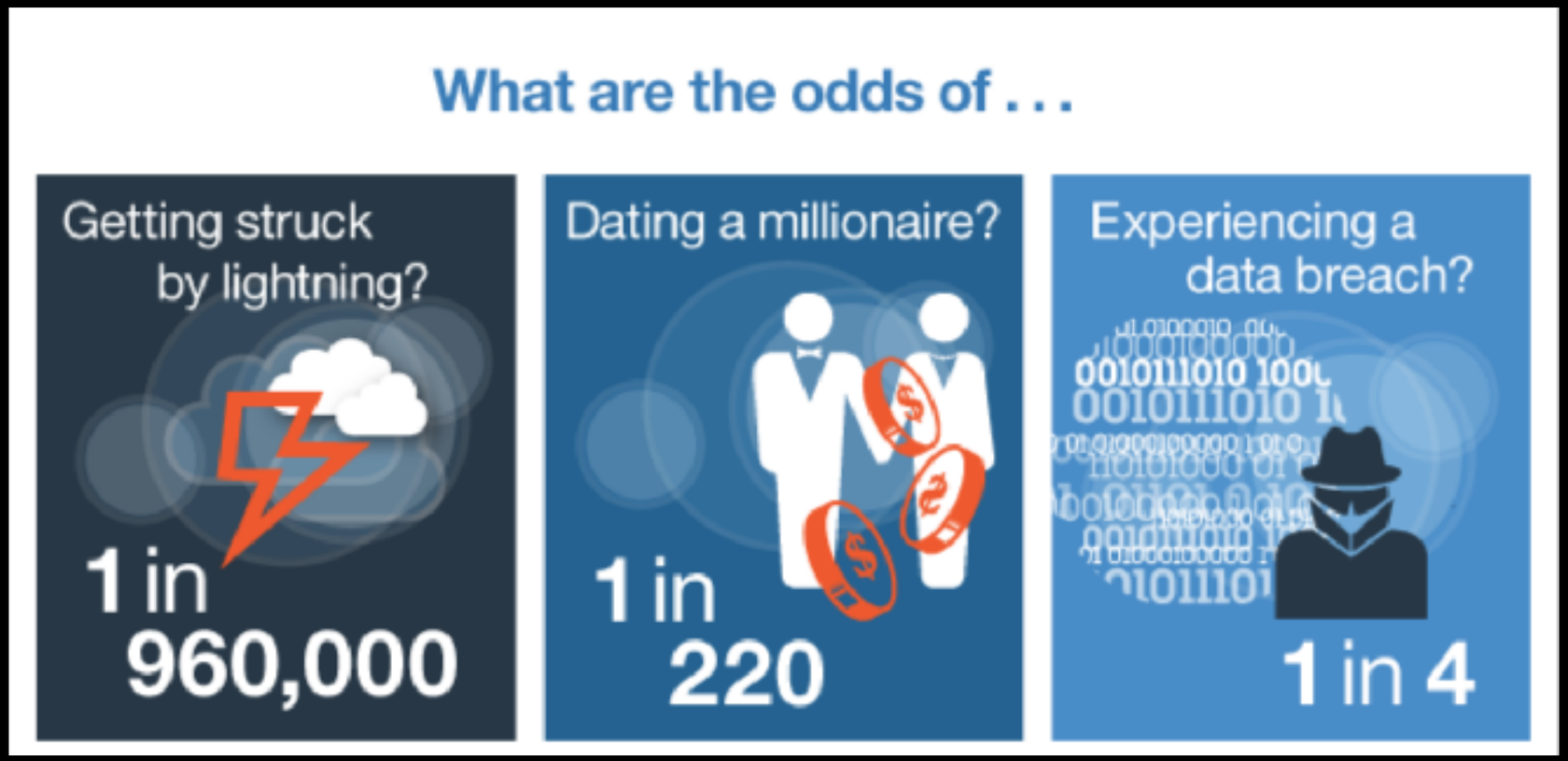
Securing Ansible

with your host, Ryan Prior
software engineer at CyberArk, Conjur team

bird site: @ryanprior
elephant site: @ryanprior@mastodon.social

A Breach Costs About $4 Million Big Ones



Source: Ponemon Institute “Cost of a Data Breach” via IBM

Security Goals

- Build on a chain of trust
- Authenticate all requests
- Authorize w/ least amount of privilege
- Audit everything

- ...and do it with code!

Human Identity

Conjur Concepts

- Machine Identity
- Establishing the unique identity of non-human actors in a network in order to control access privileges.
- Authentication
- Establishing that the requestor of access is who/what it claims to be – machine or human.
- Declarative Policies
- “Security Schema” – human readable, machine parseable documents describing the principals, resources and privileges for a deployable unit, e.g. application, tool, development team, etc.
- Secrets Injection
- Retrieving secrets on behalf of a process such that they are available to the process when/as needed, and disappear as soon as the process exits.

Security As Code

- Ansible automation is declarative
    - Playbook describes desired state (*the what*)
    - Ansible configures/remediates to that state (*the how*)
- Security tools need to follow suit w/ declarative security policies
- This has multiple benefits:
    - Versioned, like source code
    - Collaborative
    - Proactive security, not just auditing and forensics after-the-fact
    - Active Directory as a cautionary tale
    - Automated audit/compliance workflows
    - Determine if current state aligns with desired state (or not)
    - Ensures consistency across teams, environments and domains

Resources

- conjur.org - Conjur information portal
- https://www.conjur.org/integrations/ansible.html
  Ansible integration information
- https://github.com/cyberark/ansible-role-conjur
  Open source repository for Ansible integration