Flux7

White Paper

Effectively Balancing DevSecOps

How to Achieve Security with Agility in the Cloud

White Paper

How to Achieve Security with Agility in the Cloud

Security & Agility: Yin and Yang

Security and agility are often viewed as opposing forces. While there are new influences encouraging both dev and ops to build security in, security is all too often thought of as holding back progress rather than propelling it forward. Helping to shatter this myth, this paper will discuss why security and agility should be thought of as yin and yang - complementary ideas that are interconnected and support one another.

Security is not optional

Cyber threats have shaken the headlines lately. From ransomware to DDoS attacks, security incidents are on the rise. In 2016, IDG detected 38% more cybersecurity incidents than the year prior. And, the cost of these incidents is on the rise as well, with the global cost of cybercrime predicted to reach $2 trillion by 2019, from $500 billion in 2015. IBM's Ginni Rometty summed it best when she said, "Cybercrime is the greatest threat to every company in the world."

Neither is Agility

According to Information Age, 70% of the companies that were on the Fortune 1000 list ten years ago are now gone as a result of changing market conditions. And, market changes are only coming faster, which means that to stay competitive -- whether your organization is a Fortune enterprise or a startup -- you must remain agile.
PMI has identified the characteristics most indicative of organizational agility which are reflected in a strong DevOps approach. Specifically, they are the abilities to:

- Respond quickly to opportunities
- Shorten decision/production/review cycles
- Manage change
- Integrate the voice of the customer
- Manage risk
- Assign interdisciplinary project teams
- Eliminate organizational silos
- Implement contingency planning
- Use iterative project management practices
- Leverage technology

It should come as no surprise that highly agile organizations have significantly better business outcomes. Indeed, researchers at MIT found that agile firms grow revenue 37% faster and generate 30% higher profits than non-agile companies.

White Paper

How to Achieve Security with Agility in the Cloud

Balancing Perspectives

If you think of a security as a dial, on one hand it can be dialed all the way up (let's say to 11) where no one can access systems let alone make any changes. On the other hand, you could dial security all the way down to zero which would be the equivalent of leaving your front door unlocked and wide open. Obviously, neither of these scenarios are workable. Yet, finding the "just right" spot on the security dial takes significant thought and collaboration to ensure the dial meets risk and compliance objectives on one hand while empowering development on the other.

At Flux7, we recommend a balanced perspective focused less on doing cool new things in security and more on how to be more effective and efficient with the security processes and tools that are already in place. Organizations should ask themselves, what are the attack patterns we are trying to avoid, is it external threats or internal threats, data leaks or something else? How can we automate our best practice security processes, boosting our security benefits? How can we automate security to provide a greater level of control over our architecture and new elements being created?

In the cloud, security can be automated to do just this, and ideally starts by building security in. Just as you wouldn't bolt a roll cage onto your race car just before the race, neither should a mature organization bolt on security as an architectural afterthought.

Which of these two cars would you prefer to race in? While they both offer *some* level of protection, it's clear that you'll go faster in the car on the right because you'll be safer and you'll feel safer. There are other controls in these cars -- like brake systems -- that actually allow you to go faster because they provide you with the ability to drive safely. (Really, how fast are you going to drive in a car without brakes?) Similarly, security controls provide your organization with a level of security that allows developers and IT operations to increase their speed and willingness to take risks, growing your overall business agility.

Regulatory Compliance

In addition to moving faster because we have built-in security controls, we also have active security requirements -- like regulatory compliance -- with deadlines that we must meet. Security automation helps us achieve these goals as well.

While traditional security methods do not scale to DevOps based cloud approaches, new elastic platforms that use different design principles do. These platforms include elements like Amazon Web Services, immutable containers, infrastructure as code, and continuous integration and delivery. In this new landscape, the perimeter has a new definition and security policies are not applied but automated.

Moreover, this new landscape presents opportunities for continuous, automated auditing rather than periodic deadline-driven audits. In either case, the organization needs to start by identifying the list of controls they want and/or need to implement -- consider regulations and best practices like PCI, DISA STIGs, NIST 800-53, and the CIS Benchmarks. AWS, for example, has Quick Starts[1] for many of these that can help get you started on the right footing.

1 https://aws.amazon.com/quickstart/

White Paper

How to Achieve Security with Agility in the Cloud

The Role Containers Play

Containers are becoming increasingly popular in cloud based DevOps environments for many reasons, not the least of which is facilitating faster development through a microservices architecture. In addition to speeding code throughput as development teams can work on individual services in parallel, containers can also make it easier to pass security and compliance checks. From an auditability and defensibility standpoint, IS auditors care about three things:

1. **Do you have a process?**
   The answer when using containers is 'yes' because containers are built using Dockerfile and Docker commands and that is the only way to do it. A process can be easily built around it.

2. **Are you following the process? Can you guarantee that people are following the process?**
   As it so happens, with containers nobody needs to be allowed SSH access into any production instance; the only way you can actually get a container deployed is through the process you create, e.g., a Jenkins job which automates the deployment. If you go through Jenkins, the process is guaranteed to be followed because that is how Jenkins is set up.

3. **Can someone tamper with the process? How do you ensure that a Byzantine entity can't come in and tamper with your process?**
   Containers make that really easy because you are guaranteed that no one is logging in. Your container by definition needs to be stateless which means that your files are read only. So, if somebody were to get in, they would have to break through and have a file system that people can effectively write onto, because logs are being pulled out.

From a security perspective, containers make the environment a lot stronger and more defensible due to their innate processes. There are a two additional layers to consider: detecting availability and making sure you are using the right software. And since with containers once the image is built, you can't tamper with it, you can run static analysis. There are several open source tools that will help conduct a full static analysis of commonly known vulnerabilities on your containers. At Flux7 we do a lot of PCI and HIPAA projects and we find that the container pipeline is easier to defend and audit than a traditional pipeline. That is, once you prove the pipeline was built correctly, you are essentially done with the audit.

White Paper

How to Achieve Security with Agility in the Cloud

Case in Point

At Flux7 we worked with a global broadband services and technology company that has managed Wi-Fi hotspots globally. The payment processing environment for its Wi-Fi services were built with automated security built in and featured a fast and robust environment that assured immediate service to users at any time. As consistency of service is integral to business success, a fast, elastic environment that was able to scale when needed was a critical requirement. It achieved this flexibility and passed its Level 1 PCI DSS audit by an independent qualified security assessor.

By using the key tenets of DevOps as foundational building blocks, when they were built upon, the company simultaneously achieved effective operations, solid security and PCI compliance. In fact, the assessor who audited their systems was very impressed with how well the environment was described, remarking that everything he wanted was in the code. As a result, this audit was the fastest and least painless the organization had gone through.

By using the key tenets of DevOps as foundational building blocks, when they were built upon, the company simultaneously achieved effective operations, solid security and PCI compliance.

Case in Point

As an international, publicly traded organization, a Fortune 500 manufacturer of heavy equipment sought continuous auditing. The goal of this notification system was to alert operations and information security teams of any known security, risk or compliance issues.

Through a deployment pipeline, that is triggered when a change to code is made, e.g, if a function is added, removed, or changed, we were able to audit for change. Automation was used to make the process for adding a new notification really simple and was also used to log critical data. From these logs, the customer has a dashboard to view and use the audit information. This continuous auditing system is both low maintenance and extensible so that new rules can be conveniently added, and the same system can be used to audit multiple AWS accounts.

Our DevOps consultants also taught the firm how to use these advanced AWS features to ensure their ongoing success. Teaching businesses how to use, manage and extend their infrastructure helps ensure that they are equipped to effectively balance their security, development and operational objectives in support of their core business goals.

White Paper

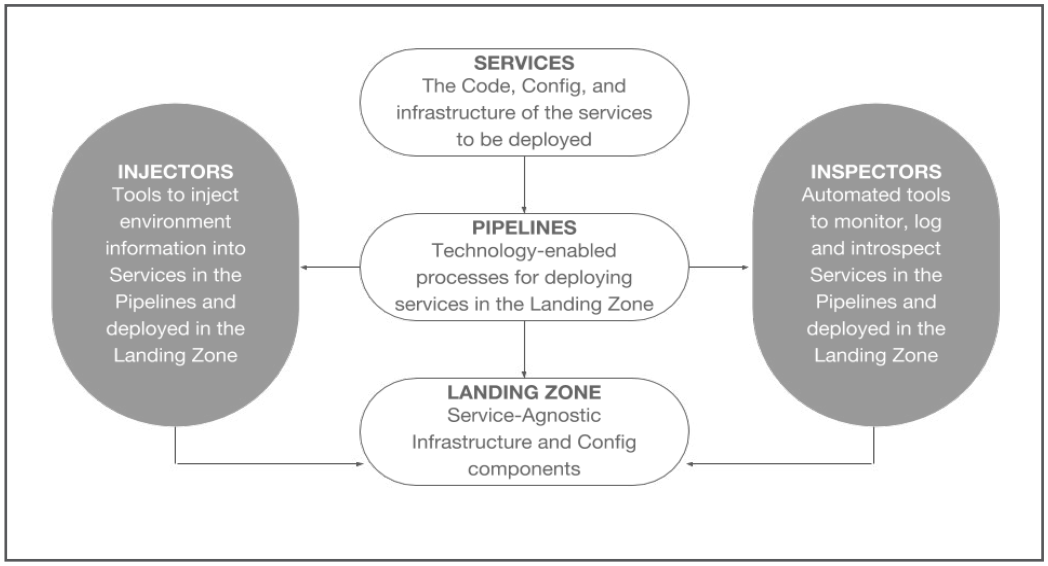
How to Achieve Security with Agility in the Cloud

The Flux7 Enterprise DevOps Framework & Security

At Flux7, we **have worked with more than 150 companies** over the years as they have gone through the DevOps transformation process. And, we've learned a lot along the way, including the patterns that emerge in the DevOps journey and where most people land and/or have the vision to land. We've captured these patterns in what we call the Enterprise DevOps Framework, or EDF, which highlights Development and Operations processes as well as the role of security.

This new patent pending framework has resulted in a significant changes. Let's examine each of these new areas, starting with services. Teams that own services own not just code, but also things like configuration (via Chef, Puppet or Ansible), infrastructure in the form of tools (like Terraform and Cloudformation) that create virtual machines, and frankly, everything specific to the services they create. In this new DevOps-driven model, the service code and all its relevant dependencies are owned by the service team. This allows the service team to move faster as they have fewer dependencies.

Conversely, traditional IT begins to provide to the service teams what we call a landing zone. The landing zone is where services deploy and as a result it is important that the landing zone be secure -- if you want your apps to be secure, you need to start with a secure foundation. To ensure the security of the landing zone, our DevOps consultants recommend starting with CIS Benchmarks for AWS. In this way you can benchmark your landing zone and embed security, building it in as you work from the ground up. Note that any decision about the landing zone needs to take security into consideration.



Looking next at the pipelines that deliver services into the landing zone, we recommend that organizations design their pipeline such that inspectors assess code, configuration and other elements as they flow through the pipeline for any potential vulnerabilities. With governance built into the pipeline via inspectors, these inspectors help ensure compliance to security requirements.

As you can see, inspectors play two roles here. They inspect components:

1. As they move through the pipeline. This more traditional inspection includes things like security and audit checks.
2. In the landing zone where they conduct image analysis to check for available software on containers. This secondary stage of inspection gives you a whole new level of security and vulnerability analysis not typically enabled in virtual machines.

In addition, we have the idea of injectors. When you have a framework like the EDF with things moving full speed, the last thing you want is for people to have to stop the process in order to communicate with each other information specific to applications landing in the landing zone. For example, imagine having to interrupt automation to pass along a subnet ID. To this end, most of the companies we work with have adopted automated tools that inject environment specific security information into their service templates on the fly as elements go through the pipeline. This level of automation helps them reach their full potential and protects against fat-finger induced errors. We encourage the use among our customers of security tools like HashiCorp Vault and HashiCorp Consul.

Cloud Common Library

Within the EDF environment it's important to have a library of best practice templates. We call a collection of these templates a cloud common library. The idea behind this library is to:

- Have all the tools your team will use in a central, known location.
- Make your team's job easier by giving them samples to learn with and from
- Collect organizational knowledge so that as one member of the team learns, the entire team has the opportunity to learn along with them.

The cloud common library helps keep your team organized and from getting in each other's way. The library provides common approaches and organizational standards that facilitate best practices and standardization. Furthering our car analogy, the library sets best practices like driving on the correct side of the road and following traffic signals.

EDF Security

Building security into the EDF environment benefits development and operations in three key ways:

- **Faster Delivery** - With this model, development and operations can focus on service throughput rather than coordination. By securing and automating the injection and inspection processes, these two teams can focus on code throughput rather than stopping to coordinate security details.

- **Reduced Risk** - With a secure foundation in the form of a secure landing zone, risk is reduced. And, with security automated, the ability for human error to be introduced is decreased, further reducing risk. When taken together, these security features empower development to move faster, with greater confidence.

- **Faster time to market** - Faster delivery with reduced risk translates into services being delivered to market more quickly, helping the business achieve its goals more quickly.

White Paper

How to Achieve Security with Agility in the Cloud

Case In Point:

We worked with a SaaS customer analytics company to build-in security in its new DevOps-based processes and technology foundation. For this company, we helped design a secure landing zone, pipelines with security embedded, and a set of inspectors that can inspect and detect any activity that will reduce security or open the firm up to a breach. Specifically, we built a process where security inspectors examined code, software, and infrastructure -- all at one time -- across pre- and post-production for security vulnerabilities.

In this instance, pipeline triggers checked configuration rules while inspectors in the pipelines checked artifacts. These inspectors were in addition to our work ensuring that the landing zone was secure and that the firm had injectors efficiently providing needed security information for services when and where required.

The result for this firm was that it was able to successfully migrate its public-facing application to AWS while increasing its scalability and availability globally. Its development productivity grew, allowing it to much more quickly and easily evolve its application, providing demanding customers with new features and functionality they had asked for. Using ISO 27001 as their security standard, the firm was able to meet these control objectives in its new environment at the same time it increased its competitiveness in the market through faster service delivery.

Created separation of duties?

✓ Consider using separate AWS accounts for Development, Production, etc.

✓ Have clearly defined roles with minimum permissions

Protected your accounts from break in?

✓ Set strong password policies

✓ Set MFA authentication

✓ Enable SSO to simplify management of accounts

AWS Security Checklist

✓ Get rid of credentials that are not automatically rotated

Protected secrets, such as API keys and passwords?

✓ Incorporate "injectors" like AWS Parameter Store and KMS Store, as well as HashiCorp Vault to manage sensitive data.
✓ Use tools like Anchore to scan assets for accidental secrets

Implemented automated testing/continuous security monitoring?

✓ Implement basic monitoring and testing with AWS Config Rules; extend to advanced monitoring and testing with custom rules configuration
✓ Enabled CloudTrail in all regions to have information available for audits
✓ Saved CloudTrail audit logs in a different secure account
✓ Enable logging on all assets including EC2 instances

Met corporate and legal compliance requirements?

✓ Understand your legal requirements
✓ Implement encryption at rest using KMS for EBS, S3, RDS, etc
✓ Follow the principle of least access on security groups
✓ Restricted access to EC2 instances
✓ Use CloudFormation to define all resources so resource creation is done in an automated, repeatable, and auditable method.

Ensured availability of services against malicious attackers?

✓ Use AWS WAF and CloudFront for protecting against external actors.
✓ Implement autoscaling to handle increased loads

Enabled CloudTrail and Config to enable audits?

✓ Make sure that the buckets which store audit logs are encrypted and any write activity on these buckets are monitored as well

Used KMS keys for encryption?

✓ Ensure all persistent stores including RDS, EBS, S3 are encrypted at rest. AWS offers it at no additional cost.

Metrics

Security teams often joke that their true measure is when nothing bad happens. While this is indeed a good thing, we have several tangible positive measures we recommend enterprises employ to determine how effective their balance is between security and agility. Specifically, consider tracking and regularly measuring:

- How your landing zone fares against the **CIS benchmarks** for AWS. Identify and prioritize areas for improvement.
- Mean time to server patching. This measure of how long your servers are in a known vulnerable state should be measured, according to CIS, in hours, not days.
- How many breaches did you get at the application level?
- How frequently is something flagged in your Amazon Machine Image (AMI) pipeline? And how long does it take your organization to address it? What is the time to roll out critical updates?
- What is your overall average time to roll out?
- Conduct resilience load testing and measure how much of a DDoS-type traffic load you can tolerate.
- If your organization uses relevant from them as additional measures for continuous improvement.

Conclusion

Over the years we've learned a couple important of things. The first is that making security easy actually improves security. The reason is that people are more likely to work within your security parameters if it's easy to do so. Unfortunately, when security presents too many barriers, we've seen people bypass it altogether effectively negating the intended security coverage. The second important lesson we've learned is that security is both a prerequisite to DevOps success and a byproduct of it. As organizations modernize their computing systems through DevOps based cloud approaches, a new way to govern systems is needed. Traditional tools and methods don't always translate into this new landscape where the perimeter has a new definition and security policies are not applied but automated. There are also new opportunities where environments can be audited continuously rather than periodic audits. Focusing on building "Security with Agility," will allow you to build secure environments without slowing down the engineering teams' work, allowing security, development and operations to simultaneously deliver on their key goals for the business.

LEARN MORE ABOUT FLUX7

As DevOps and AWS experts, Flux7 offers a suite of solutions that help organizations design, build, own and manage IT modernization projects. Focused on architecting and optimizing their clients' AWS infrastructure and training internal IT teams to manage their own infrastructure, Flux7 solutions are rooted in DevOps best practices. Flux7 has delivered hundreds of agile, right-sized projects to satisfied customers across industries, creating a well-architected core from which these business can own and expand their IT modernization.