Recurring decimal issue in CVSS V2.0 calculator

The recurring decimal value exists
under a specific condition in CVSS V2.0 calculator.
This value affects the ENVIRONMENTAL SCORE.

January 09, 2010

Masato Terada

1. Recurring decimal issue

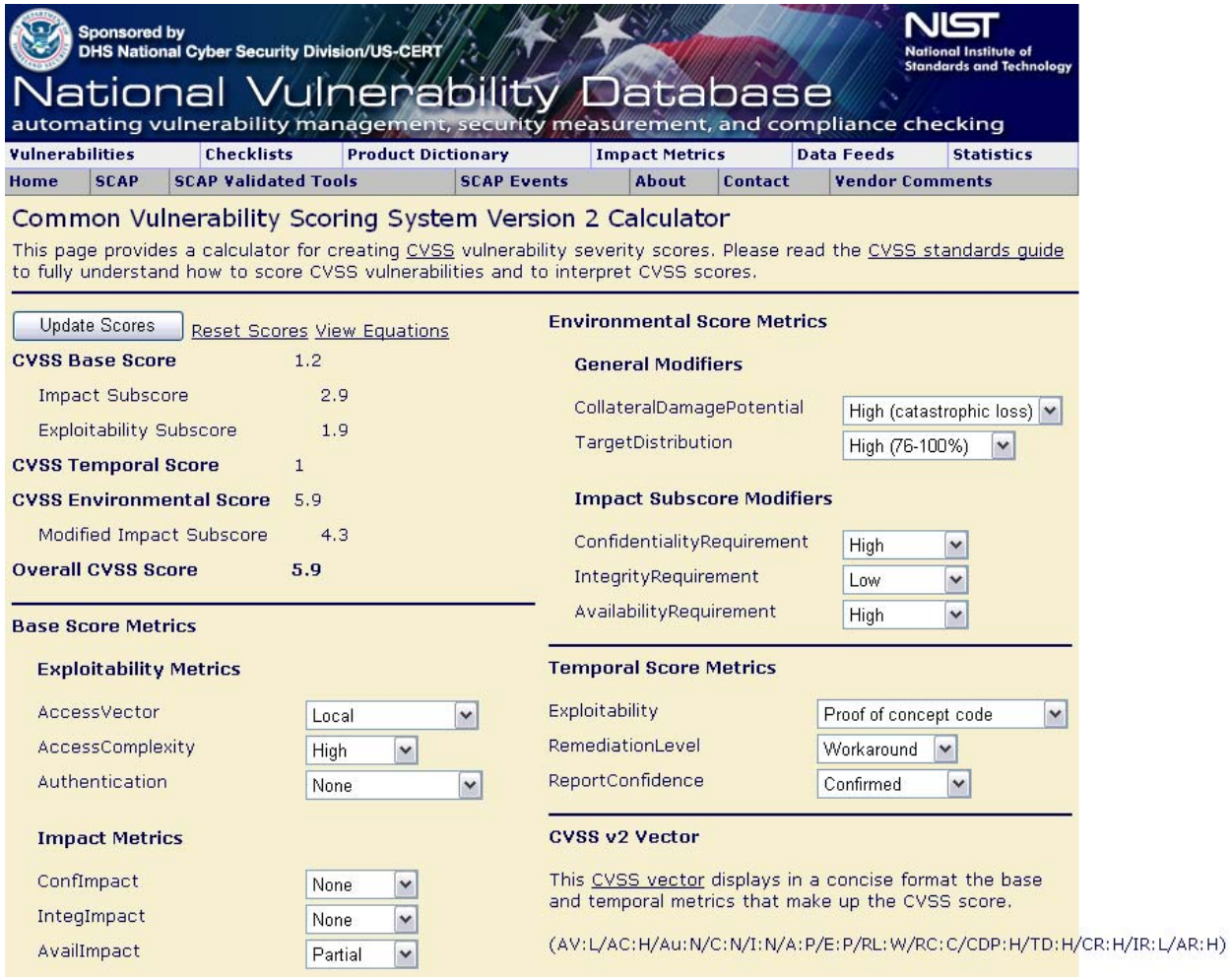
1.1 NVD ENVIRONMENTAL SCORE : 5.9

- vector=(AV:L/AC:H/Au:N/C:N/I:N/A:P/E:P/RL:W/RC:ND/CDP:H/TD:H/CR:H/IR:L/AR:H)

ENVIRONMENTAL SCORE : ~~5.9~~

↓

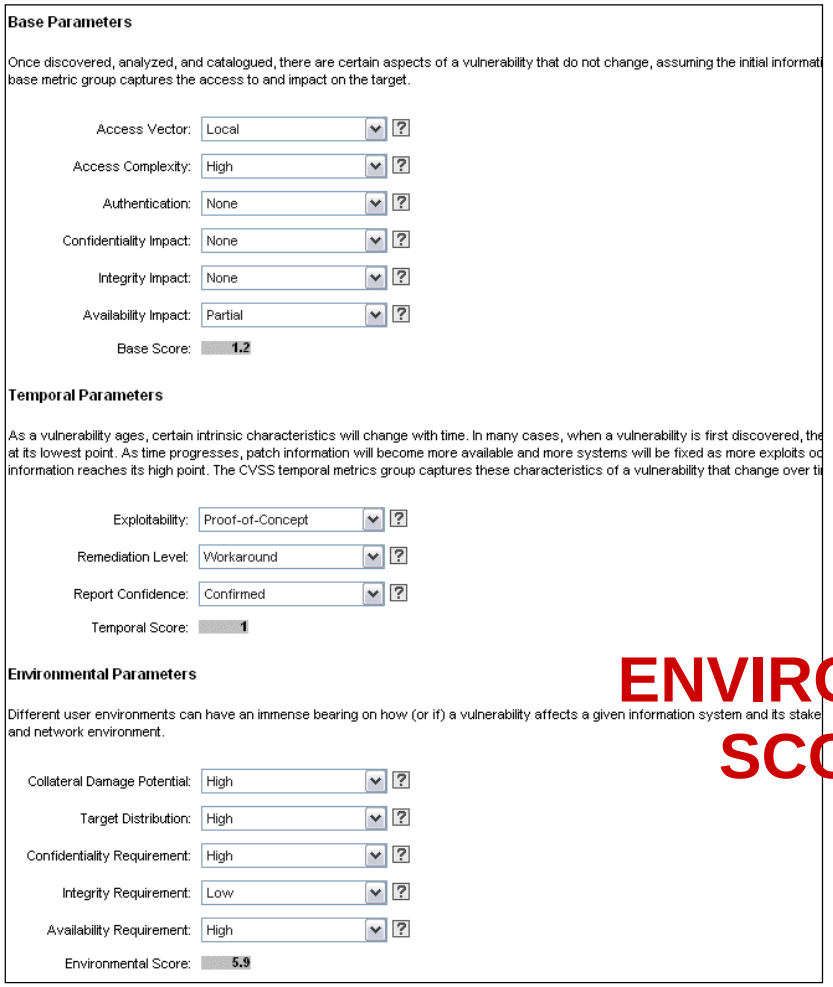
6.0



1. Recurring decimal issue

1.1 Cisco ENVIRONMENTAL SCORE : 5.9

- vector=(AV:L/AC:H/Au:N/C:N/I:N/A:P/E:P/RL:W/RC:ND/CDP:H/TD:H/CR:H/IR:L/AR:H)



ENVIRONMENTAL SCORE : ~~5.9~~

↓

6.0

1. Recurring decimal issue

1.2 CVSS score calculation

- **vector=(**
  AV:L/AC:H/Au:N/C:N/I:N/A:P/
  E:P /RL:W/RC:ND/
  CDP:H/TD:H/CR:H/IR:L/AR:H
)

Base Parameters
Temporal Parameters
Environmental Parameters

Base Parameters		
Access Vector	Local	0.395
Access Complexity	High	0.35
Authentication	None	0.704
Confidentiality Impact	None	0
Integrity Impact	None	0
Availability Impact	Partial	0.275

Temporal Parameters		
Exploitability	Proof of Concept	0.9
Remediation Level	Workaround	0.95
Report Confidence	Not Defined	1

Environmental Parameters		
Collateral Damage Potential	High	0.5
Target Distribution	High	1
Confidentiality Requirement	High	1.51
Integrity Requirement	Low	0.5
Availability Requirement	High	1.51

1. Recurring decimal issue

1.2 CVSS score calculation - Javascript

impactSub = 2.86275 = 10.41 * (1- (1 - confImpact [0]) * (1 - integImpact [0]) * (1 - availImpact [0.275]))
exploitSub = 1.94656 = (20 * accessVector [0.395] * accessComplexity [0.35] * authentication [0.704])
Base Score before round : 1.1716182240000001
= ((0.6 * impactSub [2.86275]) + (0.4 * exploitSub [1.94656]) - 1.5) * f [1.176]

BASE SCORE : 1.2

Temporal Score before round : 1.026
= baseScore [1.2] * exploitability [0.9] * remediationLevel [0.95] * reportConfidence [1]

TEMPORAL SCORE : 1

before Math.min = 4.322752500000001 = 10.41 * (1-(1 - confImpact [0] * confReq [1.51])
* (1 - integImpact [0] * integReq [0.5]) * (1 - availImpact [0.275] * availReq [1.51]))
adjustedImpact = 4.322752500000001 = Math.min(10, 4.322752500000001)

Adjusted Base Score before round = 2.2017959880000006
= ((0.6 * adjustedImpact [4.322752500000001]) + (0.4 * exploitSub [1.94656]) - 1.5) * f [1.176]

Adjusted Base Score = 2.2

Adjusted Temporal Score before round = 1.881
= AdjustedBaseScore [2.2] * exploitability [0.9] * remediationLevel [0.95] * reportConfidence [1]

Adjusted Temporal Score = 1.9

Environment Score before round = 5.949999999999999 = (AdjustedTemporalScore [1.9]
+ ((10 - AdjustedTemporalScore [1.9]) * collateralDamagePotential [0.5])) * targetDistribution [1]

ENVIRONMENTAL SCORE : 5.9

1. Recurring decimal issue

1.2 CVSS score calculation - Javascript vs Manual

- **Javascript**

  = (AdjustedTemporalScore [1.9] + ((10 - AdjustedTemporalScore [1.9])
  *** collateralDamagePotential [0.5])) * targetDistribution [1]**

  = 5.949999999999999 **Recurring decimal**

  = ENVIRONMENTAL SCORE : 5.9

- **Manual**

  = (AdjustedTemporalScore [1.9] + ((10 - AdjustedTemporalScore [1.9])
  *** collateralDamagePotential [0.5])) * targetDistribution [1]**

  = (1.9 + (10 - 1.9) * 0.5) * 1 = 5.95

  = ENVIRONMENTAL SCORE : 6.0