

Evolution of LinkedIn SRE & How Catalyzers Shaped It

SREcon18 Asia



Isha Ganeriwal

TECHNICAL PROGRAM MANAGER



TPM @ SRECon

Relevance & importance of Non-SREs in
the world of SREs

Looking back... 2010



LinkedIn Operations

2010



- Classical, stratified model: Systems, Networks, Applications, DBA
- Heavy-weight processes driven by tickets and heroes
- Culture of not trusting developers in any deployed environments
- Huge wall and growing frustration between Dev and Ops teams (and in ops itself)
- 7 engineers in total made up NOC, SRE, Release Operations: "Site Operations"
- On-call was horrible

Is the Site Up?

2010



LinkedIn Will Be Back Soon

LinkedIn is currently unavailable while we make upgrades to improve our service to you. We'll return around 8:30pm (PT).

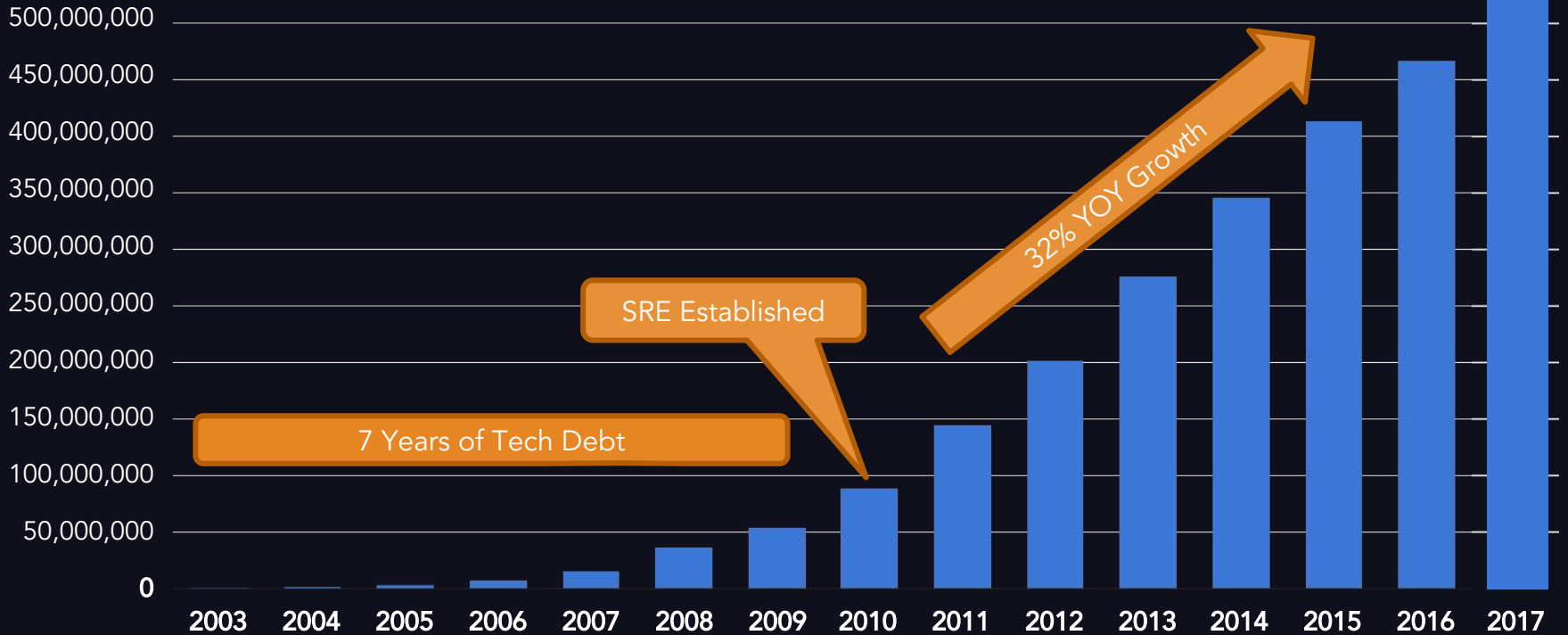
We apologize for the inconvenience and appreciate your patience. Thank you for using LinkedIn!



- Peak traffic periods Mon-Wed ~ 8am
- Regular capacity related outages Mon-Wed ~ 8am
- Zero tolerance for failure in the application stack
- Near zero instrumentation
- Bi-weekly downtime maintenances

Member Growth

■ # Members



What is SRE @ LinkedIn?



Core Principles



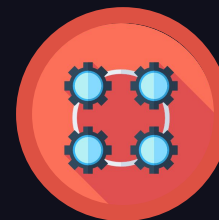
1

Site Up



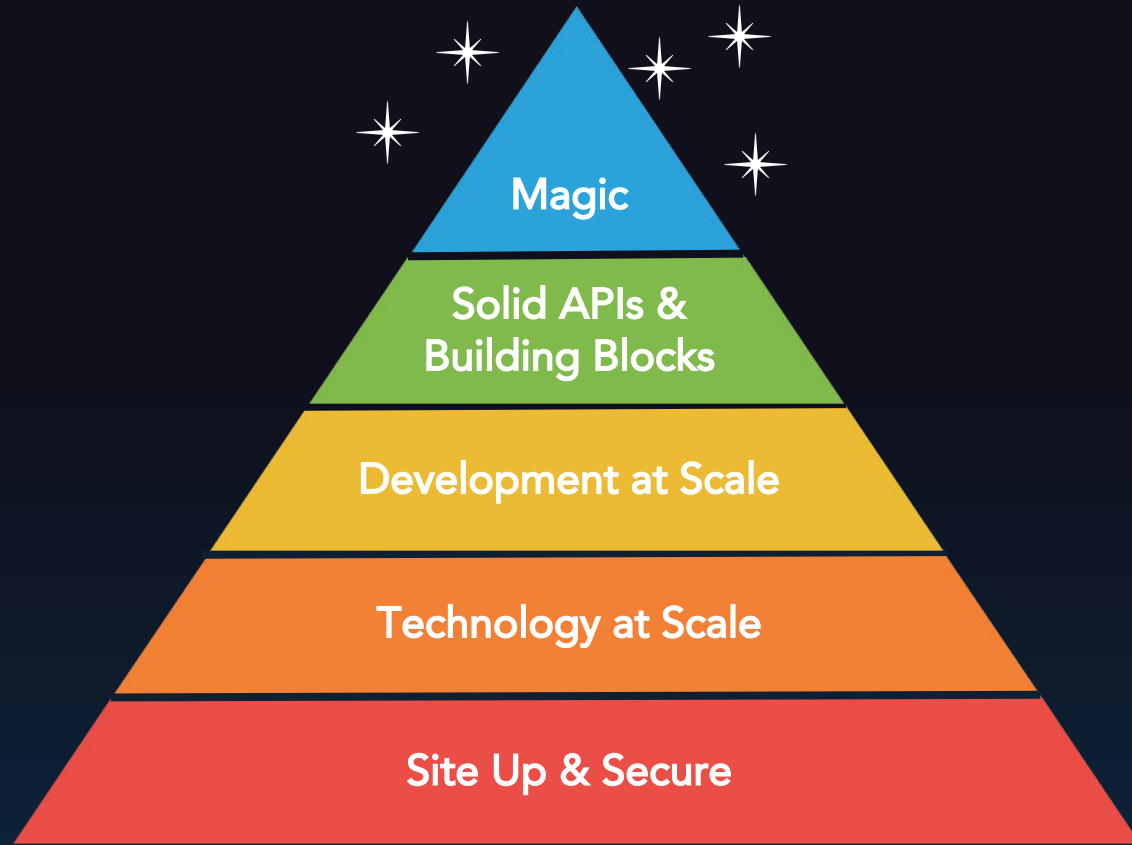
2

Empower Developer
Ownership



3

Operations is an
Engineering Problem



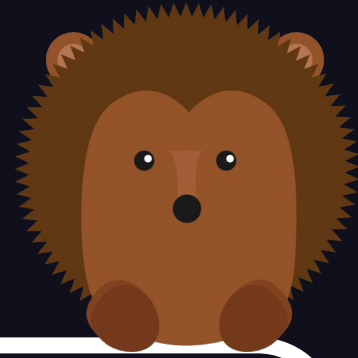
LinkedIn's Engineering Hierarchy of Needs

High Level



Composed of Software, Database, and Infrastructure Engineering generalists that make LinkedIn work

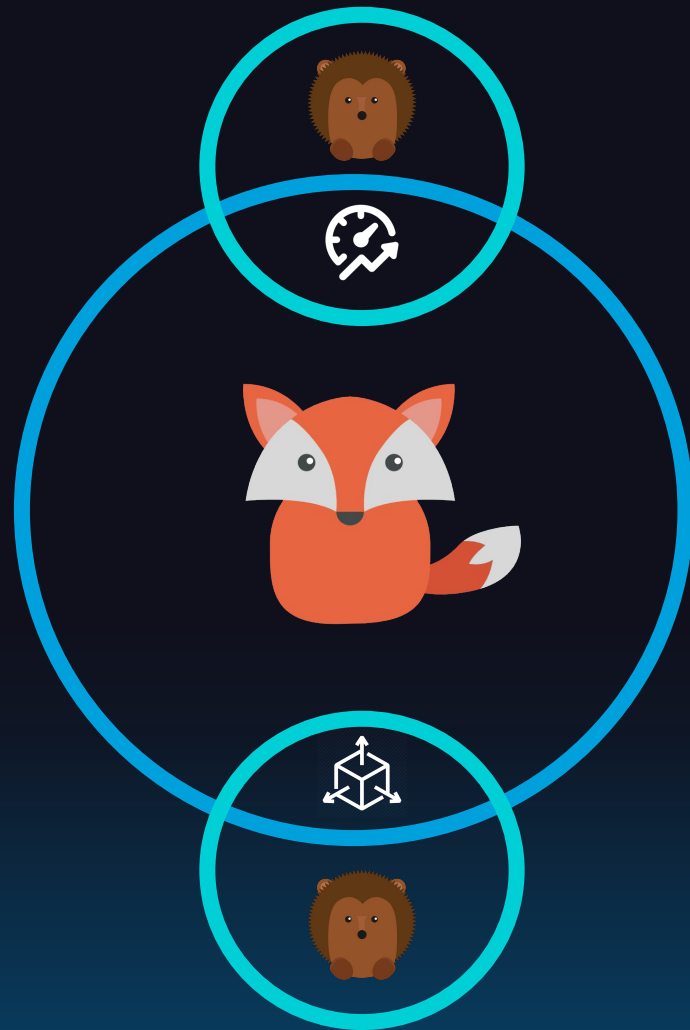
Generalists you say?



"the **fox** knows many things,
but the **hedgehog** knows one
big thing."

-- Archilochus, *Greek Poet*





"Expect the best, plan for the worst,
and prepare to be surprised."

Catalyzers: Technical Program Managers



Catalyzers: Technical Program Managers



Catalyzers: Technical Program Managers



1

Partners & Leaders in your organization



2

Execute Right & Execute what is Right



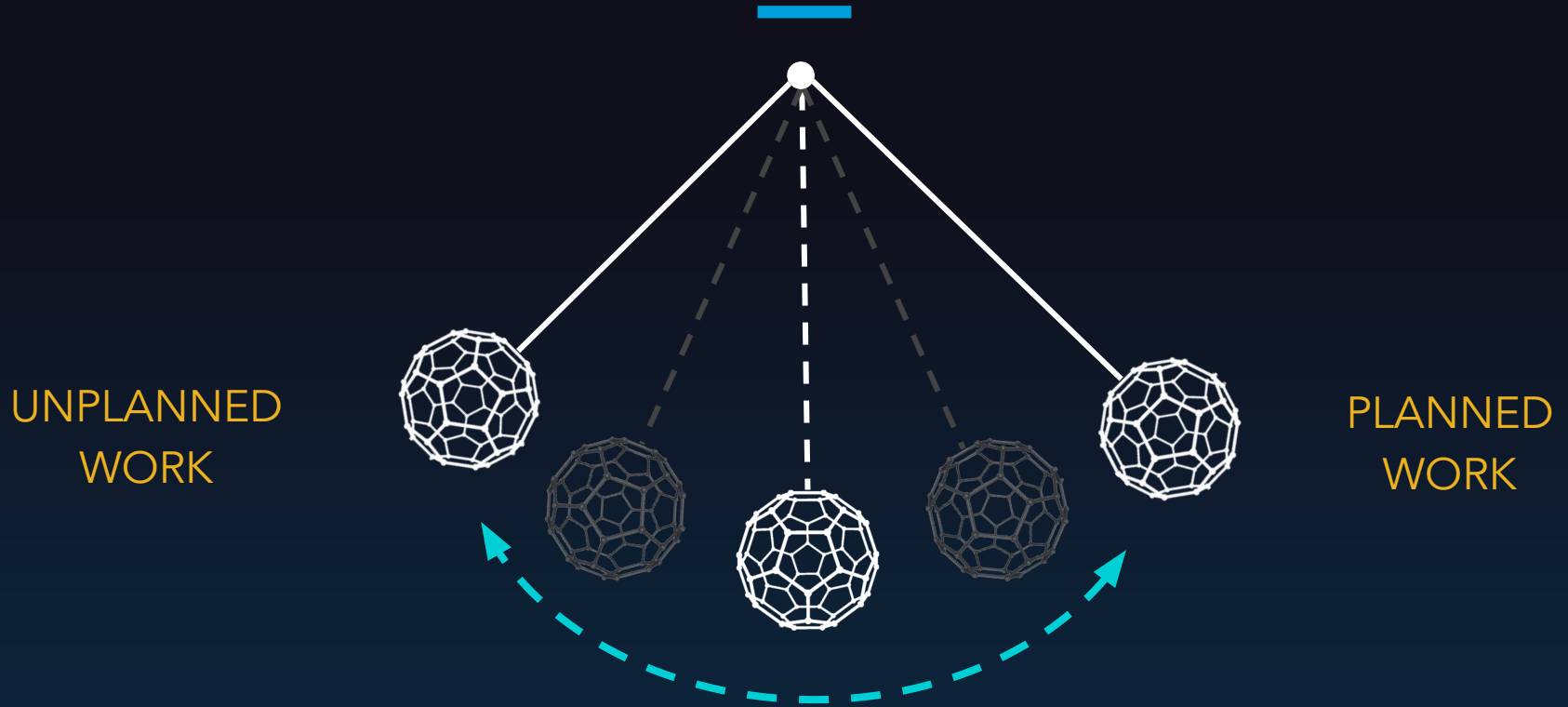
3

Metric Oriented



Poll Upcoming: Go to www.slido.com - Code: #X563

Unplanned VS Planned



Let's answer a few common SRE questions!



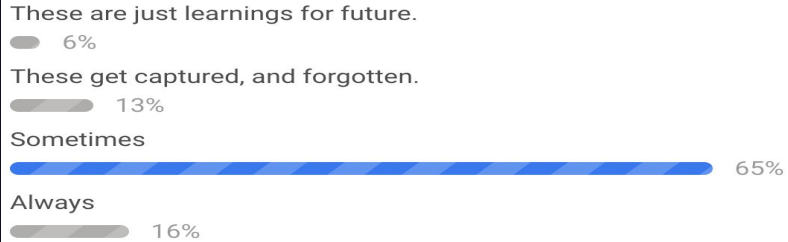
<https://wall2.sli.do/event/vdlvm7hl>

Vote at www.slido.com

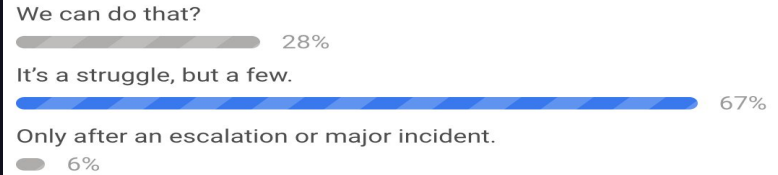
1. Enter event code - #X563/ SRECon APAC 2018
2. Answer **Polls**

Poll Results

How often does your input on a project result in an action item? 0 3 1



How many of your ideas to improve the operability or resiliency of your services turn into projects on your roadmaps? 0 1 8



How much of your time is spent on interrupt driven work? 0 0 4



Data Points TPMs Capture

LEVEL 1

- The number and severity of incidents over a month.
- The availability of your services
- Growth projections of your services / Capacity planning

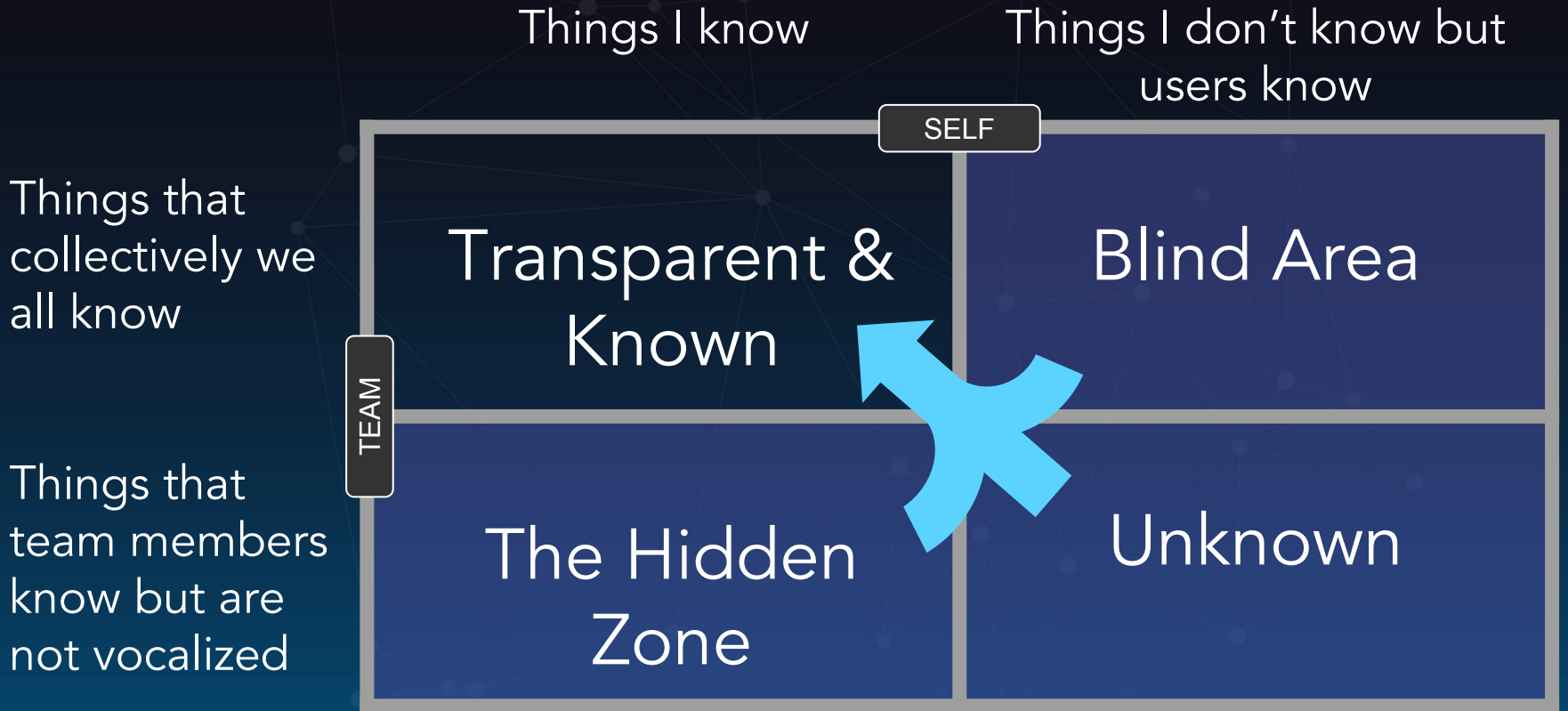
LEVEL 2

- How does your Oncall data look?
- How much of work is planned vs unplanned?

LEVEL 3

- How do you feel about your craftsmanship, how about your partners?
- How do you feel about the relationship with your partners?

The BLINDSPOTS



The FEEDBACK Loop

Iterative Feedback
Loop for course
correction








Fast Feedback
Loops



Process / Project / Tool



The 5 STEP Plan

-  1 Approach the Feedback
-  2 Remove the Facade
-  3 Isolate and Triage Issue
-  4 Know your audience
-  5 Roll with the solution

Overall Takeaways for SRE



1

Keep calm & trust your
TPM



2

What gets measured gets
fixed



3

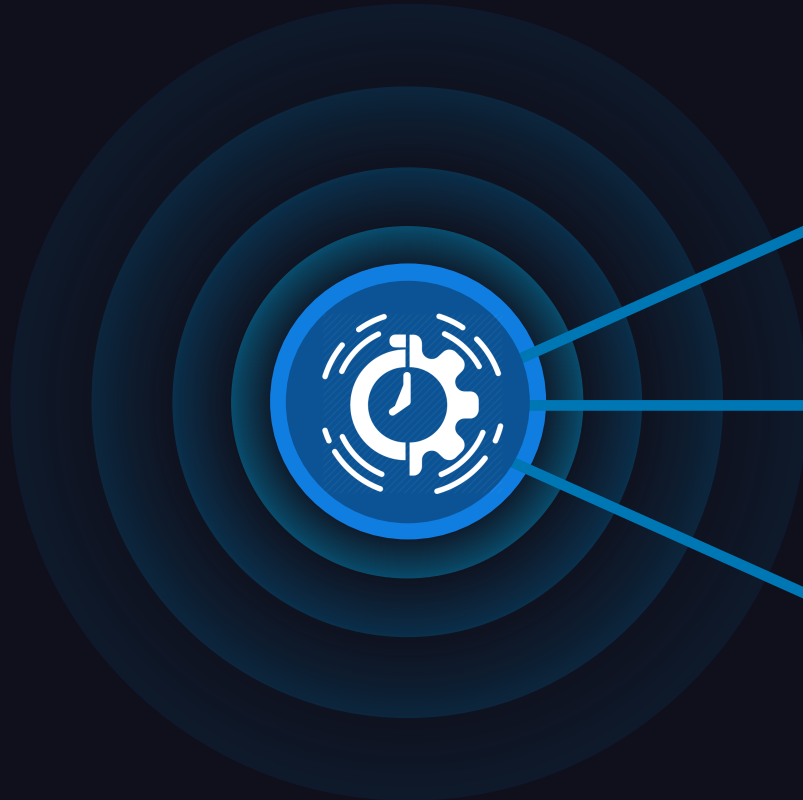
If you're not a part of the
solution, you're a part of the
problem

Catalyzers: Security



“Changes in production applications are happening at a greater rate than ever before. New product ideas can be visualized in the morning and implemented in code in the afternoon.”

Innovation and Rate Of Change



"Trust but Verify"

- Security to follow SRE "trust but verify" approach towards engineering partners



Embrace the Error Budget

- Self Healing & Auto Remediation
- Reduction of Manual Process



Inject Engineering Discipline

- Review when architecture changes reach a certain complexity point.

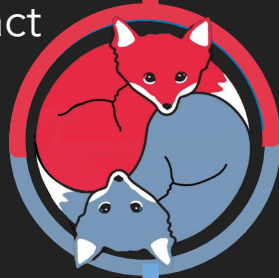
“Microservice architectures are exploding to meet scalability requirements”

Microservice Architecture

SECURITY CHALLENGES ARE SIMILAR TO SRE

SRE Challenges

- Latency & Performance Impact
- Cascading Failure Scenarios
- Service Discovery



Security Challenges

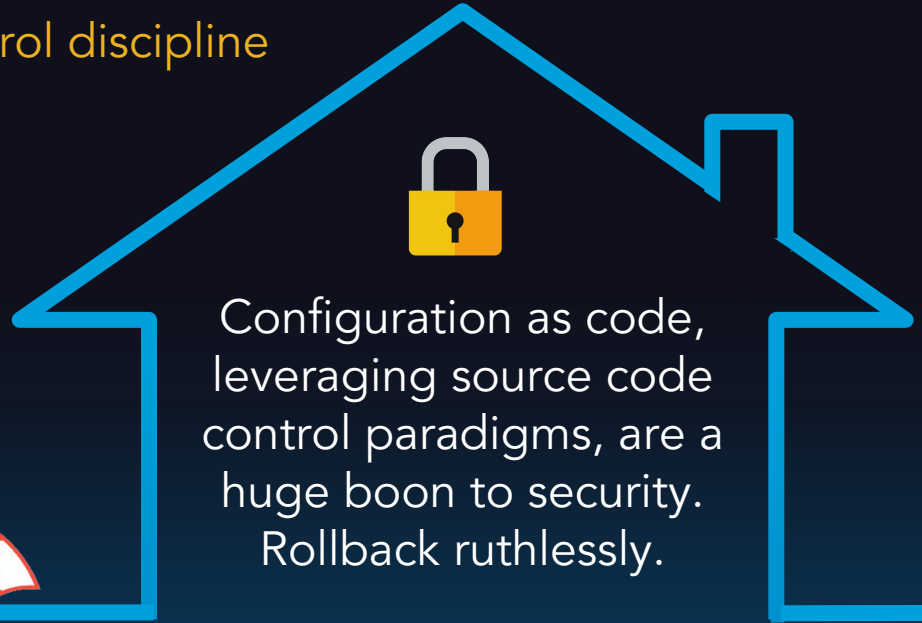
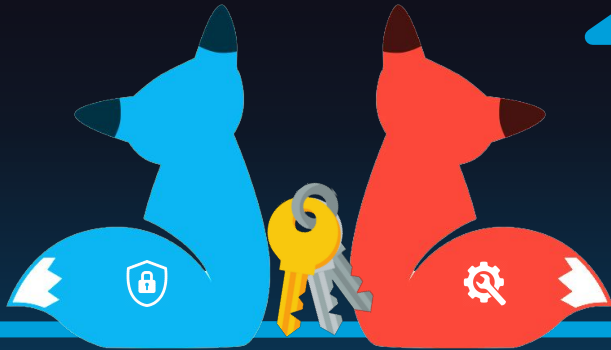
- Authentication
- Authorization
- Access Control Logic

Catalyzers: Security @ SRE



Production Access & Change Control

- Start with a known-good state
- Asset management and change control discipline
- Ensure visibility
- Validate consistently and constantly



Configuration as code, leveraging source code control paradigms, are a huge boon to security. Rollback ruthlessly.

Overall Lessons for SRE



1

Remove single points of security failure like you do for availability



2

Assume that an attacker can be anywhere in your system or flow



3

Capture and measure meaningful security telemetry

Giveaways



Align with your catalyzers, and let them help you.



TPMs and Security can help you reduce your tech debt.



Measure your data, and isolate the issues.



Failing to plan is planning to fail.



Generalists will always need specialists and vice versa -
That's how we grow together.

Q&A



iganeriwal@linkedin.com