Corsec®

Circular Reasoning: Venn Will We Agree on a Common SoF Analysis Method?

Presented by:

Nathan Lee and Amy Nicewick

Corsec Security, Inc.

The Question and the Problem

- How should labs perform SoF analysis under CC v3.1? Specifically…
  - How should the "possible password space" be calculated?
  - What should be the methodology for overall SoF analysis?

- Historically, SoF analysis under CC v2.x was **inconsistent** between schemes, labs within schemes, and even **evaluators within labs**!
  - This resulted in **"re-invention"** of SoF analysis by vendors **for each evaluation**
  - Vendors were very **frustrated** that there was no **consistency**

Corsec's Experience

- Corsec works with many different schemes, labs, evaluators, and vendors
- Corsec engineers had to perform SoF analyses differently for different schemes, different labs, and even different evaluators at the same lab
- Eventually, Corsec found a solution that satisfied every scheme, lab, and evaluator to which it was submitted
  - Just in time for CC 3.x! ☺

Inconsistent Requirements

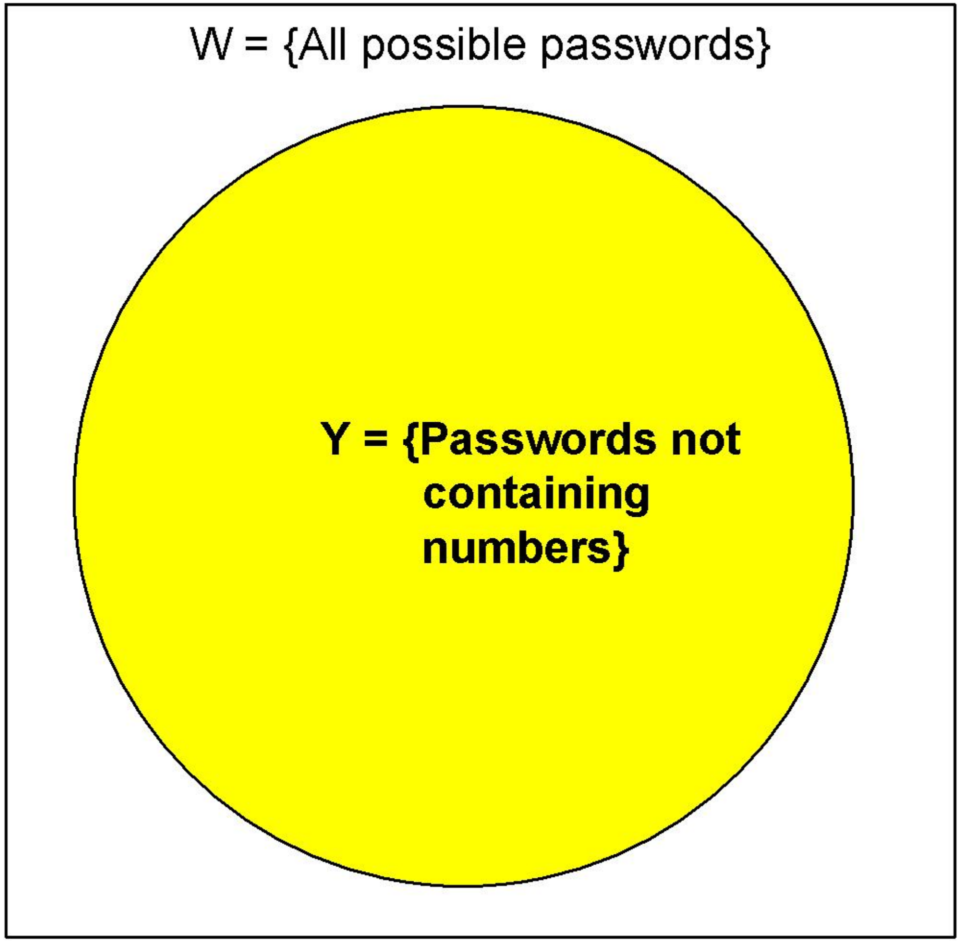
- Different evaluations had different SoF Analysis requirements imposed upon them:
  - Detailed math
  - General narrative text only
  - Analysis of likelihood of a string being chosen
    - Mathematical proof of likelihood
    - Verbal assertion
- The **same rationale** for the **same product** resulted in **different verdicts** from **different evaluators**
- The biggest/most common inconsistency: **Password Space Calculation**

The Solution:
Set Theory using Venn Diagrams

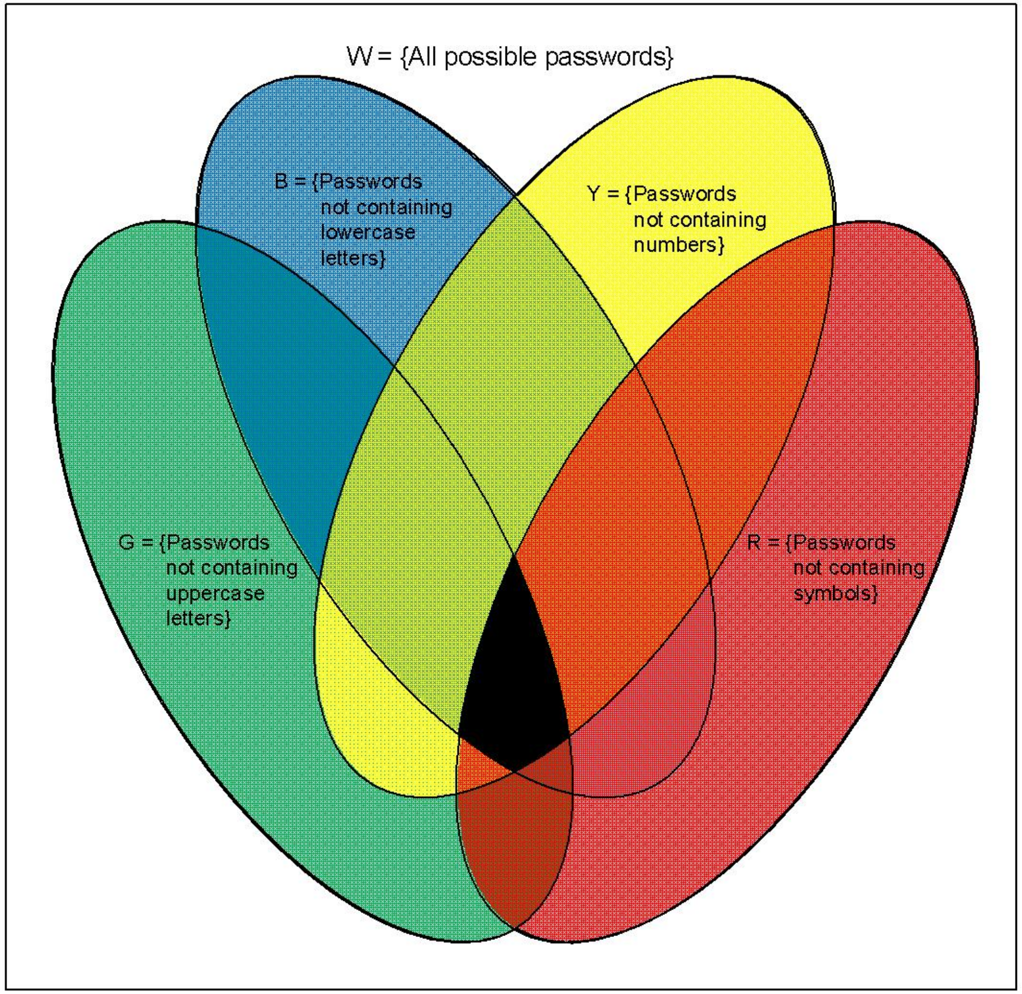
Definitions:

- **Set** – A Set is a collection of items. The items contained within a Set are called “elements” and do not repeat.
- **Intersection** – An intersection is the Set that contains all elements of Set A that also belong to Set B (or equivalently, all elements of Set B that also belong to Set A), but no other elements.
- **Venn diagram** – A Venn diagram is a drawing in which overlapping areas represent groups of items sharing common properties. A Venn diagram consists of one or more shapes, each representing a specific Set. A Venn diagram shows all of the possible mathematical or logical relationships between each Set.

Venn Diagram – "Password Must Contain a Number"


W = {All possible passwords}
Y = {Passwords not containing numbers}

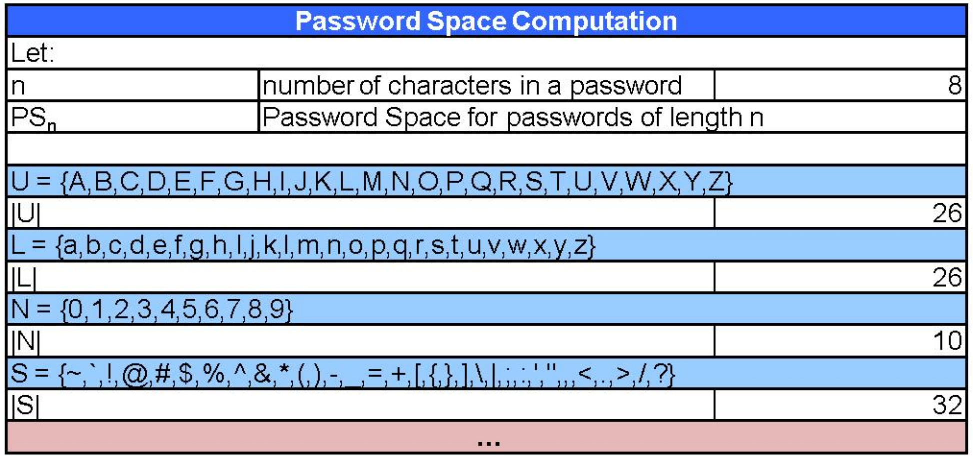
Venn Diagram – "Password Must Contain a Number, an Upper, a Lower, and a Symbol"


W = {All possible passwords}
B = {Passwords not containing lowercase letters}
Y = {Passwords not containing numbers}
G = {Passwords not containing uppercase letters}
R = {Passwords not containing symbols}

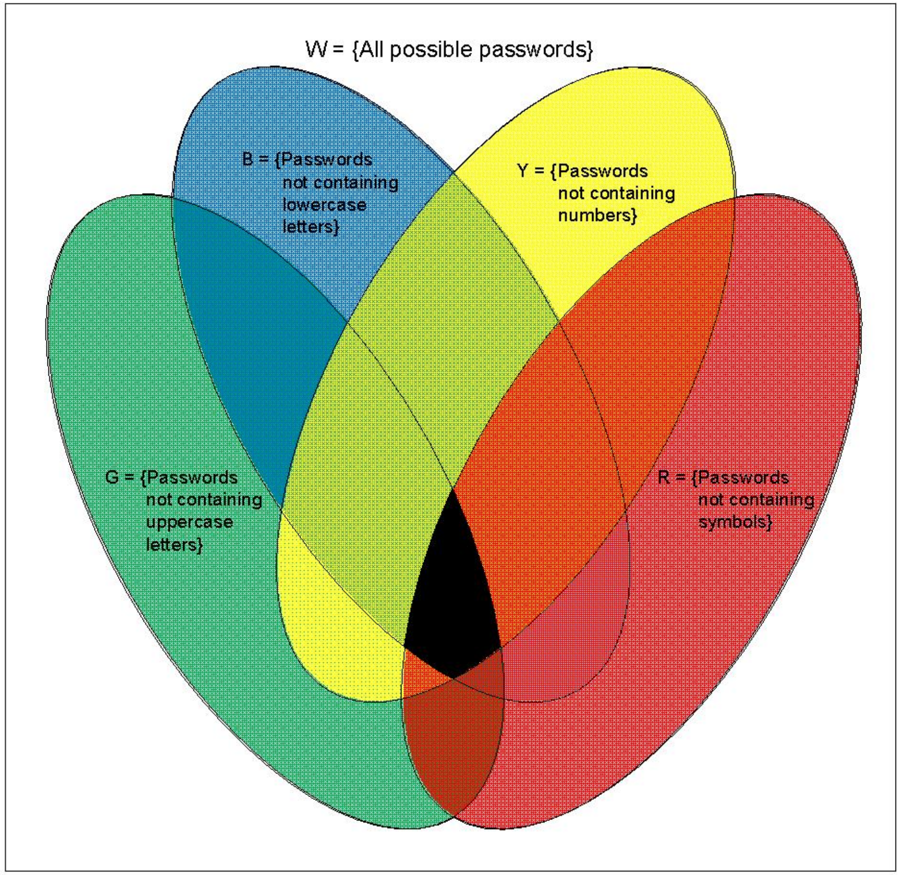
Mathematical Symbols

Example of Mathematical Symbol	Description/References
X = {1, 13, 58, 72, 96}	**A Set (called "Set X").**
X={} **X = Ø**	**The Empty Set; the NULL Set.**
|X|	**The size of Set X.** Example: X = {1, 13, 58, 72, 96}. |X| is the size of Set X. Thus, |X| = 5. Example: X = {} Thus, |X| = Ø; the Empty Set.
∩	**The intersection of two or more Sets.** Example: A = {1, 3, 5, 7, 9} and B = {2, 3, 4, 5, 6,}. A ∩ B consists of the elements in both Set A and Set B. Thus, A ∩ B = {3, 5}.

Password Space Calculation

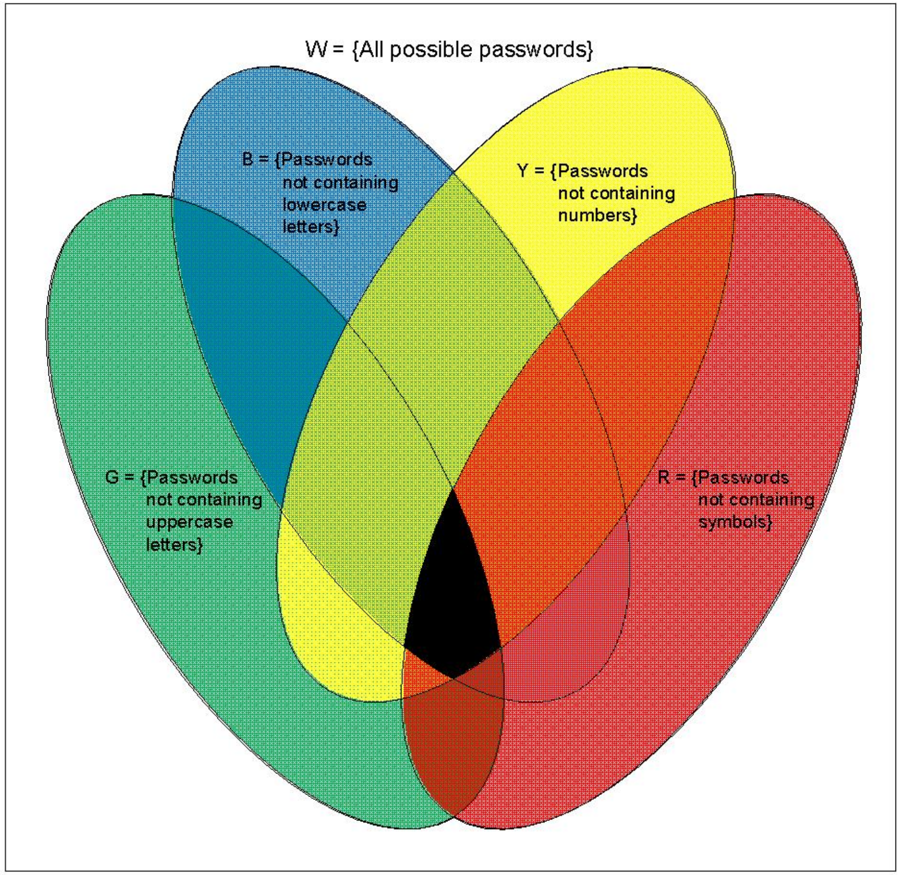


Password Space Computation		
Let:		
n	number of characters in a password	8
PS_n	Password Space for passwords of length n	
U = {A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P,Q,R,S,T,U,V,W,X,Y,Z}		
|U|		26
L = {a,b,c,d,e,f,g,h,I,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z}		
|L|		26
N = {0,1,2,3,4,5,6,7,8,9}		
|N|		10
S = {~,`,!,@,#,$,%,^,&,*,(,),-,_,=,+,[,{,},],\,|,;,:,',",,,<,.,>,/,?}		
|S|		32
...		


W = {All possible passwords}
B = {Passwords not containing lowercase letters}
Y = {Passwords not containing numbers}
G = {Passwords not containing uppercase letters}
R = {Passwords not containing symbols}

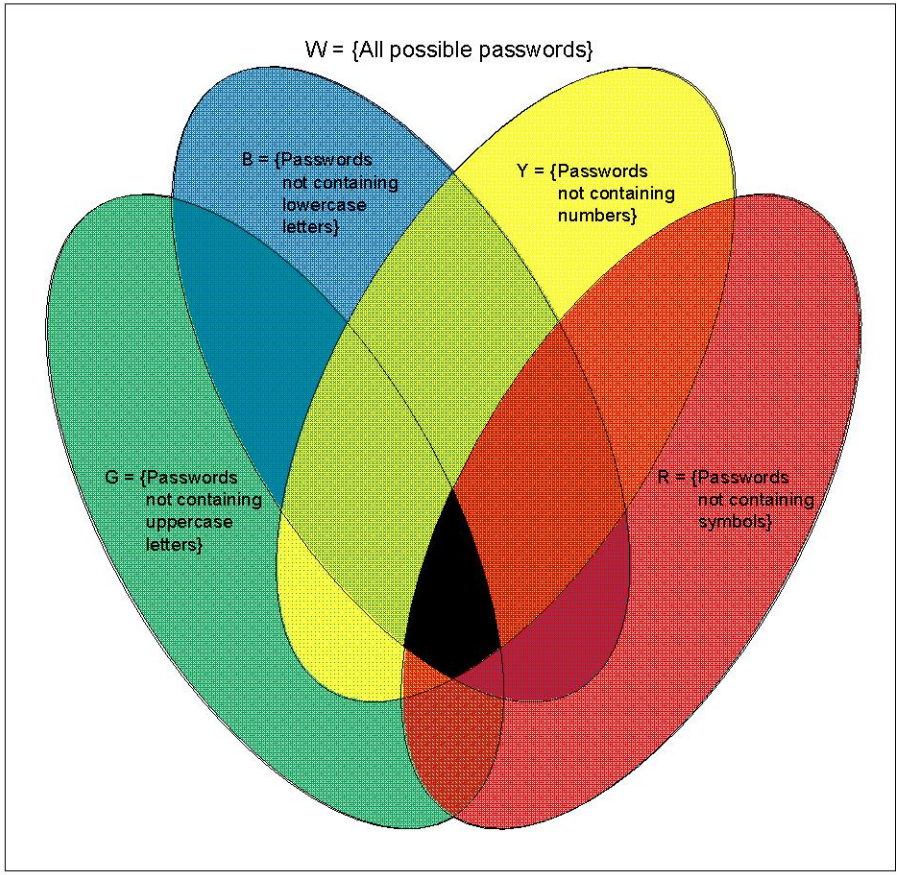
Password Space Calculation

W = {All password possibilities}	
|W|	$\|W\| = (\|U\| + \|L\| + \|N\| + \|S\|)^n$ $\|W\| = (26+26+10+32)^8 = 94^8$ 6,095,689,385,410,820
G = {passwords not containing an uppercase letter} = {passwords containing lowercase letters, numbers, and/or symbols}	
|G|	$\|G\| = (\|L\| + \|N\| + \|S\|)^n$ $\|G\| = (26+10+32)^8 = 68^8$ 457,163,239,653,376
B = {passwords not containing a lowercase letter} = {passwords containing uppercase letters, numbers, and/or symbols}	
|B|	$\|B\| = (\|U\| + \|N\| + \|S\|)^n$ $\|B\| = (26+10+32)^8 = 68^8$ 457,163,239,653,376
Y = {passwords not containing a number} = {passwords containing uppercase letters, lowercase letters, and/or symbols}	
|Y|	$\|Y\| = (\|U\| + \|L\| + \|S\|)^n$ $\|Y\| = (26+26+32)^8 = 84^8$ 2,478,758,911,082,500
R = {passwords not containing a symbol} = {passwords containing uppercase letters, lowercase letters, and/or numbers}	
|R|	$\|R\| = (\|U\| + \|L\| + \|N\|)^n$ $\|R\| = (26+26+10)^8 = 62^8$ 218,340,105,584,896
…	


W = {All possible passwords}
B = {Passwords not containing lowercase letters}
Y = {Passwords not containing numbers}
G = {Passwords not containing uppercase letters}
R = {Passwords not containing symbols}

Password Space Calculation

G ∩ B = {passwords with no uppercase letters and no lowercase letters} = {passwords with numbers and/or symbols}	
|G ∩ B|	$|G \cap B| = (|N| + |S|)^n$
	$|G \cap B| = (10+32)^8 = 42^8$
	9,682,651,996,416
G ∩ Y = {passwords with no uppercase letters and no numbers} = {passwords with lowercase letters and/or symbols}	
|G ∩ Y|	$|G \cap Y| = (|L| + |S|)^n$
	$|G \cap Y| = (26+32)^8 = 58^8$
	128,063,081,718,016
...	
G ∩ B ∩ Y = {passwords with no uppercase letters, no lowercase letters, and no numbers = {passwords containing only symbols}	
|G ∩ B ∩ Y|	$|G \cap B \cap Y| = |S|^n$
	$|G \cap B \cap Y| = 32^8$
	1,099,511,627,776
...	
The equation below gives the final computation of PS_n.	
$PS_8 = |W|-|G|-|B|-|Y|-|R|+|G \cap B|+|G \cap Y|+|G \cap R|+|B \cap Y|+|B \cap R|+|Y \cap R|-|G \cap B \cap Y|-|G \cap B \cap R|-|G \cap Y \cap R|-|B \cap Y \cap R|$	
$94^8 - 68^8 - 68^8 - 84^8 - 62^8 + 42^8 + 58^8 + 36^8 + 58^8 + 36^8 + 52^8 - 32^8 - 10^8 - 26^8 - 26^8$	
2,807,657,387,458,560	


W = {All possible passwords}
B = {Passwords not containing lowercase letters}
Y = {Passwords not containing numbers}
G = {Passwords not containing uppercase letters}
R = {Passwords not containing symbols}

“Time to Crack” Calculation

The tried-and-still-true formula:

- *((PS * ½) + 1) / (number of attempts per time unit)*
  - (Password Space * ½) because an attacker is statistically likely to find the password within the first (50% + 1) of the password space
  - Divide by the number of attempts per time unit

Conclusion

- The biggest inconsistency across all schemes, labs, and evaluators was the password space calculation
- This solution satisfied everyone who evaluated it
- Calculation of the likelihood that particular strings will be chosen as passwords is still an outstanding issue

Our Gift to You

- Since SoF analysis is now primarily a laboratory activity, we are making our diagrams and calculation tables available for laboratory use:
  - http://www.corsec.com/9ICCC.html

Contact Information

- Nathan Lee, Lead Security Engineer
  - nlee@corsec.com
  - Phone: +1 (703) 267-6050 x112
- Amy Nicewick, Lead Security Engineer
  - anicewick@corsec.com
  - Phone: +1 (703) 267-6050 x114

- http://www.corsec.com/