**Application Security Program: simple but not easy**

**TL;DR:** How to design a Service Oriented Architecture to deliver an Application Security capability: lesson learned, gotchas and everything to survive to tell the story.

### .....

To implement an Appsec Program you need to build a capability that supports the delivery of security requirements as part of the normal software development life-cycle.

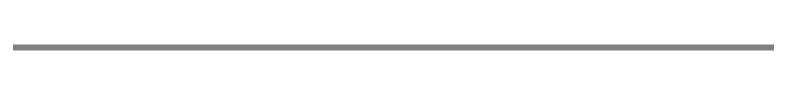
Up until now application security has been very niche, not a skill widely found part of developers’ curricula; things are changing, but not as fast as the market would like.

At the same time, the number of application security professionals is not sufficient to fulfill the request. Because of this, such a security capability can be best provided as a service, where a small Appsec team can support and enable a large group of developers.

Like with any other Service Oriented Architectures (SOA), the Appsec Service has its own specific consumers and it exists to satisfy their needs.

A good metric to define the success of any service is the adoption level, that is to say: a service is as good as its use. The users/customers are clearly main factors of the success measurement, so we will start with identifying WHO the consumers are, which roles they play, what their purpose is, and what their challenges are.

Identifying consumers is not particularly difficult: the Business, the Development and us, the Security.



**Development:** it has a dual role, **supplier** and **consumer**.

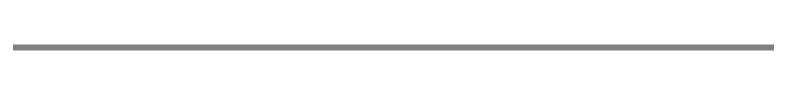
* As a **supplier**, the Development is responsible for the adherence of the input artifacts to standards defined by the consumer (Security). Here is an example of what standards may be like for a popular static analysis tool: delivery timelines (input artifact is delivered on a specific schedule), artifact composition (the artifact must be packaged in a specific way), and compilability (the artifact, in this case the code, should be in a compilable state). Note: different tools/systems may have different requirements and standards.
* As a **consumer**, the Development seeks detailed "security information" that must be available, consistent and reliable in terms of accuracy. This information is then used to plan/prioritize fixing activities.Note: Sometime the Development can split the two roles (consumer/supplier), and delegate the supplier responsibilities to the Dev-Ops team.

**Security**: it has a dual role, **supplier**and **consumer**.

* As a **supplier**it is responsible and accountable for the service that delivers security information to both the Development and the Business (it should match security information standards and requirements for both consumers).
* As a **consumer**it "consumes" the input artifact provided by the Development, and for this reason it is responsible for setting the input artifact standards.

**Business**: it has a **consumer**role.

* As **consumer**it consumes aggregated, reliable and traceable information. At the same time, it is interested in cost efficiency, from both the Security team and the Development team. The Business is looking for a balance between Appsec initiative direct costs (like licenses, maintenance, environment and people) and indirect costs (overall aggregated cost of fixing the issues, but not the cost of fixing a single issue).



To design and implement a service capable of satisfying consumer’s needs , and to keep the suppliers accountable for their responsibilities, we need to identify their goals and constraints.

**Development:**

* **Goal/s:** deliver code that implements features in production;
* **Constraint/s:** Security, among many other constraints like Code Quality, Maintainability, Velocity, and Reliability;

**Security:**

* **Goal/s:** protect the Business from security related problems (by making them manageable), so basically deliver survivability;
* **Constraint/s:** enable development velocity (avoid/minimize delays), among others like cost efficiency, tractability, replicability, availability, consistency, and reliability;

**Business**:

* **Goal/s:** deliver value to the clients;
* **Constraint:** Survive, and eventually be profitable.

Now we have a better idea of how the service will be consumed by each player, and what are the goals and challenges for each of them. We can now start thinking about how we will help them to achieve the goals and overcome the challenges.

**Note:** A big step toward success is to take in consideration that there are other teams and people that are important to your own goals: this perspective will force you to avoid self-referentiality. As an Appsec person you need to remember that you are providing a service to someone who deeply cares about what they do.

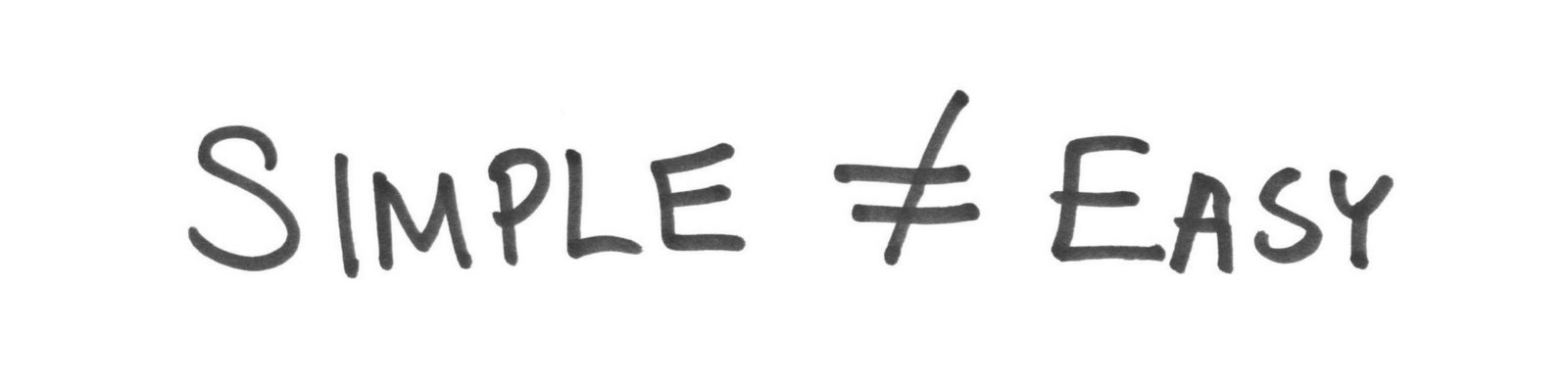
**Also remember:** if the Business doesn't invest in terms of commitment and budget, it is as bad as if the Development doesn't adopt the service at all.

As with any other Service, the success of an Appsec Service depends on the breadth and the depth of adoption.

Breadth: represented by program/application-inventory/developer-teams coverage that is limited by funding (business) and by adoption rates (development).

Depth: identified by frequency of use (per week/per month/per release of a single development pipeline) that is limited by funding (business) and by adoption rates (development).

*So, we need to design a service architecture that is CHEAP, ATTRACTIVE TO USE, and of course ...... SECURE.*



**Since it's not easy, there is a high chance of failure.** Many things can (and will) go wrong and in **my future posts** I will address these pain points one by one:

### [**INFORMATION NOISE**](https://www.linkedin.com/pulse/application-security-program-information-noise-stefano-parini/)

This noise is caused by the "false positives". Lack of accuracy, if unmitigated, drives consumers mistrust which affects both developer teams and business.

### [**COST EFFICIENCY**](https://www.linkedin.com/pulse/application-security-program-its-all-balance-stefano-parini/)

Security is not perceived as a market differentiator yet, so to be successful it must be affordable.

### [**VOLUME & VELOCITY**](https://www.linkedin.com/pulse/application-security-program-volume-velocity-stefano-parini/)

Unrealistic requirements show an essential disconnection with the customers, which in turn will not be inclined to submit to the policy.

### VISIBILITY AND REPORTING

Simply deploying an Appsec Service is not good enough; you will want to measure carefully whether the tool is functioning efficiently and effectively.

### USER EXPERIENCE

Development people are always busy, so we should minimize the effort of using the service on their part.

### 



Disclaimer: the views and opinion present on this post are those of the author and do not necessarily reflect the official policy or position of any other agency, organization, employer or company.