

**CISCO**  
**SECURE**

# All Things Vulnerabilities

May 23, 2023



The bridge to possible



# Presenters:



Lidia Attalla

Customer Success  
Engineer



Ren Ferril

Customer Success  
Manager



Jamey McGrath

Customer Success  
Manager

# Agenda



- ▶ Vuln basics – what is a vulnerability?
- ▶ Vuln Score
- ▶ Vuln details page
- ▶ Vuln statuses
- ▶ How vulns get closed in Kenna
- ▶ Using custom fields for false positive/risk acceptance workflow
- ▶ Demo and Q&A

# Vulnerability Basics



- What is a vulnerability?
  - Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source
- How is a vulnerability identified?
  - Vulnerabilities that are discovered and confirmed are given an identification. The Common Vulnerabilities and Exposures list (CVEs) are identified in the following format: "CVE-YYYY-#" where the YYYY is the year the ID was assigned or made public and the # is the order in which the vuln was found.
- How is a vulnerability scored?
  - Vulnerabilities that receive a CVE will typically receive a CVSS (Common Vulnerability Scoring System) score on a scale of 1-10. CVSS is open industry standard for assessing the severity of a vulnerability if it were to be exploited.



# Vulnerability Score

- What is the Kenna Risk Score?
  - A vulnerability score that is applied to each CVE
  - The scores range from 0 – 100
  - Estimates the probability / likelihood of exploitation for that CVE
  - The score could change daily based on many different factors

## Green 0-33

20 / 100

CVSS 2: 7

### CVE-2010-3889

Unspecified vulnerability in Microsoft Windows on 32-bit platforms allows local users to gain privileges via unknown vectors, as exploited in the wild in July 2010 by the Stuxnet worm, and identified by Microsoft researchers and other researchers.

QualysGuard Fix Available

## Amber 34-66

40 / 100

CVSS 2: 10

### CVE-2012-1721

Unspecified vulnerability in the Java Runtime Environment (JRE) component in Oracle Java SE 7 update 4 and earlier, and 6 update 32 and earlier, allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors related to Deployment, a different vulnerability than CVE-2012-1722.

QualysGuard Fix Available

## Red 67-100

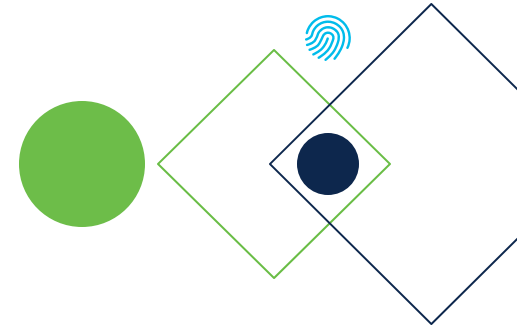
100 / 100

CVSS 2: 10

### CVE-2011-2110

Adobe Flash Player before 10.3.181.26 on Windows, Mac OS X, Linux, and Solaris, and 10.3.185.23 and earlier on Android, allows remote attackers to execute arbitrary code or cause a denial of service (memory corruption) via unspecified vectors, as exploited in the wild in June 2011.

QualysGuard Fix Available Remote Code Execution



# What impacts the Kenna Score?

 **Predicted Exploitable**

1,040

 **Remote Code Execution**

39,741

 **Easily Exploitable**

27,924

Active Internet Breach

- Identifies if the vulnerability is actively being breached in the wild
- We track the volume and velocity of exploitations by day, week and month

Easily Exploitable

- Vulnerabilities that are included in exploit kits or other public source

Pre NVD-Chatter

- Discussion of this vulnerability 3 or more sources 5 or more times

Malware Exploitable

- Identifying if this vulnerability has pieces of malware associated with it

Predicted Exploitable

- This will tell you if Kenna's algorithm predicts future exploits to develop that would leverage this vulnerability

Remote Code Execution

- Identifies if a vulnerability has remote code execution availability

Popular Target

- Vulnerabilities that are trending popularity across many customers

 **Active Net Breaches**

1,915

 **Malware Exploitable**

1,753

 **Popular Targets**

7,919

# Vulnerability Details Page

- It is this page that provides more information about the specific vulnerability, such:
  - Scanners, Scores, Description, Fixes and Exploits

AND

Do tasks such as:

- Status Changes, Editing Custom Fields and Changing Due Dates



# Vulnerability Details Page

CVE-2013-1493

Description [Fix](#) [Known Exploits 48](#) [Known Malware 2460](#) [Nexpose Enterprise... Rapid7 Nexpose Ent...](#)

The color management (CMM) functionality in the 2D component in Oracle Java SE 7 Update 15 and earlier, 6 Update 41 and earlier, and 5.0 Update 40 and earlier allows remote attackers to execute arbitrary code or cause a denial of service (crash) via an image with crafted raster parameters, which triggers (1) an out-of-bounds read or (2) memory corruption in the JVM, as exploited in the wild in February 2013.

Score: **100** / 100 [Edit](#)

CVSS 2: 10.0

**Scanner IDs**

linuxrpm-cesa-2013-0605

**Unique Identifiers**

linuxrpm-CESA-2013-0605

**Asset** 00123E6E04B2

Port80

**Vulnerability Actions**

Close Vulnerability

Accept Risk

False Positive

Wrong Fix

**Created**

3 years ago

**Last seen**

9 years ago

**Status**

open

**Due Date**

open

**Due Date**

11/08/2019

[Edit](#)

**Custom Fields**

**A Test custom field**

hello

**Rem Owner**

hahaha!

**Reason for Risk Acceptance**

asldkfjasd

[Edit](#)

Server not found, check your hostname

**ServiceNow Connector**

[ServiceNow Ticket](#)



# Vulnerability Details Page

## CVE-2013-1488

Description **Fix** Known Exploits **17** Known Malware **272** Nexpose Enterprise... Rapid? Nexpose Ent...

CESA-2013-0770: java-1.6.0-openjdk security update

- Published: 04-17-13 00:00
- Diagnosis: Deprecated

Score: **100** / 100

CVSS 2: 10.0

### Scanner IDs

linuxrpm-cesa-2013-0770

### Unique Identifiers

linuxrpm-CESA-2013-0770

Asset 00123E6ED4B2

Port80

### Vulnerability Actions

Close Vulnerability

Accept Risk

False Positive

Wrong Fix

## CVE-2013-1493

Description **Fix** Known Exploits **48** Known Malware **2460** Nexpose Enterprise... Rapid? Nexpose Ent...

Score: **100** / 100

CVSS 2: 10.0

### Scanner IDs

linuxrpm-cesa-2013-0605

### Unique Identifiers

linuxrpm-CESA-2013-0605

Asset 00123E6E04B2

Port80

### Vulnerability Actions

Close Vulnerability

Accept Risk

False Positive

Wrong Fix

### Created

3 years ago

### Last seen

9 years ago

### Status

open

### Due Date

### Metasploit

Java CMM Remote Code Execution

Reversing Labs ByteCode-JAVA.Exploit.CVE-2013-1493

Reversing Labs Win32 Exploit CVE-2013-1493

Reversing Labs ByteCode-JAVA.Trojan.CVE-2013-2465

Reversing Labs Unknown.Exploit.CVE-2013-1493

Reversing Labs ByteCode-JAVA.Exploit.CVE-2012-1723

Reversing Labs ByteCode-JAVA.Exploit.Agent

Reversing Labs ByteCode-JAVA.Exploit.Obfship

Reversing Labs ByteCode-JAVA.Trojan.CVE-2013-1493

Reversing Labs ByteCode-JAVA.Exploit.CVE-2013-2423

Reversing Labs ByteCode-JAVA.Exploit.CVE-2008-5353

Reversing Labs Script-Macro Exploit CVE-2013-1493

Reversing Labs Android Exploit.CVE-2013-1493

Reversing Labs ByteCode-JAVA.Exploit.CVE-2013-0422

Reversing Labs ByteCode-JAVA.Exploit.Lamar

Reversing Labs ByteCode-JAVA.Trojan.Konstr

Reversing Labs Unknown.Trojan.CVE-2013-1493

Reversing Labs ByteCode-JAVA.Exploit.CVE-2013-2465

Reversing Labs ByteCode-JAVA.Trojan.Agent

DotkaChef DotkaChef: Java CMM

GongDa: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Redkit 2x: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Redkit: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Neutrino: Neutrino: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Private EK: Private EK: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Rawin: Rawin: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

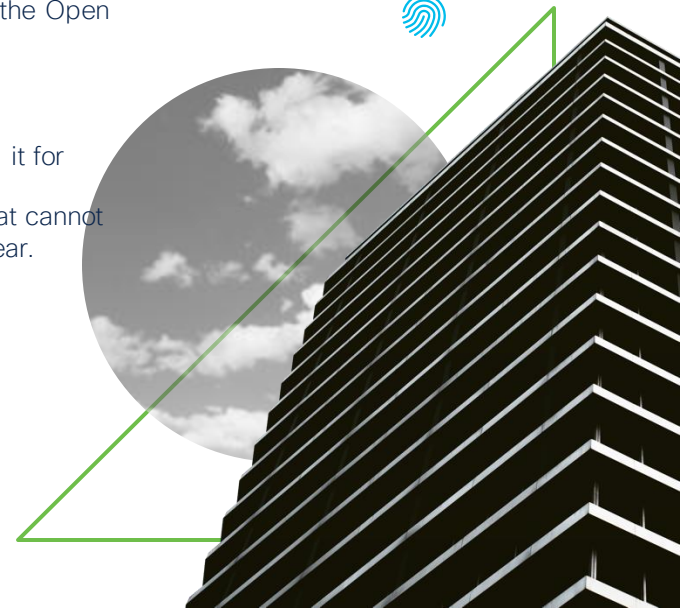
Sibhost: Sibhost: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

CoolStyxy: CoolStyxy: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

Styx 4.0: JAVA The color mgmnt (CMM) functionality in the 2D - an out-of-bounds read or memory corruption in the JVM

# Vulnerability Statuses

- **Open**
  - The vulnerability is still a risk in your organizational data and is available in the Kenna platform for remediation. This is the default status for vulnerabilities.
- **Closed**
  - The vulnerability has been remediated by your team. Once closed, it is removed from the Open vulnerability view.
- **Risk Accepted**
  - The vulnerability truly represents a risk, but the business has decided not to remediate it for some reason. A good example of a Risk Accepted vulnerability is an Internet Explorer vulnerability on a server in a data center that is not accessed or Java vulnerabilities that cannot be remediated because a legacy application will not be replaced until the next fiscal year.
- **False Positive**
  - The vulnerability identified in your scan file is not actually a vulnerability.



# How Vulnerabilities are Closed

Vulnerabilities are closed either: via Connector or Manually

**Connector:** the connector scans and analyzes which assets are reported closed or still vulnerable after a connector run imports that info.

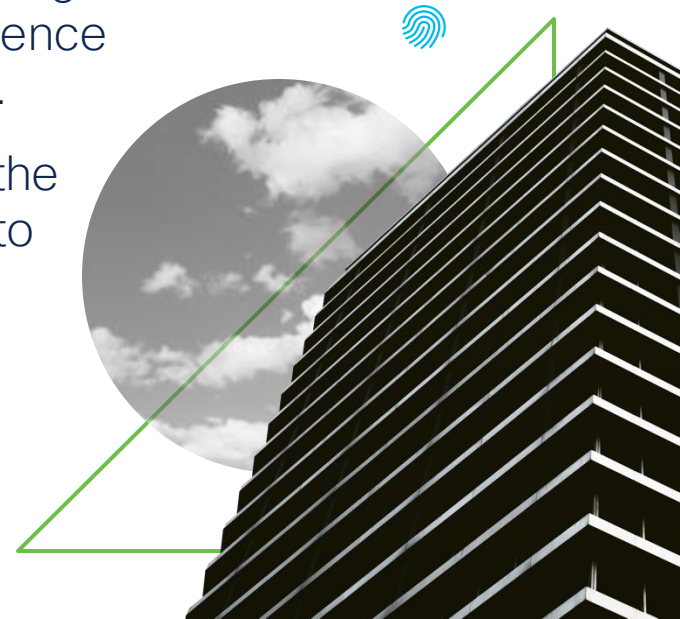
- **Advantage:** Makes it much easier to track the state of your vulns over time. No longer affects asset scores or appear on the Explore page.
- **Note:** if the vuln is seen by more than one scanner, it must be closed by each scanner before it will reflect Closed.



# How Vulnerabilities are Closed

Vulnerabilities are closed either: via Connector or Manually

- **Manually:** Manual closing generates a back-end flag once marked closed. Human action takes precedence and the vuln will not be reopened by the scanner.
- Able to open a Support ticket to reopen/remove the flag or re-set the vulnerability manually to Open, to be picked up on the next run.



# Kenna Custom Field

- What is a Custom field?
- Custom field types
- How to create a custom field
- How to add Data to a custom field
- How to display and manage a custom field in the UI



CISCO  
SECURE

Demo Time!



The bridge to possible



Appreciate your  
time and patience!



Thank You!

