Zero Days, Thousands of Nights

The life and times of zero-day vulnerabilities and their exploits

Lillian Ablon

lablon@rand.org

@lilyablon

RAND CORPORATION

black hat EUROPE 2017

Are zero-day vulnerabilities a zero-sum game?

- Zero-day vulnerabilities can be very useful to those testing defenses or planning offensive operations

- They can also lead to unsecure platforms and increase risk

Retain or disclose ?

Retain or disclose ?

Should a government keep zero-days secret?

Should a government disclose zero-days?

The decision calculus is complicated:

UNCLASSIFIED

Vulnerabilities Equities Policy and Process
for the United States Government
November 15, 2017

1. Purpose

This document describes the Vulnerabilities Equities Policy and Process for departments and agencies of the United States Government (USG) to balance equities and make determinations regarding disclosure or restriction when the USG obtains knowledge of newly discovered and not publicly known vulnerabilities in information systems and technologies. The prim
the public's interest in cybersecurity and to protect core Internet
critical infrastructure systems, and the U.S. economy through the
discovered by the USG, absent a demonstrable, overriding interes
lawful intelligence, law enforcement, or national security purpose

The Vulnerabilities Equities Process (VEP) balances whether to di
the vendor/supplier in the expectation that it will be patched, or
of the vulnerability to the USG, and potentially other partners, so
security and law enforcement purposes, such as intelligence colle
counterintelligence. The U.S. Government's determination as to
vulnerability is only one element of the vulnerability equities eval
binary determination. Other options that can be considered inclu
information to certain entities without disclosing the particular v
vulnerability by the USG in some way, informing U.S. and allied g
at a classified level, and using indirect means to inform the vendo
determinations must be informed by the understanding of risks o
benefits of government use of the vulnerabilities, and the risks ar

11/26/2017 Improving and Making the Vulnerability Equities Process Transparent is the Right Thing to Do | whitehouse.gov

the WHITE HOUSE

Improving and Making the Vulnerability Equities Process Transparent is the Right Thing to Do

NOVEMBER 15, 2017 AT 9:11 AM ET BY ROB JOYCE

There can be no doubt that America faces significant risk to our national security and public safety from cyber threats. During the past 25 years, we have moved much of what we value to a digital format and stored it in Internet-connected devices that are vulnerable to exploitation. This risk is increasing as our dependence on technology and the data we store continues to grow such that technology now connects nearly every facet of our society and the critical services that sustain our way of life. This fact

The decision calculus is complicated: there are many equities to consider

- Defense
- Intelligence, law enforcement, and operational
- Commercial
- International partnership

The decision calculus is complicated: there are many variables in play

- The product that the vulnerability is in
- The threat actor that might take advantage of the vulnerability
- The use of the vulnerability in operations
- The vulnerability itself
- Other information

These variables are a few of those that are examined as part of the U.S. Vulnerabilities Equities Process

The decision calculus is complicated: there are many variables in play

- The product that the vulnerability is in
- The threat actor that might take advantage of the vulnerability
- The use of the vulnerability in operations
- **The vulnerability itself**
- Other information

These variables are a few of those that are examined as part of the U.S. Vulnerabilities Equities Process

We focus on characteristics of the vulnerabilities

- Challenge: publicly available information about zero-days is sparse

- Goal: create some baseline metrics on the characteristics of zero-day vulnerabilities, using actual zero-day data, in order to help inform policy and technical discussions

We focus on characteristics of the vulnerabilities

Life Status

Who knows about the vulnerability?

Longevity

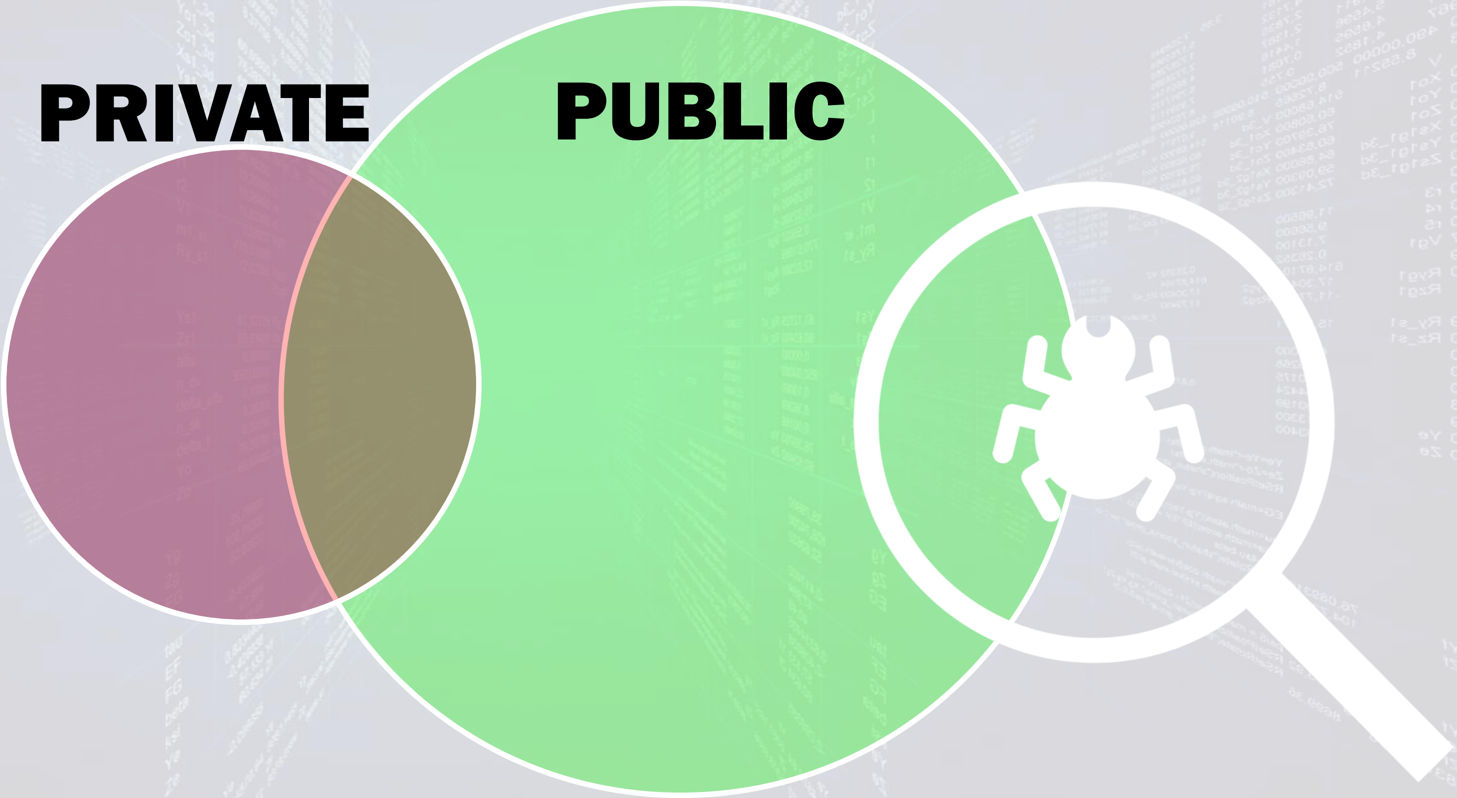
How long will the vulnerability remain publicly unknown?

Collision Rate

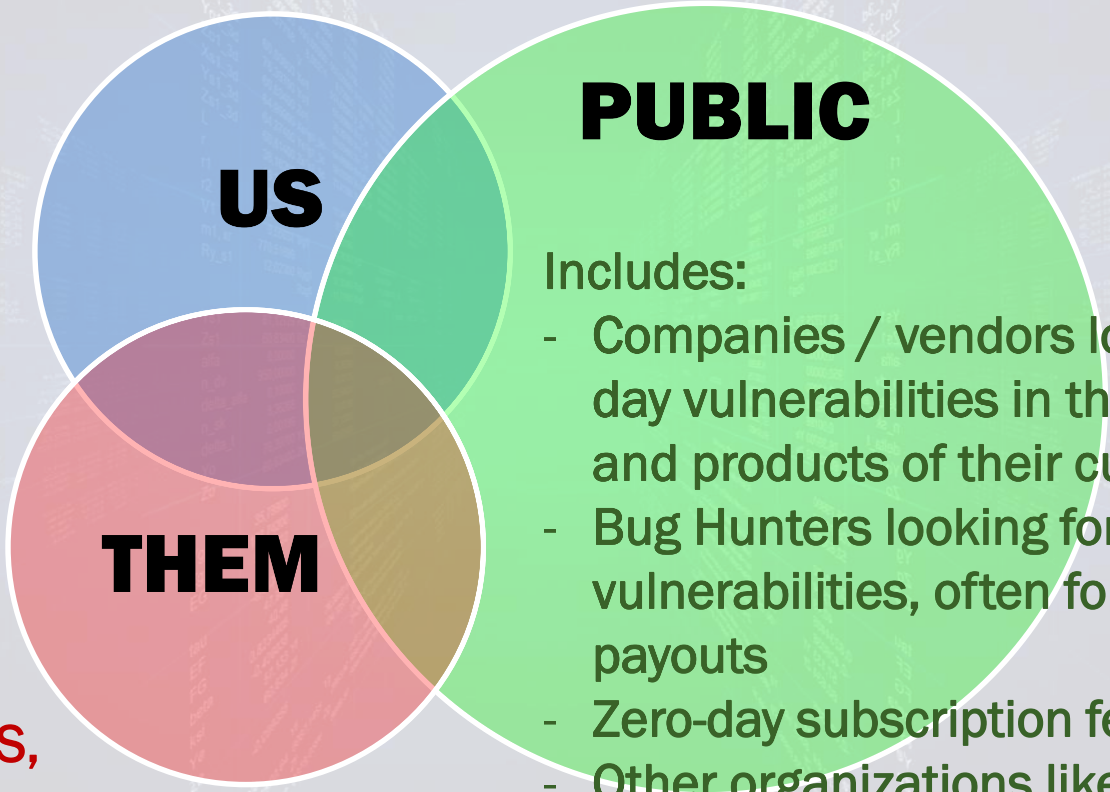
How many vulns get independently rediscovered and publicly disclosed?

- Research Focus
- Quick Dive into the Data
- Analysis & Findings
- Implications & Recommendations

Various groups search for vulnerabilities



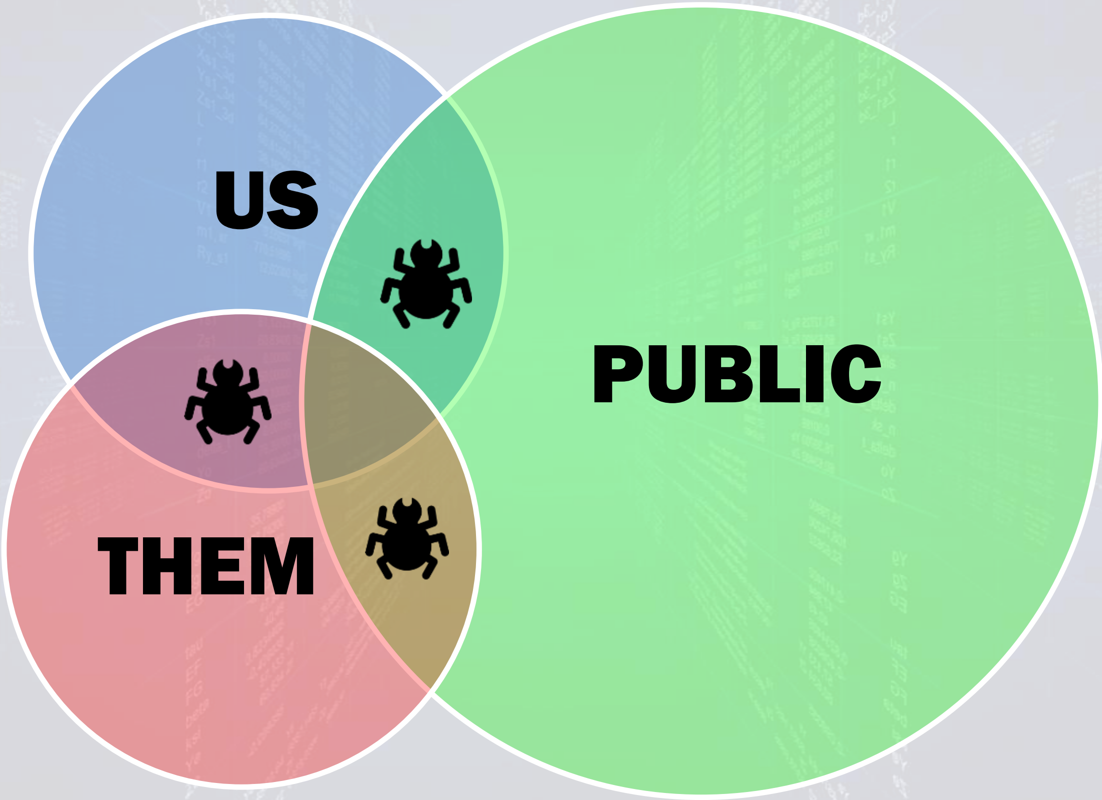
Private groups consist of ‘good’ and ‘bad’ actors



Includes:
- Companies / vendors looking for zero-day vulnerabilities in their own products and products of their customers
- Bug Hunters looking for zero-day vulnerabilities, often for bug bounty payouts
- Zero-day subscription feed businesses
- Other organizations like Project Zero

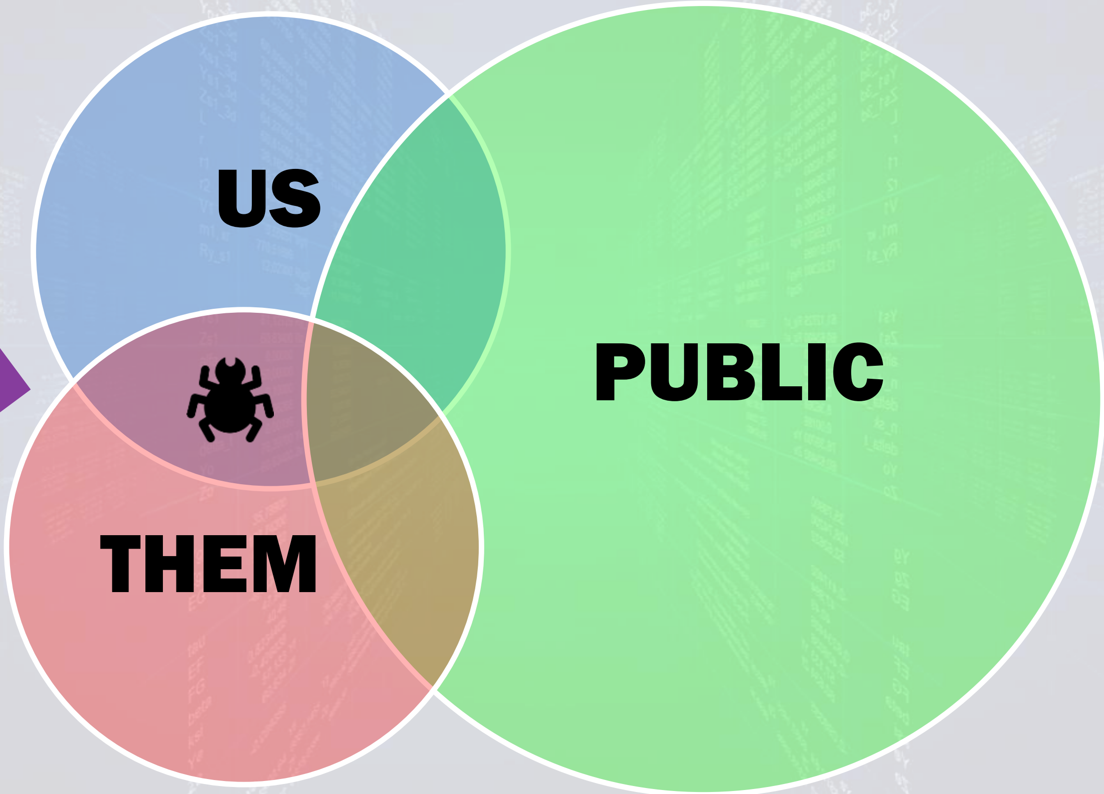
Adversaries of US, Malicious Actors

Sometimes different groups find the same vuln.



Disclosure affects each camp differently

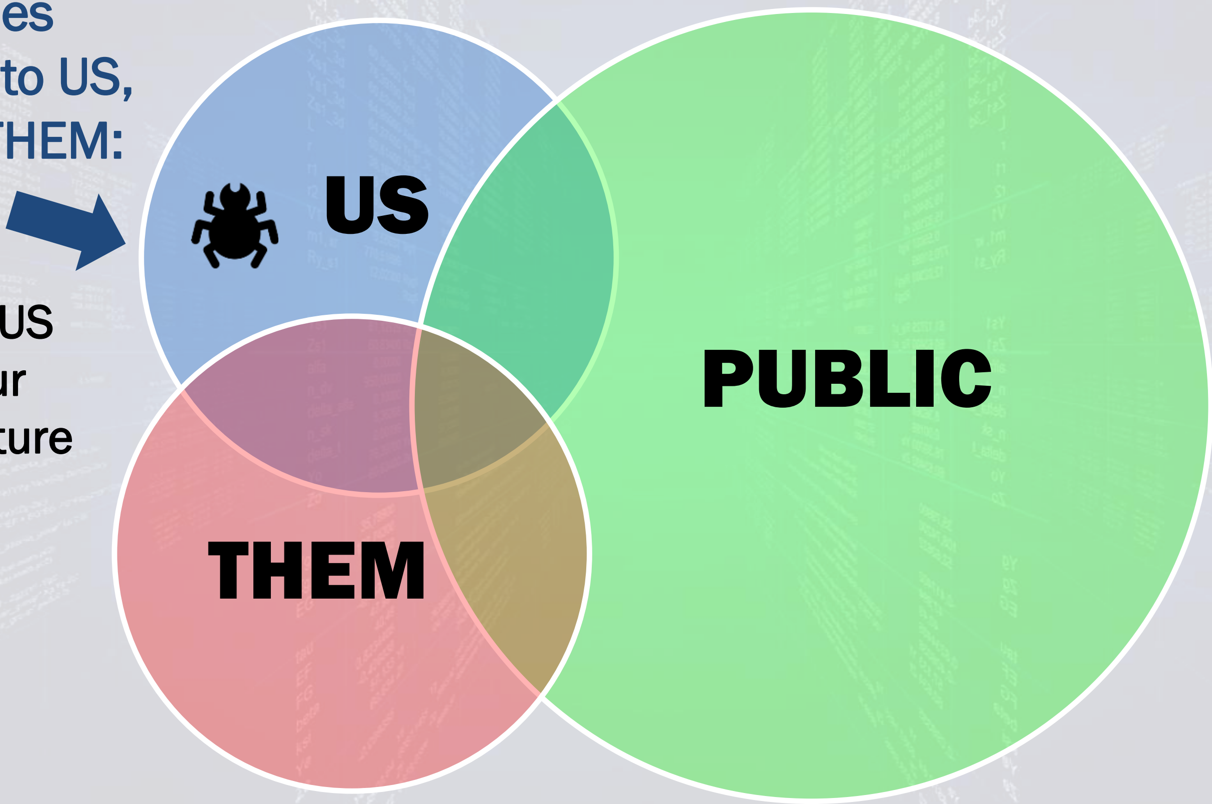
Vulnerabilities known to *both* US and THEM



disclosure by US may strengthen our defensive posture

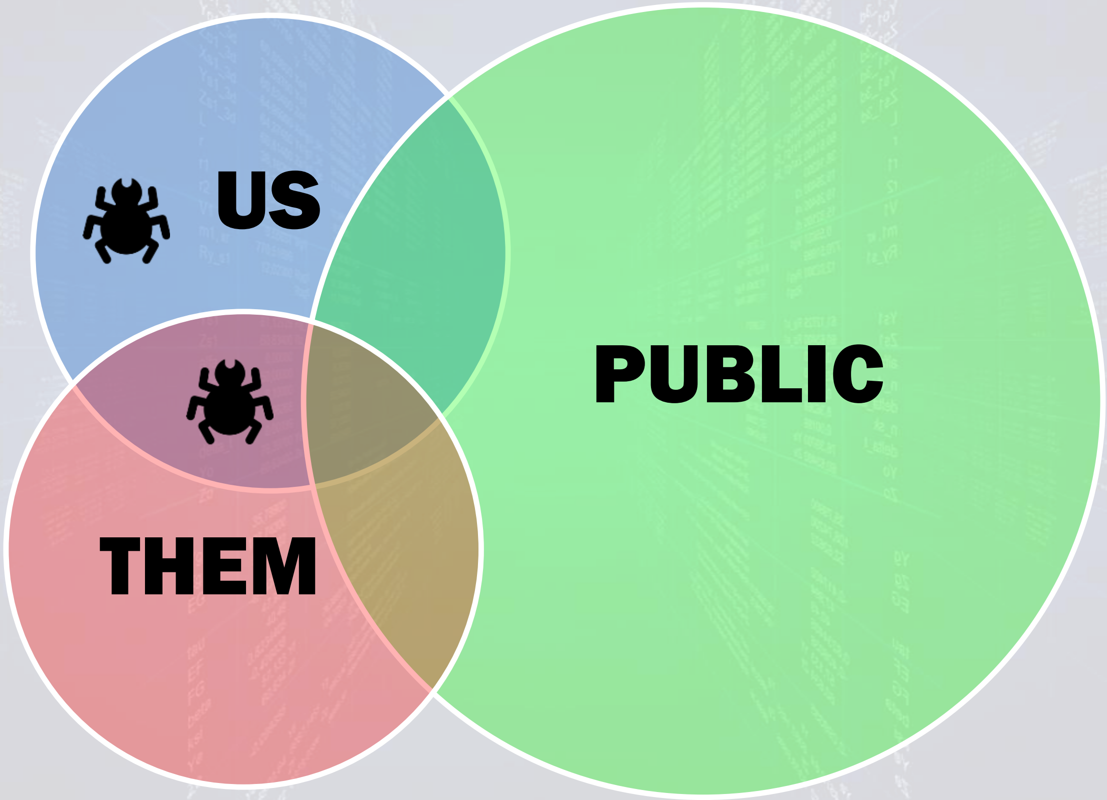
Disclosure affects each camp differently

Vulnerabilities known *only* to US, and not to THEM:

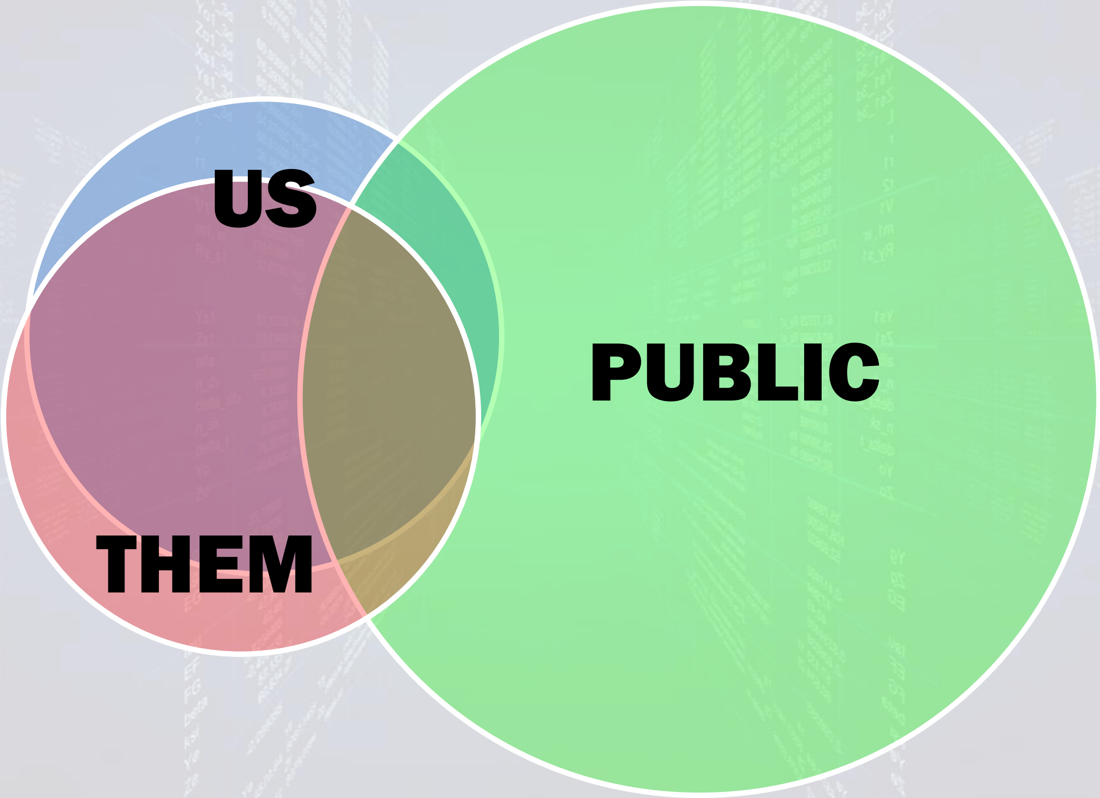


disclosure by US may hinder our offensive posture

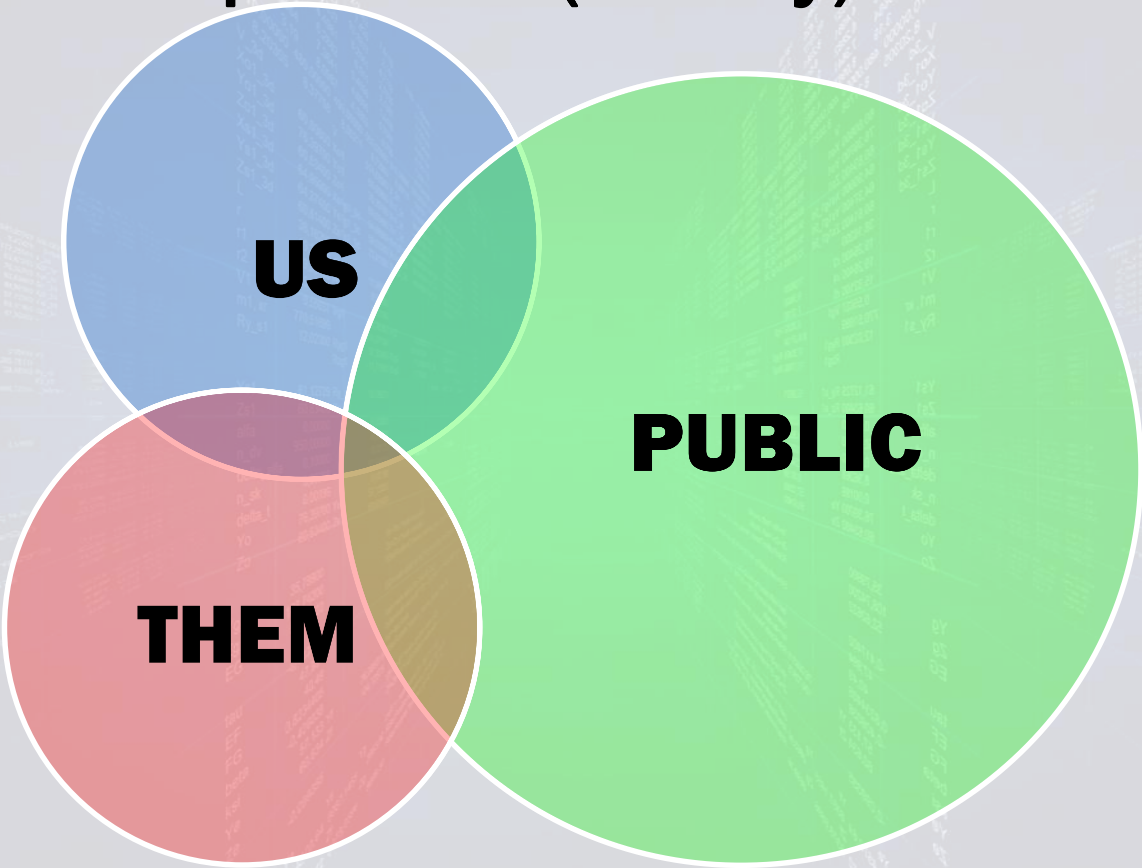
Disclosure affects each camp differently



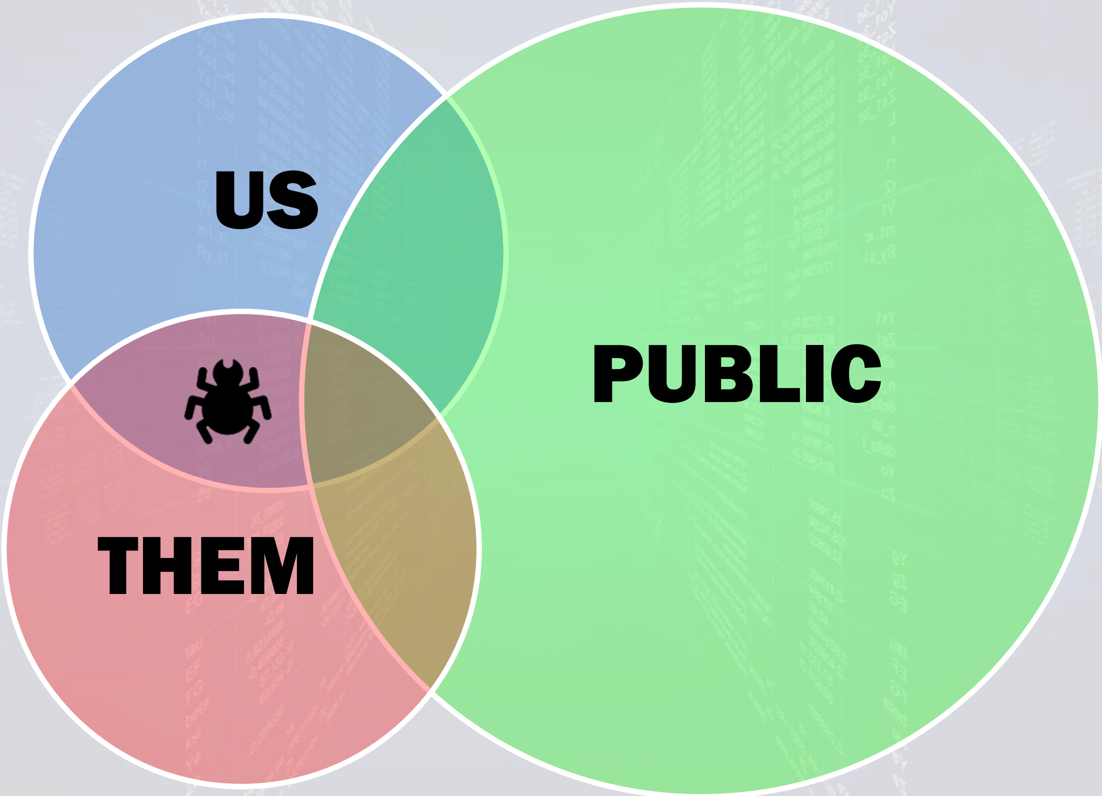
Large overlap: We're vulnerable! Disclose all!



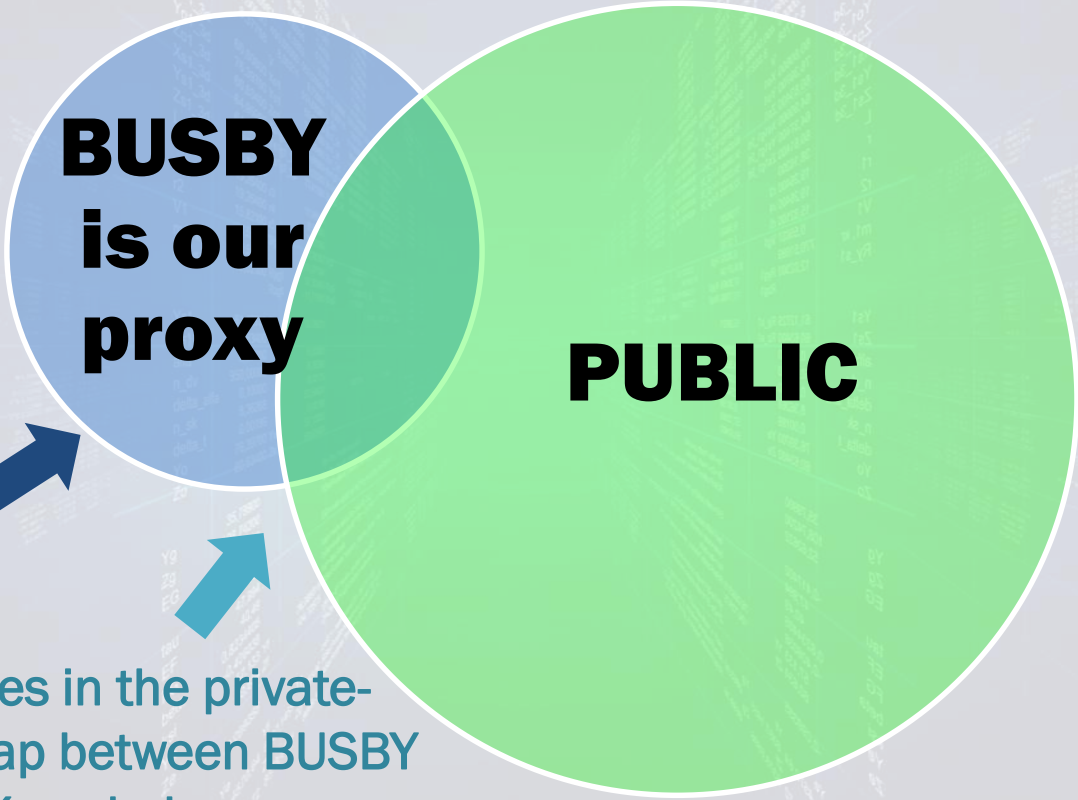
Small overlap: We're (mostly) secure; retain!



What about the overlap between us and them?



What about the overlap between us and them?



Vulnerabilities known to BUSBY; not in Public Knowledge

Vulnerabilities in the private-public overlap between BUSBY and Public Knowledge


Busby

BUSBY finds zero-day vulnerabilities, and develops exploits for them

14	207	64
Year span (2002-2016)	Vulnerabilities and their exploits	Vendors

Data consists of information about vulnerability class, source code type, exploit class type, vendor, product, exploit developer, and various dates (vulnerability discovery, exploit developed)

Data stats: three main types of vulnerabilities

Memory Corruption	Memory Mismanagement	Logic
110	41	67

Vulnerability Sub-Type: Memory Corruption

Type	Count
BSS Overflow	1
Data overflow	1
Heap Overflow	58
Integer overflow	2
Integer truncation	2
Stack overflow	40
Heap + Stack	1
Heap + Integer	1

Vulnerability Sub-Type: Memory Mismanagement

Type	Count
Remap memory	1
Information leak	4
Integer mismanagement	1
Invalid pointer dereference	2
Name validation	1
Null dereference	12
Out of bounds write	1
Privilege escalation	2
Reference count + object mismanagement	1
Type confusion + object mismanagement	1
Unsecure environment variables	1
Use after free	2
Use unverified supply pointer value	2

Vulnerability Sub-Type: Logic

Type	Count
API Misuse	3
Authentication Bypass	5
Auto execution	1
Bypass	1
Call-gate mismanagement	2
Command injection	3
Design misuse	1
Directory traversal; input validation	1
DNS Cache poisoning	1
Environment insertion	1
Executable file upload	1
File normalization error	1

Type	Count
File read primitive	2
IO control based on write primitives	1
Object injection / deserialization	4
Permissions on kernel device	1
Privilege issues: file read (1); mismanagement (2); spoofing (1)	4
Race condition	20
Reference count	3
Register / memory mismanagement	1
Remote code injection	1
SQLi	1
XSS	1

Data stats: number of vulnerabilities per source code type

Closed	Open	Mix or N/A
123	74	10

Data stats: number of vulnerabilities found and exploited, by vendor

Microsoft	Linux
55	39
Apple	**SUN/Oracle**
14	11

Other

88

- 64 vendors total
- Others include: Mozilla, LinkSys, Google, Adobe, etc.

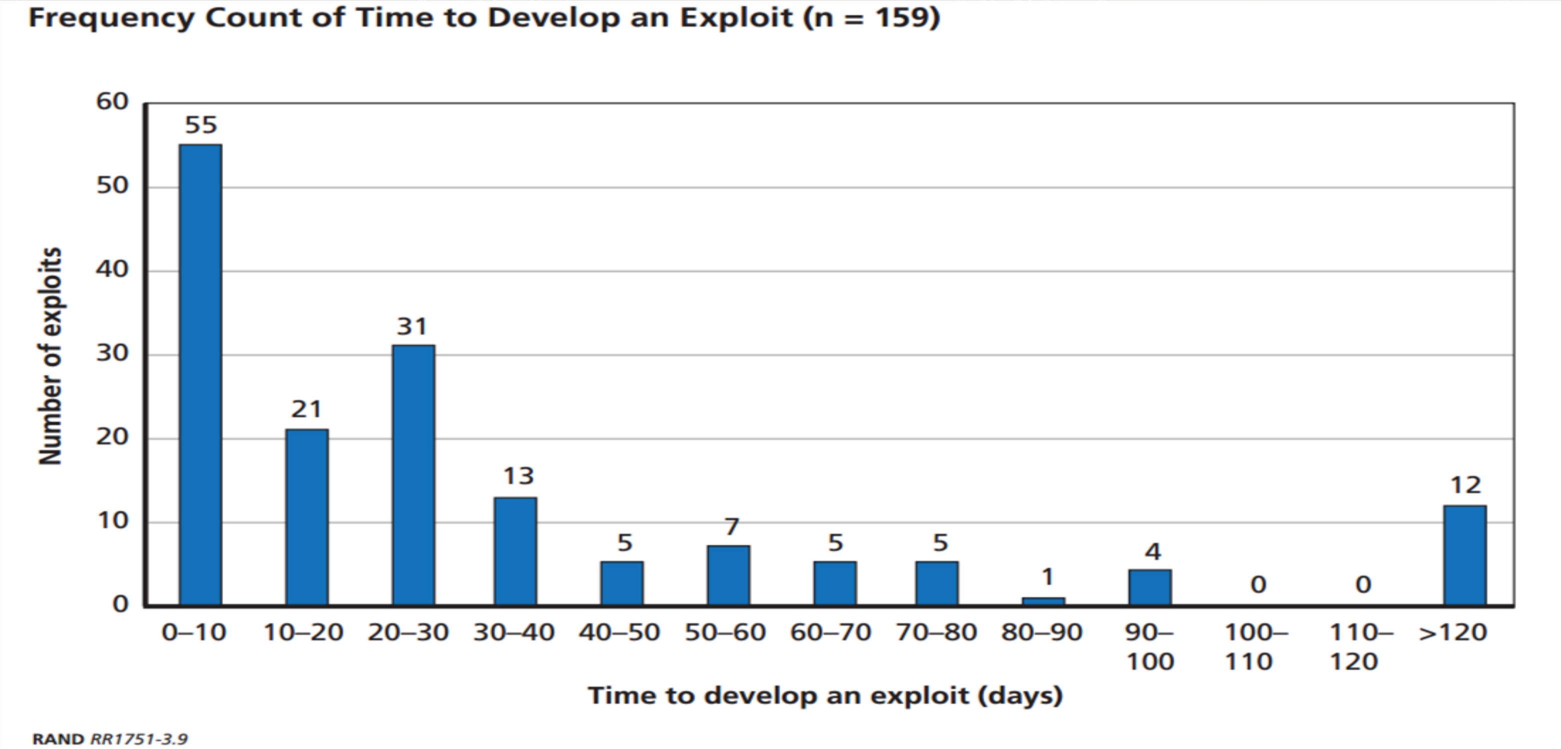
Data stats: number of exploits developed per exploit class type

Local	Client-side	Remote
76	25	71

Some other observations about the data

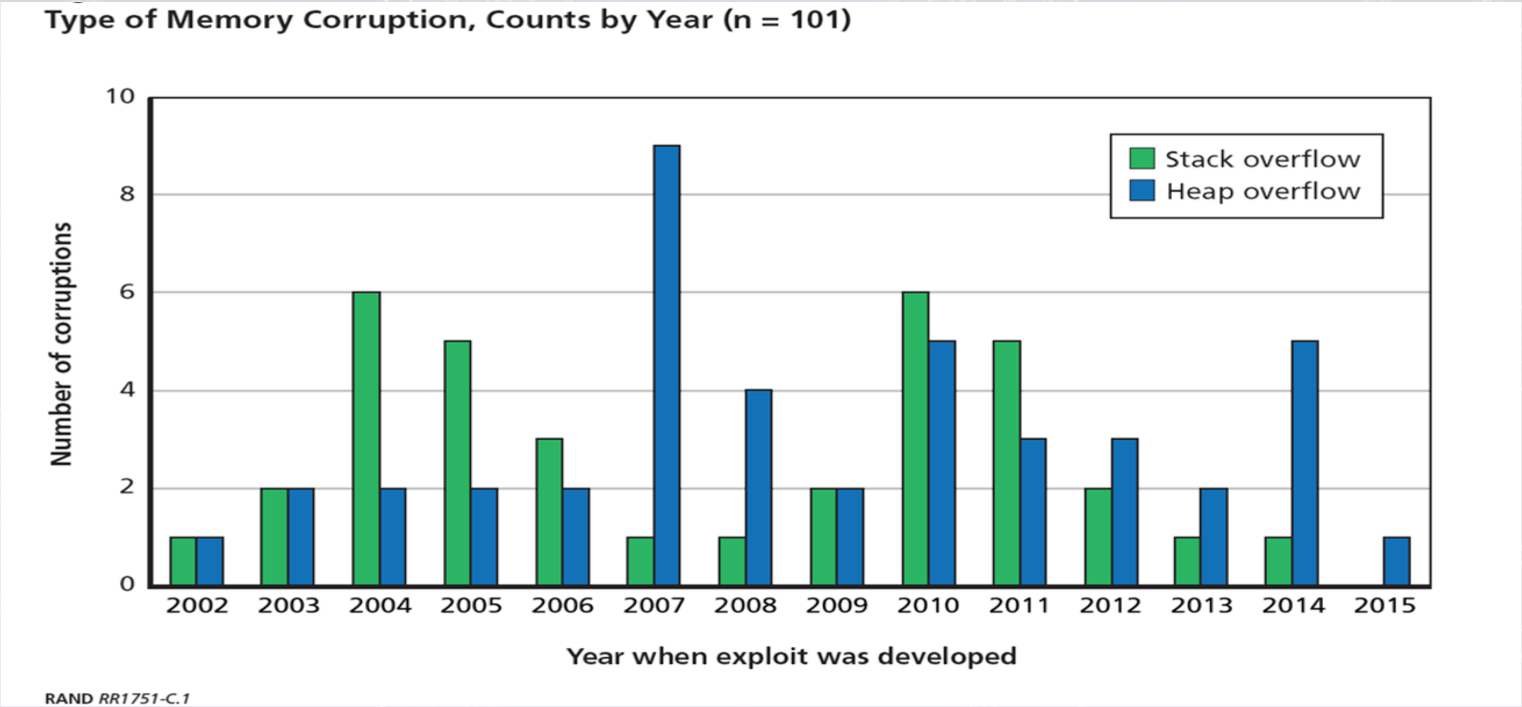
- 4% of the vulnerabilities in the dataset were purchased from an outside 3rd party
- Not all vulnerabilities were exploited
- CVEs do not always provide accurate and complete information about the severity of a vulnerability
- Exploitability is fluid, and can shift over time
- Virtual isolation (hypervisors or VMs) and anti-virus are not necessarily viable mitigations
- Other observations (charts) . . .

Exploit development time is relatively short



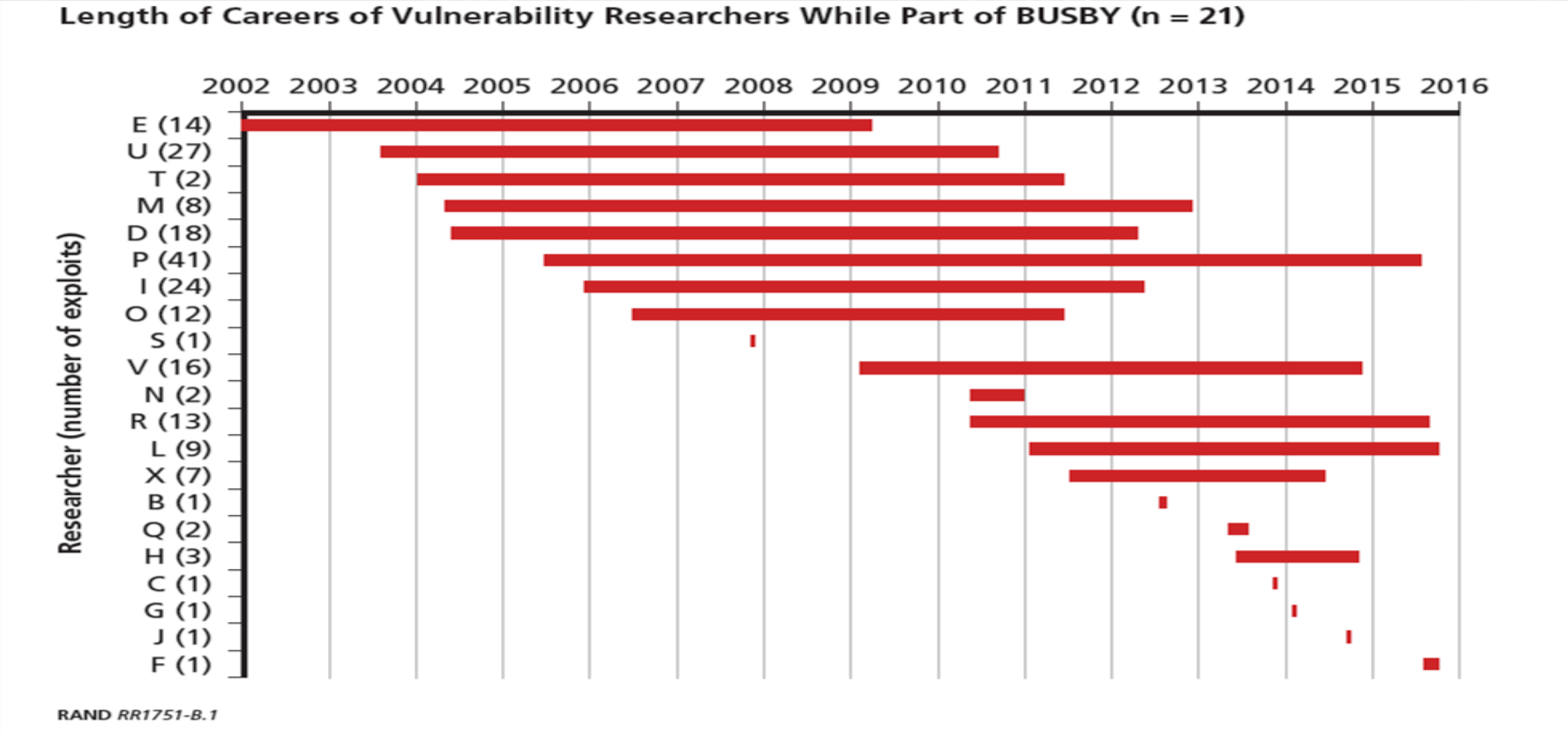
Over 70% of exploits are developed in a month (31 days) or less

Mitigations have affected exploitability (e.g., heap vs stack overflow)



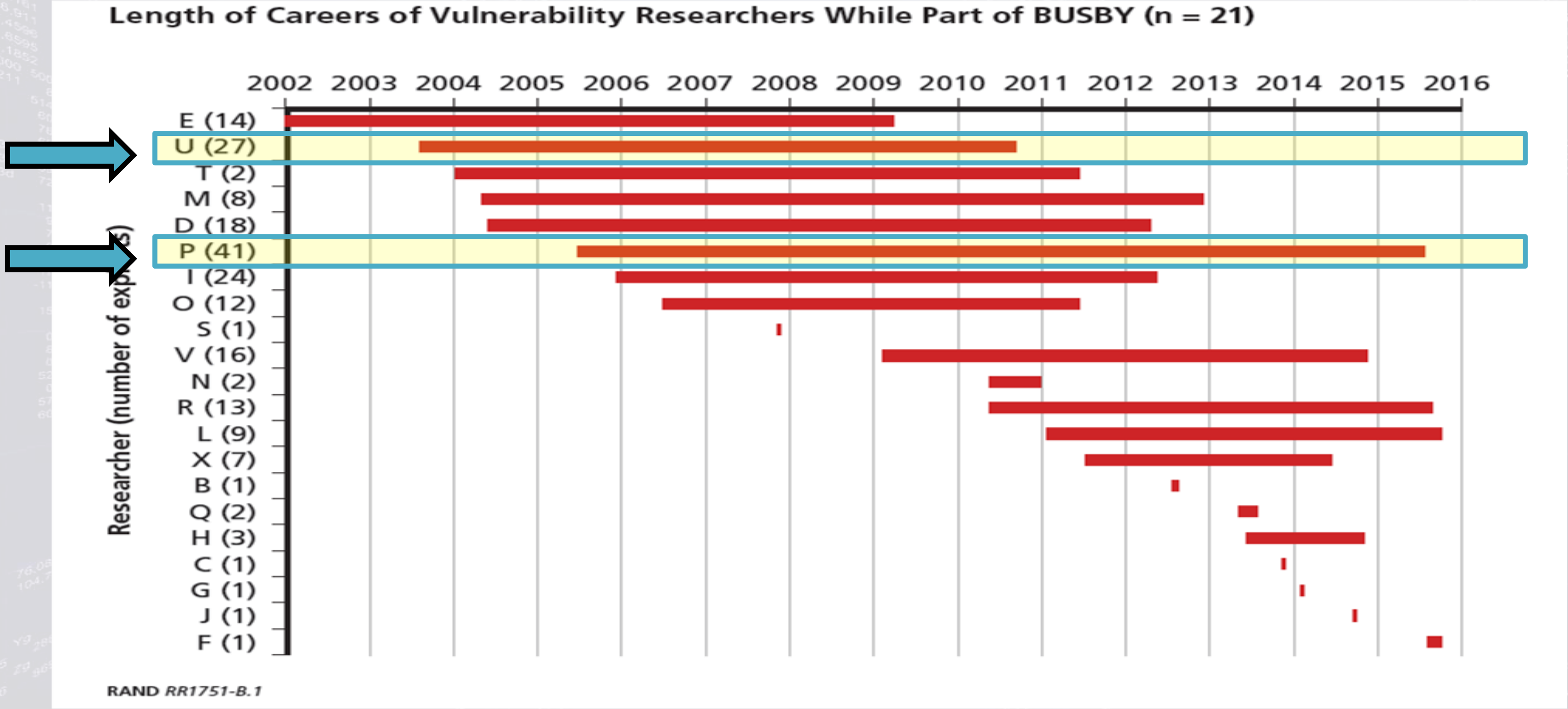
Mitigations introduced c. 2007 caused a shift in type of buffer overflow exploited

Exploit development career lengths vary



Low hanging fruit may account for a higher number of exploits developed early on

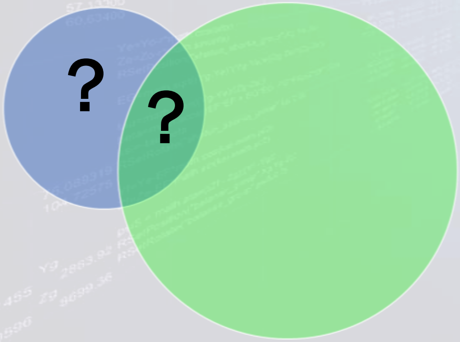
Exploit development career lengths vary



Each researcher has her/his own unique set of skills and focus

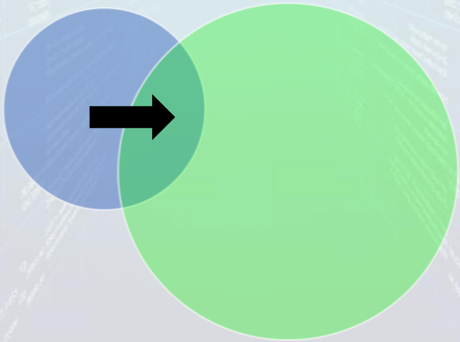
We focus on characteristics of the vulnerabilities

Life Status



Longevity

- Survival Rate
- Life Expectancy



Collision Rate

There are some caveats to our research

- Results from our research can be generalized only to similar datasets

- We are comparing private data to public data (ideal would be to compare multiple private datasets)

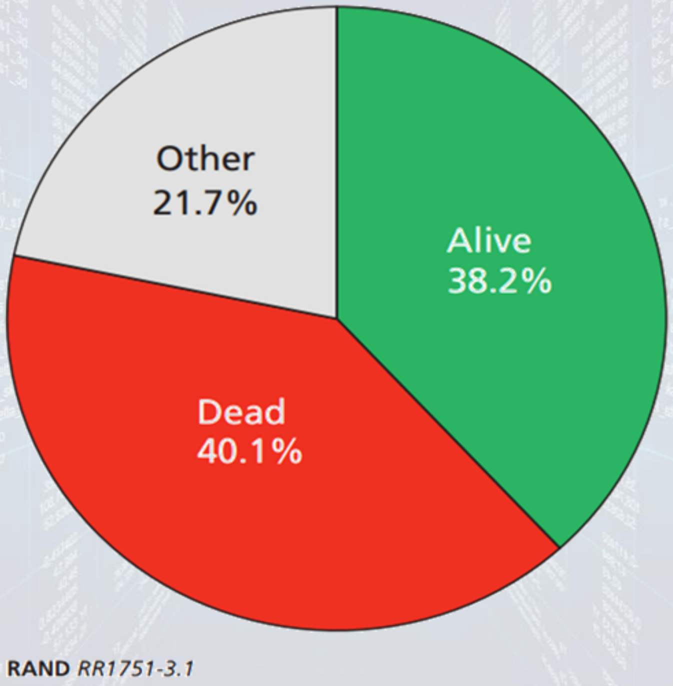
Life Status

Research Question: What are various “life stages” a zero-day vulnerability can be in?

Metric: What proportion of zero-day vulnerabilities are:

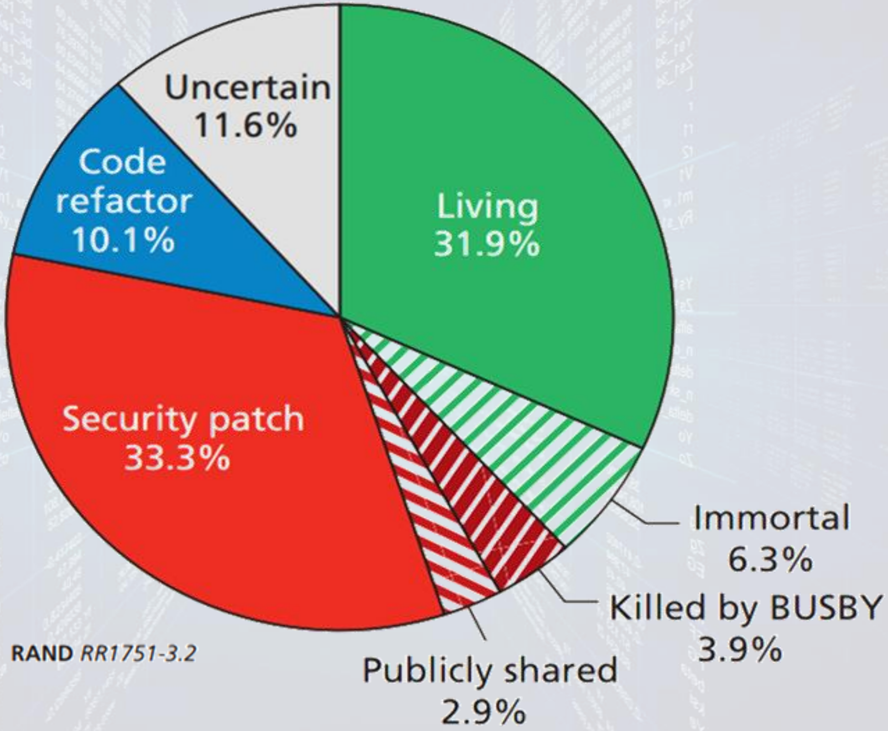
- Alive (publicly unknown / blue)
- Dead (publicly known / teal & green)
- Somewhere in between

Alive and dead are numbered about the same



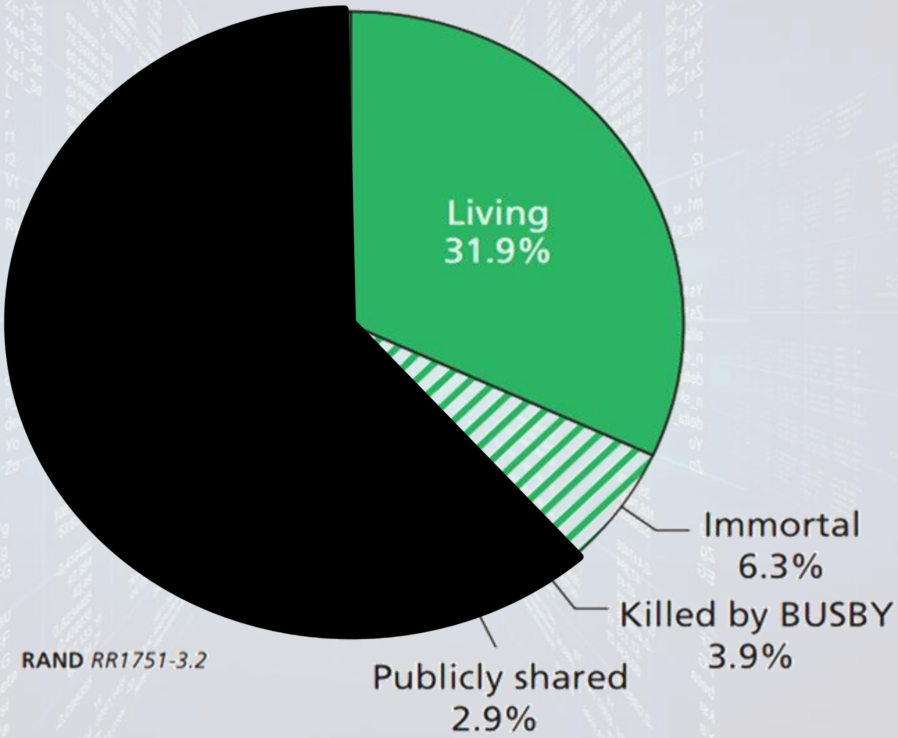
There is more granularity to a vulnerability being either alive or dead

We found more granularity in life stages



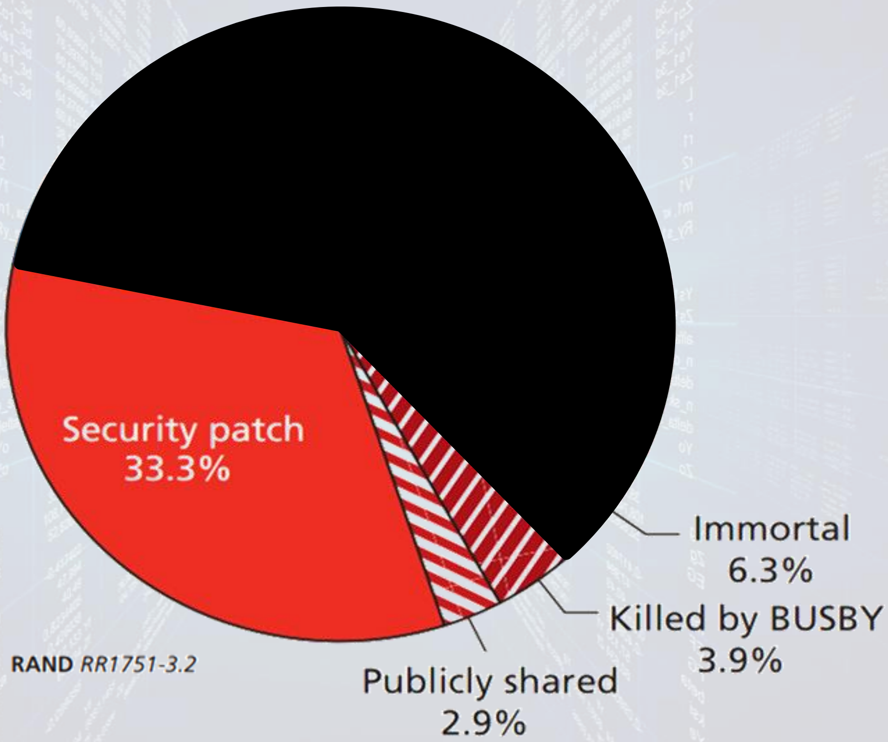
Labeling a vulnerability as either alive or dead is misleading and too simplistic

About 1 in 6 of the alive are immortal



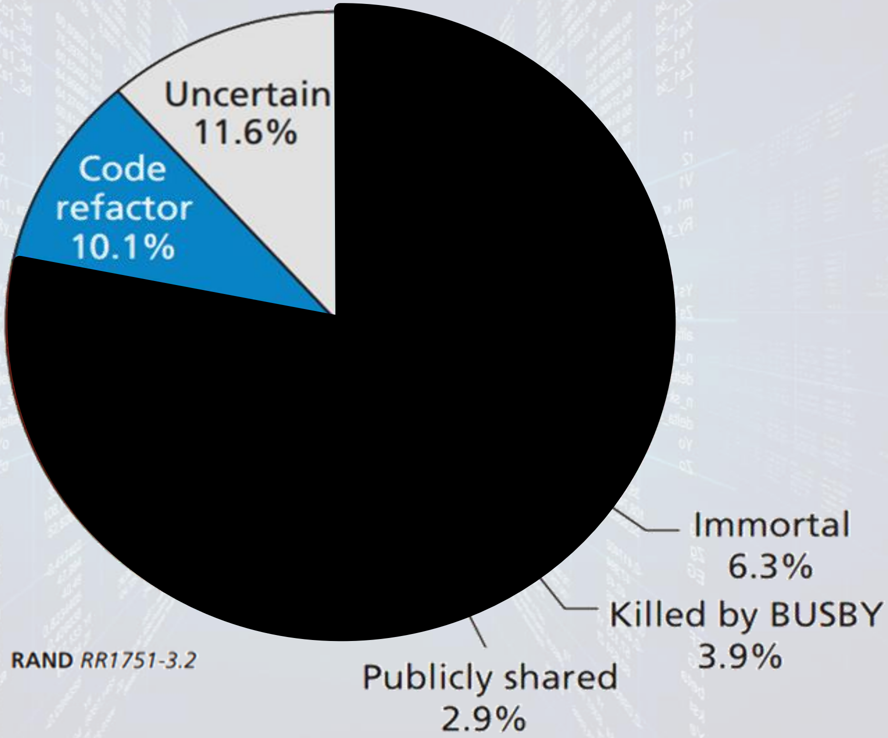
Labeling a vulnerability as either alive or dead is misleading and too simplistic

Patches killed most of the dead



Labeling a vulnerability as either alive or dead is misleading and too simplistic

Code revisions created a bunch of code refactored "zombies"



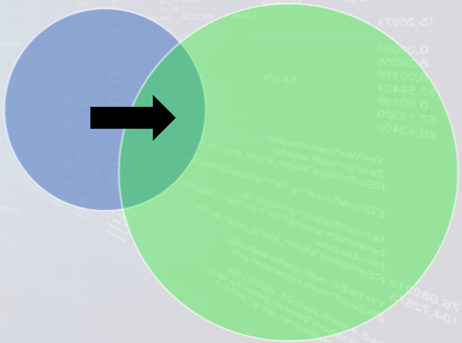
Labeling a vulnerability as either alive or dead is misleading and too simplistic

Longevity

Research Question: How long will a zero-day vulnerability remain undiscovered and undisclosed to the public?

Metrics:

- What is a short and long life for a zero-day vulnerability?
- What is the average life expectancy of a zero-day vulnerability and its exploit?

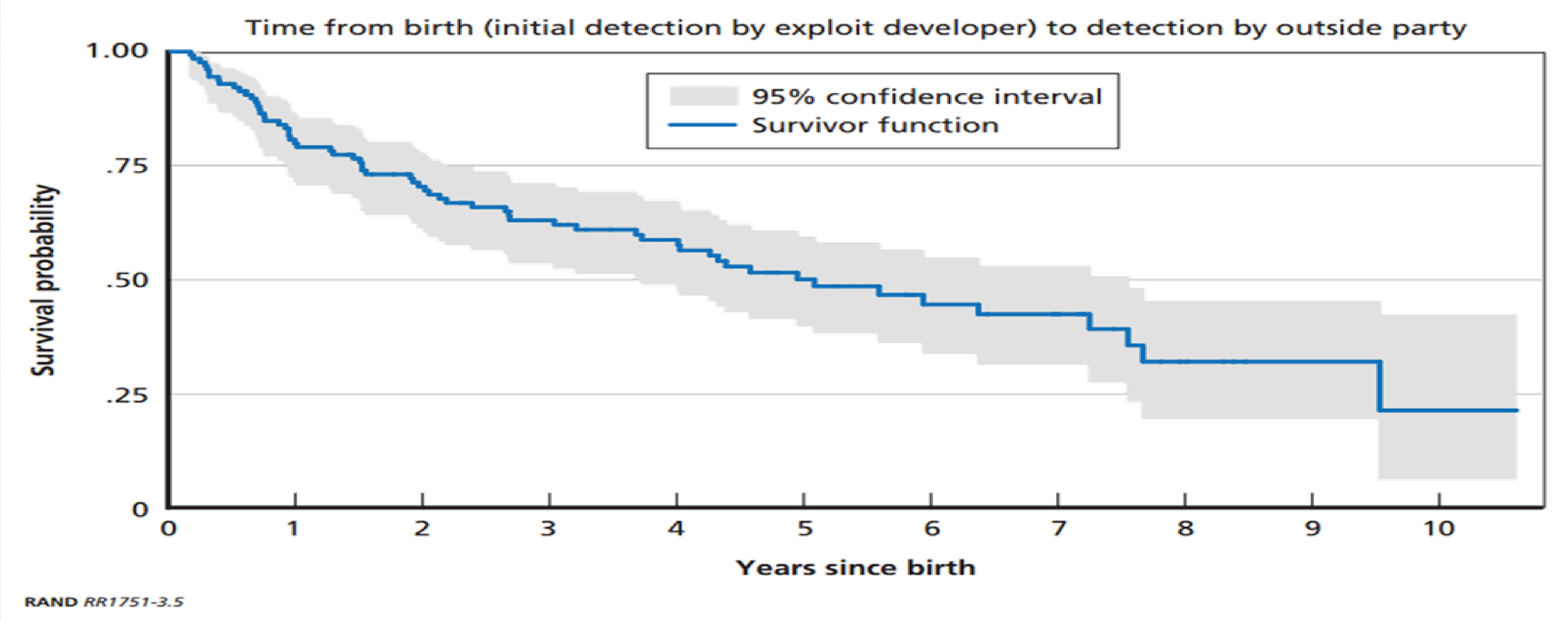


We borrowed a methodology from life insurers

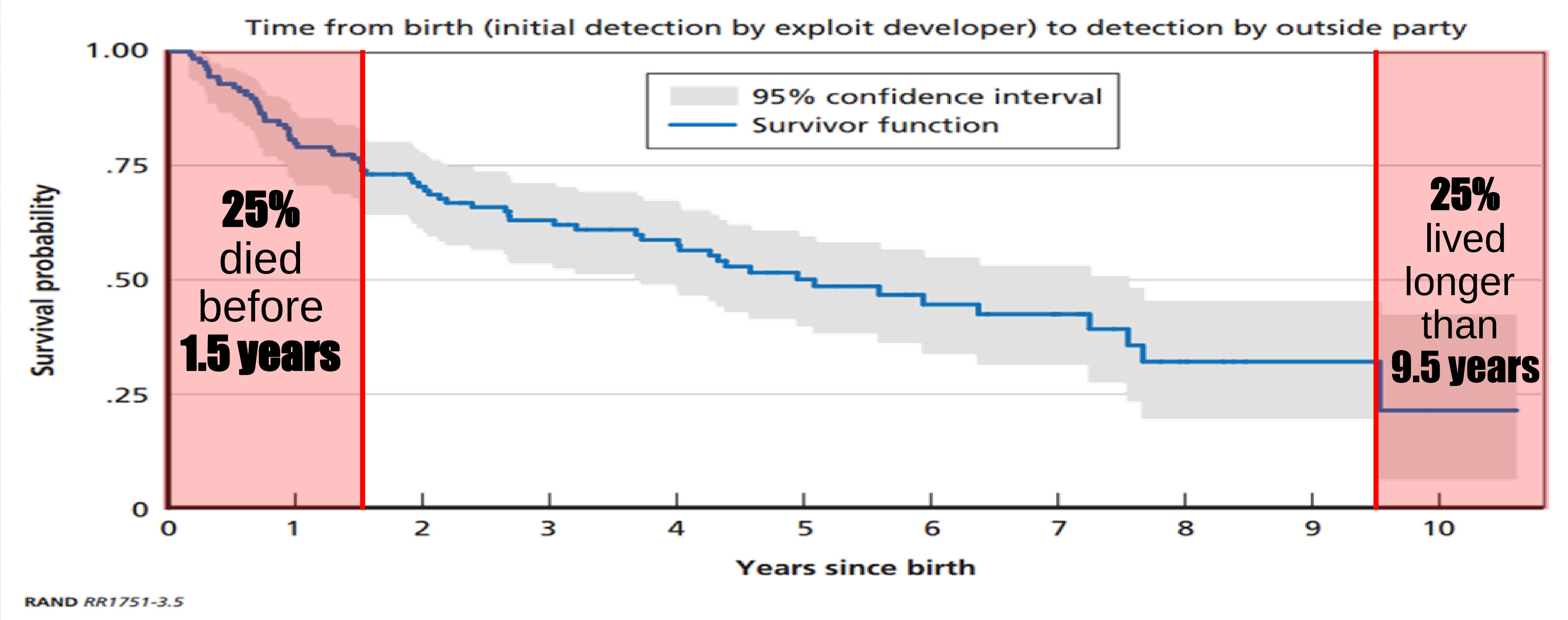
- We do not know what is going to happen to those vulnerabilities that are still currently alive
  - Calculating short life, long life, and average lifetimes requires taking into account alive vulnerabilities

- Kaplan-Meier analysis estimates the probability of surviving from some event of interest over time
  - Ex: For humans, the probability of someone having a heart attack
  - For vulnerabilities, the probability of dying and becoming publicly known

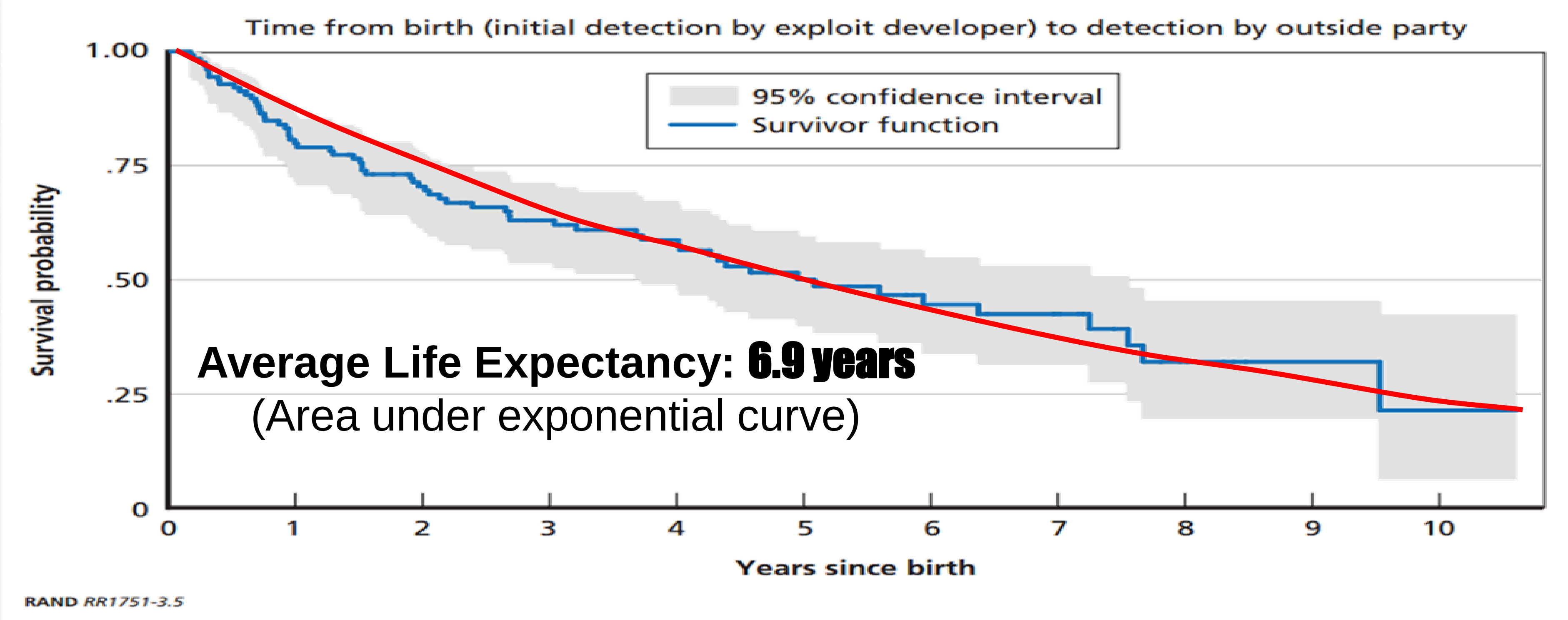
We plotted the survival probability of our data



75% lived longer than **1.5 years**



Average life expectancy is nearly 7 years

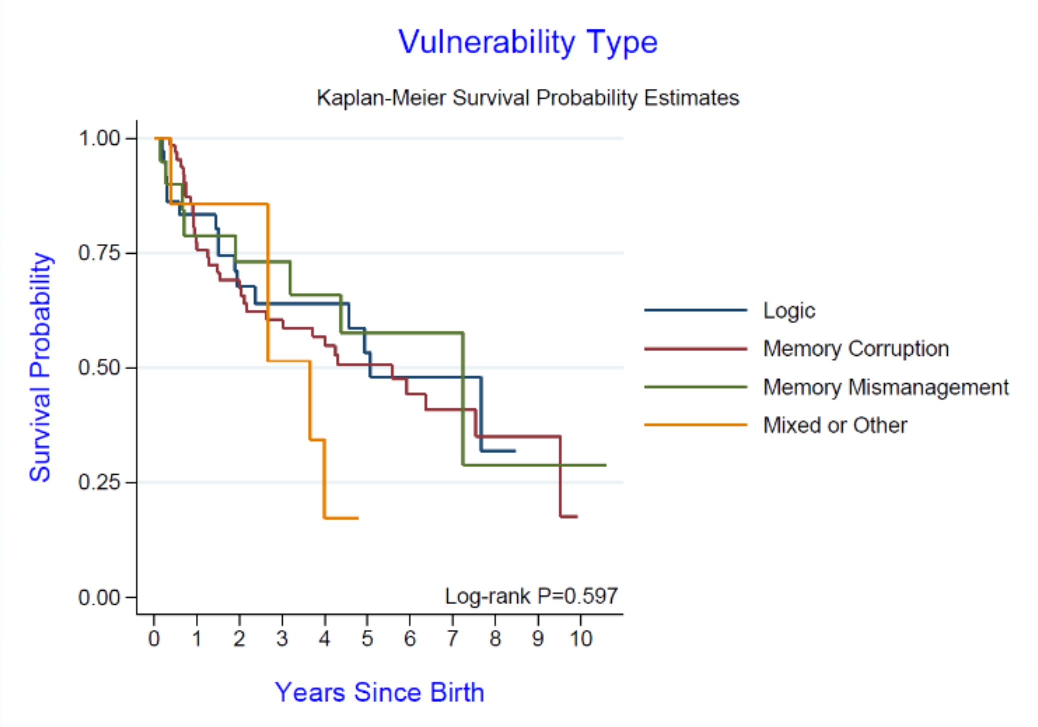


Do certain characteristics indicate a long or short life?

- Vulnerability Type
- Platform/Vendor affected
- Source Code
- Exploit Class

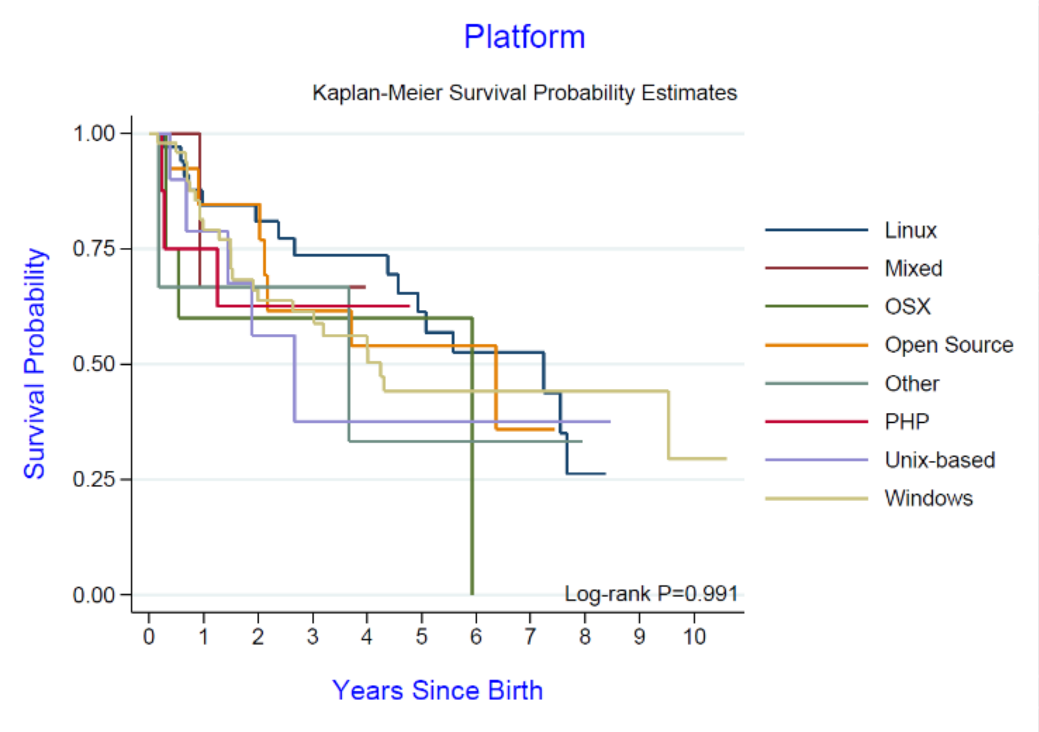
Do certain characteristics indicate a long or short life?

- Vulnerability Type
- Platform/Vendor affected
- Source Code
- Exploit Class



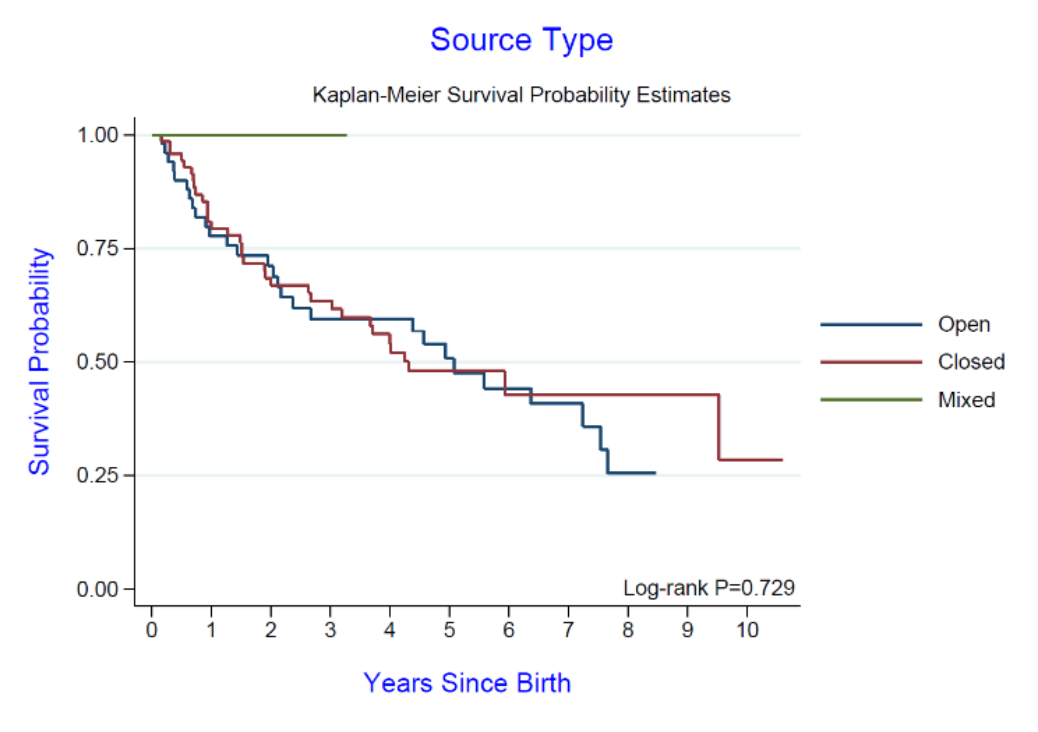
Do certain characteristics indicate a long or short life?

- Vulnerability Type
- **Platform/Vendor affected**
- Source Code
- Exploit Class



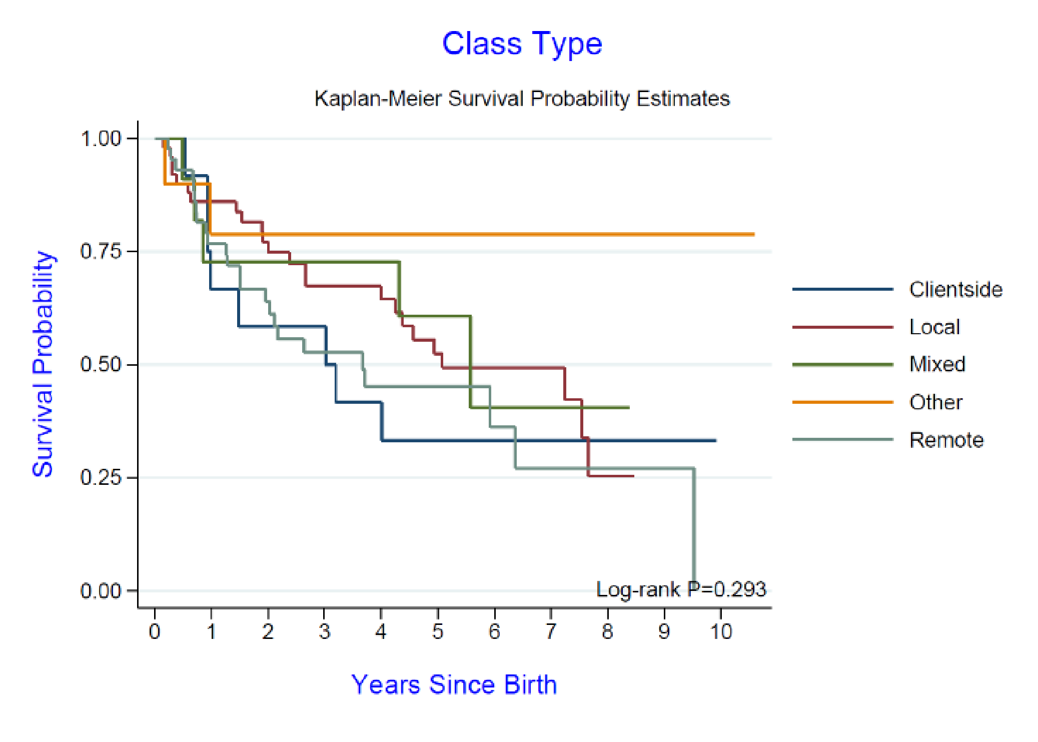
Do certain characteristics indicate a long or short life?

- Vulnerability Type
- Platform/Vendor affected
- Source Code
- Exploit Class



Do certain characteristics indicate a long or short life?

- Vulnerability Type
- Platform/Vendor affected
- Source Code
- Exploit Class



Do certain characteristics indicate a long or short life?

It’s unclear.

More data is needed to refine results.

Does life expectancy or survival probability change over time?

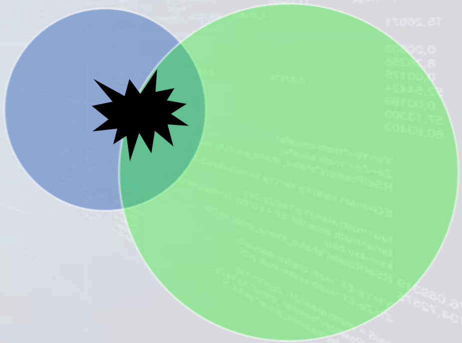
Does not appear so.
Results not statistically significant
to indicate a difference year by year.

More data could refine results.

Collision Rate

Research Question: What is the collision rate of zero-day vulnerabilities independently discovered and disclosed in a given time period?

Metric: What percentage of privately known vulnerabilities get independently rediscovered and publicly disclosed in a given time period?



Clarity about time intervals is important

Time interval:
All (14 years)

40%

We examined various time intervals

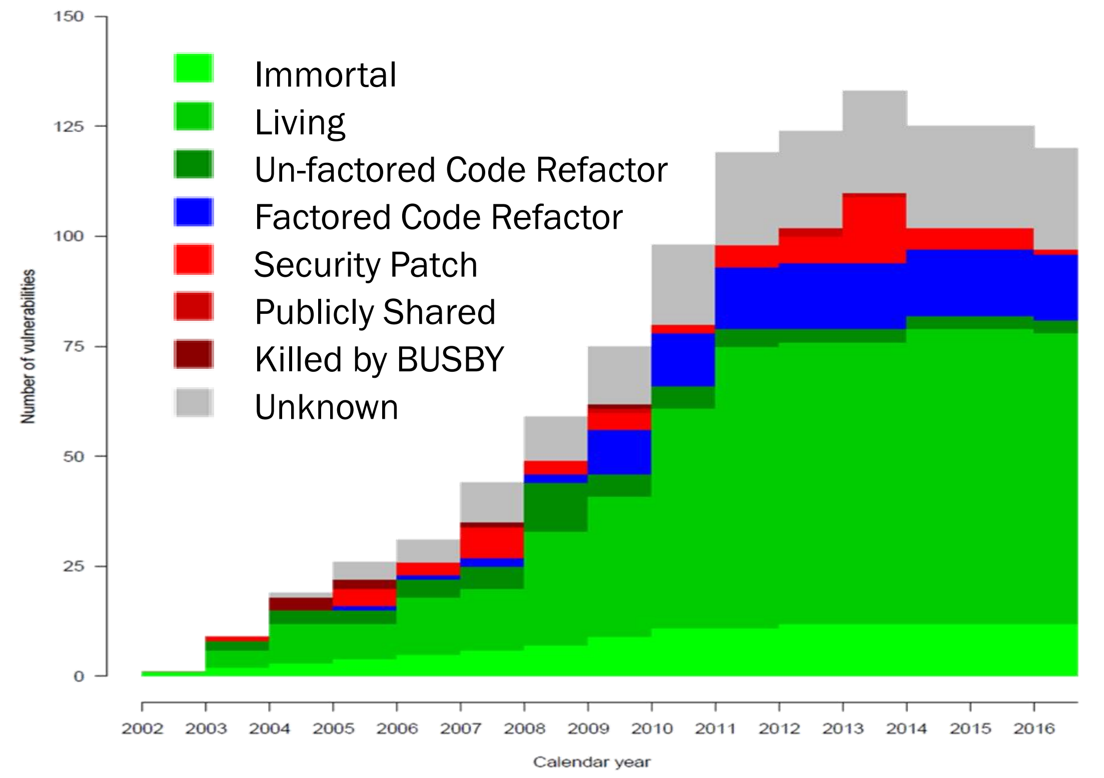
- Choose a time interval (365 days, 180 days, 90 days)
- Over that time interval, new zero-day vulnerabilities are discovered and retained
- At the end of the time interval, examine how many have been found by others and publicly disclosed (i.e. died)
  - "Throw out" those that have died
  - Keep the ones that are still alive
  - Continue to discover and retain new ones until the end of the next time interval when re-evaluation begins again

Collision rate: median percentage of those that died over all time intervals

Clarity about time intervals is important

Time interval:
365-days
(1 year)

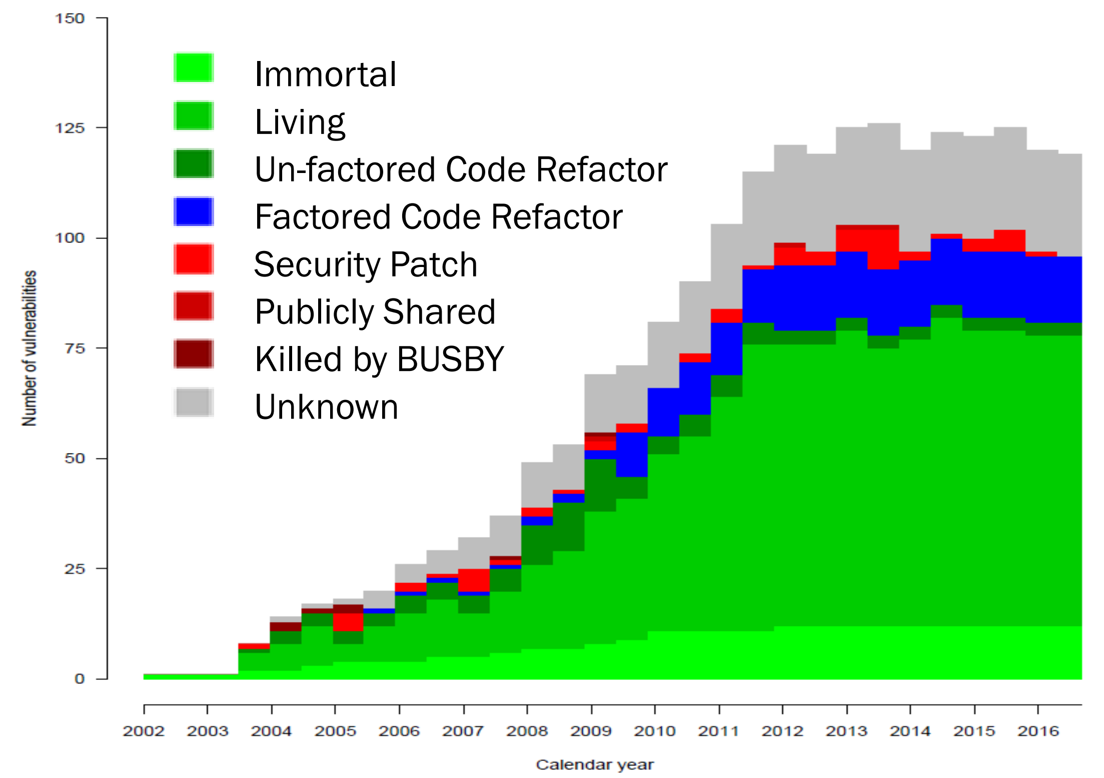
5.7%



Clarity about time intervals is important

Time interval:
180-days
(~6 months)

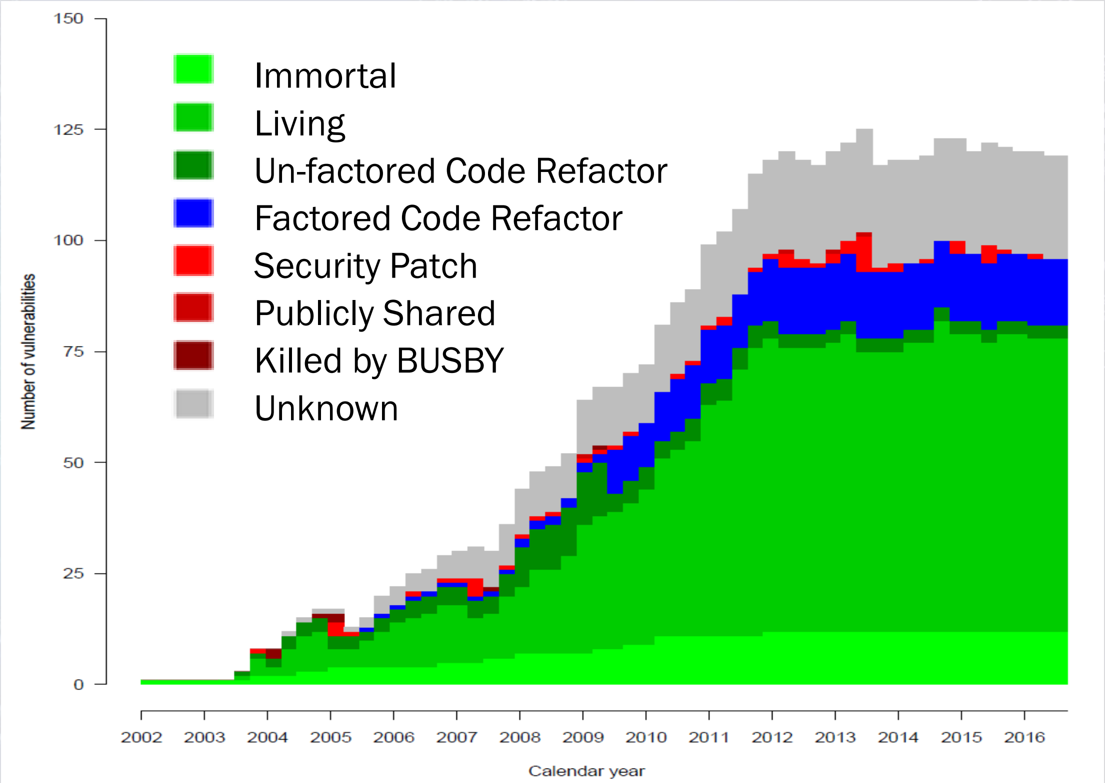
2.8%



Clarity about time intervals is important

Time interval:
90-days

0.87%



Meaning can be easily manipulated

Time interval: All (14 years)	Time interval: 365-days	Time interval: 180-days	Time interval: 90-days
40%	5.7%	2.8%	0.87%

Collision rates change significantly depending on the interval time

We explored several other research paths

- Average life expectancies based on vulnerability characteristic*
- Life expectancy variation based on birth year
- Collision rate variation based on vulnerability characteristic*
- Collision rate and timing for individual vulnerabilities
- Time to develop exploit based on vulnerability characteristic *
- Seasonality of vulnerability research
- Cost of developing an exploit

*****No statistical significance found, likely due to limited data*

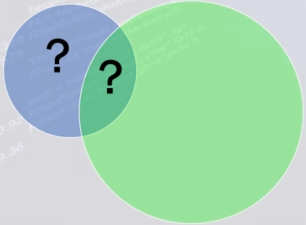
If you have data and would like to collaborate to refine this research, please contact me: lablon@rand.org or @lilyablon

Key findings

Life Status

7+ Categories

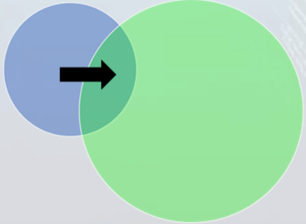
Labeling a zero-day vulnerability as either alive or dead can be misleading and too simplistic



Longevity

6.9 years

Zero-day vulnerabilities and their exploits have a rather long average life expectancy



Collision Rate

5.7% per year

Time interval examined can significantly change the percentage for likelihood of independent rediscovery

Report freely available at: http://www.rand.org/pubs/research_reports/RR1751.html

Implications and recommendations of findings

For those **defensively** focused

- Refine tactical approaches:
  - Analyze previous versions of code that are still in heavy use (e.g., ICS)
  - Harness techniques of how offense finds vulnerabilities
  - Seek better options to detect vulns
- Consider strategic approaches: mitigation, containment, accountability, and a robust infrastructure of patching
  - Employ physical isolation
  - Account for software, devices, and removable media
  - Incentivize upgrading to new versions

For those **offensively** focused

- Retain a few vulnerabilities per particular software package
- Consider immortal or code-refactored vulnerabilities for operations
- Regularly revisit vulnerabilities thought to be unexploitable
- Plan for a specific vulnerability only for short-term planning operations; expand to *any* vulnerability may extend the timeline

Our findings can help inform the retain vs. disclose discussions

Long average lifetimes and relatively low collision rates may indicate that:

vulnerabilities are dense, or vulnerabilities are hard to find

Our findings can help inform the retain vs. disclose discussions

vulnerabilities are dense, or vulnerabilities are hard to find

Pro **retention**

- The level of protection from disclosing a vulnerability may be modest
- There is a small probability of re-discovery by others

Pro **disclosure**

- Collision rates for zero-day vulnerabilities are non-zero
- A non-zero probability (no matter how small) that someone else will find the same zero-day vulnerability may be too risky

Zero-days affect many sectors, and raise policy questions

- Should we prioritize national security, or consumer safety and company liability?

- Should software companies be liable for vulnerabilities in their products?

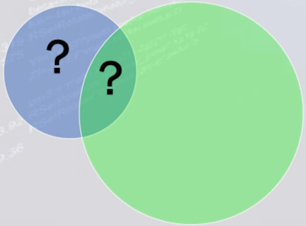
- What is the impact to a business' risk profile?

Key findings

Life Status

7+ Categories

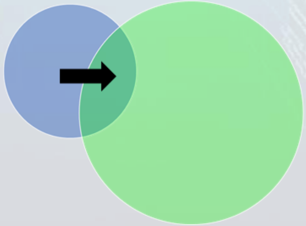
Labeling a zero-day vulnerability as either alive or dead can be misleading and too simplistic



Longevity

6.9 years

Zero-day vulnerabilities and their exploits have a rather long average life expectancy



Collision Rate

5.7% per year

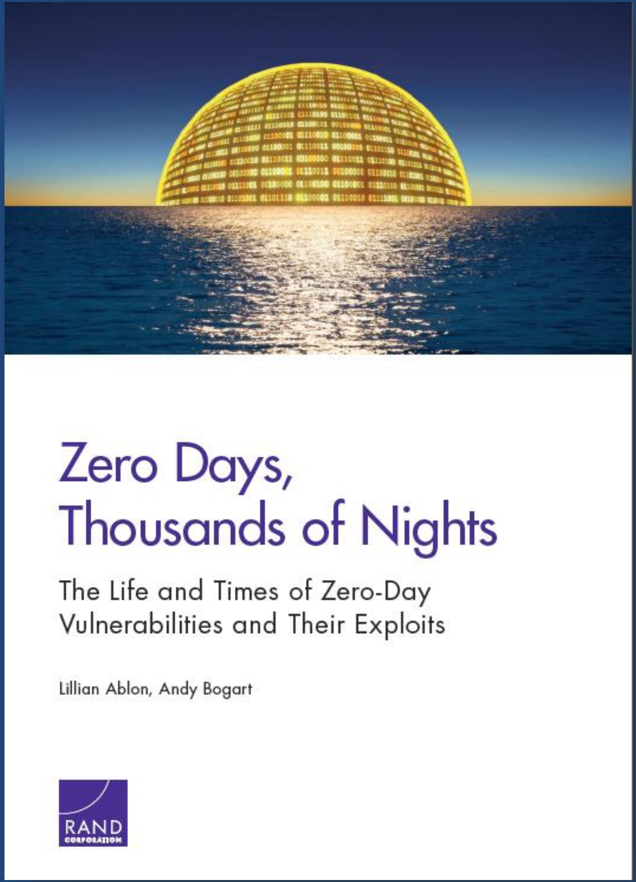
Time interval examined can significantly change the percentage for likelihood of independent rediscovery

Report freely available at: http://www.rand.org/pubs/research_reports/RR1751.html

Thank you!

Lillian Ablon

lablon@rand.org
@LilyAblon



RAND CORPORATION

Report freely available at: http://www.rand.org/pubs/research_reports/RR1751.html