

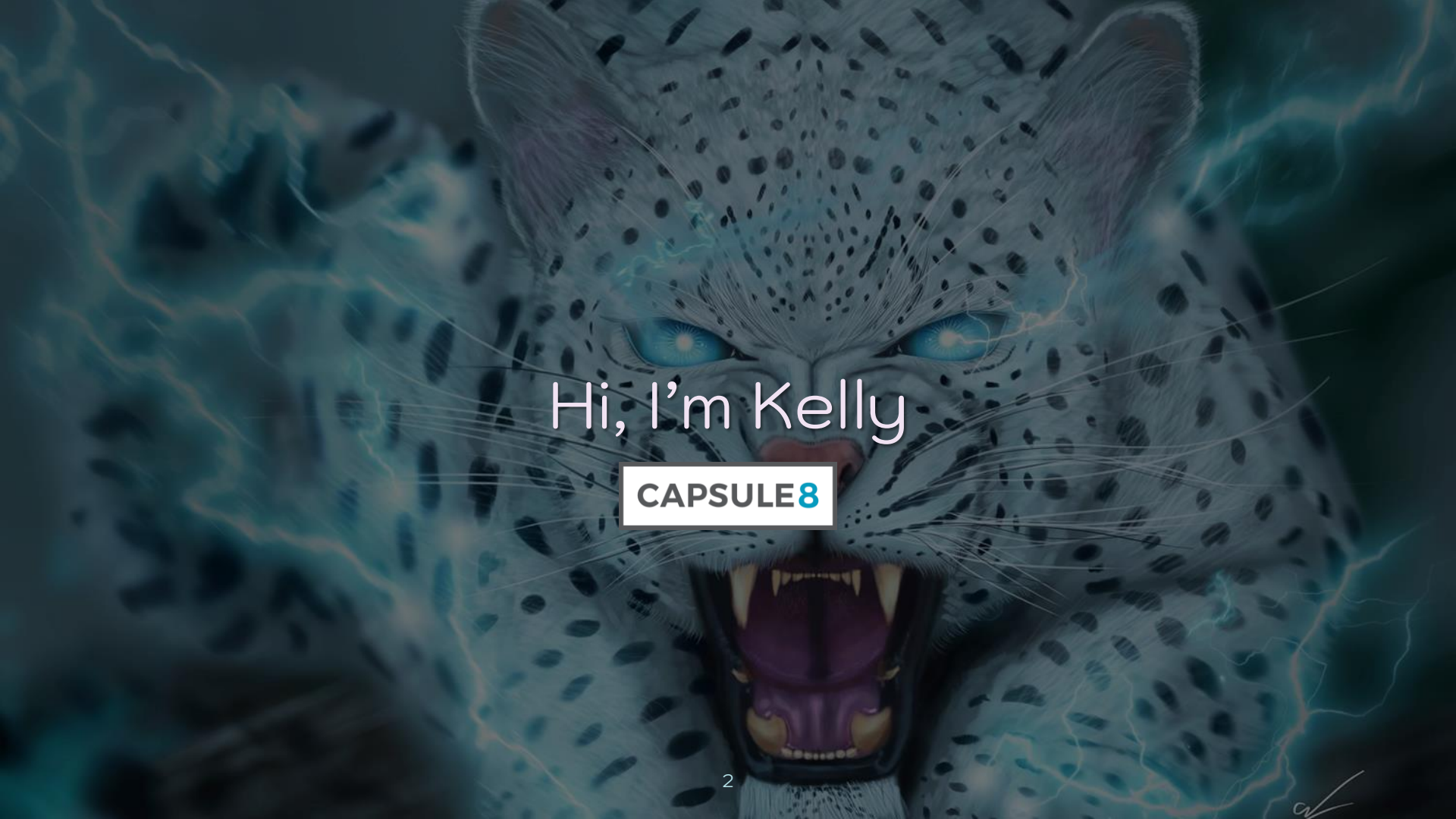


CONTROLLED CHAOS

The Inevitable Marriage of DevOps & Security

Kelly Shortridge (@swagitda_)

Velocity Berlin 2019



Hi, I'm Kelly

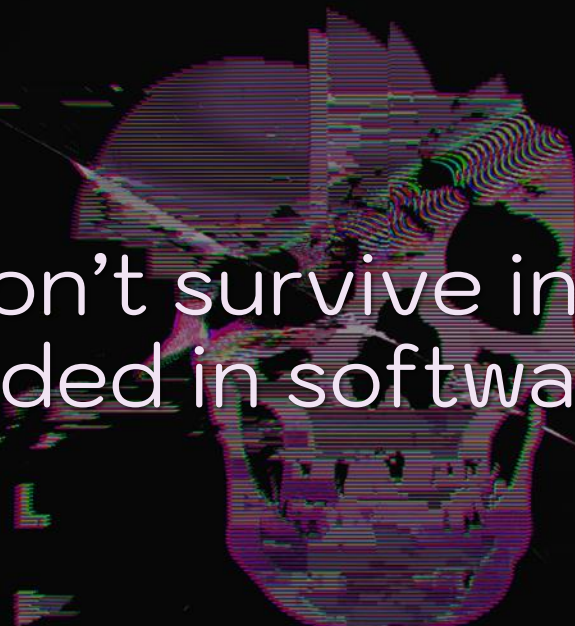
CAPSULE8



“Chaos isn’t a pit. Chaos is a ladder.”
— Petyr Baelish, *Game of Thrones*

Infosec has a choice: marry DevOps
or be rendered impotent & irrelevant





Don't survive in
ded in softwa

DevOps can learn to carve its own
path to secure software delivery



A hand holding a glowing lightbulb with concentric heart outlines. The background is dark, and the lightbulb is the central focus, with its glow illuminating the surrounding heart shapes. The heart outlines are composed of multiple concentric lines, creating a sense of depth and focus on the lightbulb.

How can controlling chaos create a marriage of infosec and DevOps?

1. Chaos Theory
2. Time to D.I.E.
3. A Phoenix Rises



Chaos Theory



Chaos engineering = continual
experimentation to test resilience

“Things will fail” naturally extends
into “things will be pwned”



HURT ME

Security failure is when security controls don't operate as intended

A dramatic night scene of a volcanic eruption. A large, dark plume of smoke and ash rises from a volcano, illuminated from below by the intense red and orange glow of the lava flow. A bright yellow lightning bolt strikes the lava flow, creating a sharp contrast with the dark surroundings. The sky is dark and filled with stars.

What are the principles of chaotic security engineering?

1. Expect that security controls will fail & prepare accordingly

2. Don't try to avoid incidents – hone your ability to respond to them

Game days: like planned firedrills



Prioritize security game days based on potential business impacts



Decision trees: start at target asset,
work back to easiest attacker paths

Determine the attacker's least-cost path (hint: it doesn't involve 0day)



Your goal is to raise the cost of
attack, ideally beginning at design

Time to D.I.E.

We need a model promoting qualities
that make systems more secure

Enter the D.I.E. model by Sounil Yu:
Distributed, Immutable, Ephemeral

An abstract background featuring a complex, interlocking geometric pattern of cubes or hexagons. The pattern is rendered in shades of blue, purple, and teal, creating a sense of depth and movement. The central area is dark, providing a high-contrast space for the text.

Distributed: multiple systems
supporting the same overarching goal

Distributed infrastructure reduces
risk of DoS attacks by design

A service mesh is like an on-demand VPN at the application level



Attackers are forced to escalate
privileges to access the iptables layer



Immutable: infrastructure that
doesn't change after it's deployed

The background is a dark, teal-colored abstract image. It features a grid of glowing blue lines that resemble circuit traces or data paths. Scattered throughout the image are numerous small, glowing red and pink dots, some of which are arranged in faint, larger-scale patterns. The overall effect is a high-tech, digital aesthetic.

Immutable infra is more secure by
design – ban shell access entirely

Patching is no longer a nightmare
with version-controlled images



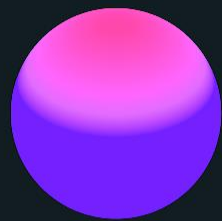
Ephemeral: infrastructure with a very short lifespan (dies after a task)

Ephemerality creates uncertainty for attackers (persistence = nightmare)

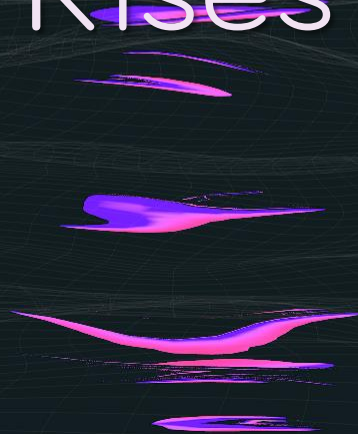


Installing a rootkit on a resource that
dies in minutes is a waste of effort

Optimizing for D.I.E. reduces risk by
design & supports resilience



A Phoenix Rises



Begin with “dumb” testing before moving to “fancy” testing

D.I.E.ing is an art, like everything else



The background of the slide is a dark blue field filled with a complex, interconnected network of thin, glowing purple lines. These lines connect numerous small, bright purple dots, creating a web-like structure that suggests a distributed system or a complex network. The lines and dots vary in brightness, with some appearing more prominent than others.

Controlling Chaos: Distributed



Distributed is mostly covered by the
existing repertoire of chaos eng tools

Repurpose these tools, but make
attackers the source of failure



Multi-region services present a fun
opportunity to mess with attackers

Shuffle IP blocks regularly to change
attackers' lateral movement game



Test: inject failure into your service mesh to test authentication controls

A glowing pink square frame is centered in the image, set against a dark, snowy landscape at night. The frame's light reflects on a body of water in the foreground. The background shows silhouettes of trees and a dark sky.

Controlling Chaos: Immutable

Immutable infra is like a phoenix – it disappears & comes back a lot



Volatile environments with continually moving parts raise the cost of attack

Create rules like, “If there’s ever a write to disk, crash the node”



Attackers must stay in-memory,
which hopefully makes them cry

Bonus: disallowing all local IO
improves service reliability

Metasploit Meterpreter + webshell:
Touch passwords.txt & kaboom

The background of the slide is a dark, atmospheric image. It features glowing, ethereal lines in shades of blue and purple that swirl and drift across the frame, resembling smoke or light trails. In the center, there is a dark, silhouetted figure of a person, possibly a dancer or a person in motion, which adds a sense of mystery and movement to the overall composition.

Build your Docker images with a
garbage-filled “bamboozle layer”

Mark garbage files as “unreadable” to craft enticing bait for attackers



A potential goal: architect
immutability turtles all the way down

Test: inject attempts at writing to disk to ensure detection & reversion



Treat changes to disk by adversaries
similarly to failing disks: mercy kill



Controlling Chaos: Ephemeral

Most infosec bugs are stated-related
– get rid of state, get rid of bugs



Reverse uptime: longer host uptime
adds greater security risk

Test: change API tokens & test if services still accept old tokens

A statue of a man in a toga, possibly a Roman figure, standing on a grid floor against a purple background with mountains. The statue is positioned in the center of the frame, facing slightly to the right. The background features a purple sky and dark, silhouetted mountains. The floor is a grid of lines that recede into the distance.

Test: retrograde libraries, containers,
other resources in CI/CD pipelines

Test: inject hashes of old pieces of data to ensure no data persistence

Leverage lessons from toll fraud –
cloud billing becomes security signal



Test: exfil TBs or run a cryptominer
to inform billing spike detection



Conclusion

The background is a deep purple with a black geometric pattern of triangles meeting at the center. A white square frame is centered on the page, containing the text.

Chaos/resilience are natural homes
for infosec & represent its future.



The future of infosec involves unified responsibility & accountability.

Security can be innovative and fuel
the engine of business as well.





“You must have chaos within you to
give birth to a dancing star.”

— Friedrich Nietzsche



@swagitda_



/in/kellyshortridge



kelly@greywire.net