Power Up/Level Up: Supercharging Your Security Program for Cloud and DevOps

Rich Mogull
@rmogull
The board is breathing down your neck, the CEO wants reports, and the CFO writes “Try Open Source?” in red pen on every budget request.

You barely have enough time to look at your to do list, never mind keep up on changing technologies.

- Looking at your Inbox feels like that scene in Se7en.

- Vendors are more than happy to educate you like a cult recruiter with an infinite bar tab.

- All your apps are moving to cloud in 2 months, and you can’t get cloud training approved until next fiscal.
And your security program be all like…
Armageddon is only bad if you want everything to stay the same

(Or you get space dementia)
Cloud and DevOps are the two most transformative forces to hit IT since… IT.

They require a fundamental rethinking of architectures and processes.

This is the biggest opportunity security has ever had since the start of our profession.

We get to start this race when the starter’s pistol goes off… not after the awards ceremony as usual.
Roadmap

- Start small, even when you can’t
- Get the right kind of education
  - Ignore your vendors and (most) consultants
- Write playbooks, not specifications
- Use compliance as a pry bar, not a speed bump
- Let your cloud providers and DevOps teams do the hard work
- Embrace architectures and automation, not scanners and loggers
The Problem

- New Tools and Applications
- Threats
- Technical Debt
The Problem

New Tools and Applications

Cloud + DevOps
Cloud and DevOps

- Cloud is a new operational model.
- It requires a re-thinking of fundamental architectures.
- DevOps is a new operational framework, highly attuned to cloud.
- Both shatter existing security approaches.
Cloud and DevOps

Abstraction breaks existing architectures

Automation breaks existing processes

Existing security patterns won’t effectively translate. It may look like they do on the surface, but sticking with the “familiar” and “manageable” will merely bring over all the liabilities and mistakes of the past.
Understand Cloud Adoption Patterns

- Developer Tethering
- New-Goes-Cloud
- Data Center Migration
Start Small

- Pick a single project
  - Even if you are losing control over everything else
  - Ideally one that’s moving fast
  - Use it to educate and build your patterns and requirements
  - Only one provider at a time
- Integrate with the team
  - Focus more on architectures and creating new processes, and less on enforcing the Old Ways
- Take the lessons and move to the next one
Get Edumacated

- **Technical** training for your specific cloud provider
- DevOps training
- Hands-On cloud security training
- Cloud and DevOps events/conferences
- Coding
  - *Be careful with vendor training. They are struggling worse than you are*
Questions to Ask

› Do they understand serverless deployments?
› Are they pushing lift and shift?
› Do they prefer a project-based approach?
› How much do they recommend third party products over inherent cloud capabilities?
› Do they start with multi-cloud or push single provider at the start?
› For vendors- how do they deal with autoscaling (native or fake), SDN, multiple accounts, etc.
Playbooks, Not Specifications

- Playbooks provide specific guidance on meeting security requirements.
  - Recommended and required controls
  - Sample architectural patterns
  - Specific, technical, integration instructions
  - Exception processes
  - And yes, specifications and policies

- Specifications/policies alone only communicate requirements
  - They often can’t keep up with the speed of cloud
  - By not providing advice they are often not fully addressed until later in a project, making compliance more difficult and costly
The Great Compliance Prybar

▪ Use compliance to engage earlier in projects
  ▪ Ideally require a review and integration at the design/architecture phase
▪ Don’t just send requirements, you need to engage and HELP the team meet the requirements
▪ Compliance is more-often used as a blocker. You need to use it as a way to get security engaged early, not as a migration control tool.

Image: https://database.wordpress.com/category/epic/page/2/
Let Your Providers and DevOps Do the Hard Work

Serverless

Deployment Pipelines

Automation
For clients to use a cloud provider, they must trust the provider.

This is especially true for anything with a sensitive data or process.

Thus security has to be a top priority for a provider or you won’t use them.

A major breach for a provider that affects multiple customers is an existential event.
PaaS providers can’t afford a preventable security failure.
- Including letting things get out of date.
- Many types of PaaS can’t rely on normal networking.
- Instead you access them via API.
- This creates an opportunity to “air gap” parts of your application.
- Kill off network attack paths (doesn’t help with logic flaws)
Network attack path?
PaaS Air Gap

No direct network connection

Internet ➔ Elastic Load Balancer ➔ Web Servers ➔ Simple Queue Service ➔ Processing Servers ➔ DynamoDB
Development Pipelines and Continuous Deployment

Source Code

Cloudformation Templates

Chef Recipes

Git

Functional Tests
NonFunctional Tests
Security Tests

Jenkins

Test

Chef Server

Prod

Chef Server

Chef Server

Chef Server
Lab: Building a Deployment Pipeline
Software Defined Security

- Attackers are automated, we are mostly manual.
- Our tools have been poor.
- We lack trustable security automation and thus need to rely on a “Meat Cloud”
- In cloud, APIs are mandatory. We can write code to automate and orchestrate, even across products and services.
Code without coding

- Work with your devs to build a library of building blocks
- Learn just enough to glue it together
- Build some core scripts
- Mix and match the blocks
Self-Healing Infrastructure (yes, for real)

<table>
<thead>
<tr>
<th>Type</th>
<th>Protocol</th>
<th>Port Range</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSH</td>
<td>TCP</td>
<td>22</td>
<td>69.2.174.98/32</td>
</tr>
<tr>
<td>HTTP</td>
<td>TCP</td>
<td>80</td>
<td>0.0.0.0/0</td>
</tr>
</tbody>
</table>

- Change a security group
- Event Recorded to CloudTrail
- Passed to CloudWatch Log Stream
- Triggers an CloudWatch Event
- Lambda Function analyzes and reverses
Code Your Security Program

- Be Agile
- Use sprints
- Use version control repositories, not Word, Powerpoint, Excel, and your Inbox
- Learn Markdown
- Love the Kanban
- Iterate iterate iterate
Power Up/Level Up: Supercharging Your Security Program for Cloud and DevOps

Rich Mogull
@rmogull