OPEN SECURITY SUMMIT

Using Wardley Mapping for Security Strategy and Architecture Development

https://open-security-summit.org/

Agenda

- Good Strategy / Bad Strategy
- Strategy Cycle
- Expanding Sun Tzu’s 5 Factors for Cyber Security
- The Strategist vs The Architect
- Strategy Development
- The changing role of the Architect
- Climatic patterns - Security Architecture
- Use cases
- Closing thoughts

“Good Strategy / Bad Strategy - The Difference and why it matters”

Richard Rumelt

A good strategy is **straightforward, simple and easy to understand**. It constitutes of “strength applied to the most promising opportunity.” Richard Rumelt

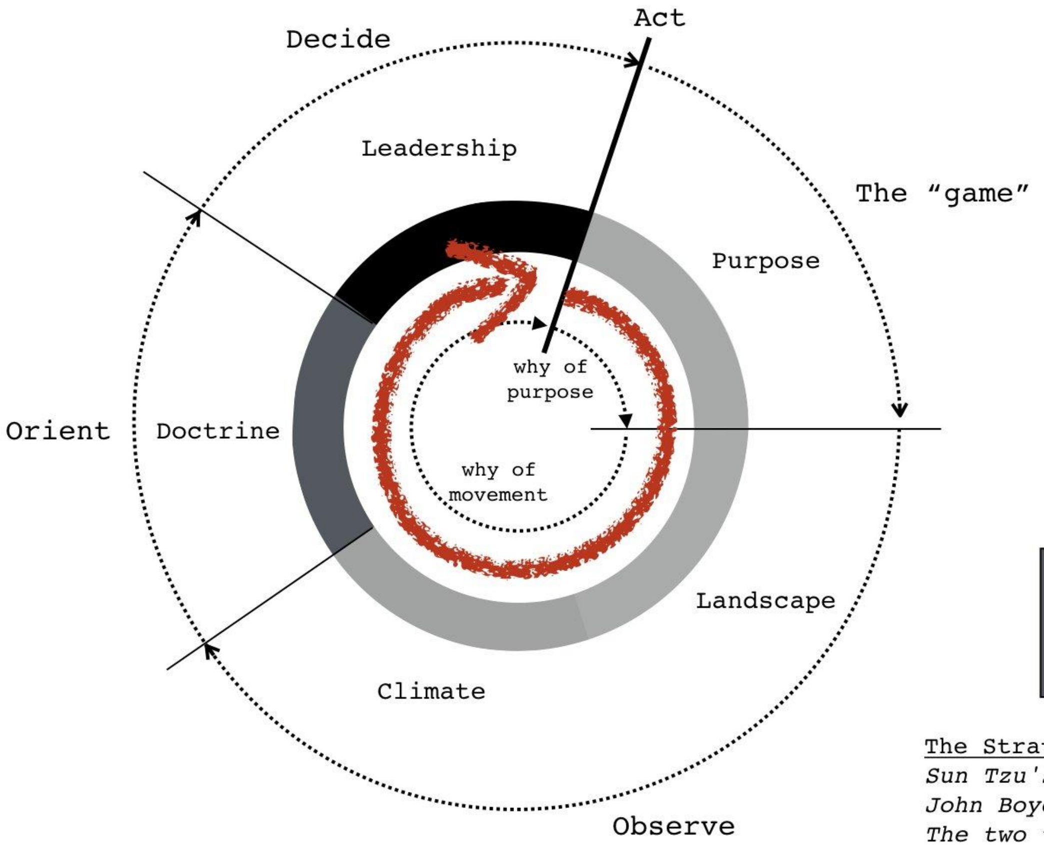
To strategize means to **identify essential issues that trouble your industry**, and to consequently **make a plan and take result-oriented action towards fixing those** critical points.

Good strategy

- Diagnosis
- Guiding Policy
- Set of Coherent Actions

Bad strategy

- Fluff / Memes
- Failure to face the challenge
- Mistaking goals for strategy
- Bad strategic objectives



Doctrine

Strategy is iterative not linear

The Strategy Cycle

Sun Tzu's five factors

John Boyd's OODA loop

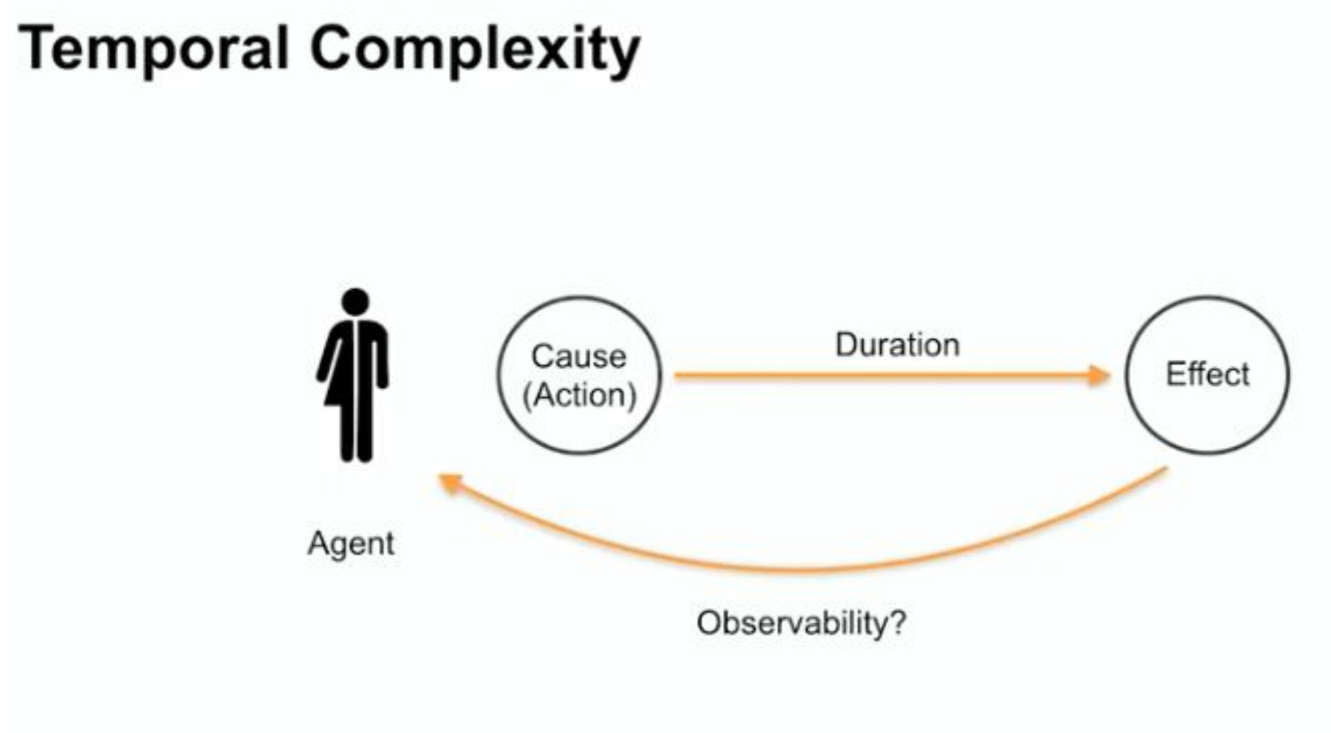
The two types of why

https://twitter.com/swardley/status/957702255842652160/photo/1

Sun Tzu’s 5 Factors and Cyber Security

5 Factors	Business	Security
Purpose	Moral imperative	Business drivers
Landscape	Environment you compete in	Sociotechnical context
Climate	Forces acting on the environment (PESTLE)	Threat landscape, vendor ecosystem and economic forces (PESTLE)
Doctrine	Training of forces, standard ways of operating	Good management, applied to context
Leadership	Strategy you choose considering purpose, landscape, climate and own capabilities. The “battle at hand”	Security Programme Management Incident Response and Crisis Management

The Strategist & The Architect



Temporal Complexity

Agent

Cause (Action)

Duration

Effect

Observability?

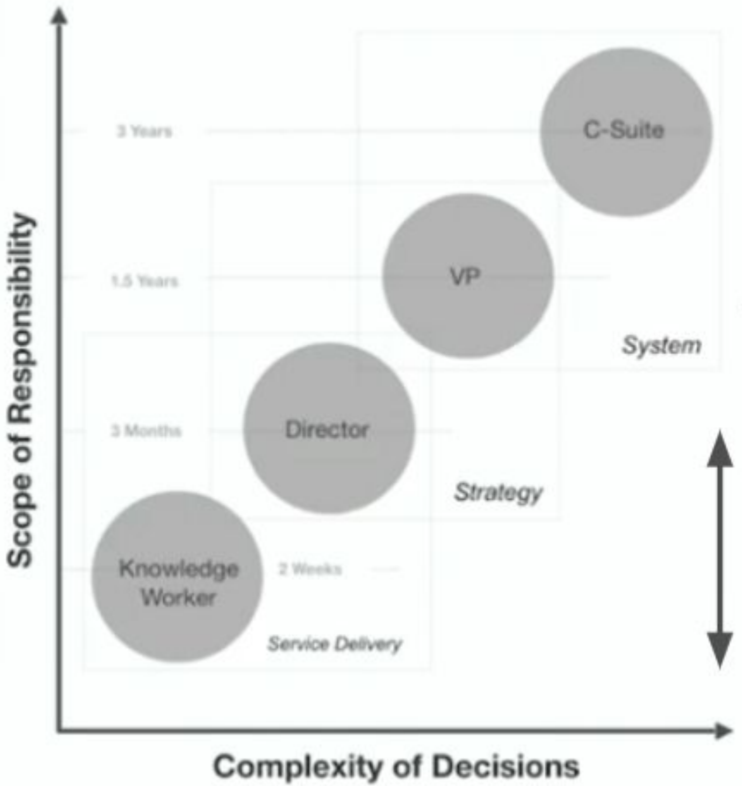
Jabe Bloom @cyetain
https://www.youtube.com/watch?v=WtfncGAeXWU

https://medium.com/@marioplatt/social-practices-and-timespan-of-discretion-in-cyber-security-cef4fdc16663

The Strategist & The Architect

Strategist

Architect



Wardley Mapping
Cynefin

Risk Management

Threat Modelling

Adapted from Jabe Bloom @cyetain
https://www.youtube.com/watch?v=WtfncGAeXWU

https://medium.com/@marioplatt/social-practices-and-timespan-of-discretion-in-cyber-security-cef4fdc16663

Developing Strategy

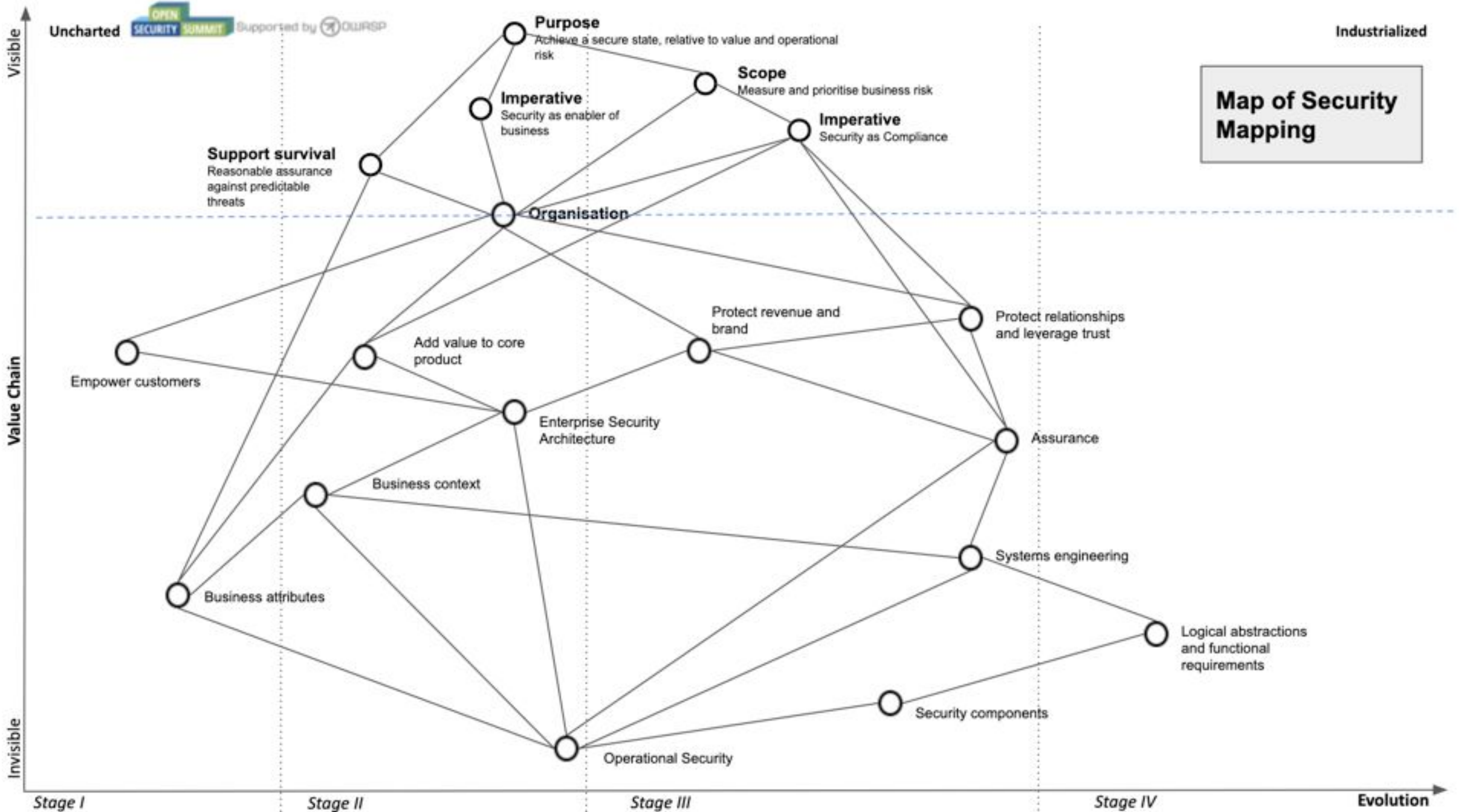
Strategy Development

OPEN SECURITY SUMMIT

- What keeps senior stakeholders awake at night ?
- Why does the company need security ?
  - Relationships with government, regulators, auditors, commercial orgs ?
  - Relationships with media, employees, activist groups ?
- Currently identified risks
- Effectiveness of risk management
- Business critical success factors ?
- People and Technological transformation ?
- Business results and competition ?

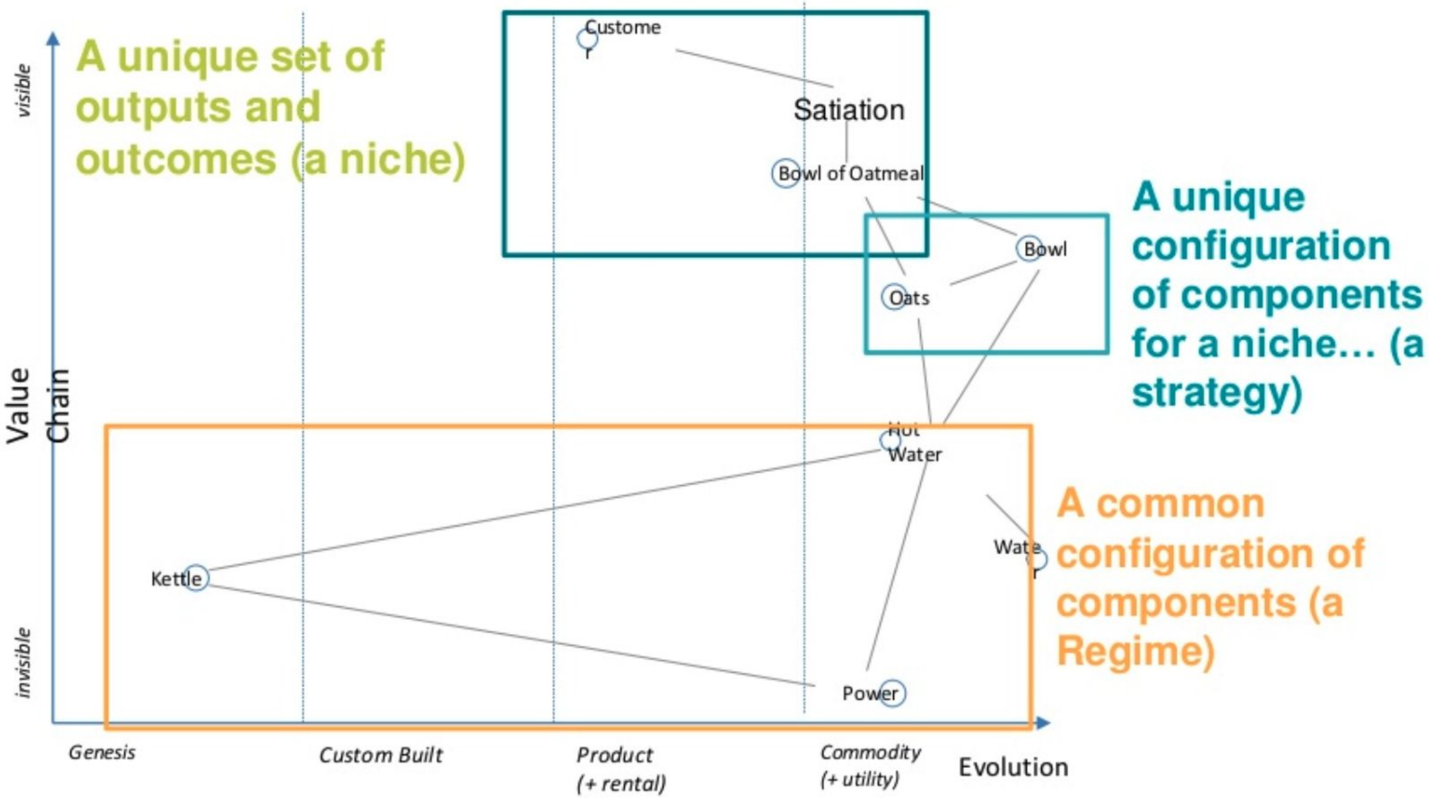
Security framed as business enablement, not as deployment of controls

Strategy Development



Visible
Value Chain
Invisible
Uncharted
Supported by OWASP
Industrialized
Map of Security Mapping
Purpose
Achieve a secure state, relative to value and operational risk
Scope
Measure and prioritise business risk
Imperative
Security as enabler of business
Imperative
Security as Compliance
Support survival
Reasonable assurance against predictable threats
Organisation
Protect revenue and brand
Protect relationships and leverage trust
Empower customers
Add value to core product
Enterprise Security Architecture
Assurance
Business context
Systems engineering
Business attributes
Logical abstractions and functional requirements
Security components
Operational Security
Stage I
Stage II
Stage III
Stage IV
Evolution

Strategy Development



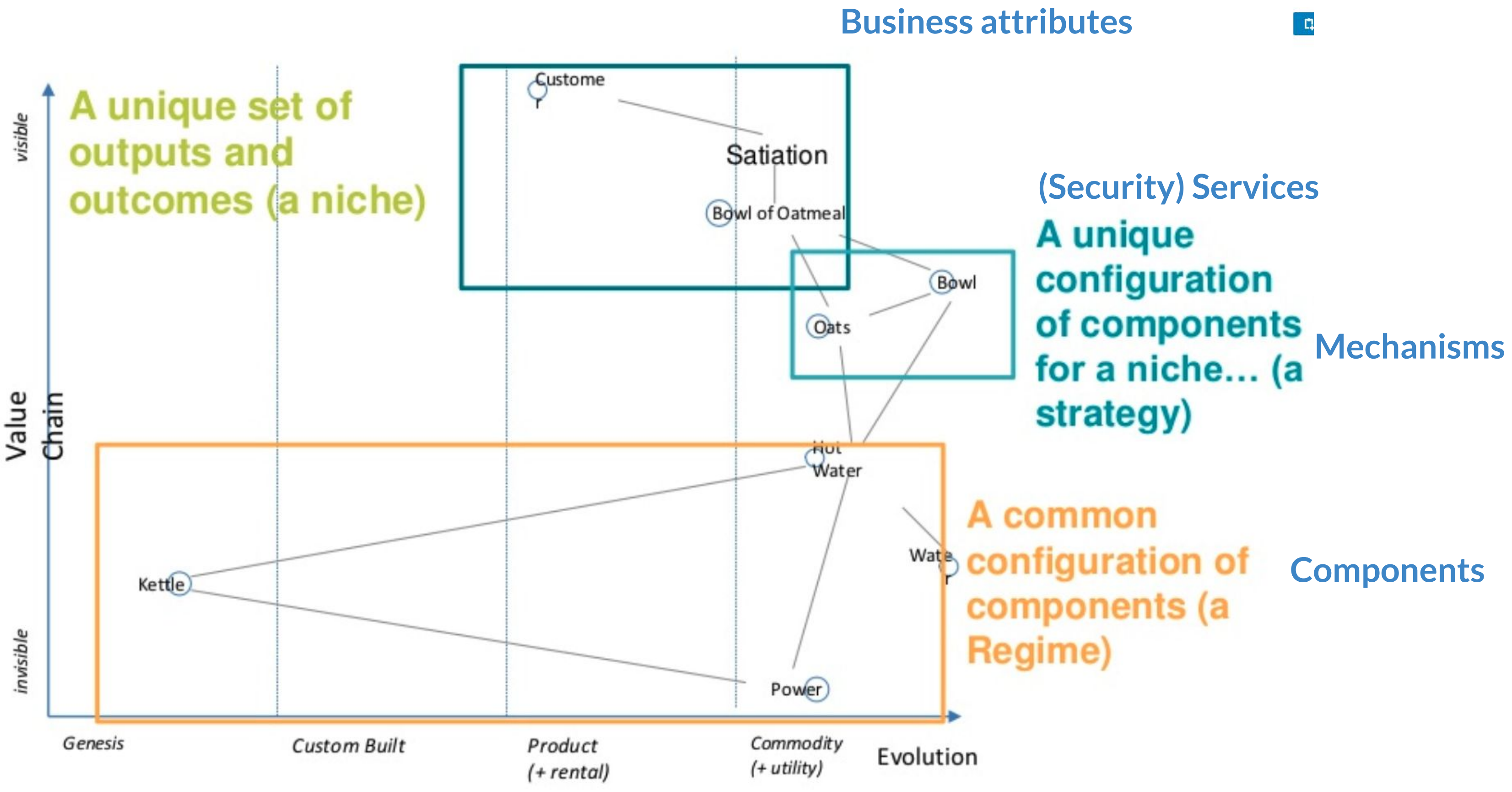
A unique set of outputs and outcomes (a niche)

A unique configuration of components for a niche… (a strategy)

A common configuration of components (a Regime)

Jabe Bloom @cyetain https://www.slideshare.net/cyetain/three-frames-devopsdays-atl/35

Strategy Development



@cyetain https://www.slideshare.net/cyetain/three-frames-devopsdays-atl/35

Key considerations

- Shorten the Feedback Loops
- Normalise attributes to business language
- Separate Services, from Mechanisms and Components (you don’t have to be part of all of them)
- Consider the implications to Operational Security

The changing role of the (Security) Architect

Organisations and Teams

OPEN SECURITY SUMMIT

“An organisation is a sociotechnical system that is shaped by the interaction of individuals and teams within it”

“The team is something that behaves differently from a mere collection of individuals”

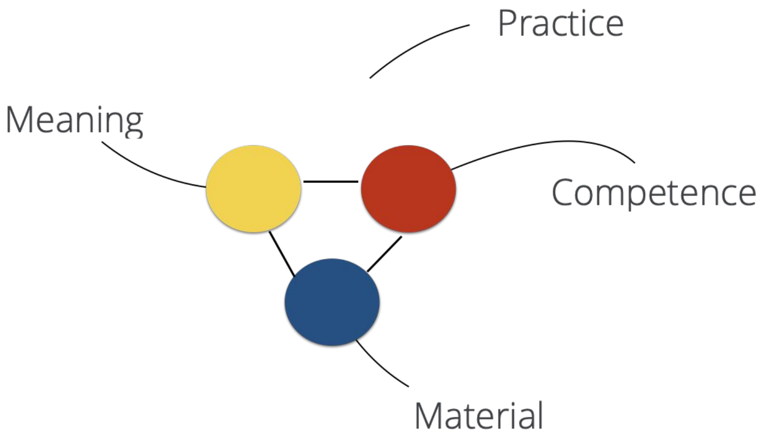
@TeamTopologies

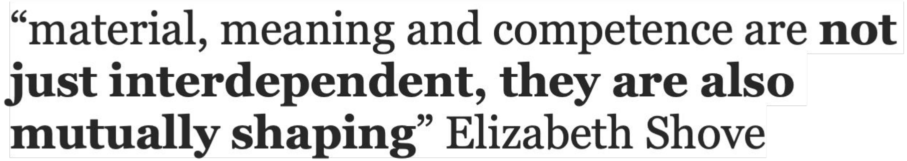
Conway’s Law

“Organisations which design systems are constrained to produce designs which are copies of the communication structures of these organisations” Conway

“If the architecture of the system and the architecture of the organisation are at odds, the architecture of the organisation wins” Ruth Malan

The Architects role




"material, meaning and competence are **not just interdependent, they are also mutually shaping**" Elizabeth Shove

Not just choosing the tech, and the standards

Designing the organisation's communication structures

Evolving the meaning of practices

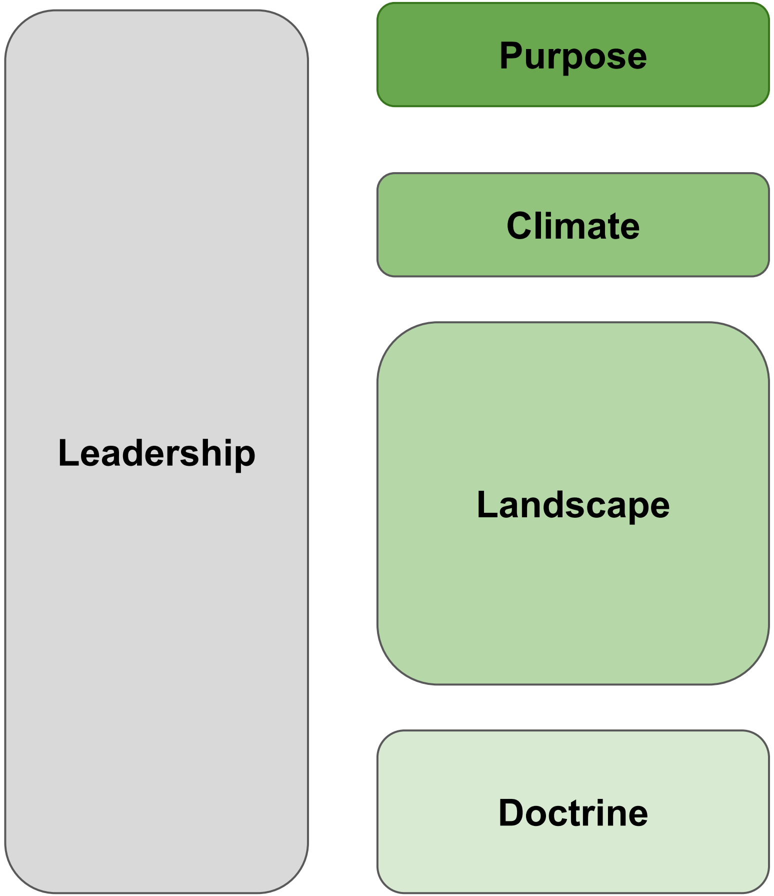
“In a DevOps world, a Pentest is not for finding security issues. It’s to improve process”

Mohammed A. Imran

~~Trust but verify~~

Trust and Informed Agency

Sun Tzu’s 5 Factors and Security Architecture



Leadership

Purpose – Business Goals, Risks, Value-chains, Governance

Climate – Business attributes, enablement and control objectives, Risk Management strategies, Process Assurance, Security & Risk Governance

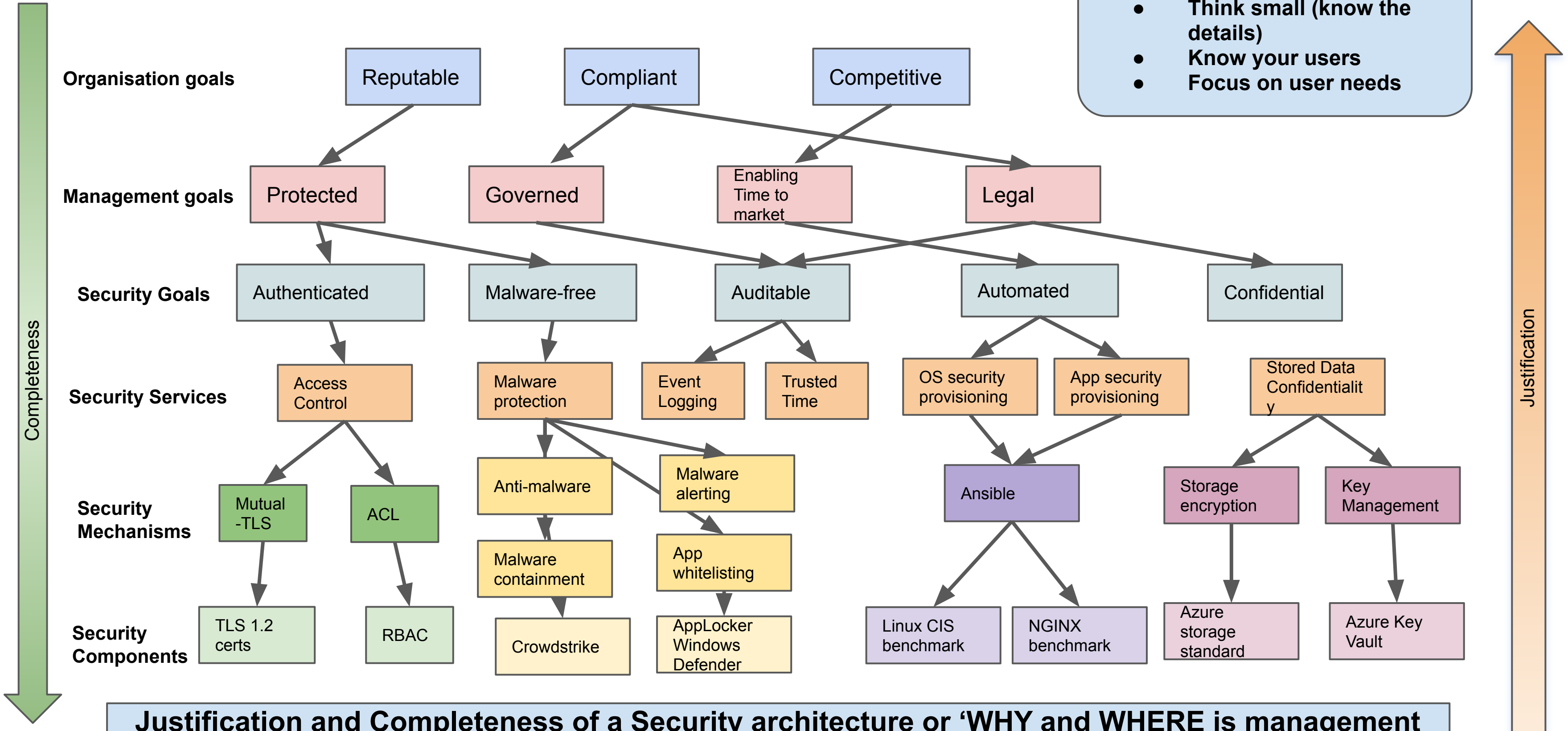
Landscape – Information & Data assets, Risk Management Policies & Practices, Process Maps/Services Trust & Domain Relationships, Human Interfaces & Teams, Component assets

Doctrine – Delivery and Continuity Mgt, Operational Risk Management, Process Delivery, Governance, Relationship and Personnel & Environment Management

Architectural traceability - SABSA layered approach

Doctrine:

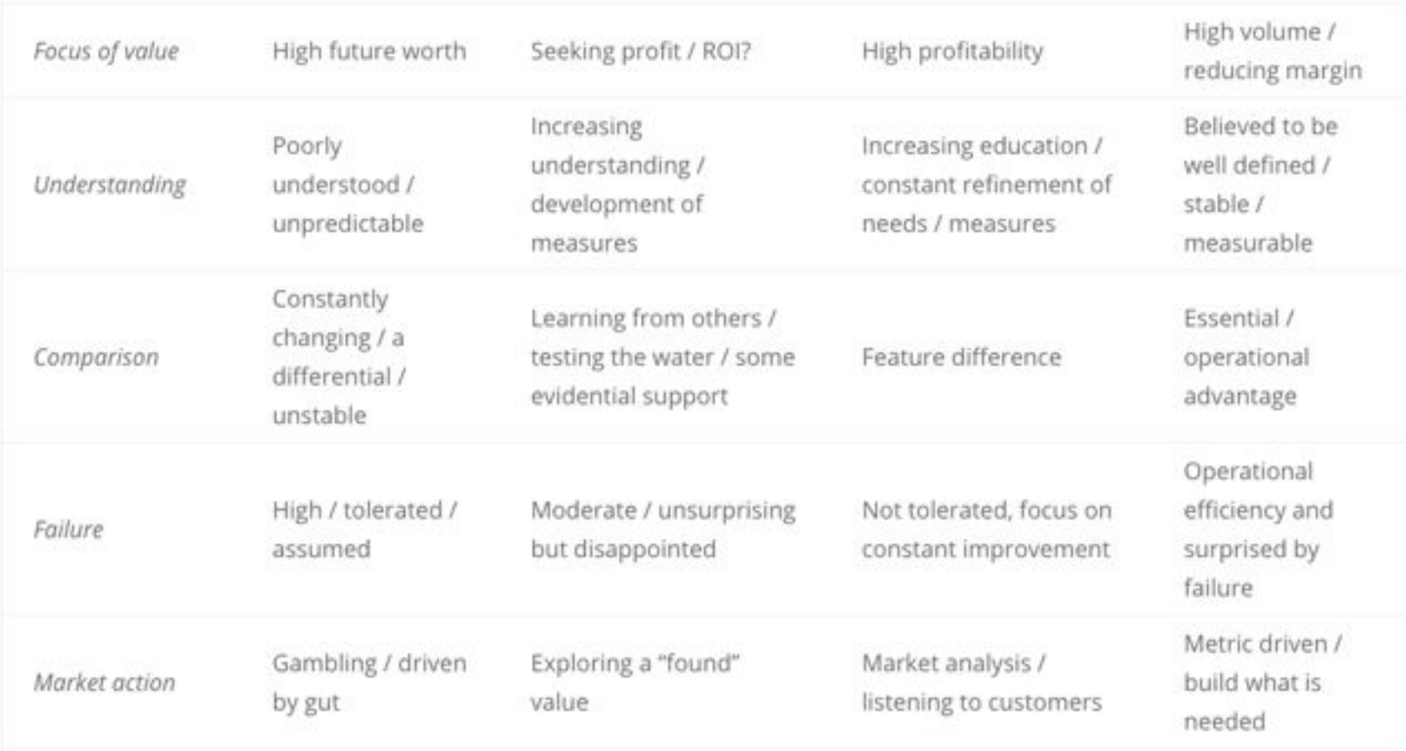
- **Think small (know the details)**
- **Know your users**
- **Focus on user needs**



Justification and Completeness of a Security architecture or 'WHY and WHERE is management spending the security budget'

Climatic Patterns - Security Architecture

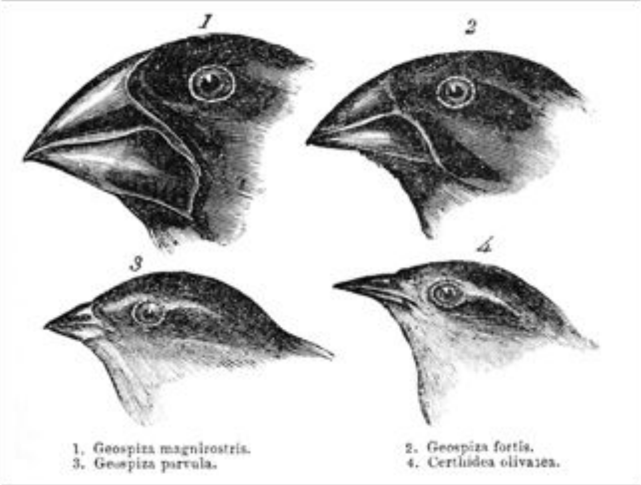
Characteristics change as capabilities evolve



	Genesis	Custom-built	Product/Rental	Commodities/ Utilities
Focus of value	High future worth	Seeking profit / ROI?	High profitability	High volume / reducing margin
Understanding	Poorly understood / unpredictable	Increasing understanding / development of measures	Increasing education / constant refinement of needs / measures	Believed to be well defined / stable / measurable
Comparison	Constantly changing / a differential / unstable	Learning from others / testing the water / some evidential support	Feature difference	Essential / operational advantage
Failure	High / tolerated / assumed	Moderate / unsurprising but disappointed	Not tolerated, focus on constant improvement	Operational efficiency and surprised by failure
Market action	Gambling / driven by gut	Exploring a "found" value	Market analysis / listening to customers	Metric driven / build what is needed

Creative Destruction

No choice over evolution



Inertia can kill an organisation

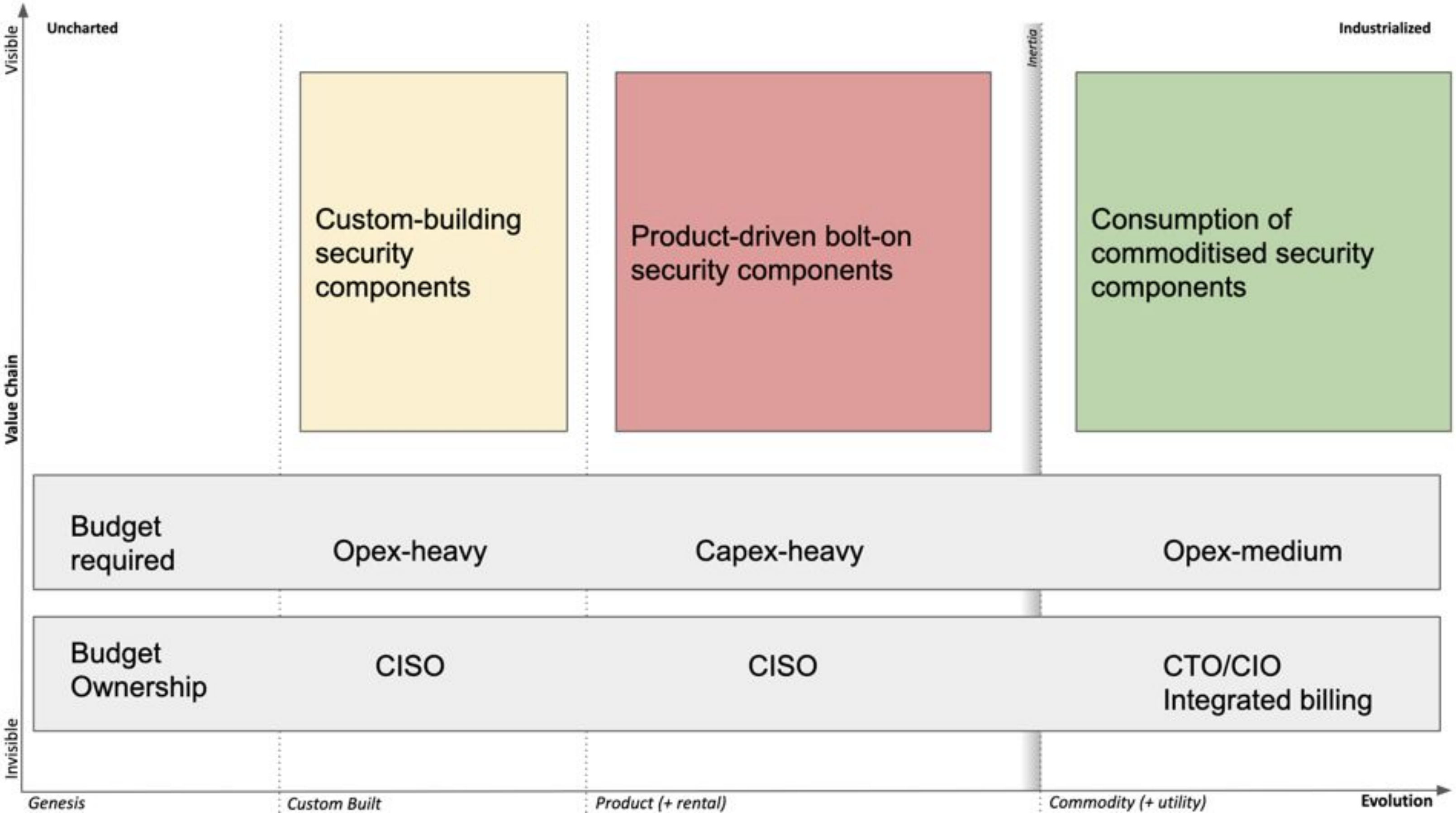
Creative Destruction

Compliance-as-Spreadsheets → Compliance-as-Code

Insecure Frameworks - Secure(r) frameworks

Security Products → Consumption of commodities or CNCF

No choice over evolution



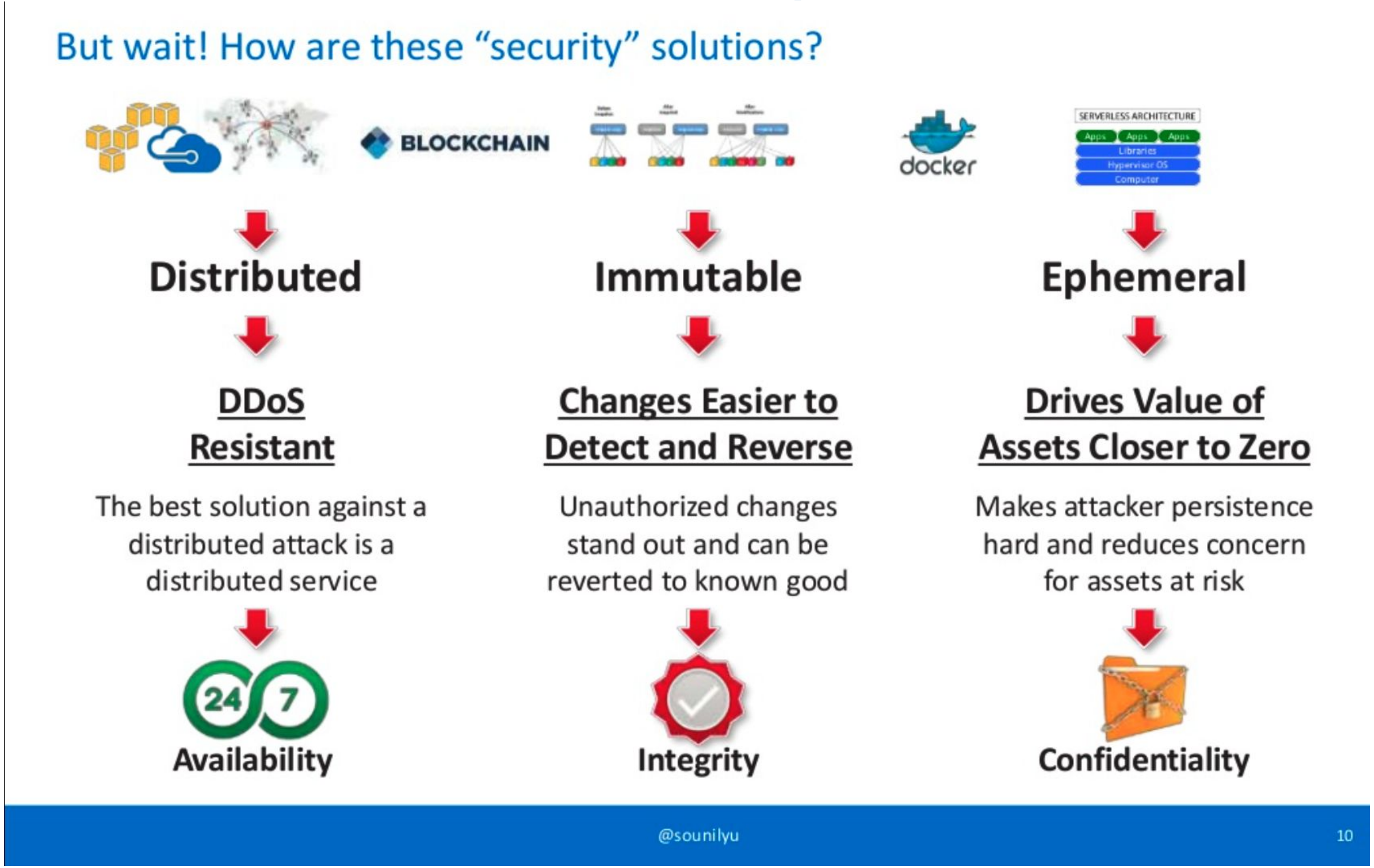
Custom-building security components

Product-driven bolt-on security components

Consumption of commoditised security components

Budget required	Opex-heavy	Capex-heavy	Opex-medium
Budget Ownership	CISO	CISO	CTO/CIO Integrated billing

Inertia due to success of past model



But wait! How are these “security” solutions?

Distributed

DDoS Resistant

The best solution against a distributed attack is a distributed service

Availability

Immutable

Changes Easier to Detect and Reverse

Unauthorized changes stand out and can be reverted to known good

Integrity

Ephemeral

Drives Value of Assets Closer to Zero

Makes attacker persistence hard and reduces concern for assets at risk

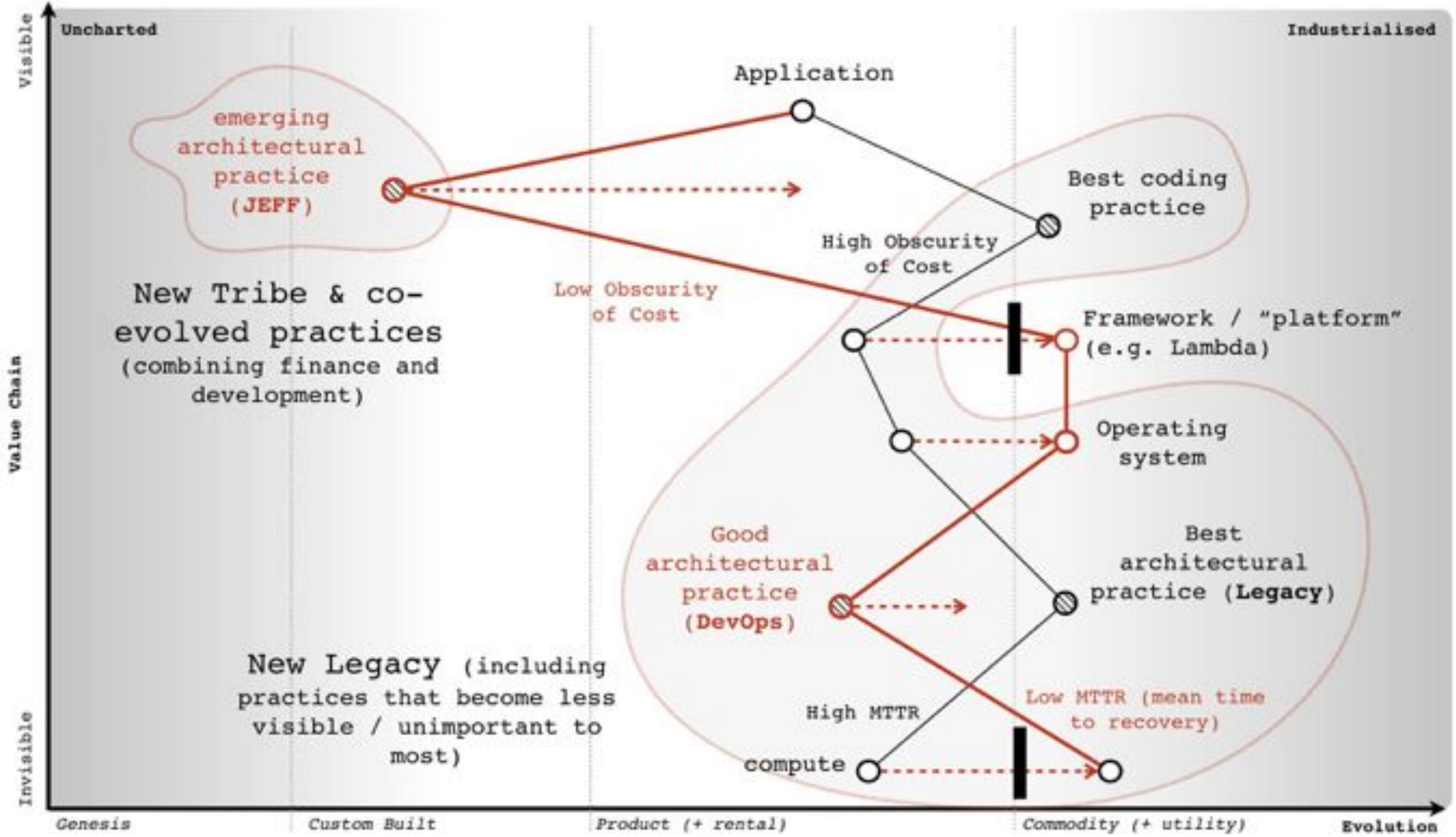
Confidentiality

@sounilyu

https://www.slideshare.net/sounilyu/distributed-immutable-ephemeral-new-paradigms-for-the-next-era-of-security

The (co-)evolution of Computing

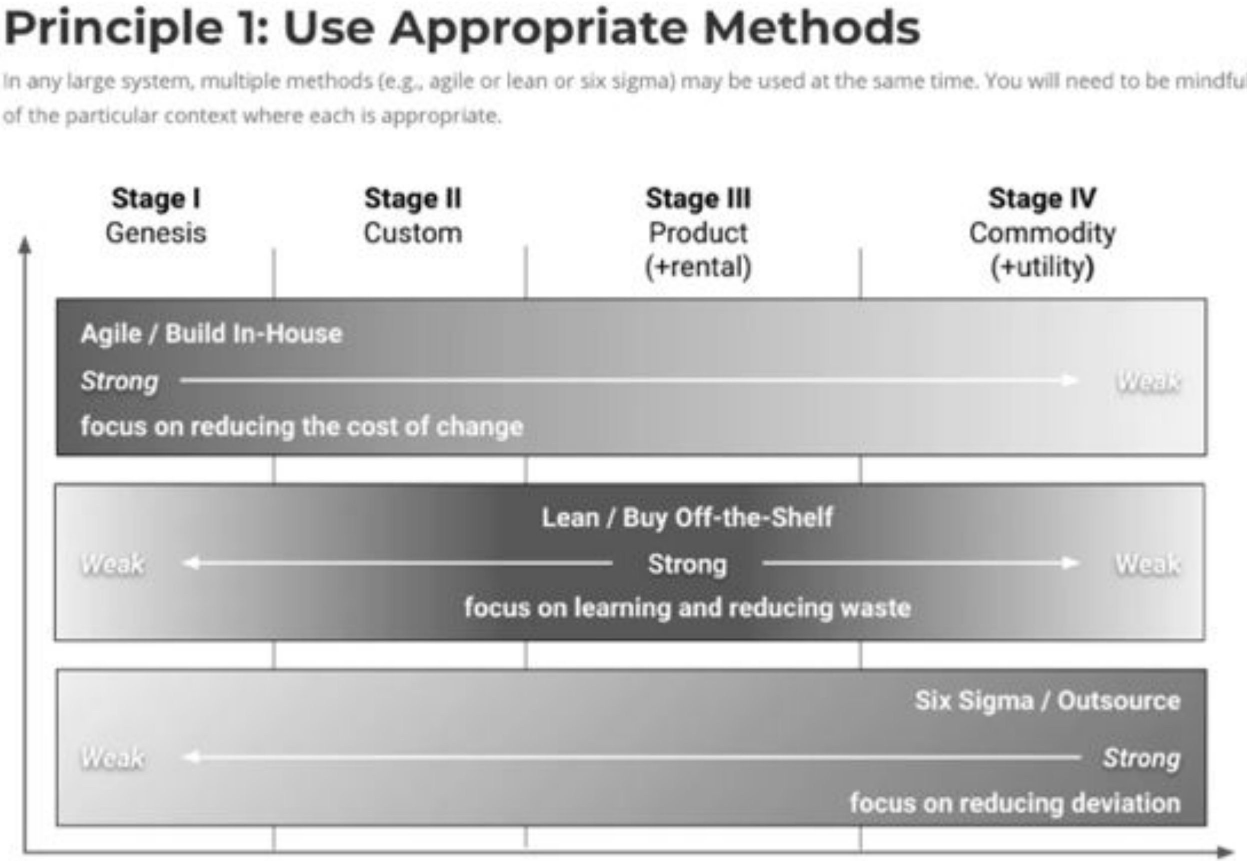
OPEN SECURITY SUMMIT


Visible
Value Chain
Invisible
Uncharted
Industrialised
Application
emerging architectural practice (JEFF)
Best coding practice
High Obscurity of Cost
New Tribe & co-evolved practices (combining finance and development)
Low Obscurity of Cost
Framework / "platform" (e.g. Lambda)
Operating system
Good architectural practice (DevOps)
Best architectural practice (Legacy)
New Legacy (including practices that become less visible / unimportant to most)
High MTTR
Low MTTR (mean time to recovery)
compute
Genesis
Custom Built
Product (+ rental)
Commodity (+ utility)
Evolution

https://twitter.com/swardley/status/914792448429297664

Use Cases in context of Strategy and Architecture

Use appropriate methods



Principle 1: Use Appropriate Methods

In any large system, multiple methods (e.g., agile or lean or six sigma) may be used at the same time. You will need to be mindful of the particular context where each is appropriate.

Stage I Genesis

Stage II Custom

Stage III Product (+rental)

Stage IV Commodity (+utility)

Agile / Build In-House

Strong → *Weak*

focus on reducing the cost of change

Lean / Buy Off-the-Shelf

Weak ← Strong → Weak

focus on learning and reducing waste

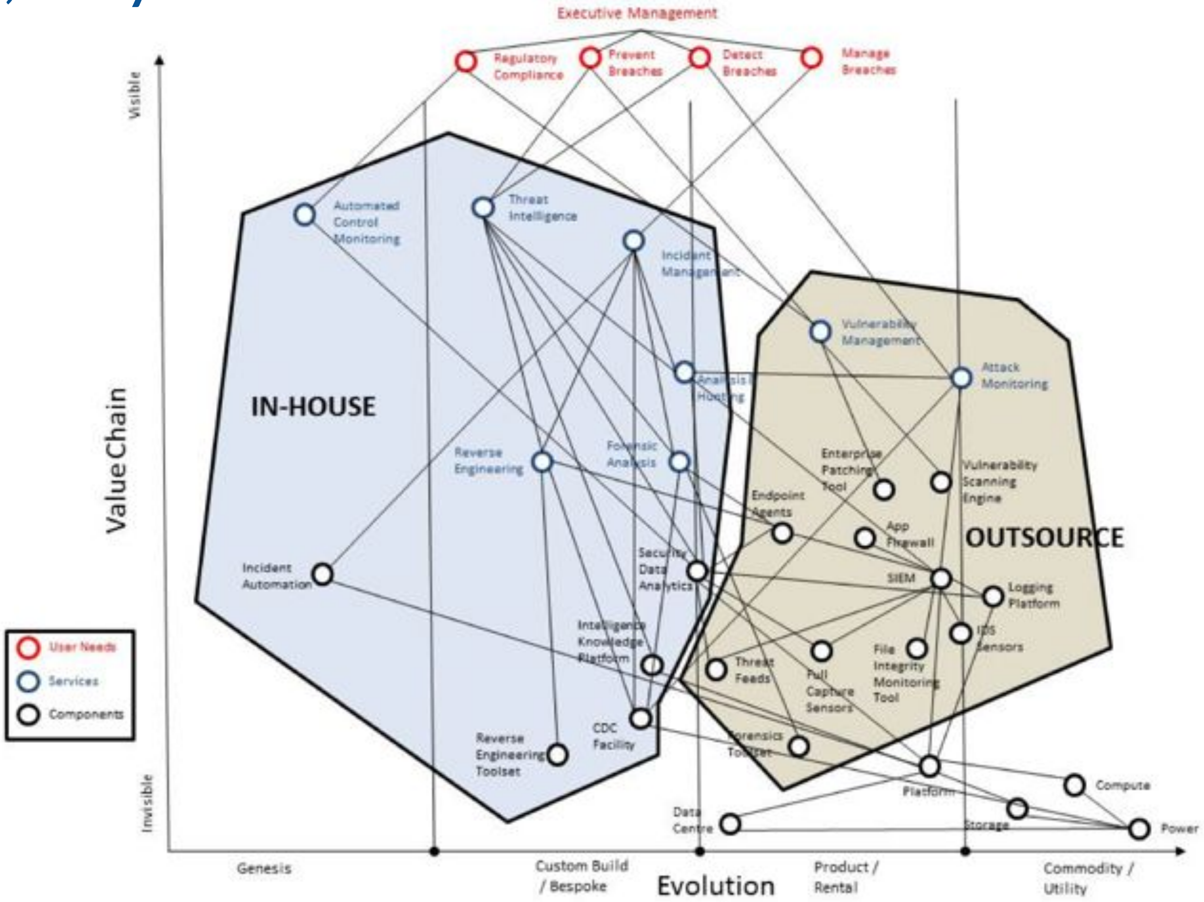
Six Sigma / Outsource

Weak ← *Strong*

focus on reducing deviation

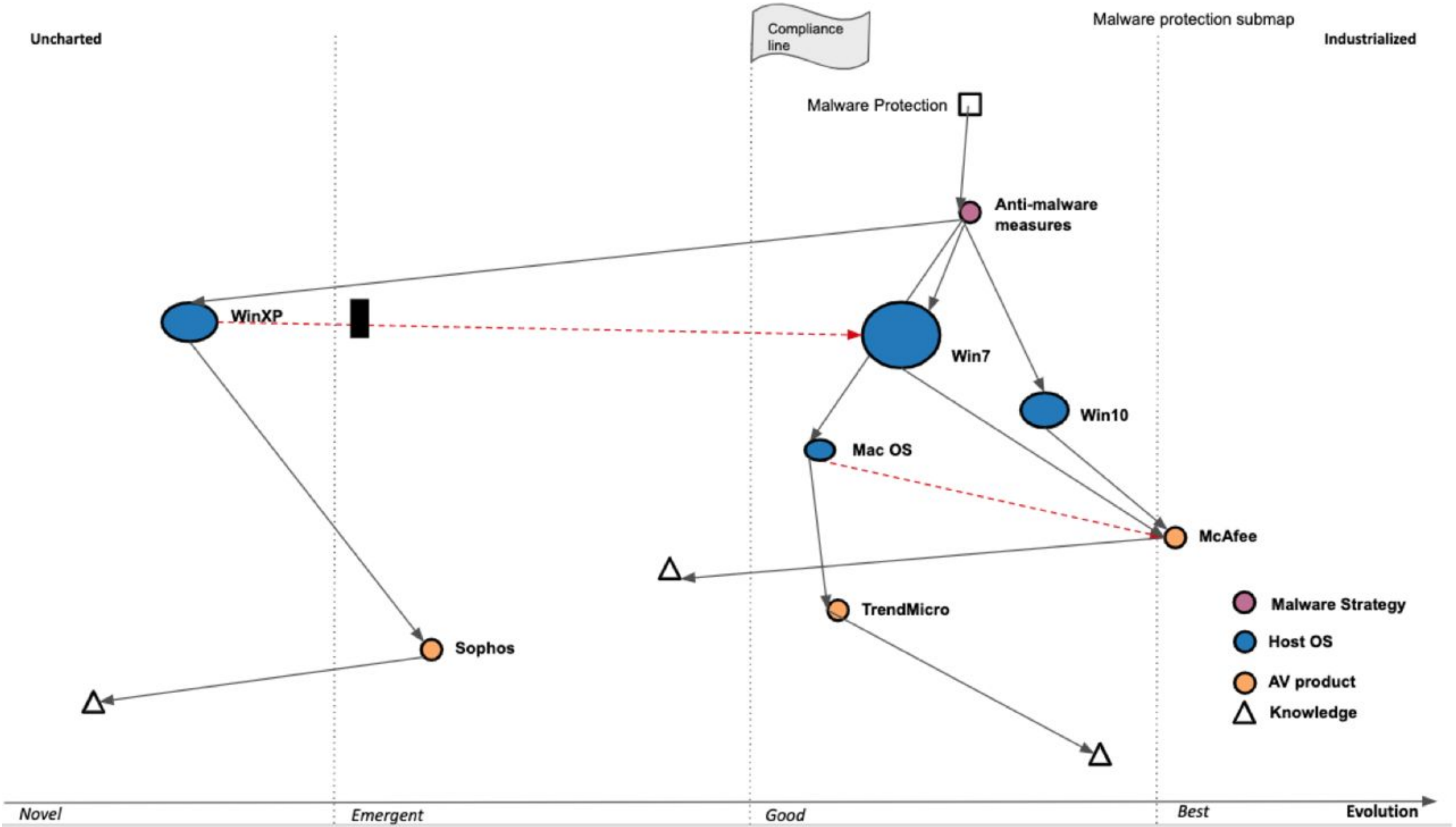
Courtesy of Ben Mosior @ LearnWardleyMapping.com

Build, Buy or Outsource

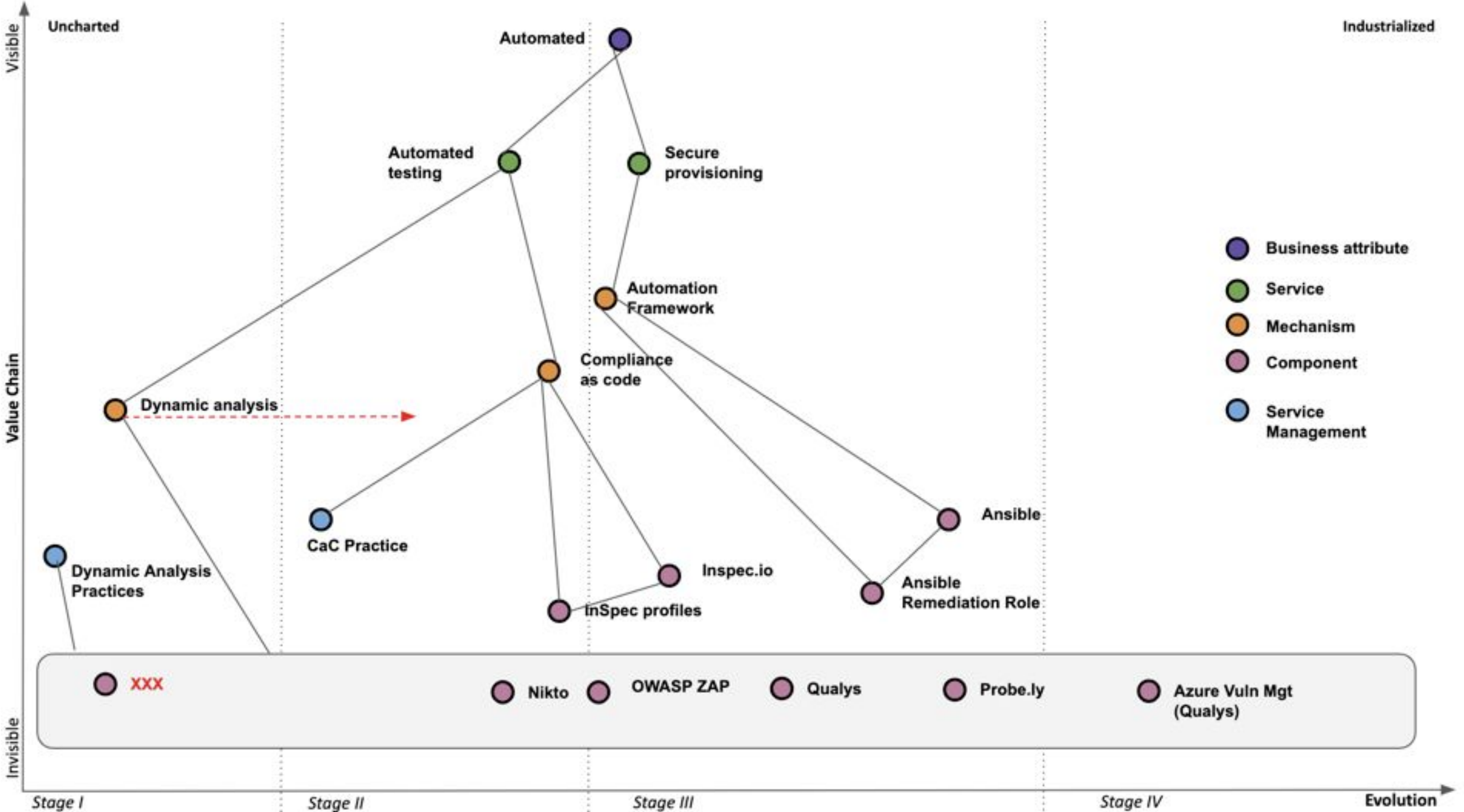


https://blog.blackswansecurity.com/2016/01/soc-value-chain-delivery-models/

Visualise a landscape

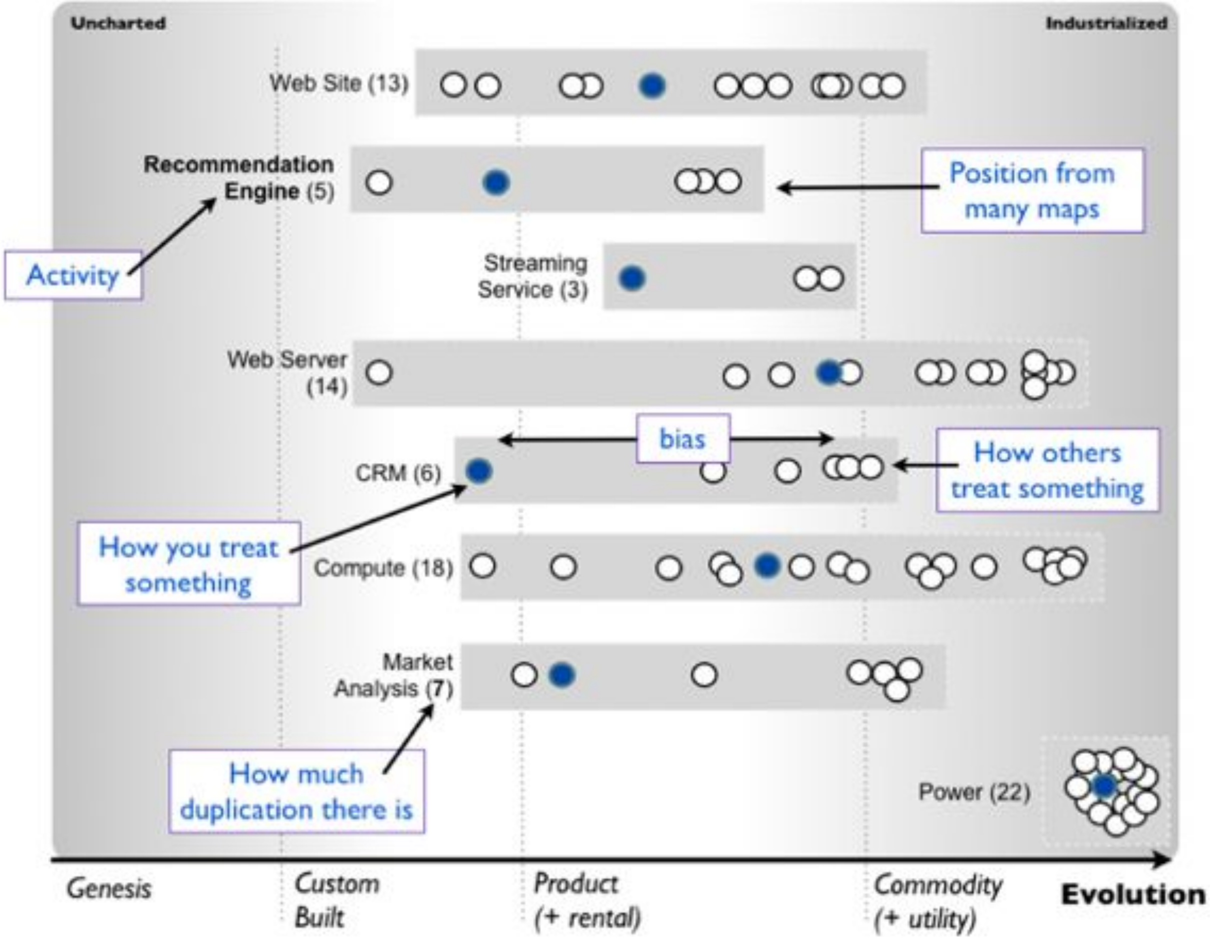


Options Analysis

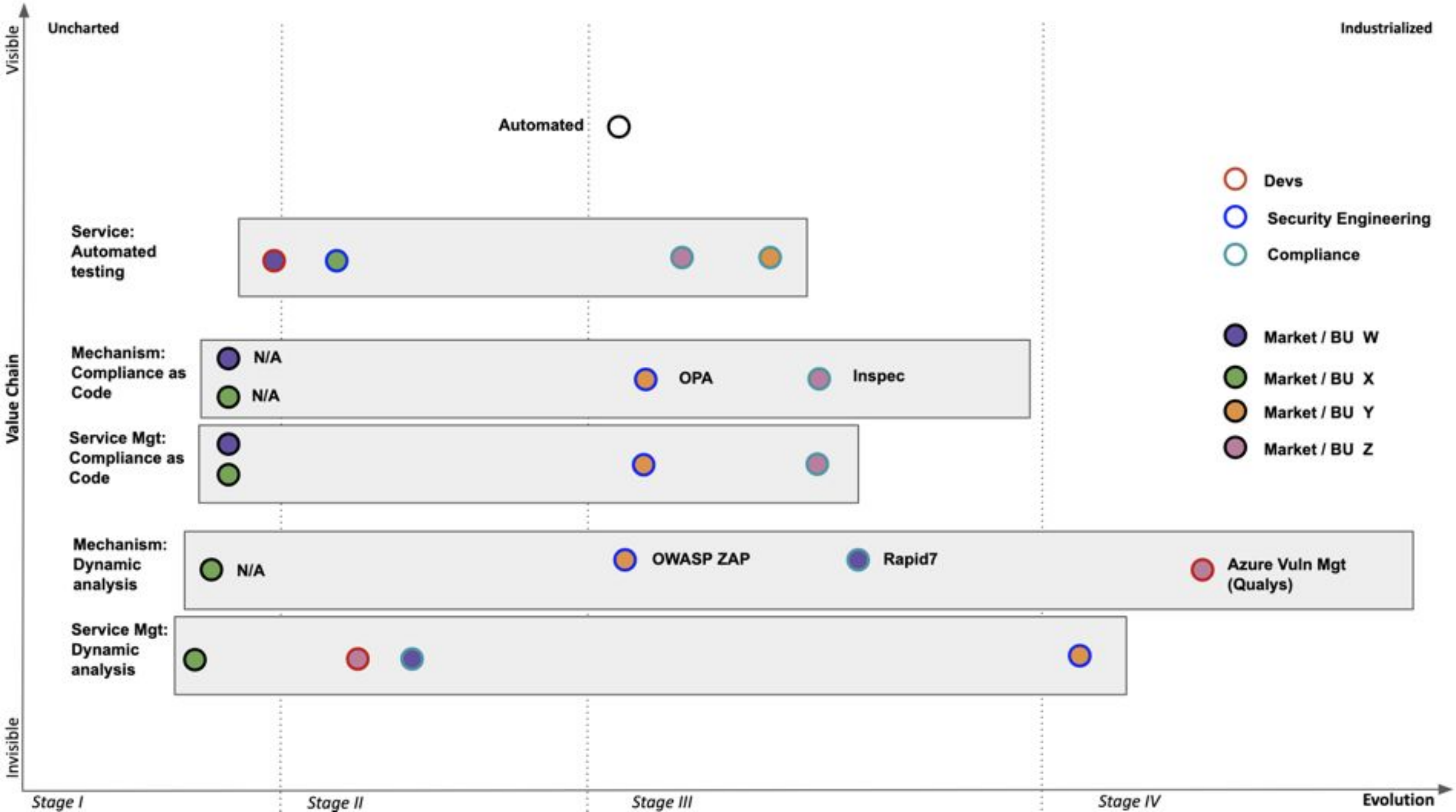


Duplication and Bias

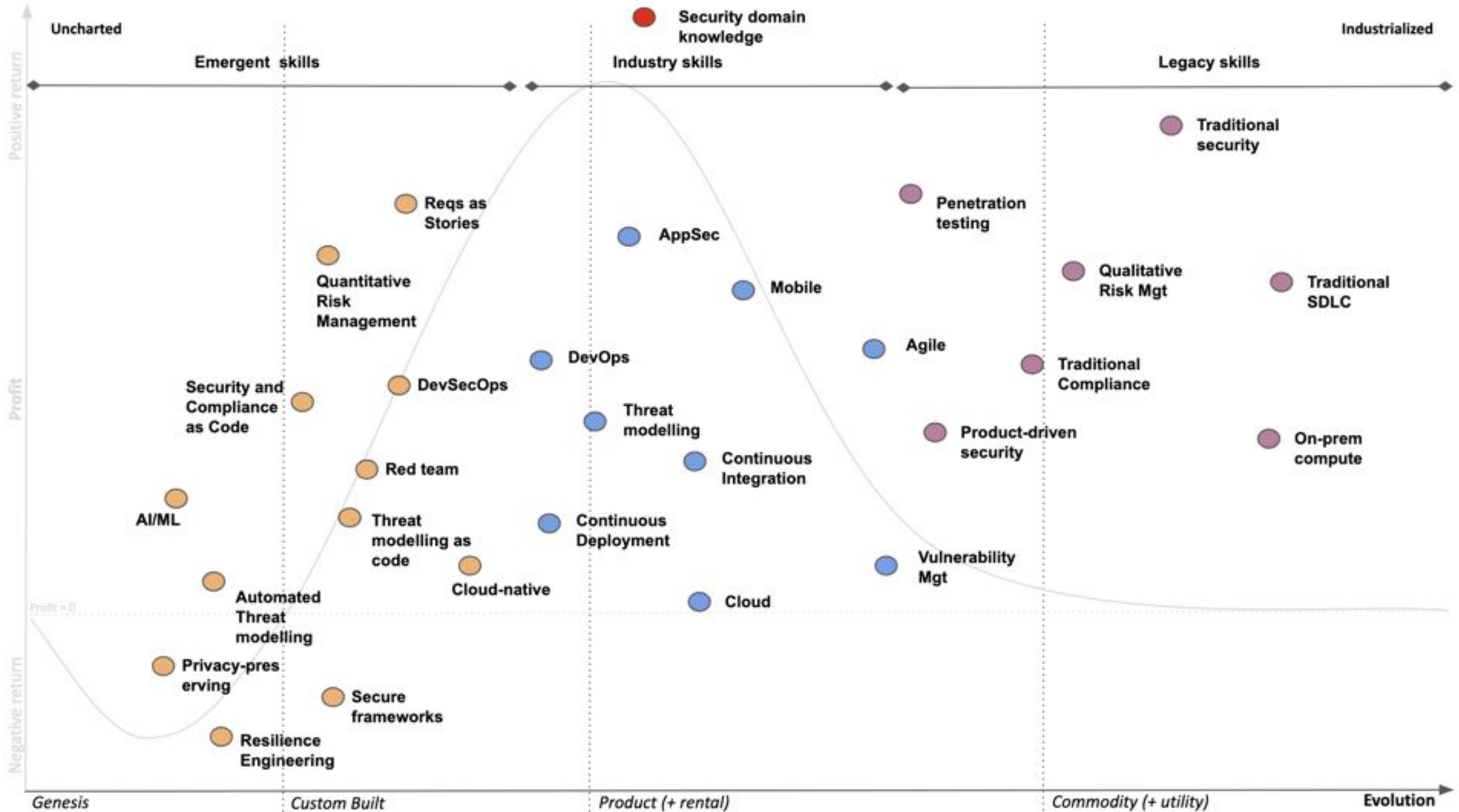
OPEN SECURITY SUMMIT



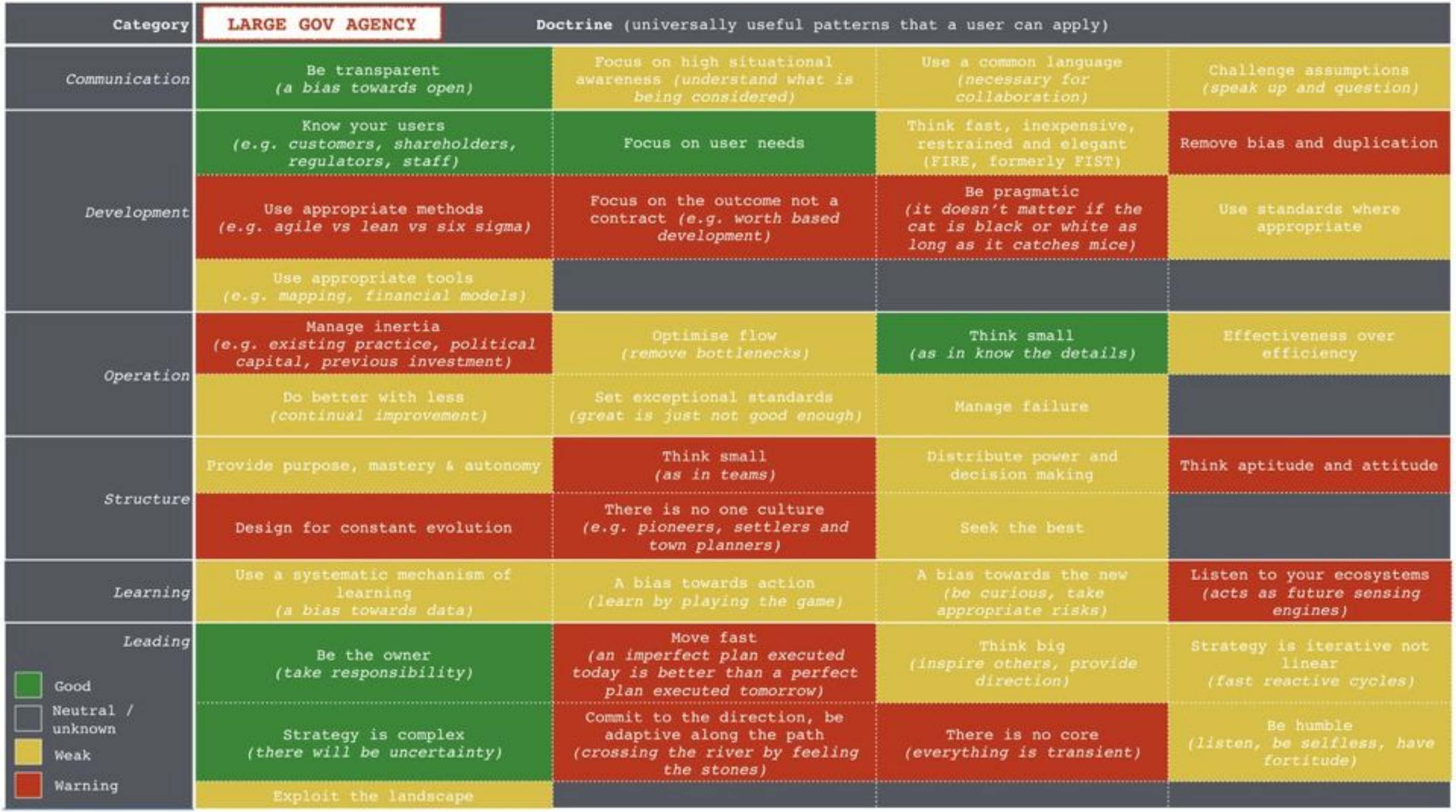
Duplication and Bias



Emergence of skills and practices



Assessing Doctrine



Category	LARGE GOV AGENCY	Doctrine (universally useful patterns that a user can apply)		
Communication	🟩 Be transparent *(a bias towards open)*	🟨 Focus on high situational awareness *(understand what is being considered)*	🟨 Use a common language *(necessary for collaboration)*	🟨 Challenge assumptions *(speak up and question)*
Development	🟩 Know your users *(e.g. customers, shareholders, regulators, staff)*	🟩 Focus on user needs	🟨 Think fast, inexpensive, restrained and elegant (FIRE, formerly FIST)	🟥 Remove bias and duplication
	🟥 Use appropriate methods *(e.g. agile vs lean vs six sigma)*	🟥 Focus on the outcome not a contract *(e.g. worth based development)*	🟥 Be pragmatic *(it doesn't matter if the cat is black or white as long as it catches mice)*	🟨 Use standards where appropriate
	🟨 Use appropriate tools *(e.g. mapping, financial models)*			
Operation	🟥 Manage inertia *(e.g. existing practice, political capital, previous investment)*	🟨 Optimise flow *(remove bottlenecks)*	🟩 Think small *(as in know the details)*	🟨 Effectiveness over efficiency
	🟨 Do better with less *(continual improvement)*	🟨 Set exceptional standards *(great is just not good enough)*	🟨 Manage failure	
Structure	🟨 Provide purpose, mastery & autonomy	🟥 Think small *(as in teams)*	🟨 Distribute power and decision making	🟥 Think aptitude and attitude
	🟥 Design for constant evolution	🟥 There is no one culture *(e.g. pioneers, settlers and town planners)*	🟨 Seek the best	
Learning	🟨 Use a systematic mechanism of learning *(a bias towards data)*	🟨 A bias towards action *(learn by playing the game)*	🟨 A bias towards the new *(be curious, take appropriate risks)*	🟥 Listen to your ecosystems *(acts as future sensing engines)*
Leading	🟩 Be the owner *(take responsibility)*	🟥 Move fast *(an imperfect plan executed today is better than a perfect plan executed tomorrow)*	🟨 Think big *(inspire others, provide direction)*	🟨 Strategy is iterative not linear *(fast reactive cycles)*
	🟩 Strategy is complex *(there will be uncertainty)*	🟥 Commit to the direction, be adaptive along the path *(crossing the river by feeling the stones)*	🟥 There is no core *(everything is transient)*	🟨 Be humble *(listen, be selfless, have fortitude)*
	🟨 Exploit the landscape			

🟩 Good

⬛ Neutral / unknown

🟨 Weak

🟥 Warning

https://allthebest.recipes/uploads/db5237/original/1X/48ac2be9d579f0df1d39e78a2acc1c007fbe8b45.jpg

Closing Thoughts

Why Wardley Map ?

DIALOGUE > ARTEFACT

PROVIDES SHARED VOCABULARY & PATTERNS IN BUSINESS

FORCES ONE TO EXPOSE ASSUMPTIONS, BIAS AND INVITES CHALLENGE

DE-PERSONALISES THE CHALLENGE

Using the Wardley Map

- Build-and-bin
- Build-and-maintain

Key benefits for Security Architecture

- See and discuss a landscape
- Assess evolution in context and anticipate change
- Patterns for effective management and for the process of managing (constant) change

Wardley mapping is a great companion and supplement to your Security Architecture and a brilliant tool to help you develop an appropriate Strategy

Q&A

Mario Platt

mario@practical-devsecops.com

Twitter: @madplatt
LinkedIn: marioplatt
Medium: @marioplatt