CVSS v3 Development Update

Seth Hanford

Chair, CVSS-SIG

Forum of Incident Response and Security Teams

CVSS v2, Brief History

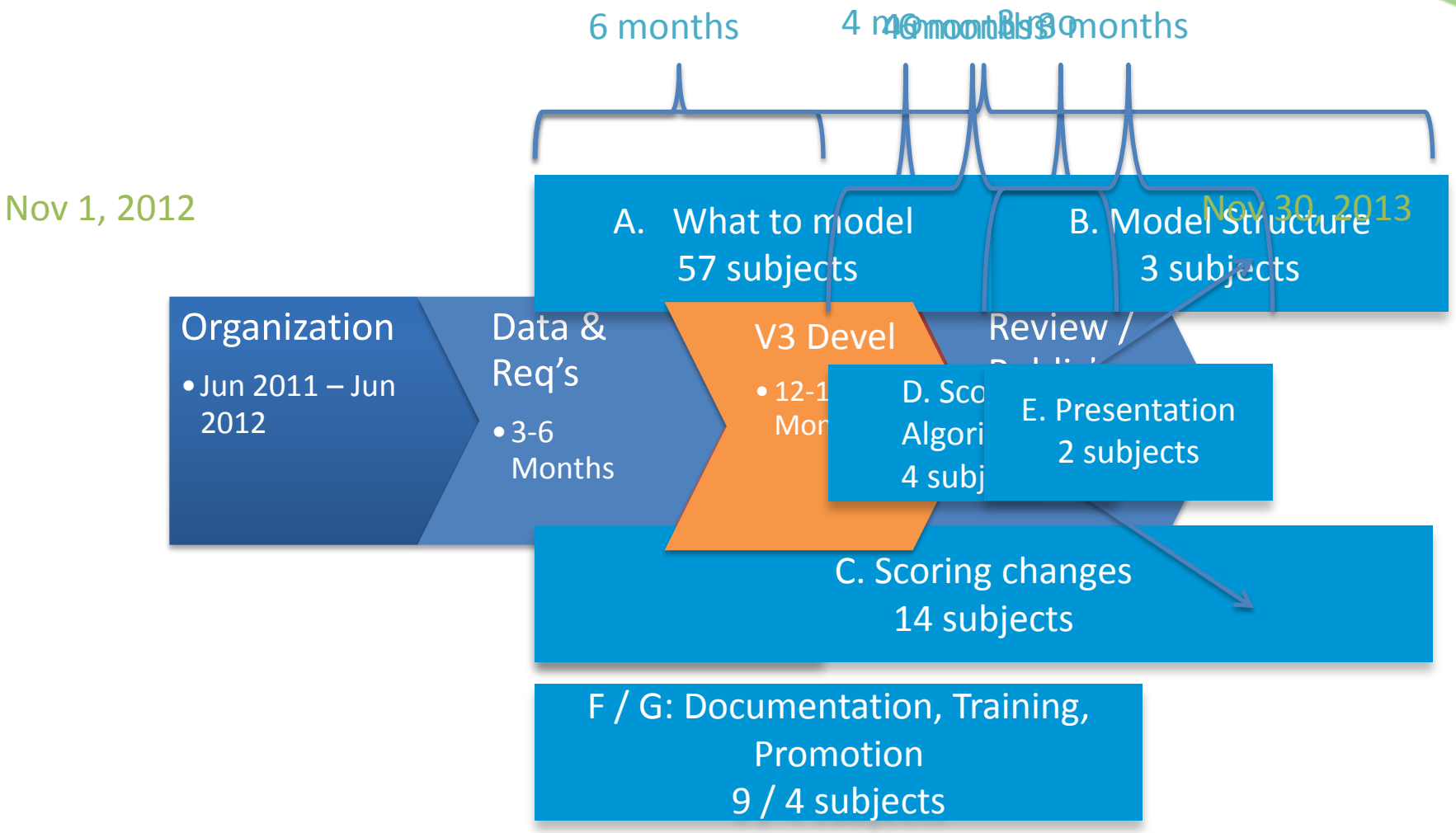
- Developed from April 2005 – June 2007
- Based on industry peer review
- Major improvements to score comparison
  - Any item with 1 Complete impact scoring higher than 3 Partials
- Included “Scoring Tips” to help remove v1 inconsistencies
- Moved “Security Requirements” to Environmental to permit independent Base calculations by 3rd party scoring providers

CVSS v3 Development

- Preliminary work June 2011 – Mar 2012
  - Seth nominated; IPR development & SIG governance work
- Work on v3: March 2012 – present
- Call for Participants (Mar – May, 2012)
  - 17 Voting Representatives from 8 constituencies
  - Banking / Finance; Government; Academic; Manufacturing / Retail; Technology; Telecommunications; CIRTs & Security Research; Energy
- Call for Subjects (Apr – Jun, 2012)
  - 93 subjects from 21 contributors
  - 4-phase development, ending in Jun 2014
- Hybrid model of read-only membership & active participants
  - IPR Agreement required for active participation; ensures CVSS output is unencumbered for all users

FIRST
Improving Security Together

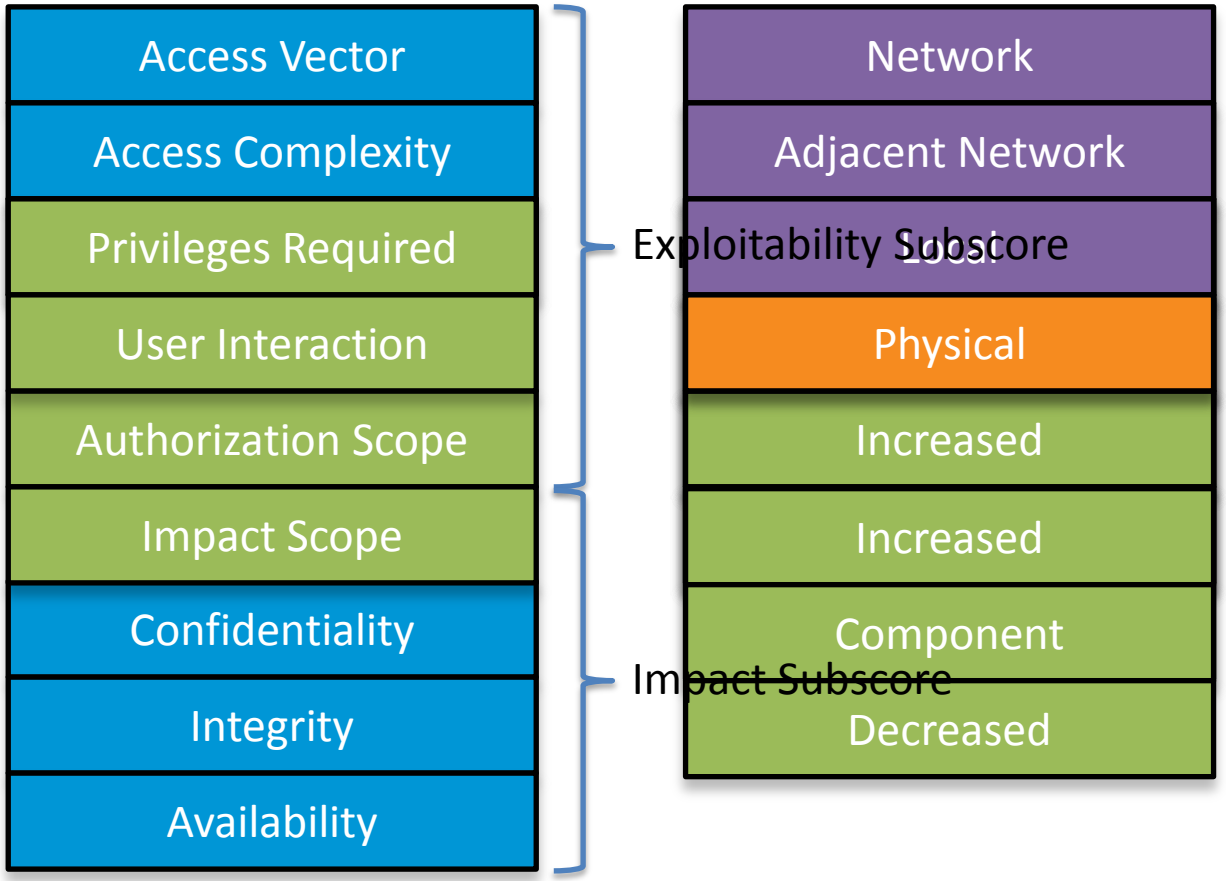
Development Timeline



Key Goals for v3

- Solve the “Scope” problem
  - 10 of 57 “Group A” subjects
  - Oracle Partial+ showing customer demand since v2 release
  - Address additional concerns for modern age: virtualization, sandboxing, etc
- Decrease subjectivity / increase objectivity & repeatability
- Better documentation and examples
- Address changes in technologies, threats, and vulnerabilities
- Increase actionable uses / decrease ineffective measures

Base Metric Group Changes, v2 -> v3



Privileges Required (A-Cisco-2) Approved

- Removes v2 “Authentication” metric
- Measures actual attacker privileges vs overloaded “local” definition
  - AV:L, Au: N == v2 locally authenticated attacker
- Allows for measurements of attacker capability, not just login counting
  - v2 “None” used > 90% of all NVD vulnerabilities (2007 – 2012)
  - v2 “Multiple” used < 1% of all NVD vulnerabilities (2007 – 2012)
- Allows for measuring “Complete” capabilities
  - Useful for corner cases involving a “root” user escalating across authorization boundaries (e.g. root on VM guest gains privilege on peer guest / VM hypervisor)

FIRST
Improving Security Together

Privileges Required (A-Cisco-2) Proposed

Metric Value	Description
None	Unprivileged
Low	Basic, low-impact capabilities; no “Complete” impacts authorized; only non-sensitive impacts
High	Significant capabilities; one or two “Complete” impacts authorized; OR “Partial” impact to sensitive resources
Complete	Fully privileged; three “Complete” impacts authorized

User Interaction (A-Citi-1) Approved

- Removes “Social Engineering” components from v2 Access Complexity definition

Metric Value	Description
None	Vulnerability requires no user interaction
Simple	Successful exploitation requires a user to take standard / expected actions (open email, click a link, view PDF, etc)
Complex	Successful exploitation requires a user to take non-standard / abnormal actions

Authorization Scope (A-Cisco-1) Approved

- First of two metrics used to answer the “Scope” problem
  - Where is the attacker coming from?
- Measure the scope of the attacker’s authorization, relative to the vulnerable component
- Removes host-centric vulnerability scoring
- Design agnostic
  - Application vs. Operating System
  - Virtualization (guest -> hypervisor, guest -> peer guest)
  - Application sandboxes
  - Multiple processor privilege separation (Proc. A Ring 0 -> Proc. B Ring -1)

Authorization Scope (A-Cisco-1) Approved

Metric Value	Description
Increased	Authorization from independent authority, or whose control includes all resources of vulnerable component
Component	DEFAULT; Authorization granted by component itself or same authority used to authorize component capabilities
Decreased	Authorization from source controlled by component, or subordinate to component

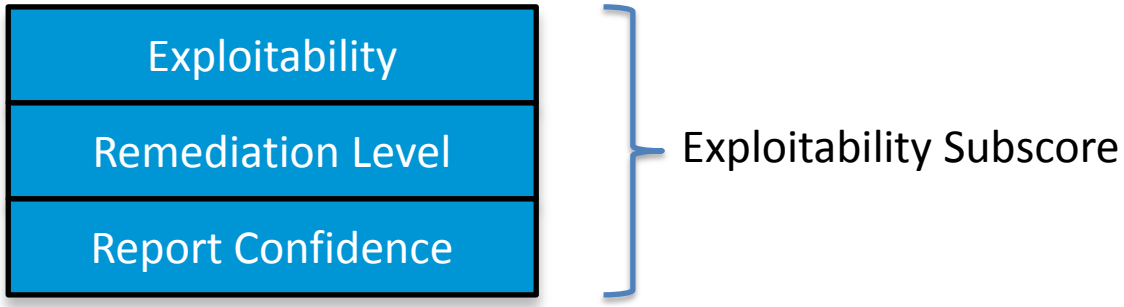
Impact Scope (A-Cisco-1) Approved

- Second of two metrics used to answer the “Scope” problem
  - Where is the attacker effecting an impact?
- Measure the scope of the attacker’s impact, relative to the vulnerable component and its scope of control
- Removes host-centric vulnerability scoring
- Design agnostic
  - Measures impact to the Vulnerable Component
  - Permits measurement of Complete control over an application, host, virtual infrastructure, etc
  - Permits measurement of impact to direct, non-device resources (network, etc)

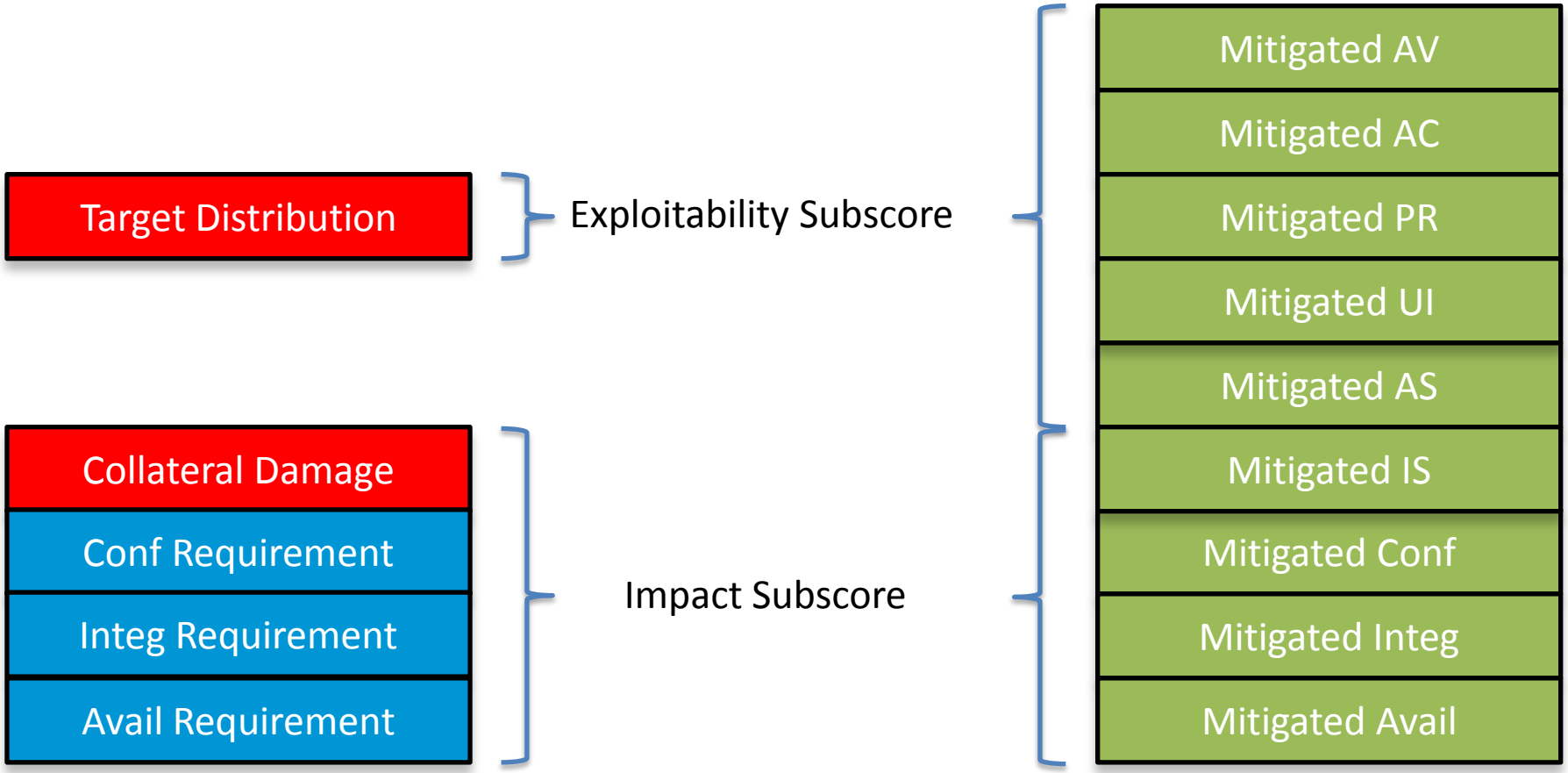
Impact Scope (A-Cisco-1) Approved

Metric Value	Description
Increased	Information resources controlled by an authority that is independent of the vulnerable component are primarily impacted
Component	DEFAULT; Resources controlled by component itself or same authority are primarily impacted
Decreased	Resources controlled by component, or subordinate to component are primarily impacted

Temporal Metric Group, v3



Environmental Metric Group Changes, v2 -> v3



Mitigated Environmental (C-Citi-1) Approved

- All BASE metrics would have an associated Mitigated Environmental metric.
- Functions similar to the Security Requirements
- Recalculates the Base metrics according to environmental mitigations
- Allows for suggested mitigations to be expressed / calculated within CVSS
- E.g. Closing a port but leaving a vulnerability unpatched; reducing effective privileges of a running service; requiring increased privileges to perform an action, etc.

Remove Collateral Damage / Target Distribution (C-Citi-2, -3)

Proposed

- Legacy CVSSv1 metrics
  - Difficult to measure
  - Do not scale well to large organizations
  - By all accounts unused
- Mitigated Environmental has shifted focus of environmental
  - Modify impact and exploitability specific to the end-user environment

Severity Categories (C-Intel-1)
Proposed

- Based on the Unofficial NIST NVD range-based assignments
  - Low: 0 – 3.9
  - Medium: 4.0 – 6.9
  - High: 7.0 – 10.0
- Adds “None” and “Critical”
  - None: 0.0
  - Low: 0.1 – 3.9
  - Medium: 4.0 – 6.9
  - High: 7.0 – 8.9
  - Critical: 9.0 – 10.0

Vulnerability Chains (C-Romanosky-1)
Proposed

- CVSS v3 still focused on scoring vulnerabilities individually
- Optional capability that removes the restriction for combining chained effects
- Requires individual vulnerabilities to have their own CVSS scores first
- Used to express 1..N vulnerabilities in order to achieve the impact of vulnerability N
- Chain has its own CVSS score
  - Exploitability is re-scored from logical combination of exploitability subscores
  - Impact is impact subscore of vulnerability N

Vulnerability Chains (C-Romanosky-1) Proposed

	Vuln 1	Vuln 2	Vuln 3	Chain
Access Vector	N	L	L	N
Access Complexity	L	M	L	M
Privileges Required	N	L	L	N
User Interaction	N	S	N	S
Authorization Scope	C	C	C	C
Impact Scope	C	C	C	C
Confidentiality	N	P	C	C
Integrity	P	P	C	C
Availability	N	P	C	C
Exploitability	F	H	U	U
Remediation Level	OF	OF	W	W
Report Confidence	C	C	UR	UR

FIRST
Improving Security Together

Further work

- Ongoing / concurrent through Nov 30, 2013
  - Document completed work
  - Collect example vulnerabilities and v2 “hard” cases
  - Plan training materials
- May 1, 2013
  - Begin Scoring Algorithm work
- Sep 1, 2013
  - Begin machine readability / presentation layer work
- Nov 30, 2013
  - First draft / public comment; FIRST approval
- June 2014
  - Release CVSS v3

FIRST
Improving Security Together

How can I help?

- Need example vulnerabilities / v2 “hard” cases
- Need examples of how you might use Vulnerability Chaining
- Contact seth@first.org
  - Subscribe to read-only cvss-sig@first.org
  - Subscribe to read / write cvss-v3@first.org (requires signed Intellectual Property Rights agreement from you / your organization)
  - Submit comments / questions
- Read and comment on the forthcoming draft
- Express interest in joining the upcoming v4 SIG

Thank you!