A Story of Breaking Things

Yusuf Hegazy

1879 AD (Logic Maketh Math ft. Frege & Peano)

- Aristotle's logic was unable to represent mathematical statements like Euclid's theorem.
- Frege came up with some good stuff like: $\forall x \, \exists y \, (y > x)$
- Peano came up axioms for arithmetic and notation like:
  - Element: $\in$
  - Subset: $\subset$
- such power, much math. (Theme 1: Unrestricted System)

~1900 AD (the year math was broken)

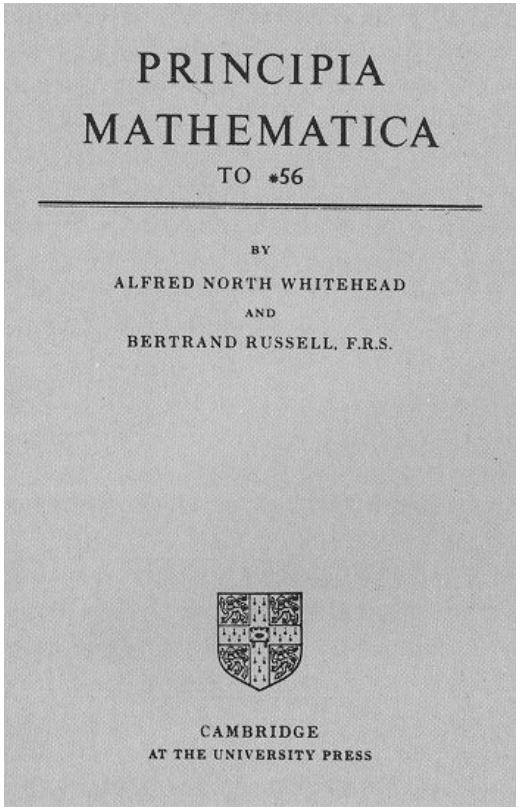
Let $R = \{x \mid x \notin x\}$. Then $R \in R \iff R \notin R$.

Theme 2: Self Reference

Theme 3: Type Confusion

~1900 AD

- featuring: Frege & Peano
- Proof that 1 + 1 = 2
- Solving Russell's Paradox
  - Russell created type theory
  - ZF set theory was founded and solved the issue using the axiom of separation
- ~~Theme 1: Unrestricted System~~



1931 AD (Math broken yet again)

- Kurt Gödel found a flaw in PM
- It is not a flaw in PM, it is a flaw of the understanding of math at the time
- Expressed "This statement is not provable in this system." in arithmetic using "Gödel Numbers"
- Any sufficiently powerful formal system cannot be both **consistent** and **complete**.

Theme 3: Type confusion

Theme 2: Self-reference

Why are you the way you are (math)?

- Formal systems are limited
  - Tarski's **undefinability theorem**
  - Church's proof that Hilbert's **Entscheidungsproblem** is unsolvable
  - Turing's theorem that there is no algorithm to solve the **halting problem**
  - ...


‘Predator’ spyware firm Intellexa resurgent after US sanctions
Revealed: leak uncovers global abuse of cyber-surveillance weapon
Spyware sold to authoritarian regimes
activists, politicians and journalis
Edward Snowden: Leaks that exposed US spy programme
Maker of Pegasus spyware told to pay $167m for WhatsApp hack
European journalists targeted with Paragon Solutions spyware, say researchers

The leaky bucket problem (Buffer Overflows)

- **C was too powerful, easy to shoot yourself in the foot.**

(Theme 1: Unrestricted System)

- Fix
  - Teach people to stop shooting themselves in the foot (aka. Use strncpy instead of strcpy, fgets instead of gets, the list goes on…)
  - Force the fixes: Compiler Protections
    - Stack Canaries/Cookies
    - ASLR & PIE
    - W^X principle
    - …

doesn't really solve the root cause, just makes an attack harder and more complex (basically raises our salaries)

Weird machines are born.

- executables contain a lot of sequences of instructions

Theme 1: Unrestricted System

- ROP: Encode executable instructions in a buffer overflow.

Theme 3: Type confusion

  - Result: we can run whatever code we want, sky's the limit.
- More oriented programming == More weird machines
    - ROP: put code in return address
    - SROP: put code in signal handlers
    - FSOP: put code in file structs
    - ...
- Data + Metadata = Weird machines

Refs

[1] Godel's Proof - Nagel

[2] The Annotated Turing - Petzold

[3] On Formally Undecidable Propositions of Principia Mathematica and Related Systems. - Godel (Braithwaite Annotation)

[4] Godel's proof in Lisp

[5] Weird Machines

[6] What hacker research taught me - Bratus