# 2019 ReInvent - Werner Vogels Keynote

There's no compression algorithm for experience

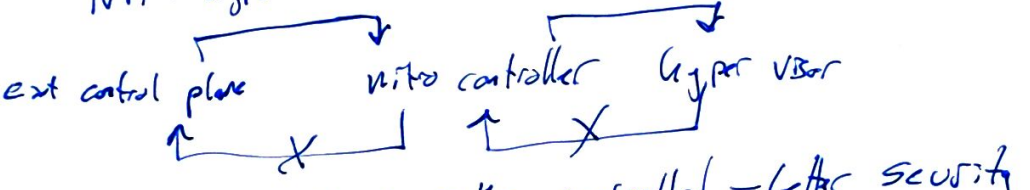
Virtualization
→ complexity, overhead, virtualization tax

classic problem: network latency per guest
How to fix? Rethink...
⇒ AWS Nitro

Trad: Monolith on Big hypervisor
Micro services, inform hardware components
APIs to virtual hardware

Nitro - solve network, moved network to sep card (2013)
- step 2 c4 - more ebs processing to sep card
- step 3 c5 - local storage i/o offloaded to card
- step 4 Nitro control plane on sep card
New hypervisor

Trust no one: kill Dom0
No Dom0 in Nitro hypervisor = No SSH




comm paths controlled - better security

Trust no one: encrypt everything: Nitro encrypts everything
Trust no one: guests not trusted: no - nonvolitile memory
needs

Nitro: base for instance types
- New updates
- new in nano
- bare metal
- outposts

* Nitro enclaves - partition off parts of memory

Firecracker - Micro VM for containers & serverless
Fargate - more responsive scaling than EKS ECS
instance spin up / spin down

Micro VM enables fargate

Fargate dataplane re-implementing using Nitro

2019 re:Invent Werner Vogels Keynote

Lambda – runs on firecracker

~35 minutes on : Vanguard

Cost to compute 30%

Cost to build 30% + Better resiliency

Deployment freq 20%

Evolvable Architecture

Amazon S3 – ~~2500~~ 262 μ services

Everything Always Fails

Reduce the blast radius

Zonal cell-based

cell size – smaller blast radius

is easier to op / more cost efficient larger cells

Physalia Millions of tiny databases

EBS cell based

Coloring of micro cells – supports use partition key

Cell – 7 nodes

running Paxos

Place cell as close to the client as possible

What if your application is stateless or soft~~state~~?

shuffle sharding Blast radius $\binom{n}{k}$ $\frac{n!}{k!(n-k)!}$

Need smart client, knows how to do retry

& Amazon Builder's Library

aws.amazon.com / builders-library

AWS Series – No Go Build

aws.amazon.com/startups/NoGoBuild