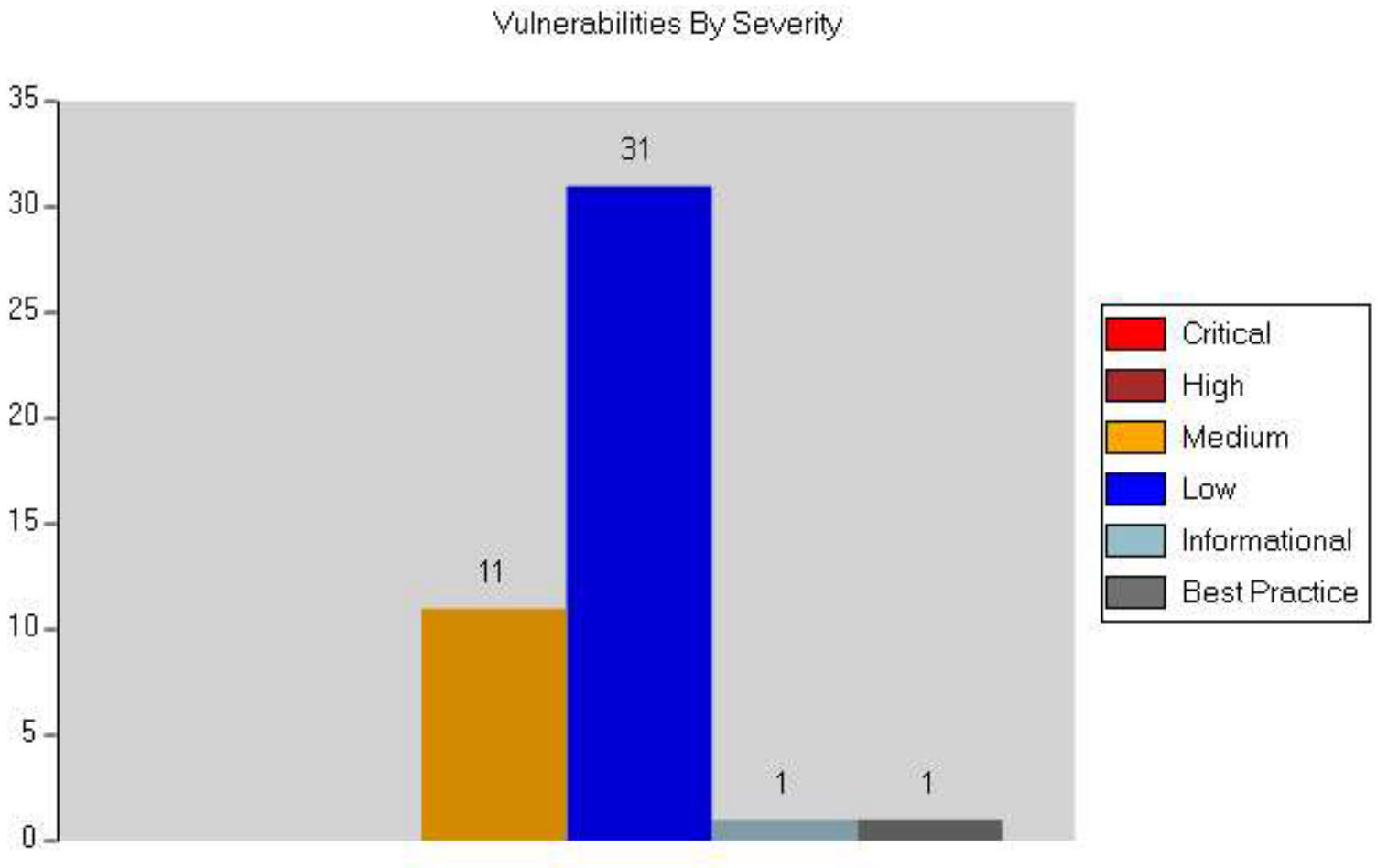
Webinspect EOS Results



Webinspect EOSstaging Results



Webinspect EOSdev Results



Categorization of Vulnerabilities using 2017 OWASP Top 10

Open Web Application Security Project (OWASP)

We categorized the critical and high risk identified under the OWASP top 10 security controls:

1. Injection – ***instances found***
2. Broken Authentication
3. Sensitive Data Exposure
4. XML External Entity
5. Broken Access Control
6. Security Misconfiguration – ***instances found***
7. Cross-Site Scripting – ***instances found***
8. Insecure Deserialization
9. Using Components with Known Vulnerabilities – ***instances found***
10. Insufficient Logging & Monitoring

Source: https://www.owasp.org/index.php/

OWASP 1 - Injection

OWASP defines this risk as: “Injection flaws, such as SQL, OS, and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker’s hostile data can trick the interpreter into executing unintended commands or accessing data without proper authorization.”

- Identified authenticated SQL injection vulnerabilities using Webinspect – 4 instances (1 COS, 3 EOS)

OWASP 6 - Security Misconfiguration

OWASP defines this this risk as: “Security misconfiguration is the most common issue in the data, which is due in part to manual or ad hoc configuration (or not configuring at all), insecure default configurations, open S3 buckets, misconfigured HTTP headers, error messages containing sensitive information, not patching or upgrading systems, frameworks, dependencies, and components in a timely fashion (or at all).”

- Adobe ColdFusion unsupported version 9 – extended support ended 12-31-2016 – 2 instances
- Adobe ColdFusion 11.x < 11u13 / 2016.x < 2016u5 Multiple Vulnerabilities (APSB17-30) – 1 instance
- Microsoft .NET Framework Unsupported – 2 instances
- Microsoft SQL Server unsupported versions – 4 instances
- MS15-124: Cumulative Security Update for Internet Explorer (3116180) and configuration change – 4 instances

OWASP 7 - Cross-Site Scripting (XSS)

OWASP defines this this risk as: “XSS flaws occur whenever an application includes untrusted data in a new web page without proper validation or escaping, or updates an existing web page with user supplied data using a browser API that can create JavaScript. XSS allows attackers to execute scripts in the victim’s browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites.”

- Identified authenticated cross-site scripting vulnerabilities using Webinspect – 2 instances (2 EOS)

OWASP 9 - Using Components with Known Vulnerabilities

OWASP defines this this risk as: “Components, such as libraries, frameworks, and other software modules, run with the same privileges as the application. If a vulnerable component is exploited, such an attack can facilitate serious data loss or server takeover. Applications and APIs using components with known vulnerabilities may undermine application defenses and enable various attacks and impacts.”

- MS KB2269637: Insecure Library Loading Could Allow Remote Code Execution – 3 instances
- Missing recent cumulative windows updates – 4 instances
- HP Version Control Agent (VCA) < 7.3.3 Multiple SSL Vulnerabilities – 1 instance
- Google Chrome < 61.0.3163.79 Multiple Vulnerabilities – 1 instance
- MS KB3074162: Vulnerability in Microsoft Malicious Software Removal Tool Could Allow Elevation of Privilege – 3 instances

OWASP 9 - Using Components with Known Vulnerabilities, continued

- MS15-011: Vulnerability in Group Policy Could Allow Remote Code Execution (3000483) – 4 instances
- Security and Quality Rollup for .NET Framework (Sep 2017) – 1 instance

Server Baseline Configuration

We had previously identified a security risk regarding the lack of defined baseline configurations for Servers. Please refer to Fiscal Year 2017 Federal Information Security Modernization Act (FISMA) Notice of Finding and Recommendation (NFR) No. 5 for further detail on this issue.

Incorrect External Connection Information

CLA identified an external connection that had inaccurate connection information documented. Specifically, the external internet protocol address had changed; however, the recently signed interconnection security agreement between GB Hawk and CNCS did not reflect the address change.

Recommendations

1. We recommend CNCS improve the effectiveness of patching all web servers.
2. We recommend CNCS update unsupported software to supported versions.
   - Ex: ColdFusion, MSSQL, .NET
3. We recommend CNCS address configuration changes required to fully address vulnerability remediation.
   - Ex: Hardened UNC Paths, CWDIllegalInDllSearch and FEATURE_ALLOW_USER32_EXCEPTION_HANDLER_HARDENING

Recommendations, continued

4. We recommend CNCS evaluate secure coding practices and remediate SQL injection and Cross-Site Scripting weaknesses.
5. We recommend CNCS update the Goldbelt Hawk Interconnection Security Agreement to reflect the correct connection information.

- Questions?