Firefox: Security Vulnerability Analysis

- Devyani Kulkarni
- Balachander Padmanabha
- Vishwesh Rege

Why Firefox ?

- Second most popular browser with around half a billion users
- Mozilla Firefox grabbed **15.6 percent** of worldwide desktop browser usage – report by ArsTechnica (April 2017)
- Open sourced code base, with active set of contributors
- Vulnerability data available publicly in CVE and Bugzilla
- Bug bounty program of up to **$3000**
- **49** versions over **11** years, included traditional and rapid release cycles

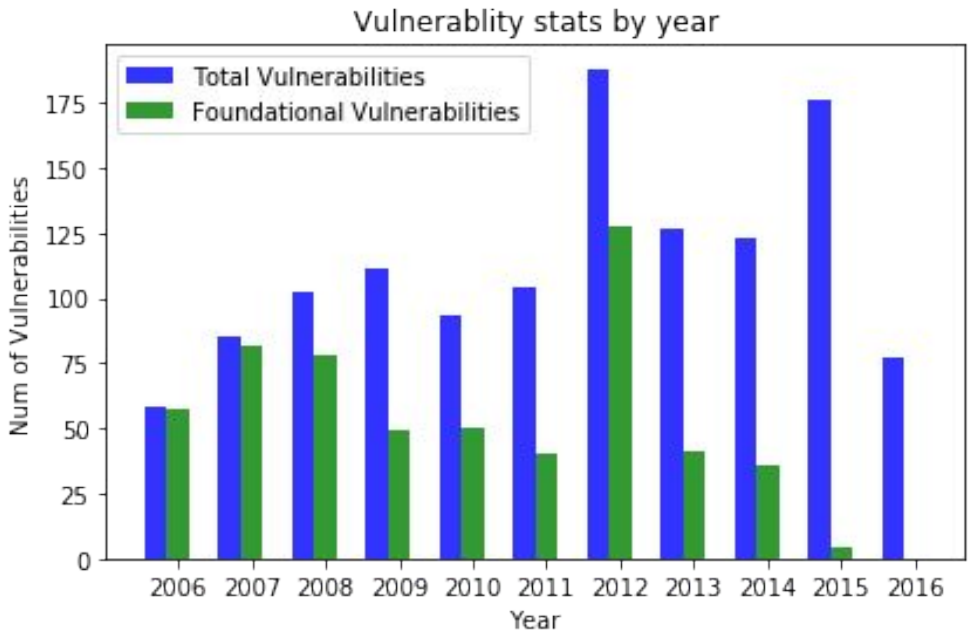
Focus of our Analysis

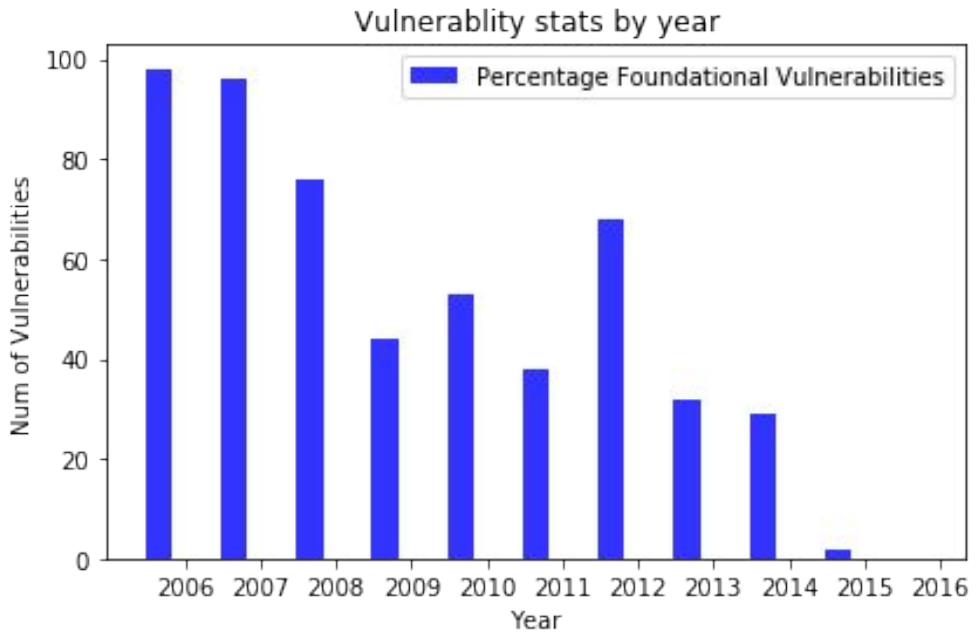
- Does Firefox browser security improve with age ?
- Does software release lifecycle affect software security ?
- Are certain components & files more vulnerable compared to others ?

Methodology

- Foundational version - Firefox 2
- Dataset for vulnerabilities
  - CVE reports (2006-2016)
  - Affected versions, bugzilla links, reported & fix dates
- Source Code information
  - Firefox release branch
  - Openhub

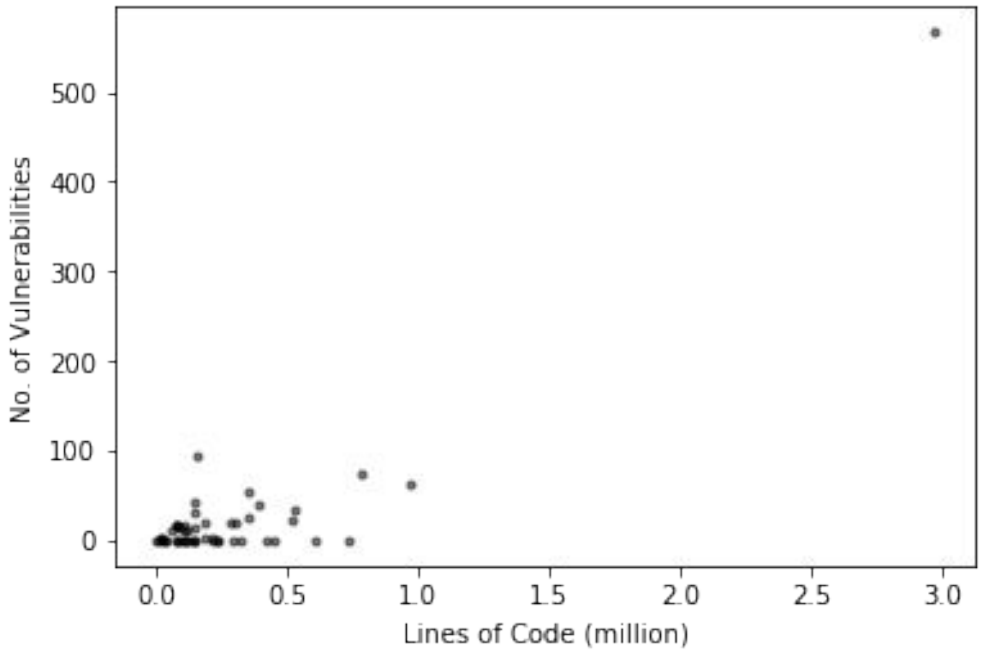
Are vulnerability reporting rates declining?





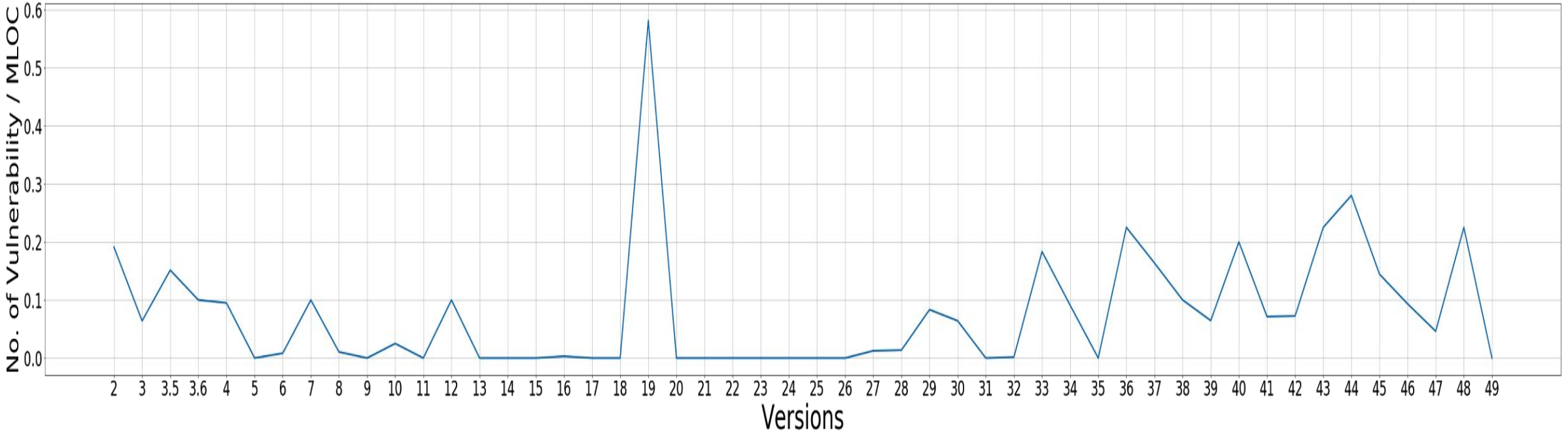
Do larger code changes cause more vulnerabilities?

- Anomaly for version 2
- The Pearson correlation coefficient of 0.0
- There is no relationship between the altered lines of code & number of vulnerabilities in Firefox



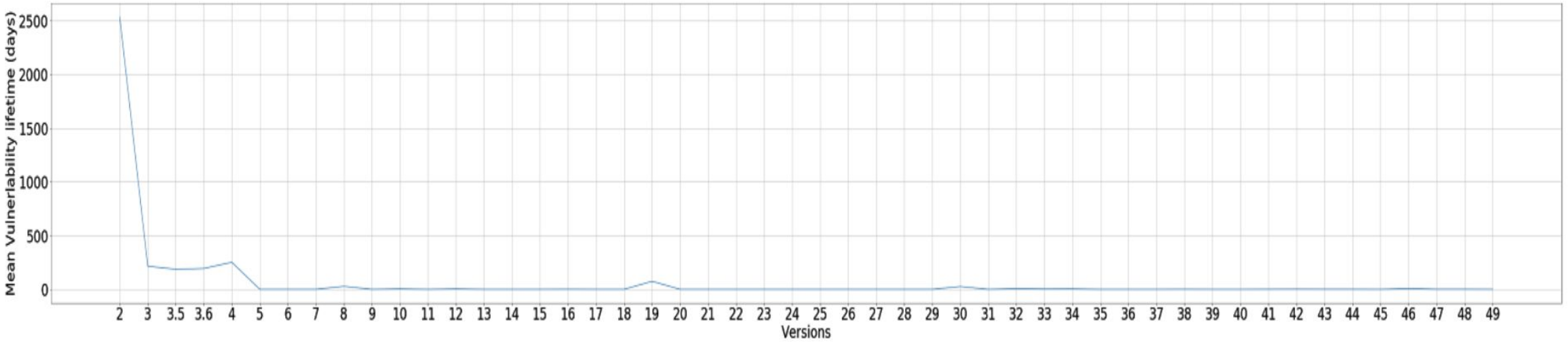
Do today's coders introduce fewer vulnerabilities per line of code?

- The vulnerability density of a version is the ratio of number of vulnerabilities to lines of code (MLOC)
- No conclusive trend in vulnerability density with time across the versions
- Vulnerability density ranged between 0 - 0.58125 & averaged at 0.0757



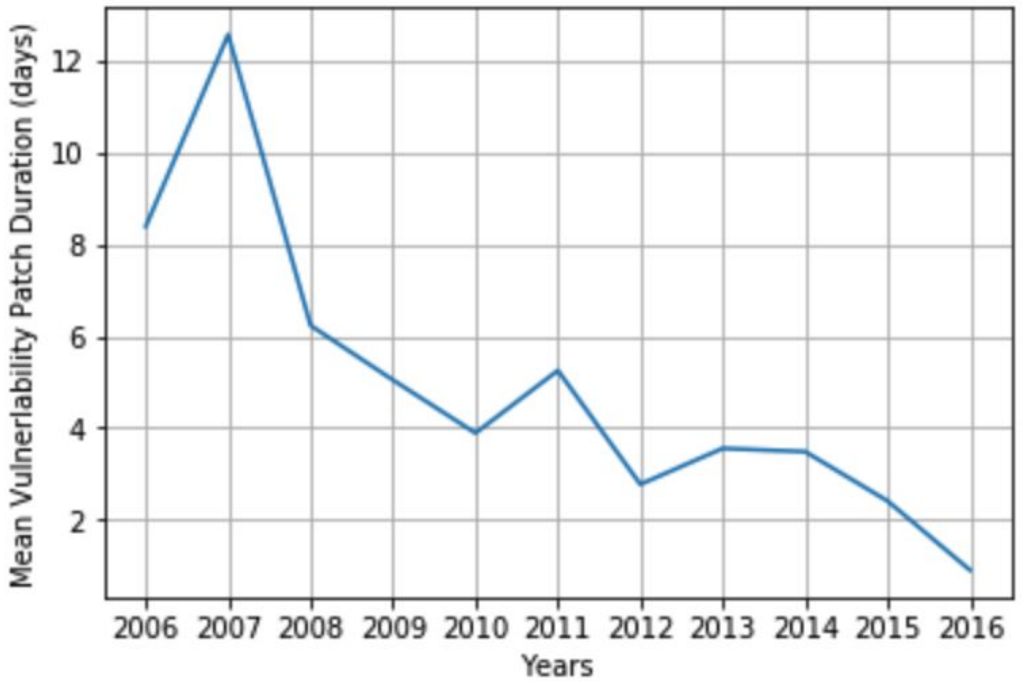
What is the mean lifetime of a vulnerability?

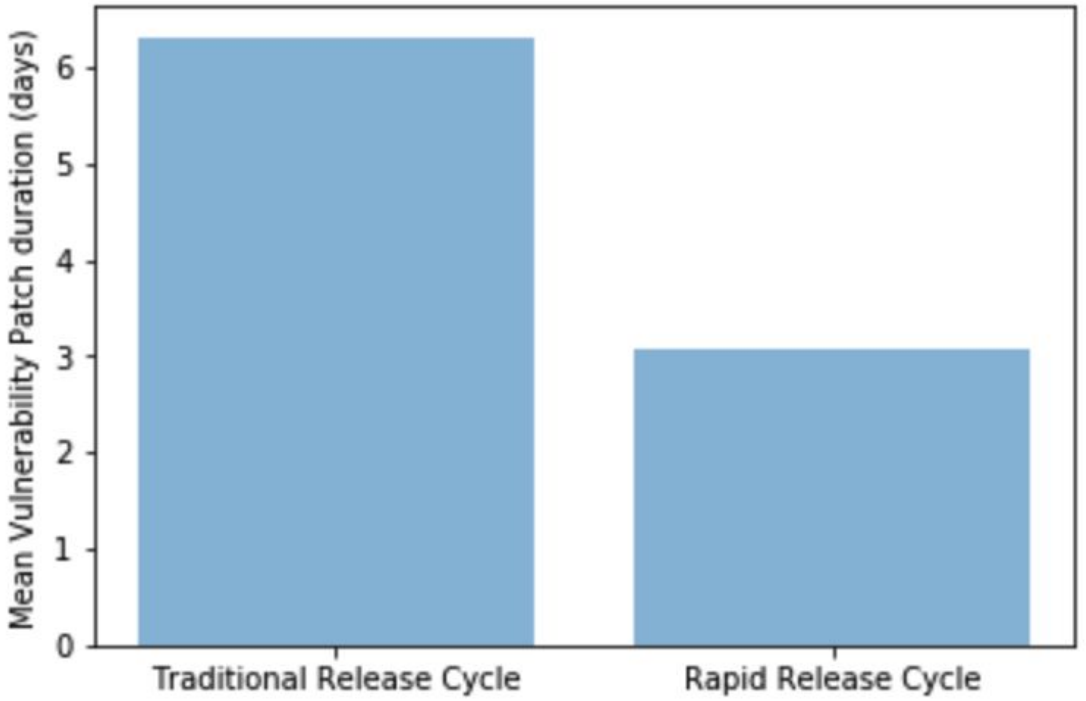
- Lifetime of a vulnerability is defined as the number of days between the release date of the version in which a vulnerability was introduced and the date on which it was reported
- The mean lifetime of a vulnerability for traditional releases is **676.6** days, whereas it is **3.6** days for the rapid releases



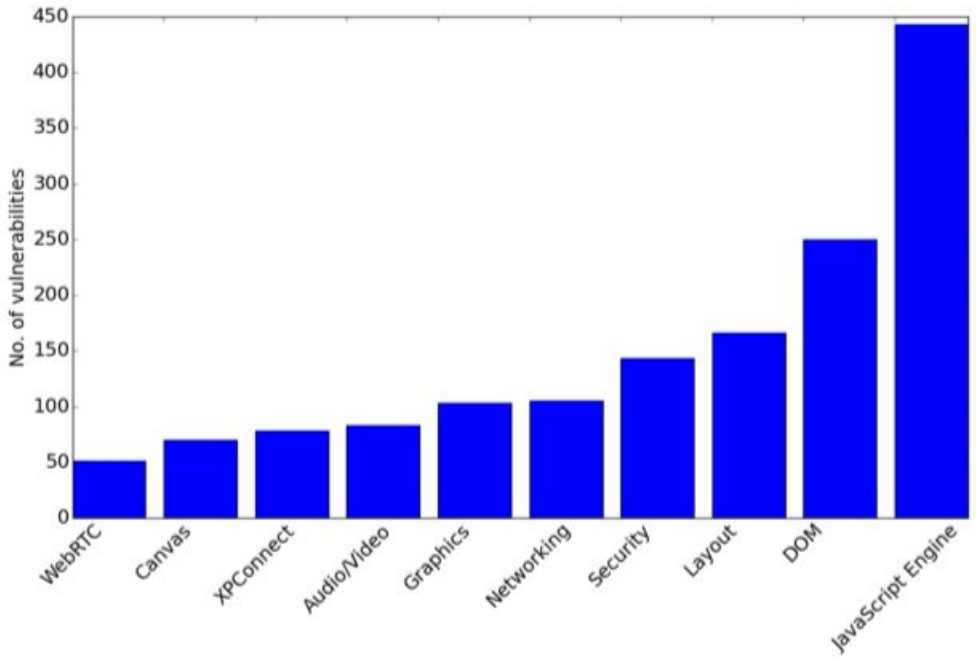
How fast are vulnerabilities fixed after being reported?

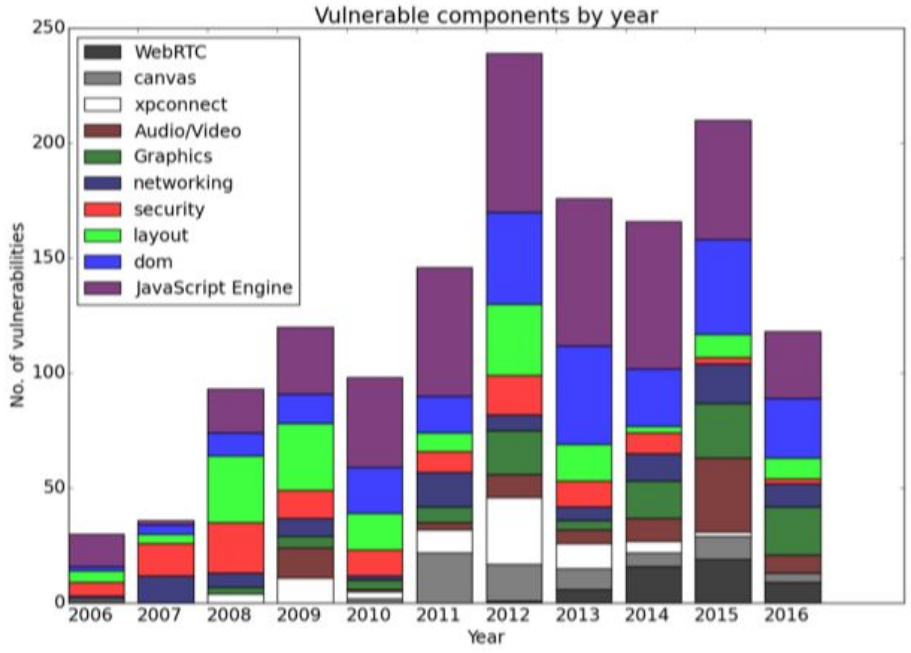
The patch duration for a vulnerability is defined as the days between date on which the vulnerability was reported and the date on which the fix was released.





Which files & components are most vulnerable?

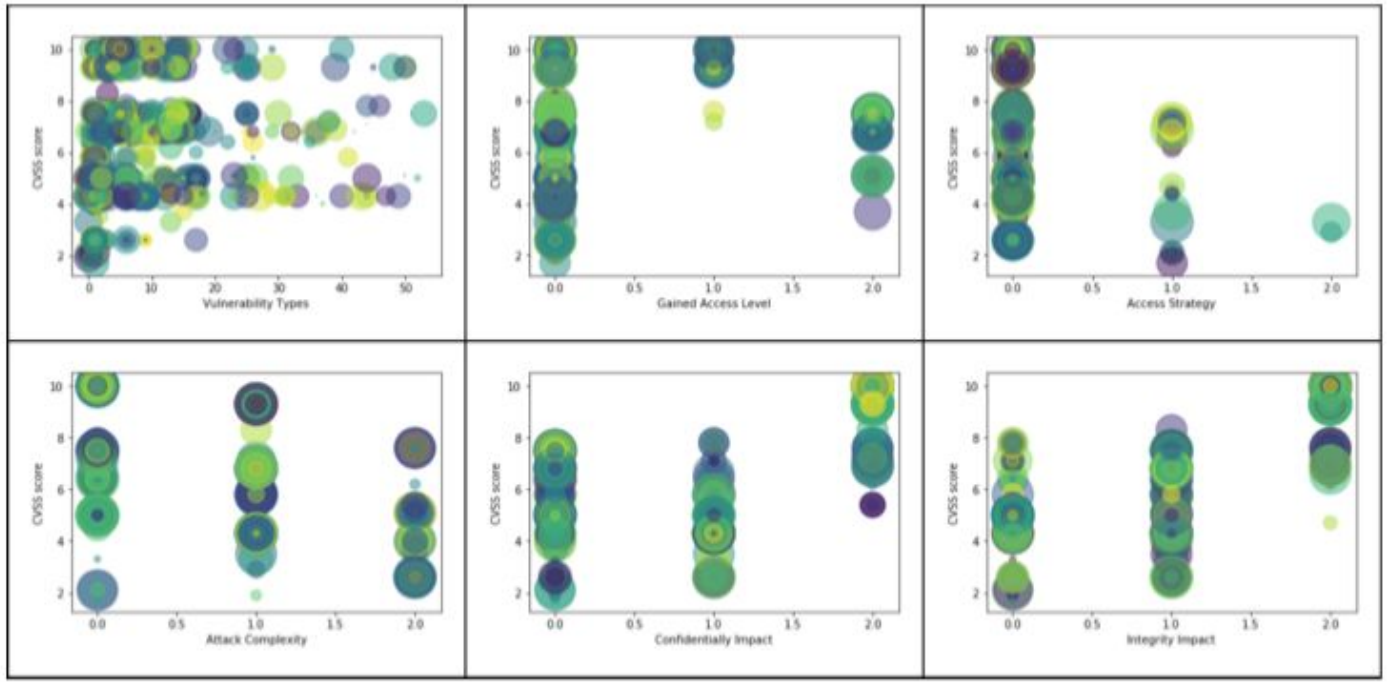




File name	Change frequency
content/base/src/nsContentUtils.cpp	28
content/canvas/src/WebGLContext.h	21
dom/base/nsGlobalWindow.h	16
js/src/vm/Stack.cpp	14
dom/canvas/CanvasRenderingContext2D.cpp	13
docshell/base/nsDocShell.cpp	13

Can we predict the CVSS score of a CVE Entry?

- Parameters:
  - vulnerability types
  - gained access level
  - access strategy
  - confidentiality impact
  - attack strategy
  - integrity impact
- Decision Tree Classifier
  - 97% accuracy
- Linear SVC
  - 88% accuracy



Conclusion

- 565 out of the total 1438 vulnerabilities were introduced in or before Firefox 2
- No strong decline of foundational vulnerabilities with age
- Vulnerabilities are reported much faster after shift to rapid release
  - Mean vulnerability lifetime: traditional 636.6 days vs 3.6 days
- Attention to security of Firefox has improved considerably after the shift to rapid releases
  - Mean patch duration is halved for rapid releases
- JavaScript & Graphic Rendering files/component are most vulnerable

Questions..


"That's all Folks!"