

RED SITE  
Comics



#1

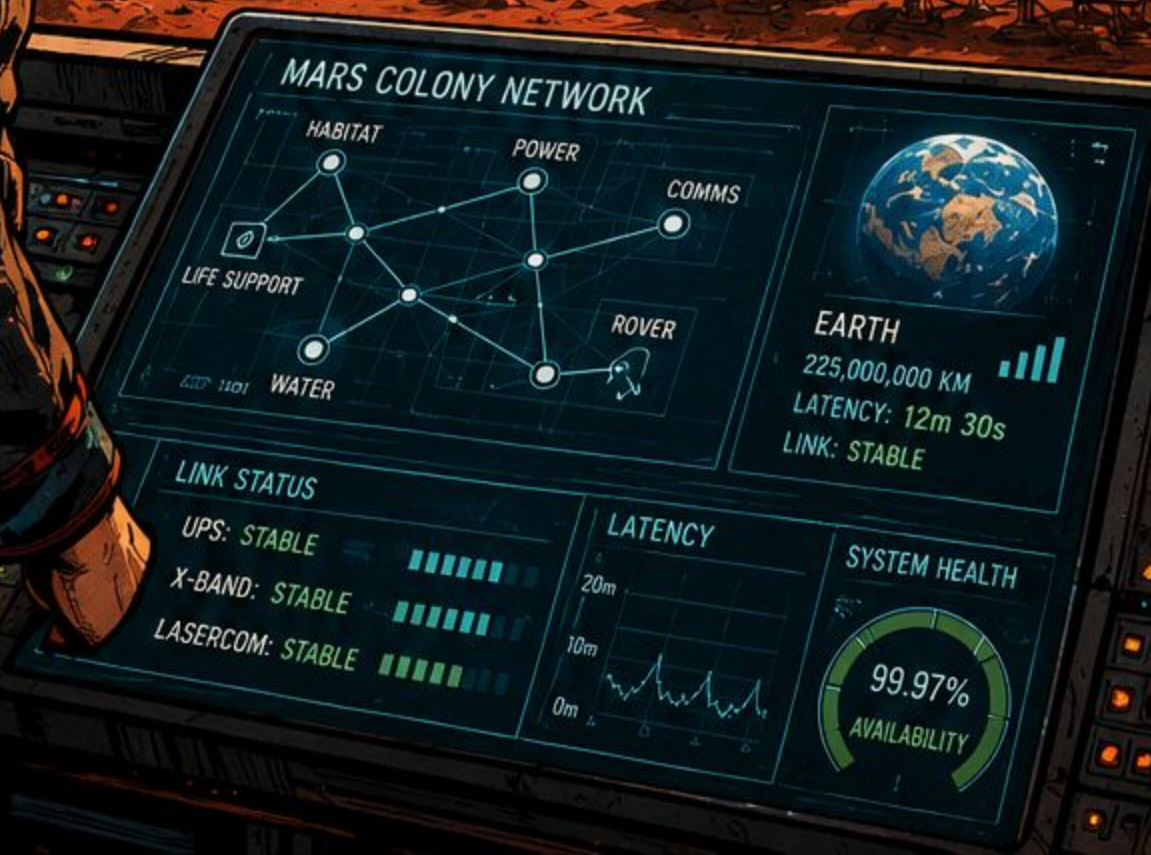
MARS COLONY  
ENGINEER

# RED SITE RELIABILITY

KEEPING THE LIGHTS ON,  
225 MILLION KM FROM HOME



RELIABILITY  
— IS —  
SURVIVAL



# EPISODE 0 — RED SITE

## Part 1 of 2

1

A young, brilliant engineer is sent to Mars to build and run the IT infrastructure for humanity's first permanent colony.



2

### ON EARTH:

A bad deploy means an apology email.



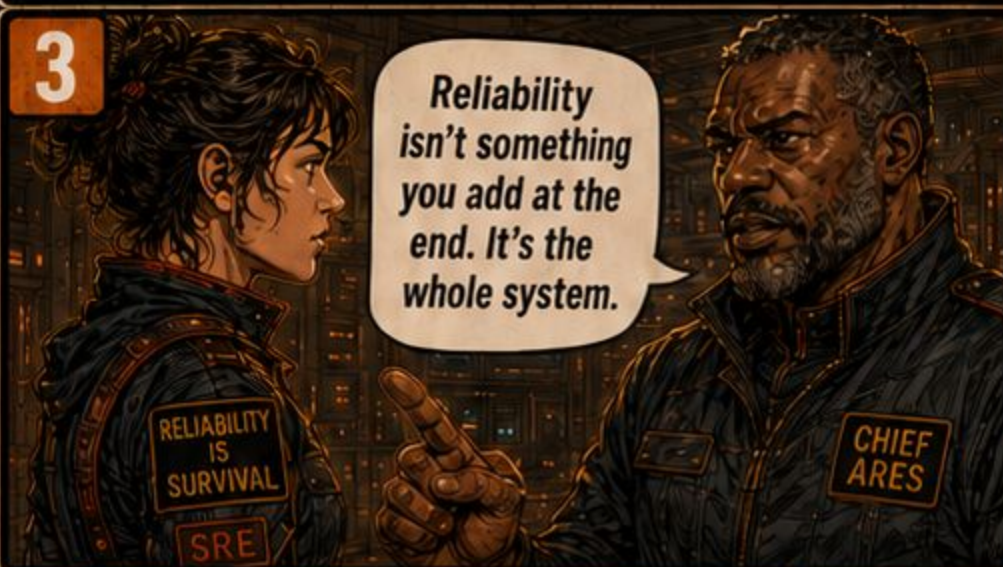
### ON MARS:

A bad deploy can make life support go dark.



3

Reliability isn't something you add at the end. It's the whole system.



4



This story teaches reliability from the concrete under the rack to the SLA signed with Earth.

5

Stakeholders demand uptime.



Physics doesn't negotiate.



Mars is the real antagonist.



12 EPISODES +  
PROLOGUE + EPILOGUE.



ADVENTURE +  
GENTLE HUMOR.



TECHNICALLY  
HONEST.

SWIPE FOR PART 2 →

MEET THE CREW

# EPISODE 0 — RED SITE

Part 2 of 2

## MEET THE CREW



### DANA REED

Brilliant new engineer.  
Great code, no ops  
scars yet.



### CHIEF ARES

Veteran SRE mentor.  
Teaches through  
war stories.



### ROVER

AIOps assistant.  
Comic relief and  
glossary voice.



### COMMANDER PHOBOS

The stakeholder.  
Wants 100% uptime.



### THE STORM

Mars itself.  
Dust, delay, radiation.  
Physics doesn't  
negotiate.

## WHAT YOU'LL LEARN



### MEASURE & PROMISE

SLI, SLO, SLA,  
ERROR BUDGET,  
RSL



### FAILURE & RECOVERY

RPO, RTO, MTTR,  
MTBF, DR,  
N+1, 2N



### TOOLS & PRACTICE

MONITORING,  
OBSERVABILITY, SIEM,  
GITOPS, CHAOS  
ENGINEERING

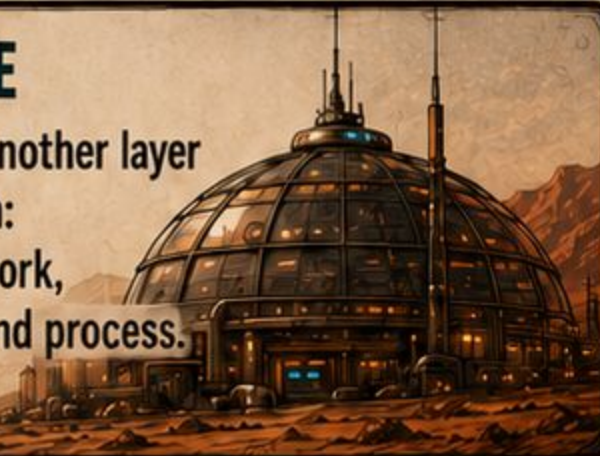


### CULTURE & PROCESS

TOIL, RCA,  
BLAMELESS  
POSTMORTEM,  
DEFENSE IN DEPTH,  
COMPLIANCE

## THE BIG PICTURE

Every episode reveals another layer  
of infrastructure design:  
facility, hardware, network,  
platform, application, and process.



## WHAT DANA DIDN'T KNOW YET...

Every layer has its own  
failure modes.  
Every component will  
eventually fail.



**NEXT: PROLOGUE — THE 14-MINUTE DELAY →**

# EPISODE 1 – YOU CAN'T PROMISE 100%

## Part 1 of 2

Welcome.  
I need 100%  
uptime on life  
support systems.  
Sign here.

SERVICE AGREEMENT  
SYSTEM: LIFE SUPPORT  
REQUIRED UPTIME:  
**100.00%**  
SIGNATURE: \_\_\_\_\_

RELIABILITY  
IS  
SURVIVAL  
SRE

Stop.  
No system  
promises  
100%.

CHIEF  
ARES

RELIABILITY  
IS  
SURVIVAL

Affirmative.  
Reality detected.



### DOWNTIME PER YEAR (approx.)

**99.9%** = ~8.7 hours/year

**99.99%** = ~52 minutes/year

**99.999%** = ~5 minutes/year

EACH ADDED NINE =  
~10x LESS ACCEPTABLE DOWNTIME

EACH EXTRA NINE  
USUALLY COSTS MUCH MORE.

RELIABILITY  
IS  
SURVIVAL  
SRE



**SLI** = what we measure



**SLO** = internal target



**SLA** = promise with  
consequences



**ERROR  
BUDGET** =  
allowed unreliability



**BUDGET SPENT =  
FEATURE FREEZE**

If the error  
budget is gone,  
we slow releases  
and fix stability  
first.

Swipe for Part 2 →

# EPISODE 1 – YOU CAN'T PROMISE 100%

Part 2 of 2

## LIFE SUPPORT — 99.999% REDUNDANT HARDWARE

- ✓ MULTIPLE POWER PATHS
- ✓ GEOGRAPHIC REDUNDANCY
- ✓ AUTO-FAILOVER
- ✓ RIGOROUS TESTING
- ✓ MONITORED 24/7

LIFE SUPPORT STATUS  
ALL SYSTEMS NOMINAL  
99.999%

## MOVIE STREAMING — 99% BEST EFFORT



Mars can survive a night without cat videos.

- ✓ SINGLE REGION
- ✓ BEST EFFORT NETWORK
- ✓ CAN TOLERATE OUTAGES
- ✓ NOT LIFE-CRITICAL

## ROVER GLOSSARY

- ⚙️ **SRE:** Site Reliability Engineering
- 🕒 **Uptime:** how much of the time a system stays available
- ★ **The nines:** reliability targets like 99.9%, 99.99%, 99.999%



100% UPTIME  
REQUIRES:  
∞ INFINITE BUDGET  
+  
REPEAL OF ENTROPY



Thanks, ROVER.

He's not wrong.

Great. We have targets. Now... where do we physically **PUT** the servers?

DATA CENTRE SHELL  
SITE: ALPHA-1  
BAY D1A

WHAT DANA DIDN'T KNOW:  
A DATA CENTER IS **MOSTLY NOT COMPUTERS.**

**NEXT: POWER, COOLING, FLOOR SPACE, FIRE SUPPRESSION.**

# EPISODE 2 — CONCRETE, COLD AIR & KILOWATTS

## Part 1 of 2

### 1 SITE IDEA: NEXT TO THE HAB.



### 2 CHIEF ARES VETOES.



### 3 THE BETTER SITE.

## LAVA TUBE DATACENTER



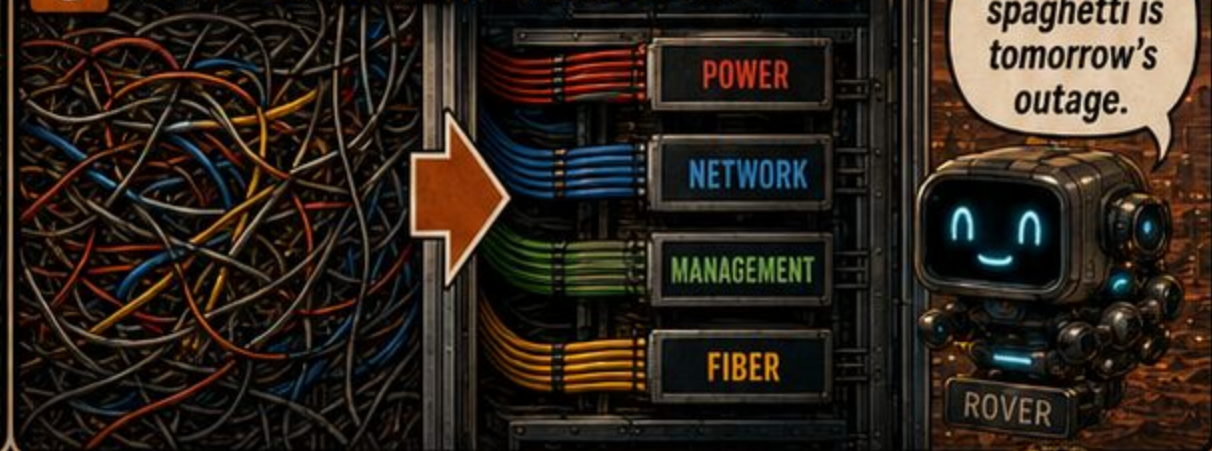
#### ON EARTH WE ALSO ASK:

- FLOODS
- EARTHQUAKES
- AIRPORTS
- GRID QUALITY

### 4 DESIGN FOR AIRFLOW



### 5 CABLES ARE INFRASTRUCTURE



### 6 PHYSICAL SECURITY MATTERS.

**PHYSICAL SECURITY:**  
ACCESS CONTROL,  
CAMERAS, MANTRAPS.

BADGE  
READER



CAMERA

MANTRAP  
VESTIBULE

The best  
firewall can't  
stop someone  
with a crowbar.

CHIEF  
ARES

**Swipe for Part 2 →**

# EPISODE 2 — CONCRETE, COLD AIR & KILOWATTS

## Part 2 of 2

### 1 POWER CHAIN

From source to server. Every link matters.



REDUNDANCY + DIVERSITY = SURVIVAL

CHIEF ARES

Redundancy keeps the lights on when the unexpected happens.

### 2 UPS: THE SHORT BRIDGE

Batteries bridge the gap.

Seconds matter.

UPS

STATUS

Normal



LOAD  
42%

UPS CARRIES THE LOAD UNTIL GENERATORS TAKE OVER.

### 3 HEAT IS THE ENEMY

If cooling dies, temperature rises fast.

COLD AIR SUPPLY

HOT AIR RETURN

COOLING PLANT

AIRFLOW IS PART OF RELIABILITY.

### 4 BREAKER TRIPPED!

Oops!

ALARM

UPS ON BATTERY

GENERATOR STARTING...

TIME ON BATTERY

00:30

UPS BEEPING



Oh no!

### 5 COUNTDOWN TO CRITICAL

THERMAL SHUTDOWN IN

00:27

INLET TEMP

31°C

RACK TEMP

47°C

TREND



This is why we design for failure.

Not great, Rover...

### 6 FAILOVER WORKS. MISSION CONTINUES.

The failover worked because we designed it.

GENERATORS ONLINE



POWER RESTORED



RACKS STAY ON



ALARMS NORMAL

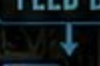


DUAL A/B POWER FEEDS

FEED A



FEED B



We built it right.

Phew!



WHAT DANA DIDN'T KNOW:  
EVERY SINGLE COMPONENT  
IN THIS ROOM WILL EVENTUALLY FAIL.



POWER



COOLING



NETWORK



STORAGE



HUMANS

THE SOLUTION?  
REDUNDANCY. DIVERSITY.  
OBSERVABILITY. PRACTICE.

NEXT UP: EPISODE 3 — OBSERVABILITY: SEE IT BEFORE IT HURTS →

# EPISODE 3 – EVERYTHING FAILS, ALL THE TIME

Part 1 of 2

A disk died!

**ALERT**  
**DISK FAILED**  
DRIVE-17  
STATUS: OFFLINE

Disks are consumables, like coffee filters.

Panic level: unnecessary.

## SPARES MATTER ON MARS.

On Earth, a spare part is 2 days away.  
Here, it's 9 months.  
Capacity planning includes shelves.

### SPARE PARTS — USE WISELY

DISKS

NICS

PSUS

CABLES

3 LEFT

2 LEFT

2 LEFT

4 LEFT



**MTBF =**  
MEAN TIME  
BETWEEN FAILURES

**MTTR =**  
MEAN TIME  
TO REPAIR



You can't prevent every failure. You can shrink recovery time.

**N**



JUST ENOUGH

**N+1**



ONE SPARE

**N+2**



TWO SPARES

**2N**



FULLY DUPLICATED

**MORE REDUNDANCY = MORE RESILIENCE + MORE COST.**

DATA CENTRE  
SHELL  
BAY D1A

ROVER

CHIEF  
ARES

Now let's hunt the real danger: hidden single points of failure.



# SWIPE FOR PART 2



# EPISODE 3 — EVERYTHING FAILS, ALL THE TIME

Part 2 of 2

## 1. PROTECT THE DATA.

### RAID + HOT SPARE



One disk can fail without data loss.

### ERASURE CODING



Data + parity spread across drives or nodes.

## 2. FAILURE DOMAINS.



DISK



SERVER



RACK



ROOM



SITE

Contain the blast radius.

Small failures should stay small.

## 3. HUNT THE SPOF (Single Point of Failure).

### SINGLE SWITCH



### SINGLE UPLINK



### SINGLE POWER FEED



### SINGLE COOLING LOOP



I found a suspicious cable.



## 4. PROTECT BY DESIGN.

Build it redundant. Sleep a little better.

### DUAL POWER SUPPLIES



### DUAL NICs



### RAID WITH HOT SPARE



### N+1 COOLING



### WHAT DANA DIDN'T KNOW:

Redundant servers are useless if there's only one cable between them.



**NEXT: NETWORKS, PATHS, AND WHY ONE CABLE CAN RUIN EVERYTHING.**

# EPISODE 4 – THE ROADS BETWEEN MACHINES

Part 1 of 2

## 1. THE PROBLEM: NO REACHABILITY.



The greenhouse can't reach the water-control system.

## 2. THE ROOT CAUSE: A SINGLE POINT OF FAILURE.

ONE SWITCH. ONE CABLE. ONE PATH.



**THIS IS A SINGLE POINT OF FAILURE.**  
When it breaks, everything stops.



One switch, one cable, one path. Single Point of Failure.

## 3. THE REDESIGN: REDUNDANCY BUILDS RESILIENCE.

TWO PATHS. MULTIPLE CHOICES. KEEP TRAFFIC FLOWING.



Redundancy gives packets another way through.



**LACP / BONDING**

Bundle links.  
More bandwidth.  
No single link risk.



**SELF-HEALING ROUTING (OSPF / BGP)**

Finds new routes automatically when links fail.



Design for failure. Expect it. Survive it.

## 4. DIVERSE PATHS: NOT JUST REDUNDANT — DIVERSE.



Two tunnels. Two cables. Two chances to succeed.

If one tunnel collapses, packets take the other tunnel — literally.

## 5. THE MISSION: BUILD SYSTEMS THAT OUTLIVE THE UNEXPECTED.



RESILIENT NETWORKS KEEP LIFE SUPPORT, POWER, AND HOPE FLOWING.

**THAT'S OUR JOB.**



**SWIPE FOR PART 2** →

**DNS, LOAD BALANCERS, AND THE PRECIOUS EARTH UPLINK.**

# EPISODE 4 – THE ROADS BETWEEN MACHINES

Part 2 of 2

## 1. DNS: THE PHONE BOOK OF THE COLONY.

Services talk by name, not by numbers.  
It's always DNS — even on Mars.



Names fail,  
nothing  
finds  
anything.

It's  
always  
DNS.

## 2. LOAD BALANCER: SHARE THE WORK.

Health checks keep traffic flowing.

Don't  
overload the  
good ones.



Resilient  
by  
design.

## 3. LATENCY & BANDWIDTH: USE THE GOOD LINK WISELY.

MARS — LOCAL TRAFFIC



EARTH UPLINK



Cache what  
you can.  
Queue what  
can wait.

## 4. SELF-HEALING NETWORKS.

When a path dies, routing chooses another.



ADAPT.  
REROUTE.  
KEEP GOING.

We don't  
panic. We  
path around.

## 5. THE BOTTOM LINE.

- ✓ DNS makes names mean something.
- ✓ Load balancing spreads the pain.
- ✓ Use the link like it's gold.
- ✓ Self-healing keeps the colony alive.

The network  
doesn't just  
connect us.  
It keeps us  
alive.

**WHAT DANA DIDN'T KNOW:**  
HARDWARE AND CABLES ARE THE SKELETON —  
NOW IT NEEDS A **NERVOUS SYSTEM.**

**NEXT: OBSERVABILITY — SEEING THE WHOLE COLONY THINK.** ➔

# EPISODE 5 – THE COLONY RUNS FROM A GIT REPO

Part 1 of 3

## 1. HIDDEN LAPTOP. HUGE PROBLEM.



## 2. PLATFORM STANDARDIZATION WEEK.



### PLATFORM STANDARDIZATION WEEK.

Every team running services their own way creates **SNOWFLAKE SERVERS** and **OPERATIONAL CHAOS**.



Different configs. Different tools. Different failures.  
*Chaos at scale.*

## 3. CIS BENCHMARKS. GOLDEN IMAGES.



A CIS Benchmark is a checklist to harden an OS.

### GOLDEN IMAGE HARDENED OS



- ⚙️ **MINIMAL PACKAGES**  
Only what you need.
- 🔒 **LOCKED PERMISSIONS**  
Least privilege.
- 🔒 **SECURE DEFAULTS**  
Safe by default.
- 📋 **AUDITED SETTINGS**  
Traceable and reviewed.
- ⚙️ **DRIFT SCANNING**  
Detect. Report. Remediate.

- ✓ Consistent.
- ✓ Secure.
- ✓ Repeatable.
- ✓ Compliant.

## 4. PLATFORMS BEAT SNOWFLAKE SERVERS.



### SNOWFLAKE SERVER (ONE-OFF)

- ✗ Manual setup
- ✗ No standard
- ✗ Hard to scale
- ✗ Fragile
- ✗ Hard to secure

VS.

### STANDARDIZED PLATFORM (REPEATABLE)

#### VIRTUALIZATION (OpenStack)



#### CONTAINERS (Kubernetes)



- ✓ Automated
- ✓ Standardized
- ✓ Scales easily
- ✓ Resilient
- ✓ Secure by design

KUBERNETES / OPENSTACK PLATFORMS > UNMANAGED SNOWFLAKE SERVERS.

## 5. SELF-HEALING BEATS PANIC.

### POD CRASHES



pod: data-collector  
STATUS: CrashLoopBackOff



Uh oh!  
I'll fix it—



### KUBERNETES RESTARTS IT

#### RESTARTING...

New container provisioned



#### RUNNING

pod: data-collector  
STATUS: Running



Fixed before  
you finished  
panicking.  
That's the  
platform.



Kubernetes watches, restarts, and keeps things running.

Self-healing beats panic.



# SWIPE FOR PART 2

## INFRASTRUCTURE AS CODE AND PIPELINES.



# EPISODE 5 – THE COLONY RUNS FROM A GIT REPO

Part 2 of 3

## 1. INFRASTRUCTURE HAS A RECIPE.

The infrastructure has a recipe, not a legend.

CODE IN GIT

main.tf  
variables.tf  
network.tf  
deploy.yaml  
...

INFRASTRUCTURE



Infrastructure as Code =  
repeatable, reviewable,  
versioned, testable.

Everything the colony runs on is described in code.

## 2. IAC IN ACTION.

CODE (DECLARATIVE)



PROVISIONED INFRASTRUCTURE



Code builds VMs, networks, and config – automatically.

## 3. PIPELINES KEEP CHANGES SAFE.

Humans are inconsistent at 3 a.m. Pipelines aren't.

CI/CD PIPELINE



TEST



SECURITY SCAN



CANARY DEPLOY



METRICS CHECK



ROLLBACK IF UNHEALTHY



## 4. ERROR BUDGET CHANGES THE RULES.

PIPELINE BEHAVIOR ADAPTS TO ERROR BUDGET

BUDGET HEALTHY = SHIP FAST



- ✓ Faster approvals
- ✓ Smaller test sets
- ✓ More automation
- ✓ Move quickly, safely



BUDGET BURNED = TIGHTER GATES



- ✗ More approvals
- ✗ Wider test coverage
- ✗ Extra security scans
- ✗ Slow down to recover

Protect the colony first. Speed follows stability.

## 5. CHIEF ARES' LAW.

“If you did it by hand, you’ll do it wrong the second time. Automate it, review it, version it.”

Automate. Review. Version. Survive.



SWIPE FOR PART 3



GITOPS, FLUX,  
AND REBUILDING FROM GIT.



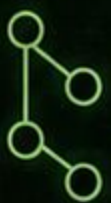
# EPISODE 5 – THE COLONY RUNS FROM A GIT REPO

Part 3 of 3

## 1. GIT IS THE SOURCE OF TRUTH.



### GIT REPOSITORY



- ✓ YAML
- ✓ Kustomize
- ✓ Helm
- ✓ Policies

### FLUX CD



Observe  
Diff  
Reconcile

### KUBERNETES CLUSTER COLONY PLATFORM



Git  
is the truth.  
The cluster is  
just the  
reflection.

The cluster continuously pulls desired state from Git  
and reconciles itself.



## 2. DRIFT GETS CORRECTED.

### SOMEONE MAKES A MANUAL CHANGE (SSH, KUBECTL EDIT, ETC.)



### FLUX DETECTS DRIFT



Detects difference  
from Git.



Reconciles back  
to desired state.

### DESIRED STATE FROM GIT



replicas: 1

Manual SSH  
changes are  
a code smell.



## 3. REBUILD WHILE EVERYONE EATS DINNER.

Wait... it  
just fixed  
itself?



### ! NODE FAILED



### NODE REBUILT FROM GIT



Self-healing in progress...

If the  
datacenter  
burns down,  
we git clone  
the colony.



## 4. WHY AUTOMATION MATTERS.



SPEED WITH  
SAFETY



CONSISTENCY



AUDITABILITY



LESS TOIL



EVERY CHANGE  
HAS AN AUTHOR

## 5.



WHAT DANA DIDN'T KNOW:  
SERVICES CAN BE REBUILT FROM CODE.  
**DATA CAN'T.**



NEXT: THE COST OF OBSERVABILITY. SEEING IS SURVIVING. →

# EPISODE 6 – DATA IS THE COLONY’S MEMORY

## Part 1 of 4

### 1 DANA HITS A BAD SCRIPT.



### 2 ARES EXPLAINS THE FAILED FIRST RESTORE ATTEMPT.



Replication is not backup.

Replication copies your mistakes at the speed of light.

CHIEF ARES

### 3 RECOVERY FROM A REAL SNAPSHOT.



### 4 COMMANDER PHOBOS ASKS: WHY WAS THAT ACCEPTABLE?

Why was four hours okay?

**RPO =**  
Recovery Point Objective  
How much data loss you can afford.

**AGREED RPO:**  
**4 HOURS**

✓ Accepted and budgeted.

CMDR PHOBOS

ROVER



**SWIPE FOR PART 2 → CEPH, ARRAYS, TAPE, AND TIERING**

# EPISODE 6 – DATA IS THE COLONY’S MEMORY

## Part 2 of 4

### 1 THE STORAGE LANDSCAPE.



### 2 CEPH: SOFTWARE-DEFINED STORAGE.



### 3 EXTERNAL ARRAYS: SAN / NAS.



### 4 TAPE: COLD ARCHIVE.



### 5 TIERING MATCHES DATA TO COST.



 **SWIPE FOR PART 3 → BACKUP VS REPLICATION, 3-2-1, AND RPO/RTO/RSL**

# EPISODE 6 – DATA IS THE COLONY’S MEMORY

## Part 3 of 4

### 1 BACKUP VS REPLICATION.



**REPLICATION =**  
LIVE COPY FOR AVAILABILITY



PROTECTS AGAINST  
HARDWARE/SITE FAILURE



COPIES DELETIONS,  
CORRUPTION, RANSOMWARE FAST

*Replication keeps  
you running.  
It doesn't save you  
from yourself.*



**BACKUP =**  
POINT-IN-TIME COPY FOR RECOVERY



PROTECTS AGAINST  
HUMAN ERROR, BUGS, MALICE



CAN BE OFFLINE



ACCEPTS SOME DATA LOSS  
AND RESTORE TIME

### 2 YOU NEED BOTH.



*Different  
failure modes  
need different  
tools.*



### 3 THE 3-2-1 RULE – MARS EDITION.



**3**  
COPIES



**2**  
DIFFERENT  
MEDIA



**1**  
OFFSITE  
COPY



*One copy trickles to Earth over the uplink.*

### 4 RPO, RTO, RSL.

**RPO –**

HOW MUCH  
DATA LOSS  
YOU CAN AFFORD.



**RTO –**

HOW LONG  
DOWNTIME  
CAN LAST.



**RSL –**

HOW MUCH SERVICE  
MUST STILL WORK  
IN DEGRADED MODE.



### 5 TEST YOUR BACKUPS.

#### QUARTERLY RESTORE DRILL

- ✓ SELECT DATA SET
- ✓ RESTORE TO ISOLATED ENVIRONMENT
- ✓ VERIFY INTEGRITY
- ✓ DOCUMENT RESULTS



*A backup you  
haven't test-  
restored is just  
a hope.*



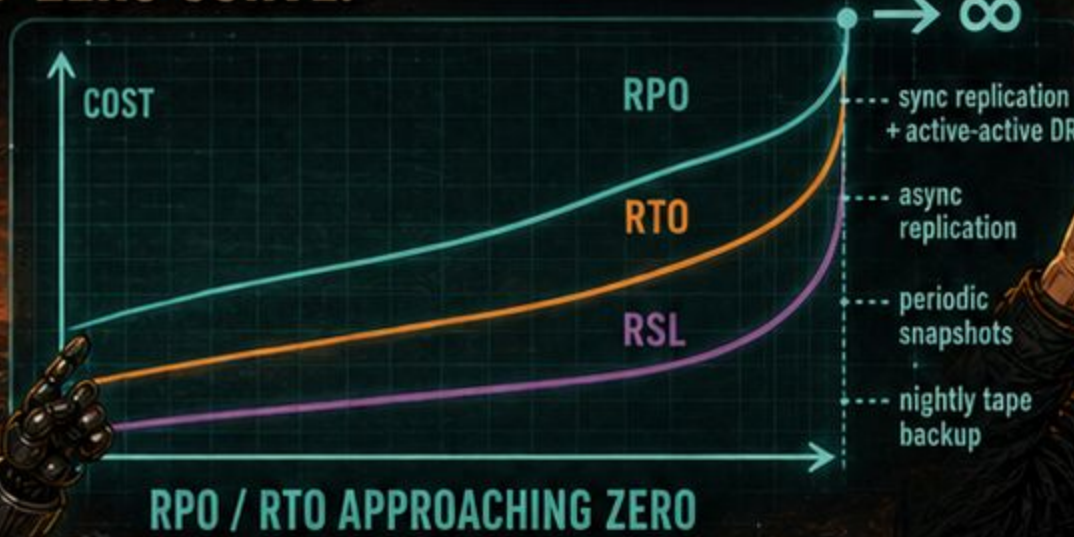
**SWIPE FOR PART 4 → THE COST-TO-ZERO CURVE AND THE VAULT**

# EPISODE 6 – DATA IS THE COLONY’S MEMORY

## Part 4 of 4

### 1 THE COST-TO-ZERO CURVE.

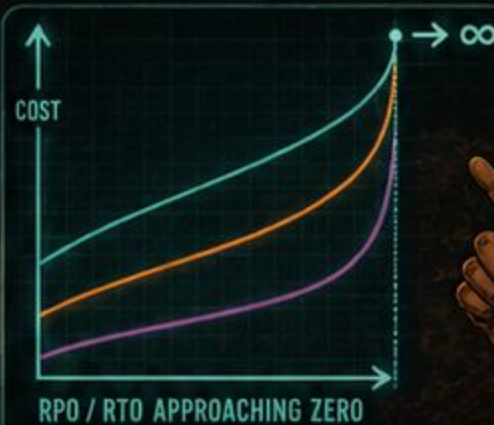
This is the same economics as the nines.



Zero data loss, zero downtime, zero degradation – bring an infinite checkbook.

### 2 ENGINEERING IS CHOOSING THE POINT YOU CAN AFFORD.

We pick the point that fits the risk, the budget, and the mission.



Tradeoffs aren't bugs. They're how we stay alive.

### 3 THE VAULT.

Tape? What century is this?

Radiation-tolerant. No power for 30 years. Cheap. Ransomware can't encrypt a cartridge on a shelf.

VAULT-1  
AIR-GAPPED  
SHIELDED  
COLD STORAGE  
OFFLINE

### 4 WHAT DANA DIDN'T KNOW.

NOT EVERY THREAT IS AN ACCIDENT. SOME THREATS HAVE INTENT.



NEXT: SECURITY, MALICE, AND DEFENSE IN DEPTH →

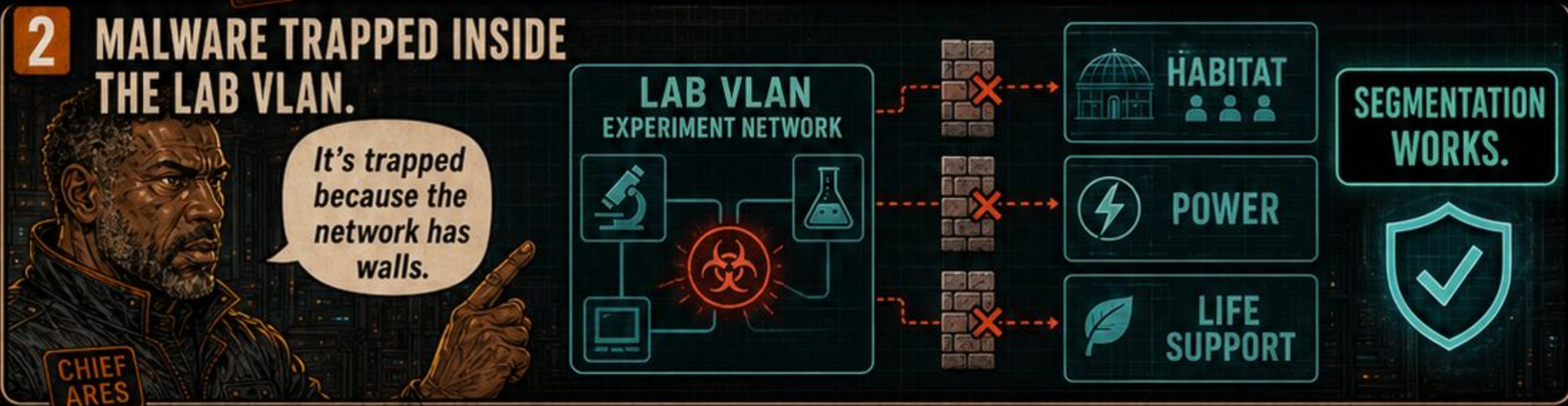
# EPISODE 7 – THE WALLS HAVE LAYERS

## Part 1 of 4

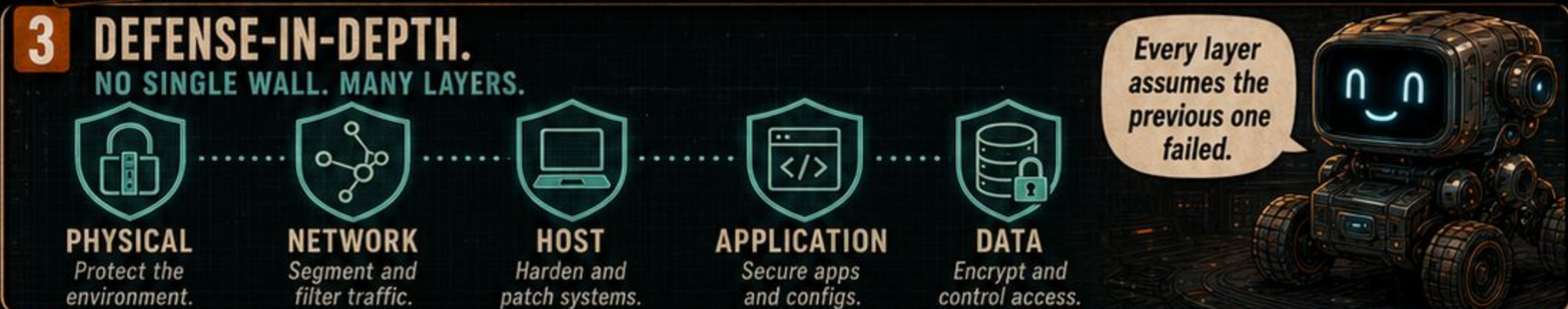
### 1 DANA REED AND ROVER DETECT SUSPICIOUS TRAFFIC.



### 2 MALWARE TRAPPED INSIDE THE LAB VLAN.



### 3 DEFENSE-IN-DEPTH. NO SINGLE WALL. MANY LAYERS.



### 4 THE WALLS ARE BY DESIGN.



**SWIPE FOR PART 2 →**

**MFA, ZERO TRUST,  
AND WHY STOLEN PASSWORDS FAIL**



# EPISODE 7 – THE WALLS HAVE LAYERS

## Part 2 of 4

### 1 2FA / MFA EVERYWHERE

#### OS / SSH



Password +  
Hardware Token

#### APPLICATIONS



Password +  
TOTP / WebAuthn

#### VPN ACCESS



Password +  
Approval

#### NETWORK GEAR ADMIN



Password +  
Hardware Token

2



A password  
is something you  
know – it can be  
stolen silently.  
A second factor is  
something you have  
or are.

3



So even  
a stolen  
password  
hits a wall.

No  
second  
factor.  
No  
access.



### 4 ZERO TRUST

Being inside the base doesn't mean you're trusted.



AUTHENTICATE  
Verify identity



AUTHORIZE  
Check permissions



LEAST PRIVILEGE  
Grant minimum  
necessary access



VERIFY CONTINUOUSLY  
Monitor and re-validate  
at every step



Every  
request must  
prove itself.



SWIPE FOR PART 3 → LOGGING, OFF-HOST LOGS, AND SIEM CORRELATION

# EPISODE 7 — THE WALLS HAVE LAYERS

## Part 3 of 4

### 1 LOG EVERYTHING. SHIP IT OFF-HOST.



### 2 SIEM: CONNECT THE DOTS



### 3 SAME LOG LINE. DIFFERENT QUESTIONS.



### 4 CORRELATION TURNS NOISE INTO SIGNAL.



SWIPE FOR PART 4 → SECRETS, PATCHING, AND THE FINAL LESSON

# EPISODE 7 – THE WALLS HAVE LAYERS

## Part 4 of 4

### 1 SECRETS MANAGEMENT



No more secrets in scripts or Git. Ever.

#### BAD IDEA

##### PASSWORD IN SCRIPT OR GIT

```
# connect.sh
DB_PASSWORD="MarsRocks2024!"
export DB_PASSWORD
...
```

```
git log
commit a1b2c3d
Author: dana.reed
- add connection script
- update password
```



#### GOOD IDEA

##### FETCH FROM SECRETS MANAGER

```
# connect.sh
DB_PASSWORD=$(vault read -field=password
secret/data/db/app)
export DB_PASSWORD
...
```

APP / SCRIPT



SECRET RETRIEVED AT RUNTIME

### 2 PATCH MANAGEMENT



Scheduled.  
Tested.  
Reviewed.  
Not chaotic.

### 3



The triangle we balance every day.

Security protects. Reliability keeps us running. Compliance proves we did it right.



### 4 WHY IT MATTERS

- DETECT THREATS EARLY
- LIMIT BLAST RADIUS
- SLOW ATTACKERS DOWN
- PROTECT THE DATA
- SUPPORT COMPLIANCE



You can't fix – or defend – what you can't see.

### 5 WHAT DANA DIDN'T KNOW:

# YOU CAN'T FIX – OR DEFEND – WHAT YOU CAN'T SEE.



### 6 NEXT: OBSERVABILITY – SEEING THE WHOLE COLONY THINK. →

## MONITORING vs. OBSERVABILITY

You need both.

### MONITORING

Answers known questions.  
Predefined checks & dashboards  
for known failure modes.

"Is CPU > 90%?"  
Is the service up?"

CPU



62%

MEMORY



71%

SERVICE



UP

- ✓ Predefined checks
- ✓ Dashboards
- ✓ Alert on thresholds
- ✓ Known failure modes



GOOD FOR FAST DETECTION

### OBSERVABILITY

Answers unknown questions.  
Rich telemetry you can slice,  
correlate & explore.

"Why is Module C slowly  
leaking memory after  
Tuesday's deploy?"

METRICS



LOGS



TRACES



- ✓ Collect everything
- ✓ Correlate across signals
- ✓ Slice • Correlate • Explore
- ✓ Find unknown issues



GOOD FOR FAST DIAGNOSIS

### YOU NEED BOTH:

**MONITORING** tells you that something is wrong.  
**OBSERVABILITY** lets you find out why.

But we **ARE**  
monitoring  
everything!

You're monitoring  
the questions you  
thought to ask in  
advance.  
**ASK NEW ONES.**  
That's how you  
see the truth.



# EPISODE 8 – EYES EVERYWHERE

PART 2 OF 4

## ALERTING PHILOSOPHY

Page on user-impacting symptoms and SLO burn, not on every hiccup.

### PAGE (WAKE SOMEONE UP)

-  User-impacting issues
-  SLO burn rate high
-  Service unavailable
-  Data loss risk



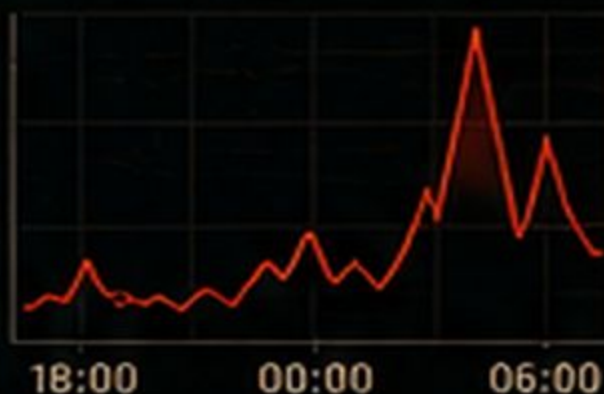
### TICKET (CREATE A TASK)

-  Resource trending
-  Single test failure
-  Low-severity errors
-  Expected noise

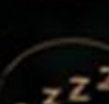
### ALERT FATIGUE IS DANGEROUS.



**47**  
PAGES  
LAST NIGHT



### REMEMBER:

-  Too many pages = soon zero get taken seriously.
-  Page on symptoms and SLO burn.
-  Ticket the rest.
-  Noise today, blindness tomorrow.

*Quality alerts  
save sleep...  
and save  
missions.*



# EPISODE 8 – EYES EVERYWHERE

PART 3 OF 4

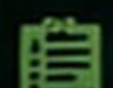
## ON-CALL • TRIAGE • ESCALATION

A system, not a punishment.

### 1 ON-CALL

 Humane rotations

 Hand-offs

 Runbooks ready

 Comp time earned

On-call is a system.  
Healthy people  
run healthy systems.



### 2 TRIAGE

(WHEN PAGED)

 What's the  
USER IMPACT?



 What's the  
SEVERITY?



 Mitigate first,  
diagnose later.



STOP THE BLEEDING,  
THEN FIND THE WOUND. 

### 3 ESCALATION

(CAN'T MITIGATE IN N MIN?)

Primary On-Call  
(You)

 N min



Secondary On-Call  
(Backup)

 5 min



Specialist  
(Subject Matter Expert)

 10 min



Incident Command  
(IC)



ESCALATING LATE IS A BIGGER FAILURE  
THAN ESCALATING WRONG.

RUNBOOKS AREN'T SUGGESTIONS.  
THEY'RE MEMORY FOR 3 A.M. YOU.



# EPISODE 8 – EYES EVERYWHERE

PART 4 OF 4

## TOIL: THE SILENT KILLER

Repetitive. Manual. Boring.

Measure it.  
Automate it.  
Eliminate it.



## MONITOR THE MONITORS

Who watches the watchmen?

- ✓ Check metrics pipeline
- ✓ Check log pipeline
- ✓ Check alert pipeline
- ✓ Synthetic checks



## THE THREE PILLARS OF OBSERVABILITY

### METRICS



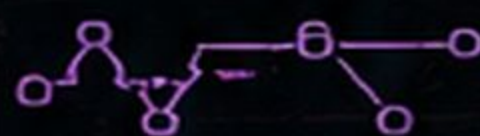
What's happening?

### LOGS



What happened?

### TRACES



Why did it happen?

## FIRST NIGHT ON-CALL...

*Half-asleep.  
Too many pages.  
Escalated the  
wrong thing to  
Ares at 04:00.*

PAGES RECEIVED:

47



*You'll learn.  
We all did.  
The system  
makes you  
better.*



## WHAT DANA DIDN'T KNOW:

A WALL OF SCREENS GOES RED.

A DUST STORM — THE BIGGEST IN A DECADE — IS COMING.

EVERYTHING SO FAR WAS REHEARSAL.



# EPISODE 9 – THE STORM, PART I: INCIDENT!

## 1. THE STORM HITS.

Part 1 of 3

SOLAR OUTPUT  
COLLAPSE



COMMS MAST  
FAILURE



Multiple  
systems down.  
Alarms are  
cascading.

HVAC INTAKE  
DUST DETECTED



INTAKE 3

DATACENTER TEMP  
CLIMBING



ALARMS CASCADING



## 2. DECLARE THE INCIDENT.

We have an  
incident. Roles  
now. Ego later.

### INCIDENT COMMAND STRUCTURE

INCIDENT  
COMMANDER



ARES

Decides.  
Communicates.  
Owns the call.

OPS LEAD



DANA

Executes.  
Coordinates.  
Keeps systems up.

COMMS LEAD



UPDATES  
COMMANDER  
PHOBOS &  
THE COLONY.  
Shares what's right.  
No surprises.

SCRIBE



ROVER

Logs decisions.  
Tracks actions.  
Preserves truth.

## 3. SINGLE SOURCE OF TRUTH.

One channel.  
One truth.  
Zero duplicate  
updates.

### INCIDENT CHANNEL: #INCIDENT-01

#### DECISIONS

- ✓ Declare SEV-1
- ✓ Isolate Intake 3
- ✓ Failover Comms
- ✓ Throttle Non-Essential

#### ACTIONS

- ① Ops isolating intake
- ② Comms on backup link
- ③ Thermal scripts running
- ④ User comms in progress

#### STATUS

- IN PROGRESS
- IN PROGRESS
- COMPLETE
- IN PROGRESS

#### TIMELINE (UTC)

- 14:02 Incident Declared
- 14:04 Intake 3 Isolated
- 14:06 Backup Comms Up
- 14:08 User Notice Sent

CURRENT STATUS: STABILIZING | NEXT UPDATE: 14:15 UTC

## 4. SEVERITY LEVEL: SEV-1 (CRITICAL).

SEV-1  
CRITICAL



WIDESPREAD USER IMPACT.



MULTIPLE SYSTEMS AFFECTED.



ALL HANDS FOCUSED.

### PROCESS BEATS PANIC.

- ✓ Follow the plan.
- ✓ Communicate clearly.
- ✓ Protect the mission.
- ✓ Take care of each other.

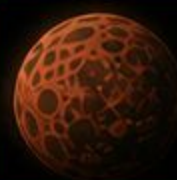


## 5. EVEN COMMANDERS GOTTA FOLLOW THE RULES.

I'm a Commander.  
I can fix a cable  
faster than you  
can say 'latency'.  
Let me in.

No, Commander,  
you may not help  
in the server  
room.

Rules apply  
to everyone.  
Especially  
Commanders.



# SWIPE FOR PART 2



## TRIAGE, ESCALATION, AND RUNBOOKS UNDER FIRE



# EPISODE 9 – THE STORM, PART I: INCIDENT!

Part 2 of 3

## 1. TRIAGE UNDER FIRE.

Dana works through an alert flood using Episode 8 discipline.



### TRIAGE FLOW

#### USER IMPACT?



Are users experiencing impact?

#### SEVERITY?



How severe is it?

#### MITIGATE FIRST, DIAGNOSE LATER



Stabilize service. Dig in after.

Stop the bleeding, then find the wound.

Triage first. Perfectly human. Perfectly right.



## 2. ESCALATION: BETTER EARLY THAN LATE.

A junior tech speaks up. Ares backs them up.



**“Escalating late is a bigger failure than escalating wrong.”**

We build trust.  
We own outcomes.  
No blame. Just better.

Culture keeps us alive.



## 3. RUNBOOKS PAY OFF.

The storm/HVAC contamination runbook gets us moving fast.

### RUNBOOK: STORM / HVAC CONTAMINATION

-  **1. CONFIRM ALERT & SCOPE IMPACT**  
Identify affected zones and systems. ✓
-  **2. CONTAIN CONTAMINATION**  
Close intakes. Activate isolation dampers. ✓
-  **3. STABILIZE ENVIRONMENT**  
Maintain pressure. Adjust filtration. ✓
-  **4. FAIL OVER SERVICES**  
Shift to healthy paths. Verify redundancy. ✓
-  **5. COMMUNICATE & MONITOR**  
Update status. Watch metrics. Stay looped. ✓

HVAC CONTAMINATION  
⚠️ DETECTED



Runbooks turn chaos into check-boxes.



## 4. GAP FOUND.

We log what's missing—live—to make the system stronger.

### OBSERVATION (LIVE) ⚠️

During intake revert, dust seal degradation wasn't explicitly checked.

### POTENTIAL IMPACT

Dust intrusion risk on reversion path.

### ACTIONS TAKEN NOW

- ✓ Seal inspected & reseated  
No active leak detected.
- ✓ Filters verified  
Within safe range.
- ✓ Intake path stabilized  
Contamination contained.

### DOCUMENT FOR LATER



Add explicit step:  
“Inspect dust seals before intake revert.”

OWNER: HVAC TEAM  
DUE: POST-INCIDENT REVIEW ✓

Write it down now. Save someone later.



Documenting gaps during the incident improves the system for everyone.

# PROCESS > PANIC. TRAINING > HEROES.

## SWIPE FOR PART 3 ➡

## LOAD SHEDDING, THE SCRIBE LOG, AND THE FAILOVER CLIFFHANGER

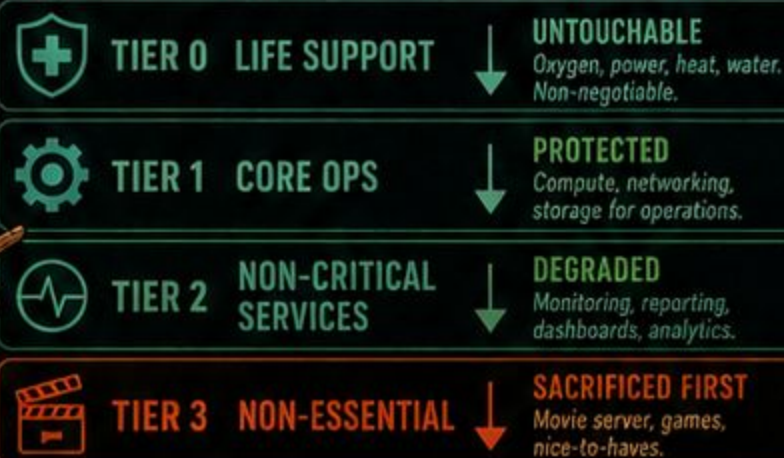
# EPISODE 9 – THE STORM, PART I: INCIDENT!

Part 3 of 3

## 1. SHED LOAD. PROTECT WHAT MATTERS.



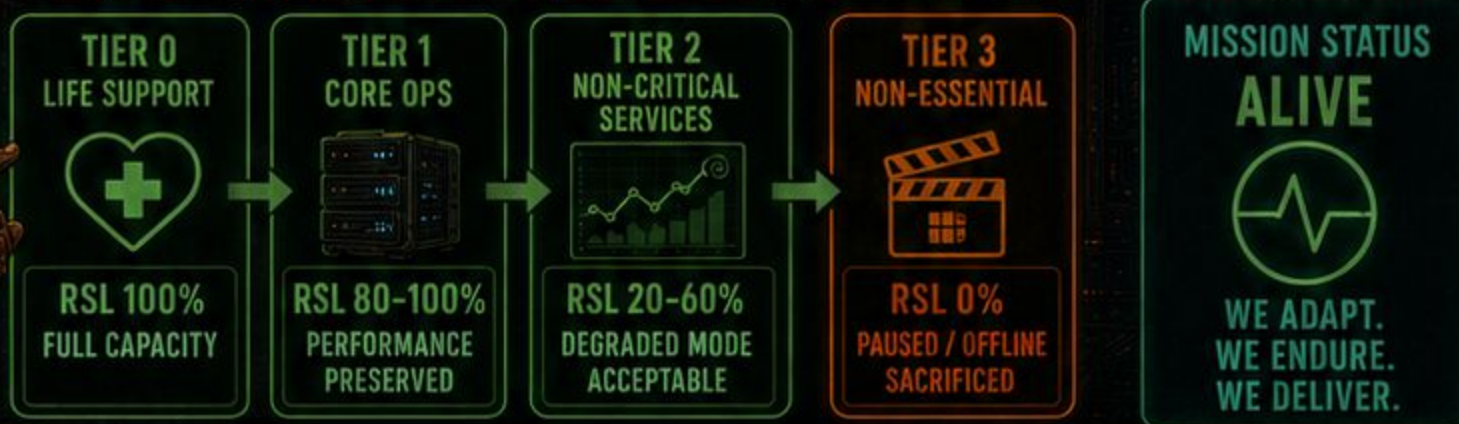
### GRACEFUL DEGRADATION – PRIORITY ORDER



## 2. RSL IN ACTION.



### RECOVERY SERVICE LEVEL (RSL) – MISSION SAYS ALIVE



## 3. DOCUMENT EVERYTHING.



### LOG LIKE YOUR FUTURE SELF DEPENDS ON IT. (BECAUSE IT DOES.)



Memory is unreliable. The log is not.

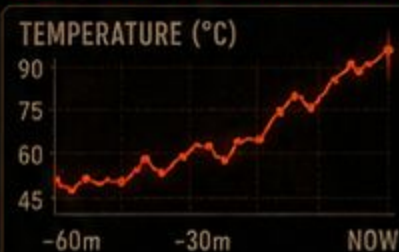
## 4. THE CLIFFHANGER.

Prepare to fail over to Site B.

...the DR site we've never fully tested?

### PRIMARY DATACENTER — COOLING

**LOSING THE BATTLE**



**STATUS CRITICAL**

COOLING CAPACITY  
**62%**  
AND DRIPPING



**WHAT DANA DIDN'T KNOW:  
EVERYTHING SO FAR WAS REHEARSAL.  
THE STORM IS THE TEST.**



➔ **NEXT: EPISODE 10 — FAILOVER, DR, AND WHETHER SITE B CAN REALLY SAVE THEM**

# EPISODE 10 – THE STORM, PART II: FAILOVER

Part 1 of 4

## 1. COOLING IS LOSING.

The storm is straining the primary site. Systems are falling fast.



PRIMARY SITE:  
LAVA TUBE ALPHA

COOLING CAPACITY



18%

STATUS: CRITICAL



POWER: STABLE



NETWORK: DEGRADED



## 2. SITE B: THE DR SITE.

Different cave.  
Different power.  
Different failure domain.



SEPARATE SITE



INDEPENDENT POWER



SEPARATE COOLING



SEPARATE NETWORK PATH

## 3. WHY SEPARATION MATTERS.

Both sites must not fail the same way.

POWER



Both sites must not lose power the same way.

COOLING



Both sites must not lose cooling the same way.

NETWORK



Both sites must not lose network the same way.

SITE



Both sites must not be impacted by the same event.

## 4. FAILOVER STARTS NOW.

Detect. Decide. Execute.



PRIMARY UNSTABLE



SWITCH TO SITE B



Independent failure domains are the point of DR.



# SWIPE FOR PART 2



DR STRATEGIES AND WHY  
MARS USES WARM STANDBY



# EPISODE 10 – THE STORM, PART II: FAILOVER

Part 2 of 4

## 1. DR STRATEGIES.



### HOT STANDBY



- ✓ Fastest recovery
- ✓ Near-zero data loss
- ✓ Highest cost

### WARM STANDBY



- ✓ Balanced recovery
- ✓ Some data catch-up
- ✓ Practical cost

### COLD STANDBY



- ✓ Slowest recovery
- ✓ More data risk
- ✓ Cheapest

## 2. ACTIVE-ACTIVE VS ACTIVE-PASSIVE.



### ACTIVE-ACTIVE



- ✓ Both sites live
- ✓ Higher complexity
- ✓ Higher cost

VS.

### ACTIVE-PASSIVE



- ✓ One site active
- ✓ One site ready
- ✓ Simpler failover

Good balance for Mars.



★ OUR CHOICE: ACTIVE-PASSIVE + WARM STANDBY

## 3. PROMISES VS PRICE.

### TIGHTER PROMISES COST MORE.

Lower RPO, RTO, and RSL mean higher cost.



RPO

(Recovery Point Objective)



RTO

(Recovery Time Objective)



RSL

(Recovery Service Level)

Want the gap gone? Here's the price.

FASTER. LESS RISK. HIGHER COST.



## 4. DR ONLY COUNTS IF TESTED.



DR THAT ISN'T TESTED ISN'T DR.

Test regularly. Fix early. Trust when it matters.

Test now, or fail later.



DR DRILL SCHEDULED



SWIPE FOR PART 3

FLUX, BACKUPS, DATA RECOVERY, AND THE LEGACY SERVICE FAILURE



# EPISODE 10 – THE STORM, PART II: FAILOVER

## Part 3 of 4

### 1. GIT REBUILDS THE PLATFORM.



#### GIT REPOSITORY

- ✓ YAML
- ✓ Kustomize
- ✓ Helm
- ✓ Policies

#### FLUX CD

- Observe
- Diff
- Reconcile

#### DR CLUSTER

- New place.
- Same platform.

Same Git repo. Same desired state. New place.

Because we built it in code, we can rebuild it under pressure.



### 2. DATA RETURNS.

#### CEPH ASYNC REPLICA



Remote site

#### LAST NIGHT'S BACKUPS



Object storage

#### DATA RESTORED



Back in service.  
Ready for traffic.

Backups are patience you can restore.



### 3. THE LEGACY SERVICE FAILS.



#### HARDCODED IP. NOT DOCUMENTED.

Connection refused  
10.1.42.17:8080

Service failed to start

Dependency check failed

One legacy service won't start.

Undocumented assumptions are outages in disguise.



### 4. THE LESSON.



#### DR THAT ISN'T TESTED ISN'T DR.



INVENTORY  
HIDDEN  
DEPENDENCIES



REPLACE  
HARDCODED  
ENDPOINTS



TEST FAILOVER  
FOR REAL

Boring prep is what makes recovery possible.



## SWIPE FOR PART 4



RPO/RTO/RSL RESULTS,  
FAILBACK, AND WHY  
BORING PREP WAS HEROIC.



# EPISODE 10 – THE STORM, PART II: FAILOVER

Part 4 of 4

## 1. MEASURED AGAINST THE PROMISE.

Not perfect.  
But alive.



|                                                                                                                        | PROMISED | ACTUAL  | RESULT                                                                                                          |
|------------------------------------------------------------------------------------------------------------------------|----------|---------|-----------------------------------------------------------------------------------------------------------------|
|  <b>RPO</b> Recovery Point Objective | ≤ 5 min  | 2.6 min |  CRITICAL SYSTEMS MET TARGET |
|  <b>RTO</b> Recovery Time Objective  | ≤ 30 min | 18 min  |  DEGRADED MODE HELD          |
|  <b>RSL</b> Recovery Service Level   | ≥ 95%    | 96.8%   |  CORE SERVICES ALIVE         |

## 2. FAILBACK IS A PLANNED OPERATION.

Planned.  
Verified. Boring.  
That's how we  
go home.



## 3. THE BORING WORK WAS THE HEROIC WORK.

This team  
kept the colony  
alive. Well done,  
everyone.



COLONY STATUS  
**STABLE  
AND  
SECURE**



WHAT WE BUILT MADE  
THE DIFFERENCE.

- ✓ GitOps
- ✓ Backups
- ✓ Runbooks
- ✓ Tiers
- ✓ Failover design

Preparation saved us.

## 4. WHAT DANA DIDN'T KNOW.



Surviving the storm  
was easier than  
proving it on paper.

Auditors  
incoming.



# NEXT: AUDITORS.



# EPISODE 11 – THE PAPERWORK PLANET

Part 1 of 4

## 1. EARTH AUDITORS ARRIVE.

The audit team lands at the colony.



I thought this would be paperwork and yawning.



## 2. THE AUDIT FINDS REAL ISSUES.

Not theoretical. Real.

### 1. EX-CREW ACCOUNT STILL ACTIVE



Access that shouldn't exist.

### 2. UNDOCUMENTED EMERGENCY CHANGE



Changed during the storm. Not recorded.

### 3. MEDICAL DATA IN THE WRONG ENCLAVE



Sensitive data. Wrong home.

## 3. COMPLIANCE IS PROCESS RELIABILITY.

CHIEF  
ARES

It's not bureaucracy.  
It's reliability for processes —  
proof we do what we say,  
every time, even at 3 a.m.

## 4. WHY THIS MATTERS.



CATCHES  
HIDDEN RISK



PROTECTS  
PEOPLE & DATA



PROVES  
CONSISTENCY



IMPROVES  
THE SYSTEM



# SWIPE FOR PART 2



CHANGE MANAGEMENT,  
ACCESS REVIEWS,  
AND MFA EVIDENCE



# EPISODE 11 – THE PAPERWORK PLANET

Part 2 of 4

## 1. CHANGE MANAGEMENT.

Every change gets recorded.



PLANNED CHANGE



1

REVIEW



2

APPROVE



3

DEPLOY



4

## 2. EMERGENCY CHANGES TOO.

Even emergency changes must be recorded and reviewed after the fact.



EMERGENCY NOW, REVIEW AFTER.

EMERGENCY CHANGE



RESTORE STABILITY



REVIEW & DOCUMENT



## 3. ACCESS REVIEWS & LEAST PRIVILEGE.

Right access. Right time. Remove it when roles change.



EX-CREW ACCOUNT

jordan.r@drf.mars

STATUS: ACTIVE

REVOKE ACCESS

JOINER



Grant only what they need.

MOVER



Adjust access to the new role.

LEAVER



Remove access on day one.

## 4. MFA EVIDENCE.

A password alone is weak evidence. MFA proves the second factor was there.



SSH LOGIN



- ✓ PASSWORD
- ✓ MFA CODE

MFA: ENABLED

APP LOGIN



- ✓ PASSWORD
- ✓ MFA CODE

MFA: ENABLED

VPN LOGIN



- ✓ PASSWORD
- ✓ MFA CODE

MFA: ENABLED

Evidence we can prove.



COMPLIANCE FEEDS DESIGN – IT'S NOT A BOLT-ON.



SWIPE FOR PART 3



SIEM LOGS, GIT HISTORY, AND AUDIT EVIDENCE



# EPISODE 11 — THE PAPERWORK PLANET

Part 3 of 4

## 1. SIEM = EVIDENCE GOLD.

Complete trail.  
Searchable.  
Timestamped.



Logs don't lie. They prove.



## 2. GIT HISTORY = EVIDENCE GOLD.

Every infrastructure change has an author, reviewer, and time.

| GIT HISTORY  |              |         |                      |
|--------------|--------------|---------|----------------------|
| AUTHOR       | REVIEWER     | COMMIT  | TIMESTAMP            |
| Dana         | Cmdr. Phobos | a1b2c3d | 2025-05-03 10:21:33Z |
| Chief Ares   | Dana         | d4e5f6g | 2025-05-03 11:47:02Z |
| Cmdr. Phobos | Chief Ares   | 7h8i9j0 | 2025-05-03 13:12:48Z |
| Dana         | Cmdr. Phobos | k1l2m3n | 2025-05-03 15:36:19Z |

Version control is accountability in code.



## 3. RETENTION MATTERS.

Keep it long enough. Keep it trustworthy.



- ✓ RETAIN LOGS PER POLICY
- ✓ PROTECT INTEGRITY
- ✓ KEEP EVIDENCE AVAILABLE

Retention today is defense tomorrow.



## 4. CIS HARDENING EVIDENCE.

Baselines applied.  
Controls proven.  
Risk reduced.



Auditors love proof, not promises.



# GITOPS + SIEM = FREE AUDIT GOLD.



## SWIPE FOR PART 4 →

## STANDARDS, DATA ENCLAVES, AND THE FINAL LESSON



# EPISODE 11 – THE PAPERWORK PLANET

Part 4 of 4

## 1. THE STANDARDS LANDSCAPE.

Compliance isn't one thing. It's a whole constellation.

RELIABILITY IS SURVIVAL  
SRE

ISO 27001



Information Security Management

SOC 2



Trust Services Criteria

PCI-DSS



Payment Card Industry Security

HIPAA / GDPR



Privacy & Data Protection

CIS BENCHMARKS



Secure Configurations & Hardening

## 2. DATA MUST LIVE IN THE RIGHT ENCLAVE.

Sensitive data stays where policy says it must stay.



MED-BAY ENCLAVE



Right data.  
Right place.  
Right rules.



ROVER

## 3. DESIGN CHANGES BECAUSE OF COMPLIANCE.

Compliance shapes the system. It doesn't get stapled on later.

CHIEF  
ARES

CONTROLS



Define what must be true.

ARCHITECTURE



Build with controls in mind.

EVIDENCE



Prove it works. Keep it continuously.

REVIEW



Assess. Adjust. Improve.

## 4. THE PARTING LINE.

Impressive recovery. But you got lucky the storm tested you before something worse did.

Who says we wait to be tested?



ROVER

WHAT DANA DIDN'T KNOW: SURVIVING THE STORM WAS ONLY HALF THE TEST.

NEXT: WE TEST OURSELVES FIRST.



# EPISODE 12 – BREAK IT ON PURPOSE, WRITE IT ALL DOWN

Part 1 of 4

## 1. THE POSTMORTEM BEGINS.

We're here to understand, not to assign fault.



## 2. THE TIMELINE TELLS THE TRUTH.

Memory is unreliable. The log is not.



## 3. 5 WHYS – WHY #1

QUESTION:



Why did the DR failover stall?



ANSWER:

Because the legacy telemetry service failed to start at Site B while everything else came up cleanly.



EVERYTHING ELSE



TELEMETRY SERVICE

## 4. 5 WHYS – WHY #2

QUESTION:



Why did the telemetry service fail to start?



ANSWER:

Because it was trying to reach a hardcoded IP address that only exists at the primary site.

**HARDCODED IP MISMATCH**

CONFIGURED IP

10.10.1.42

(PRIMARY SITE ONLY)

SITE B

NO ROUTE

IP DOES NOT EXIST

**CONNECTION FAILED**



# SWIPE FOR PART 2



THE REST OF THE 5 WHYS AND THE REAL ROOT CAUSE



# EPISODE 12 – BREAK IT ON PURPOSE, WRITE IT ALL DOWN

Part 2 of 4

## 1.5 WHYS – WHY #3

Why did it have a hardcoded IP instead of using DNS or service discovery?

Because it was configured by hand years ago and was never migrated into the IaC / GitOps platform.



### UNMANAGED HAND-CONFIGURED SERVICE

```
service.conf
IP = 10.42.7.19
PORT = 8080
```



Fragile. Manual. Forgotten.

VS.

### MANAGED GITOPS SERVICE



Discoverable. Declarative. Durable.

## 2.5 WHYS – WHY #4

Why was it never migrated?

Because no inventory of legacy services existed, so nobody knew it was still running unmanaged.



### UNLABELED SERVICE

No owner.  
No visibility.  
No migration.  
No one knew.

## 3.5 WHYS – WHY #5

Why was there no inventory?

Because the colony had no onboarding and audit process requiring every service to be registered, owned, and managed as code.



NO ONBOARDING  
STANDARD



NO AUDIT  
REQUIREMENT



SERVICES NOT  
REGISTERED



NO OWNER.  
NO ACCOUNTABILITY.

Without process, good intentions don't scale.

## 4. THE ROOT CAUSE



INCIDENT: PAYMENT SERVICE OUTAGE

TRIGGER: ~~JUNIOR TECH~~

ROOT CAUSE:

**PROCESS GAP**

The trigger was technical.  
The root cause was systemic.



Blame hides problems.  
Process exposes them.



# SWIPE FOR PART 3



ACTION ITEMS,  
DOCUMENTATION,  
AND SHARED LESSONS



# EPISODE 12 – BREAK IT ON PURPOSE, WRITE IT ALL DOWN

Part 3 of 4

## 1. EVERY FINDING GETS AN OWNER.



### ACTION ITEMS

| ITEM                                                                                                           | OWNER                                                                                     | STATUS                                                                                          |
|----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
|  Build service inventory     |  DANA  |  DONE        |
|  Require IaC onboarding      |  REYES |  IN PROGRESS |
|  Add config-review checklist |  MAYA  |  DONE        |
|  Run the next DR test        |  ARES  |  PLANNED     |

## 2. FIX THE DOCS.

*"Scars are only valuable if they're shared."*



RUNBOOKS



ARCHITECTURE  
DOCS



DECISION  
RECORDS



POSTMORTEM  
REPORTS



GIT

DOCUMENTATION LIVES IN GIT.

## 3. BLAMELESS VS. BLAME CULTURE

### BLAME CULTURE



*People stay silent.  
The same incident  
happens again.*



VS.

### BLAMELESS CULTURE



*People report honestly.  
We fix the system.  
Fewer repeats.*



*Honesty is the raw material of reliability.*

## 4. UNDOCUMENTED KNOWLEDGE LEAVES.



*Undocumented  
knowledge leaves on  
the next shuttle home.*



Searchable.



Versioned.



Maintained.



# SWIPE FOR PART 4



GAME DAY, CHAOS TESTS,  
CAPACITY PLANNING,  
AND DANA'S NEXT MISSION



# EPISODE 12 – BREAK IT ON PURPOSE, WRITE IT ALL DOWN

Part 4 of 4

## 1. GAME DAY

ROVER gleefully runs safe chaos tests.



PULL THE PLUG



POWER CUT

KILL A POD



POD TERMINATED

BLOCK A LINK



LINK ISOLATED

SIMULATE A ZONE FAILURE



ZONE DOWN

Hypothesis → Controlled experiment → Learn

## 2. SOME THINGS HEAL. SOME SURPRISE US.

SELF-HEALING SUCCESS



- ✓ DETECTED
  - ✓ ISOLATED
  - ✓ FAILED OVER
  - ✓ RECOVERED
- SLO MET

SURPRISING FAILURE

- ✗ CASCADE
  - ✗ DATA LOSS
  - ✗ MANUAL REPAIR
  - ✗ SLO BREACHED
- NOT EXPECTED

Every surprise becomes a fix and a doc update.

## 3. CAPACITY PLANNING

Reliability includes the future.



MARS YEAR FORECAST

CREW



DATA



STORAGE



POWER



SRE IS A LOOP, NOT A PROJECT.

## 4. THE NEXT ASSIGNMENT

You're the Chief SRE now.



NEW MISSION - SECOND COLONY PLANNED



END OF SEASON – BREAK IT. LEARN. IMPROVE. REPEAT.



# EPILOGUE – THE NEXT ENGINEER

Part 1 of 4

## 1. NEW SHUTTLE ARRIVES.

Every cycle brings new engineers.



## 2. DANA AND ARES EXCHANGE A LOOK.

Experience doesn't argue. It just observes.



## 3. ROVER SIGHS.

Some things never change.



## 4. THE REAL LESSON.

Remember this. Build for it. Rely on it.



# WELCOME TO MARS.

IT'S NOT THE AI THAT KEEPS THE LIGHTS ON.

# IT'S THE SYSTEM.



# SWIPE FOR PART 2



THE MAP OF RELIABILITY.



# EPILOGUE — THE NEXT ENGINEER

## THE FULL-STACK MAP OF RELIABILITY

Part 2 of 4

RELIABILITY IS A STACK. EVERY LAYER CAN FAIL.

MARS COLONY  
SITE A

A RELIABLE SYSTEM  
IS MORE THAN TECH.  
IT'S ENVIRONMENT.  
INFRASTRUCTURE.  
PEOPLE. PROCESS.  
AND CONTINUOUS  
IMPROVEMENT.  
THIS IS OUR MAP.  
LIVE IT. IMPROVE IT.

— DATA FLOW  
— CONTROL PLANE  
— OBSERVABILITY  
— SECURITY LAYER  
— RESILIENCY LINK

EVERY LAYER  
PROTECTS THE  
ONES ABOVE.  
EVERY FAILURE  
TEACHES US.

### 1. SITE & ENVIRONMENT

Mars is hostile.  
Our first perimeter.

Ep. 2

### 2. POWER / UPS / GENERATORS / PDUS

Keep the lights on.  
No power, no mission.

Ep. 2

### 3. COOLING / HVAC / AIRFLOW

Heat kills faster  
than code.

Ep. 2

### 4. RACKS & SERVERS

Compute is the muscle.  
Healthy, redundant,  
replaceable.

Ep. 2-3

### 5. CEPH / STORAGE NODES / TAPE VAULT

Data survives  
everything—if we  
design it to.

Ep. 6

### 6. NETWORK FABRIC / SWITCHES / LINKS / DNS / LOAD BALANCING

Packets in. Packets out.  
Always a path.

Ep. 4

### 7. OS HARDENING / CIS LAYER

Lock it down.  
Baseline everything.

Ep. 5

### 8. KUBERNETES + FLUX / PLATFORM LAYER

Automate the boring.  
Make drift impossible.

Ep. 5

### 9. APPLICATIONS & SERVICES

Deliver value.  
Measure it.

Ep. 5

### 10. SECURITY LAYERS + SIEM

Assume breach.  
Detect. Contain. Learn.

Ep. 7

### 11. OBSERVABILITY WALL METRICS • LOGS • TRACES

If you can't see it,  
you can't save it.

Ep. 8

### 12. DR SITE / SITE B

When Site A can't,  
Site B must.

Ep. 10

### 13. COMPLIANCE OFFICE / EVIDENCE / AUDIT

Prove it or  
it didn't happen.

Ep. 11

### 14. INCIDENT / POSTMORTEM / DOCS / GAME DAY LOOP

Break it. Learn it.  
Make it better.

Ep. 12

### 15. CULTURE & PEOPLE

The multiplier.  
Everything else  
depends on this.

Ep. 12



SWIPE FOR PART 3

WHAT THIS COMIC  
WAS REALLY ABOUT.

# EPILOGUE — THE NEXT ENGINEER

Part 3 of 4

## RELIABILITY IS A STACK.

Every layer can fail.



### GROUND / SITE

Facilities, environment, physical security



### POWER

Generation, distribution, UPS, redundancy



### COOLING

HVAC, chillers, temperature control



### CABLES / NETWORK

Physical links, switches, routers, DNS



### OS

Kernel, drivers, system services



### PLATFORM

Compute, storage, orchestration



### CODE / APPS

Services, dependencies, config, features



### DATA

Storage, replication, backups, integrity



### HUMANS

Skills, judgment, communication



### PROCESSES

Runbooks, change, reviews, audits



**SRE EXPECTS FAILURE AT EVERY LAYER.**

*There is no perfect layer. There are only weak links we haven't found yet.*



**WE DESIGN FOR REALITY.**

*Redundancy, isolation, observability, and automation reduce the blast radius.*



**WE MAKE FAILURE BORING.**

*Systems heal, traffic shifts, alerts fire, and you keep moving.*



**WE EARN TRUST EVERY DAY.**

*Reliability isn't a feature. It's the product. It's the team. It's the culture.*

**“ SRE IS THE DISCIPLINE OF EXPECTING FAILURE — AND DESIGNING FOR IT. ”**



## SWIPE FOR PART 4



**THE MAP IS NOT THE TERRITORY.**



**BUILD RELIABLE THINGS. HELP HUMANS. LEAVE IT BETTER. ///**

# EPILOGUE – THE NEXT ENGINEER

Part 4 of 4

## 1. THE MAP, NOT THE TERRITORY.

Each panel hid deep subjects. This comic showed the map, not the territory.



### SLO MATH

Error budgets, SLIs, SLAs, and trade-offs.



### NETWORKING

Packets, routing, latency, DNS, and reliability.



### CEPH

Distributed storage, replication, OSDs, and failure domains.



### BACKUPS

Snapshots, retention, restores, and disaster readiness.



### CIS HARDENING

Secure baselines, least privilege, and risk reduction.



### SIEM

Logs, detection, correlation, and response.



### GITOPS

Declarative infra, drift detection, and automation.



### INCIDENT COMMAND

Triage, comms, coordination, and clarity.



### RCA

Go beyond the symptom. Find the real cause.



### CHAOS ENGINEERING

Break on purpose. Build systems that survive.



### COMPLIANCE

Frameworks, evidence, audit, and assurance.

## 2. IF THIS MADE YOU REALIZE HOW MUCH YOU DON'T KNOW – GOOD.

That feeling is the starting line.  
Curiosity beats confidence.  
Stay humble.  
Stay hungry.  
Keep going.  
That's where every great SRE starts.

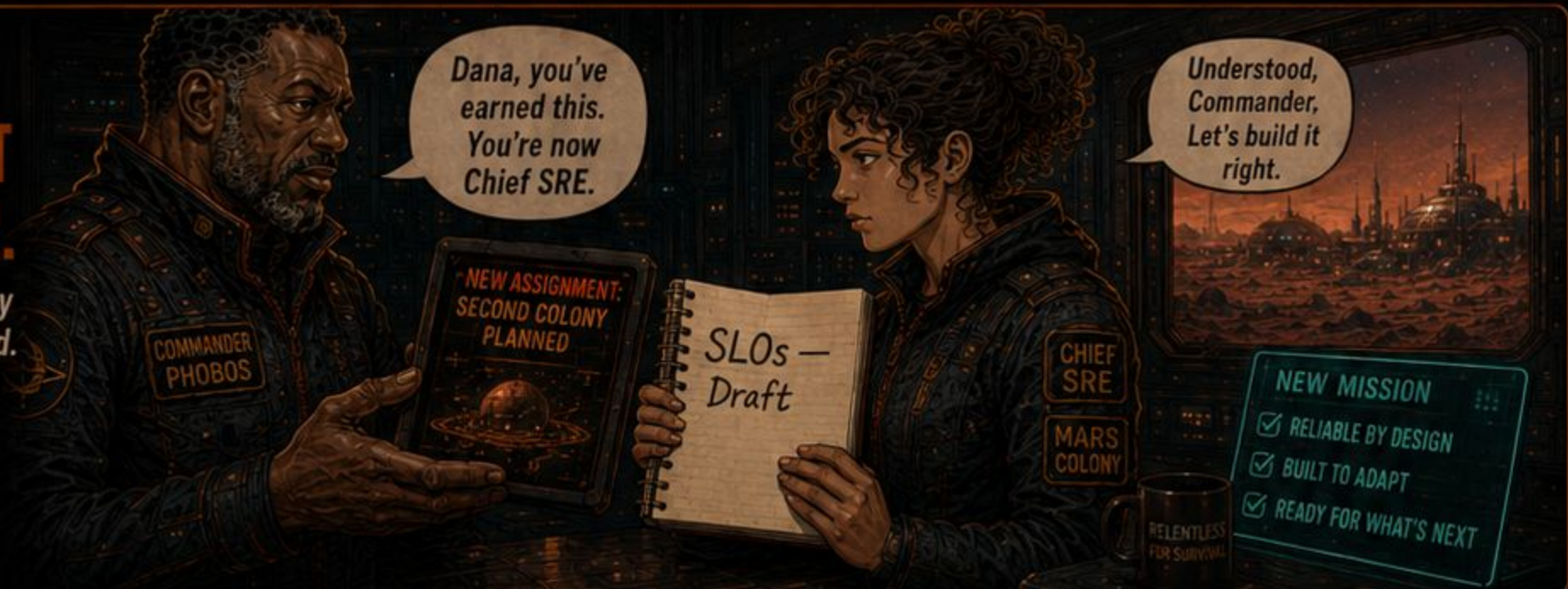


### YOU DON'T NEED ALL THE ANSWERS.

- ☒ You need systems that learn. ☒
- ☒ A team that trusts. ☒
- ☒ The courage to own the truth. ☒
- ☒ And the discipline to improve. ☒

## 3. THE NEXT CHAPTER.

A second colony is being planned.  
New systems.  
New risks.  
Same mission.  
Higher stakes.



## 4. RELIABILITY IS A JOURNEY. KEEP LEARNING. KEEP BUILDING.



— END OF BOOK —