THE MATHEMATICS OF OBSCURITY

HERMANN HÄRTIG,
CLAUDE-JOACHIM HAMANN,
MICHAEL ROITZSCH
I will tell about…

- the process of finding security errors
- our mathematical model for it
- the comparison of open and closed source

I will not tell about…

- whether open or closed source is better
Open Source  Closed Source
Open Source

- everyone has access to source code
- everyone can search for bugs
- more defenders find more bugs
- easier to find bugs

Closed Source

- only the company has the source code
- attackers have a harder time
- bugs are prevented from being exploited
- harder to find bugs
Attackers only need only one error. Defenders need to find all errors.
3 Errors:

No Error:

\( e = 3 \)

\( p,q \)
ATTACKERS

\[ a = 3 \]
$p_A = 1 - (1 - ep)^a$
DEFENDERS

\[ d = 5 \]
The Static Model

DEFENDERS

The Mathematics of Obscurity

\[ p_D = e! \cdot \sum_{i=0}^{d-e} \binom{d}{i} q^{d-i} (1 - eq)^i S_{d-i,e} \]
- 20 errors
- 1% probability to find an error
  \( p = q = 0.01 \)
- 75% desired winning chance
  \( p_A = p_D = 0.75 \)
- How many attackers?
  \( a = 7 \)
- How many defenders?
  \( d = 424 \)
- What happens if both sides lose?
- ... or win?
- Do the defenders really lose if they do not find all errors?
- They just have to find the errors first.
- Instead of a snapshot, model a race.
Defenders need to find any error earlier than the attackers.
Figure 3: The Dynamic Model

The dynamic model considers the probability of finding an error in a software system through a series of steps. Each step involves drawing a ball from an urn, where the urns are divided into two categories: attackers and defenders.

- **Attackers (represented by the black ball)**: The probability of finding the black ball in the attacker's urn is denoted by $p$.
- **Defenders (represented by the white ball)**: The probability of finding the white ball in the defender's urn is denoted by $q$.

The probability of finding the black ball in the $m$th step for the attacker is given by:

$$p_{m,n} = (1 - p)^{m-1} p \cdot (1 - q)^{n-1} q$$

Where:
- $m$ is the number of steps for the attacker.
- $n$ is the number of steps for the defender.

The model accounts for the probabilities of finding the black ball in each step, considering the possibility of the error being found in the current step or not, and then moving to the next step. The urns are considered to be replaced after each draw, ensuring that the probabilities remain constant across all turns.

This dynamic model allows for the calculation of probability distributions over an unbounded number of turns, providing insights into the effectiveness of the attackers and defenders in finding errors.
We represent this situation: What are the odds for the defenders to find exactly this error before the attackers find it? We continue with an example software project, which we define in the next section. The uniform distribution of errors is missing from the list of vulnerabilities; thus, this particular outcome is favorable for the group claiming the vulnerabilities. The drawing steps are synchronized and will eventually find the ball; hence the probability is $\frac{m}{m+n}$.

Multiple Errors

The compound probability to find at least one attacker to find the black ball in one step and the defender finding the ball in the second step is thus $p_{m,n} = \sum_{n=1}^{\infty} \sum_{m=n+1}^{\infty} p_{m,n} = \frac{q (1 - p)}{q (1 - p) + p}$.

defenders win for $n < m$
The Mathematics of Obscurity

open source

more defenders

higher q

closed source

harder for attackers

lower p

\[ p_W = \sum_{n=1}^{\infty} \sum_{m=n+1}^{\infty} p_{m,n} = \frac{q(1-p)}{q(1-p)+p} \]
- 1 million lines of code, 15 security errors \( e = 15 \)
- probability for a single defender to find an error \( q_{single} = 0.002\% \)
- the same for attackers in open source case \( p_{single,open} = 0.002\% \)
- closed source factor 2 harder \( p_{single,closed} = 0.001\% \)
- 500 attackers
- How many defenders do we need?
### RESULTS

<table>
<thead>
<tr>
<th>Source</th>
<th>$p_w = 0.6$</th>
<th>$p_w = 0.9$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Closed</strong></td>
<td>7815</td>
<td>62088</td>
</tr>
<tr>
<td><strong>Open</strong></td>
<td>17133</td>
<td>impossible</td>
</tr>
</tbody>
</table>
CONCLUSION

No matter how many defenders, there’s always a window for attackers.
■ urn model for discovery of security errors
■ race between attackers and defenders
■ there is an upper bound for the defenders
■ this bound may be hit in reality