YOU ARE HERE

Leading your Security Program with Wardley Maps

John Duffy - duffy.dev

So you are the key security person

- Every major security decision is run through you
- Every incident is your responsibility
- Understanding of risk to the companies operations comes down to you
- You have a plan – and you are executing it.

*And then…a **big** incident.*

- You get hit by Cryptolocker malware – and you start to wonder
  - Are the teams resourced properly to respond? - $$$, time, planning, skillsets
  - Did I do everything I could to protect against this incident?

- When you have a chance to breathe – you wonder some more
  - Are we focused on the right priorities?
  - Will I get more resources to handle the situation - or less?

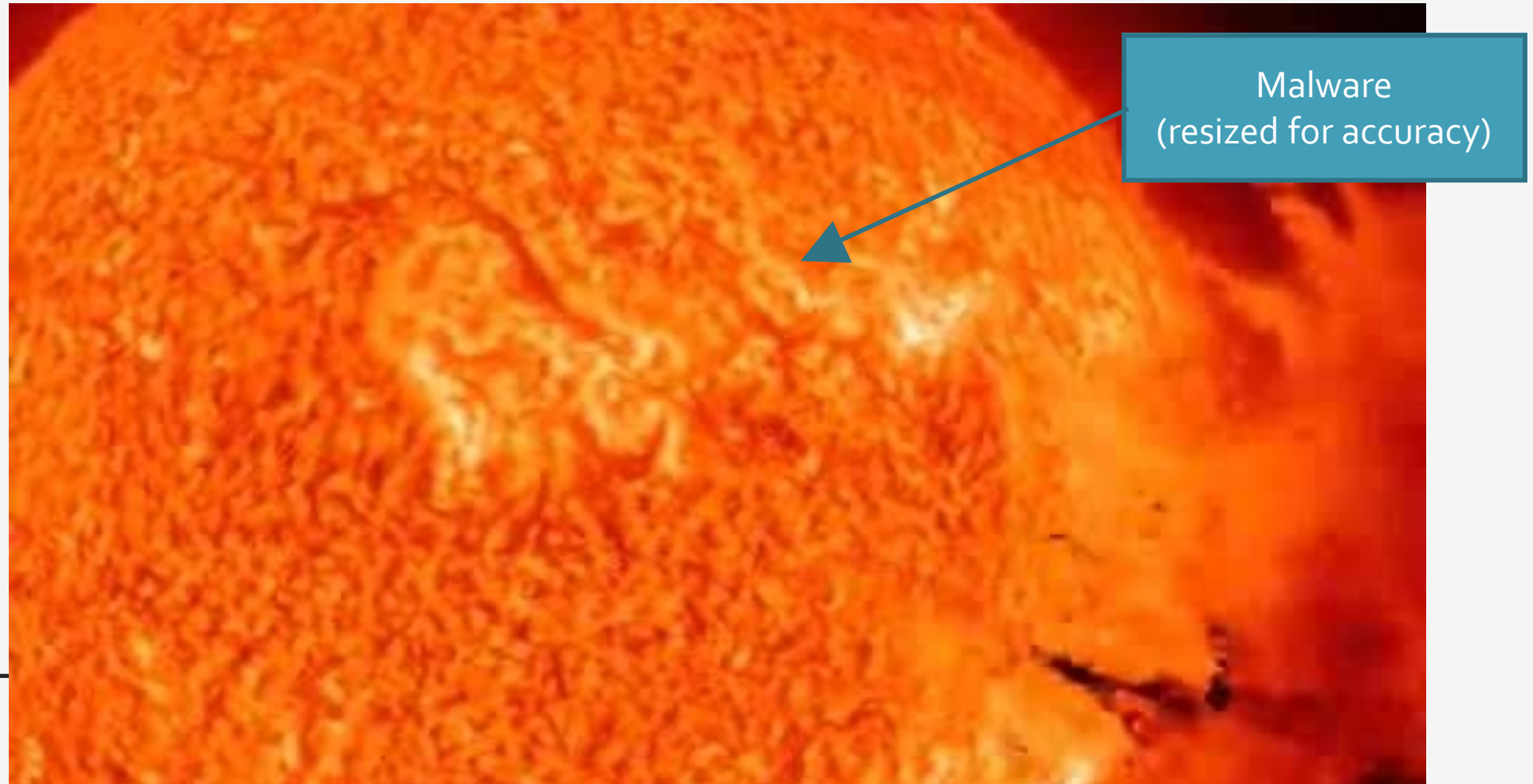
- **Is the plan still valid? Am I doing a good job? Who gets to judge this?**

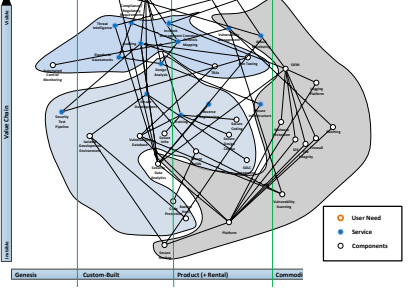
Cryptlocker malware is a great test-case

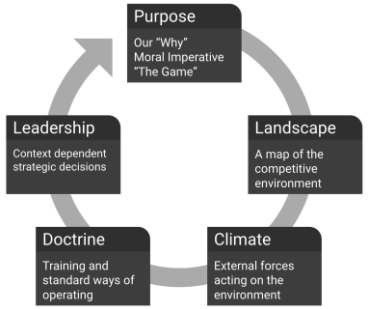
- At the master-plan level – it's a dot in the sky

Cryptlocker malware is a great test-case

- At the master-plan level - if it occurs…..







BSIMM10

This talk is about True North

- Wardley Maps are our best effort at answering these questions
  - Are we focusing on the right priorities?
  - What needs to be in-group, in-house and outsourced?
  - How do we put on-fire and new topics into alignment with our overall vision?

- It's the story of how we used different methods to answer these questions:
  - Best Practice
  - Control Based Approach (ISO 27001)
  - Maturity Model Approach (BSIMM)
  - Wardley Maps & The Strategy Cycle
- It's the story of how I was finally able to sleep at night.

Who am I

- Developer, Architect, Cryptography, Marketing, Management
  - I'm a builder with a researching problem
- Handed Keys to Security Program
  - "Responsible for establishing, implementing and maintaining the security management strategy, research, technologies and implementation that protects all products, systems and information"
- Out of the gate - I was confident had a clear direction
  - "If you want to make the security-gods laugh, tell them about you have a plan"
  - "When you're out at sea, all landmarks are blue"

Who is CBN

- We provide secure solutions for Payment, Identity and Lottery markets worldwide
  - Couple hundred technical staff located in multiple locations in Canada, US and Europe
  - Deployed in 80 countries, millions of daily-users and systems with 50,000+ VMs
  - Users range from technical teams to front-line service to plant-operations personnel
  - Capabilities include: firmware, cryptography, biometrics, pen-testing, classified network security, malware analysis, manufacturing systems, physical product development, digital printing features and projects that include RF and optics.

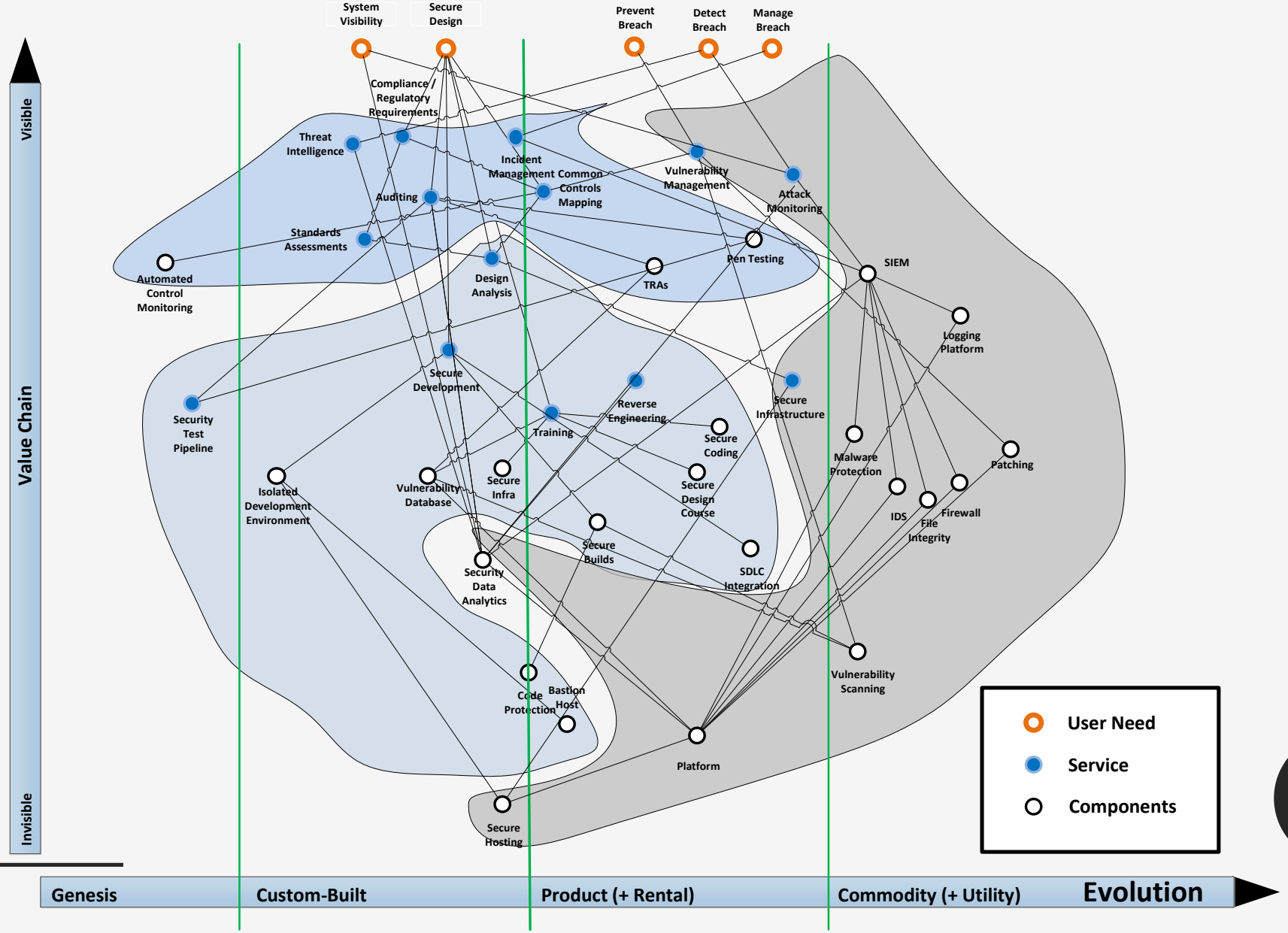
- CBN is really, **really** customer focused – this creates challenges for Security
  - This means we do a great deal of customization for each customer
  - We need to work in multiple jurisdictions and implement multiple standards
  - Customer's range from "Let's make our own standard" to "We'll deploy the tried and true"
  - I've worked on mobile travel credentials and 2D bar codes on printed paper

Setting up a Security Program

Welcome to my world

- Allows us to understand:
  - What needs to be done
  - Who needs to do it
  - What our users want
  - Where we're going
  - Where components are going
  - What's valuable to us

- We fit everything into this map



Approaches to Security Program Strategy

- Security Program is really about your present and future security priorities

- Several methods to developing a Security Program

Method	Description	Example Techniques
Best Practice	A large risk-assessment and controls based on best practice	Experience in the trenches
Controls based	Align to a standard and using the evidence to show competence	ISO 27001
Maturity Model	Structured form of "best practice" that incorporates metrics	BSIMM
Dashboarding	Tracking KPIs or Activities	Mapping

- Security programs also have different phases [from BSIMM 10]:
  - **Emerging**: Formalizing processes and ad-hoc security activities into a holistic strategy
  - **Maturing**: Existing security approach with executive expectations for risk and investment
  - **Optimizing**: Fine-tuning security capabilities with clear view into operational expectations (and business value)

Best practice - 100 Day Plan

- Numerous "100 First Days Roadmaps" out there
  - Establish the program's maturity
  - Understand Security's role in the business
  - Identify security experts and champions
  - Deliver a vision for the program

Phase	Key Activities	Key Outcomes
Observe	Interviews	Know the players, Risk Register
Orient	Tabletop, Risk Assessments	Establish 5 priorities
Decide	Two short-term priorities (6-12mth) Two long-term priorities (12-24mth)	Strategy that links priorities to business
Act	Communicate the strategy	Get buy-in

“You can’t just do a good job. You have to communicate you are doing a good job”

Start Phase 2 – Ongoing Maintenance

- In theory, you keep running Risk Assessments, Security Testing (Pen Tests) and Incident Response
  - Address issues as they surface
  - Feed recommendations back to teams

- In practice, everything hits at once with varying impact
  - You're biggest issue might be a software defect class (XXE)
  - Running pen-tests on 1000 applications might be a resource issue
  - Teams may run DevOps, Agile, Waterfall across applications in R&D, Product and Support phases
  - Fixes are applied differently by different teams and deploy at different times

- You need something more structured to break out of this maintenance cycle

Controls-Based Security Program

Controls based approach: ISO27001

- We have deployed ISO27001 at several sites and were thinking of expanding to provide the structure

PROs	CONs
With well-defined processes it's very effective	Scoping can stop improvements by kicking problems to others
Map to other schemes very easily (such as PCI, SP800-53)	Can be expensive to implement - in both $$ and Effort
Business Development folks like to put it on brochures	What are the drivers? Are there benefits for others not requiring it?

- Corporate IT Security outlines dependence on Controls in TRAs
  - Showed Identity Management and Security Event Handling (SIEM) were solely mitigating very high risks
  - Effective for illustrating where investment should occur
  - Drove thinking towards a control-based approach to security

Controls based approach

- Security for us is fundamentally tailored
  - If a customer isn't interested in certs - a cert-based approach causes significant, and unnecessary, overhead
- I would argue most things are tailored at some level, so more variance = more complex compliance
- When operating multi-nationally, different jurisdictional requirements apply. No certifications/frameworks are universally required.
  - Clouds solve this by "doing everything"
  - Watch a hosting providers responsibility matrix. You might be responsible for everything hard.
- Products that achieve strict compliance still have vulnerabilities (some of which are managed poorly)
- Ultimately – given the variance, this didn't have an answer to "why should we do this?"

Maturity-Model Based Security Program

Maturity Model approach: BSIMM

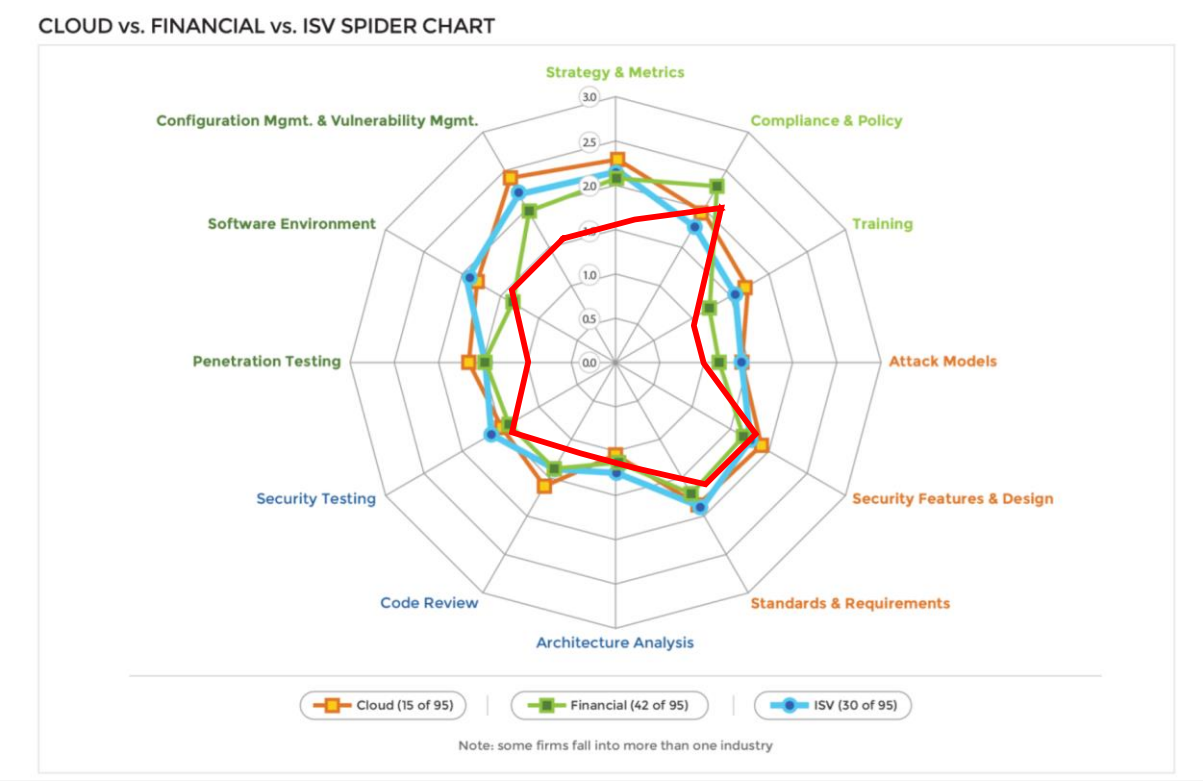
- Build Security In Maturity Model (BSIMM) answered the "Why should we do it?" – we could compare and contrast against other companies and determine based on best practice

- BSIMM is a software study done for 10 years on Cigital/Synopsis on the most common tasks performed by their customers

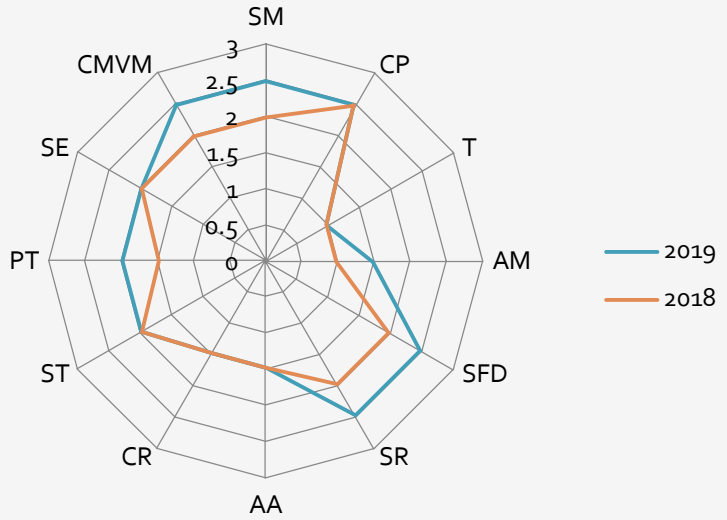
PROs	CONs
Easy to compare to others in your industry or outside	Primarily self audited so results may vary against model
Provides a great context for security briefings	Not generally a requirement from customers
	Tricky to "map" to another specification such as ISO 27001

Maturity Model approach: BSIMM

Comparison Dashboard



Future Looking Dashboard



Sanitized. Not real.

Maturity Model Pitfalls

- Most, if not all, maturity models are context free.
  - Presume a perfect path from start to finish – yet every project path is somewhat different
  - Maturity for a 20 person company is much different from a 2000 person company

- The model tasks and objectives are perfect and knows all
  - Encourage "gap thinking": where are you in relation to the perfect model and what needs to be done?
  - Encourage "level thinking": what do metrics such as "level 2" mean in a self-assessed framework?

- Order is prioritized by doing easier things first when harder items may be more important.
  - Obviously the higher the level means more advanced program right?

- To be fair - BSIMM however doesn't make many of these claims. It simply reports what others are doing.

Maturity Model approach: BSIMM

- Storytime: I had done the self-assessment and crunched the numbers. I mapped out what I took to be a firm and objective view of the program.
  - Time to communicate the good work I had done

- The executives from both technology and business pointed out two key problems
  - Question #1: Are we JP Morgan or Amazon?
    - Neither of these logos are on the sign out front.
  - Question #2: Great work. How does this help the customer? Response:
    - "It's for us to know our posture"
    - "We can brief customers with it"
    - "If we're in line with best practice we won't look foolish"

They were right

- **So What?**

The Dark Period

- Left with some big questions:
  - Are we handling the priorities?
  - Are the priorities right for what we are seeing?
  - What should the group be working on?
  - What should be handled externally?
  - Where were we on the continuum?

Revolution Evolution Maintenance

Guidance & Wisdom

- Advice came to me from three different viewpoints
  - Specific to their segment
    - You need to focus on "network-security" or "blah-security"
    - The rest will come along – don't worry about it
  - This is what you signed up for
    - Grab a shovel and do the Risk Assessments / Incident Response
  - Just pick a road and follow it
    - Any compliance standard is good enough

"Look, do something - anything - because all security heads get fired when something bad happens."

"You want to be able to say you were turning things around at the next interview. Also plug your incident experience."

MOTIVATIONAL ADVICE FROM SECURITY PROFESSIONALS

(Title slide from my next talk)

VISION

- I like the sailing analogy. I know I want to go to Europe.
- I'm in a good place with the day-to-day "maintenance" activities
- When I got out to sea I got lost in all the blue.

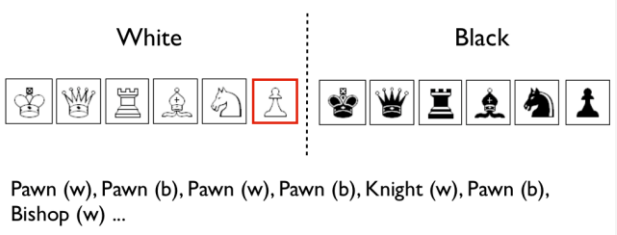
This is all about situational awareness

Wardley Mapping

- Wardley realized business strategy is often copied and repeated. Context is sometimes lost.
  - Lots of extra baggage attached for other companies.
  - Survivor Bias: since it worked for someone else it will work for us
  - Loss of context in translation – we might not have Facebook-type problems

This is really about situational awareness

- This thought experiment caught my attention: take a game of chess where you can't see the whole board - what happens if you copy and repeat the same moves as the grandmasters?
  - Might not work out well as you have no context for those moves.

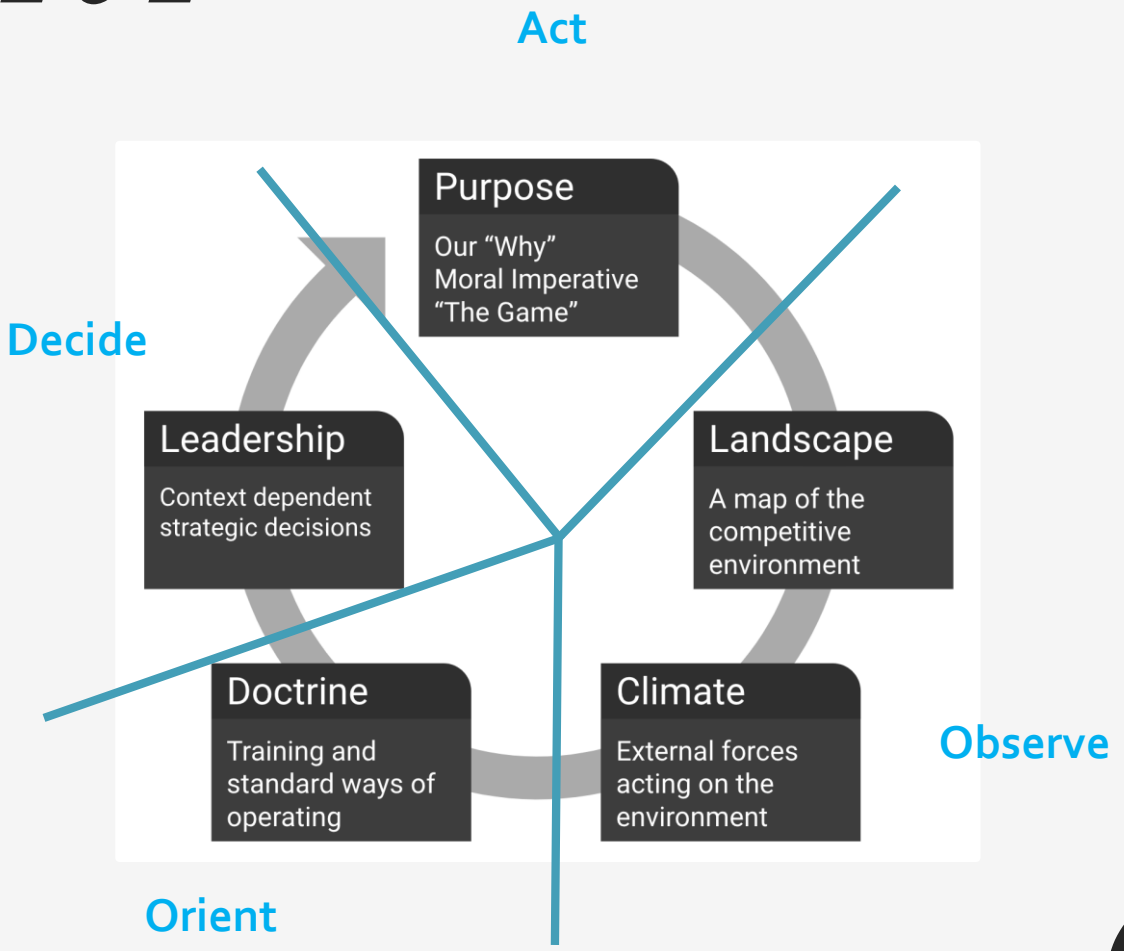


- Security is much like playing chess without seeing the whole board
  - "Best practice" might be just repeating the same moves as those grandmasters without the context

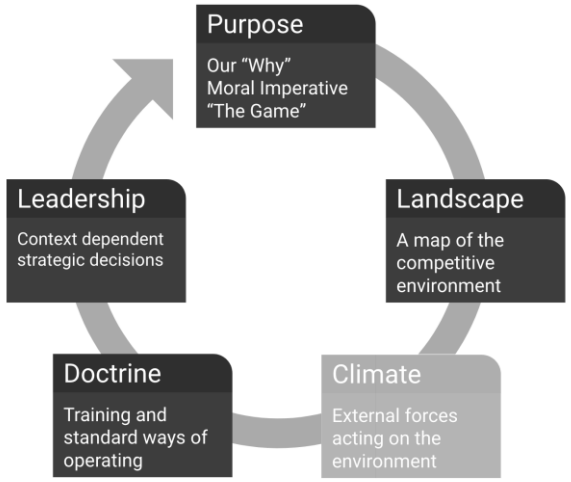
Wardley Mapping: 101

- **Leadership** is making strategic decisions based on:
  - The **Purpose** of your program
  - Description of the competitive **landscape**
  - External forces or **climate** acting on the landscape
  - Training of your team (or **doctrine)**

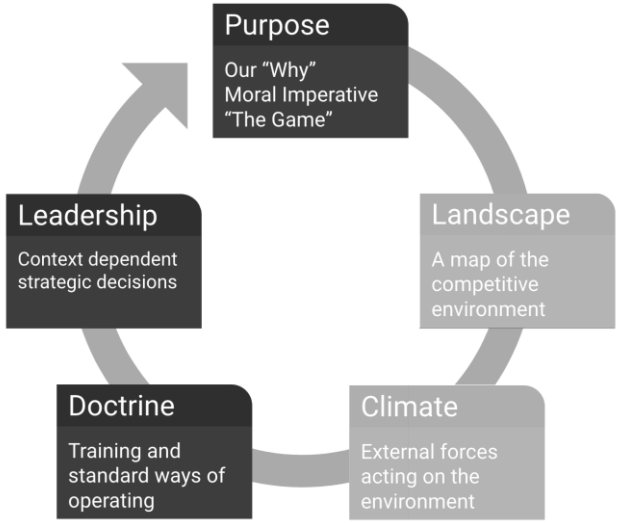
- **This is also an OODA loop for those in Incident Response**
  - Observe, Orient, Decide, Act



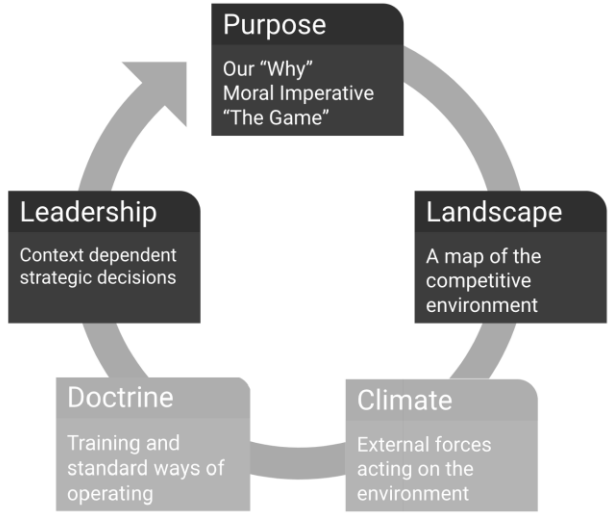
Images: Simon Wardley



"Best Practice"



Compliance & Frameworks

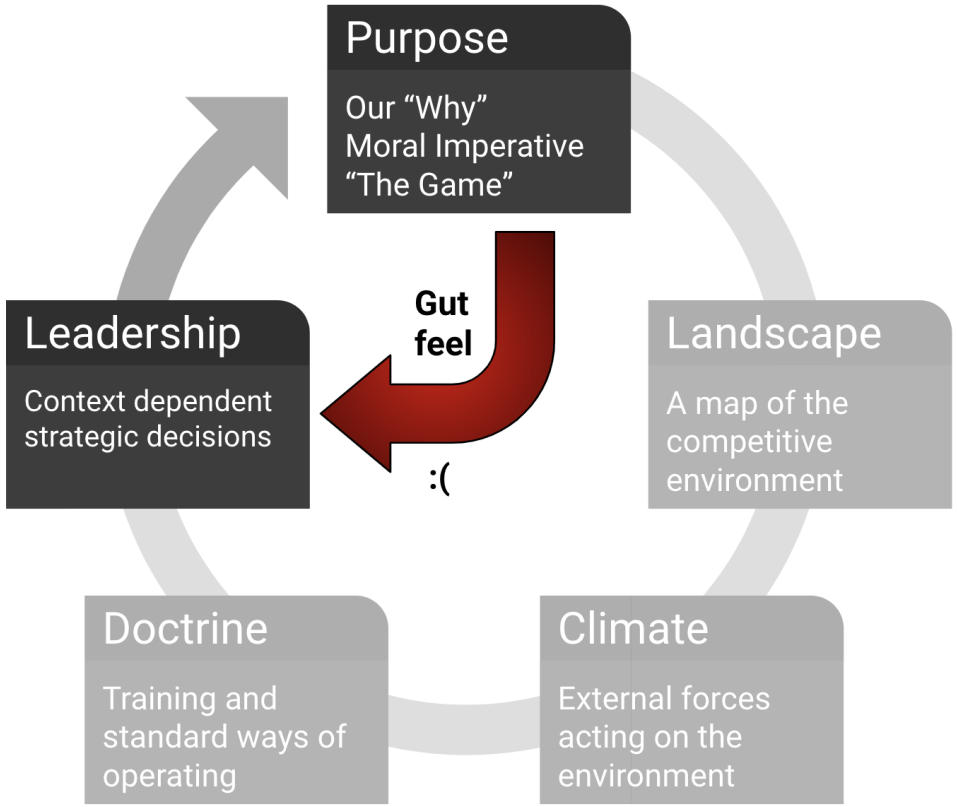


Maturity Model

The problem began to emerge

- Debatable, but to us it seemed our strategies ranged in completeness
  - 100-day plan covered much of the cycle but lacked depth due to its short run
  - Compliance & Frameworks didn't care about context
  - Maturity Model didn't tell us the standard ways of operating

Images: Simon Wardley



We couldn’t help but feel - when picking security strategy it was really this

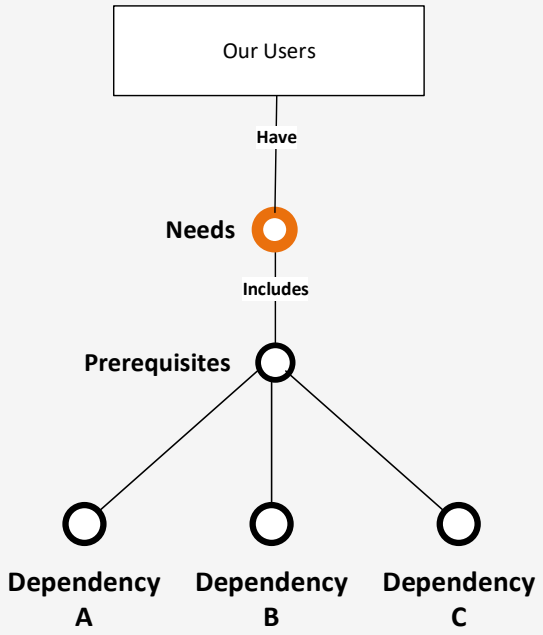
Images: Simon Wardley

Getting Started with Wardley Maps: Value Chaining 101

Description

- Start with your users
- They have needs
- Include Prerequisites
- Which have dependencies

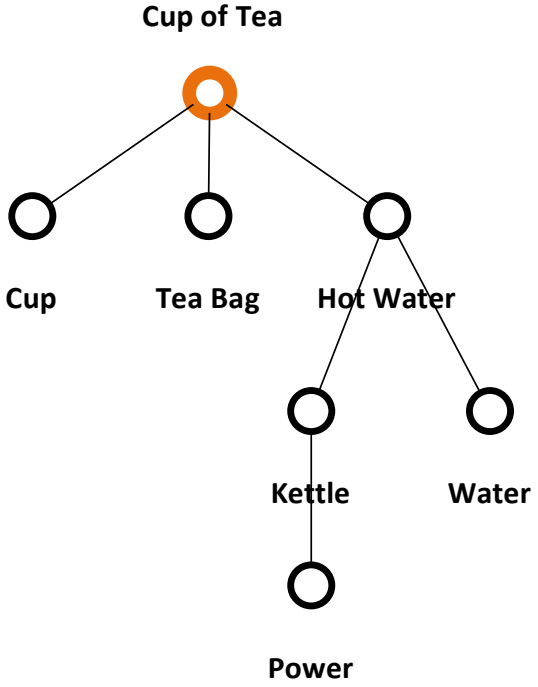
Define your purpose



Example: Cup of Tea

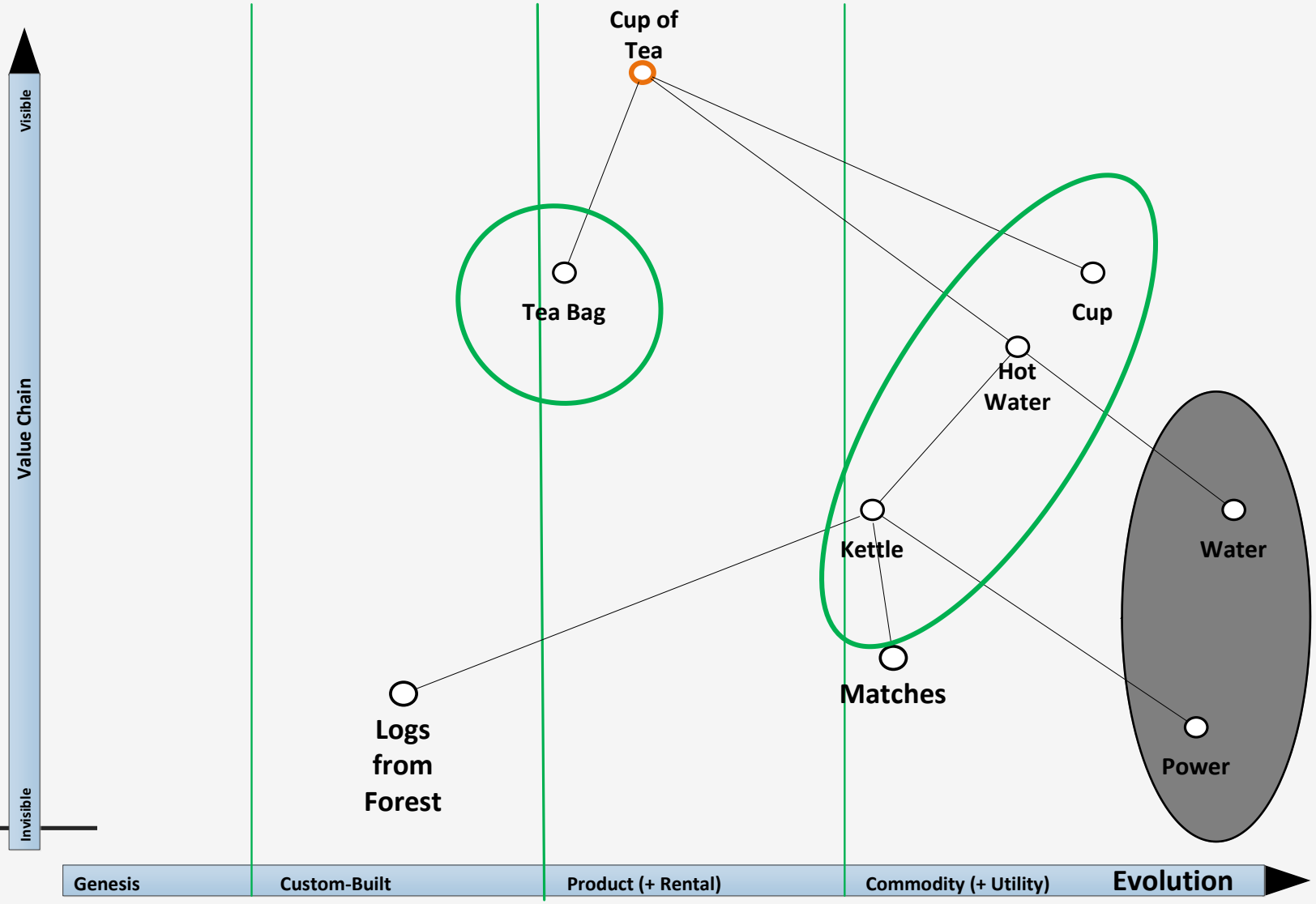
- We need a cup of tea
- Tea needs a cup, teabag and hot water
- Hot water needs a kettle & tap water
- Kettle needs power

Friend wants cup of tea



Situational Awareness is about understanding Change. Change happens in a context.

- Our context is the Evolution Scale
  - How does a prerequisite occur in the world?
  - How do users perceive it?
  - What does the market look like for it?
- What if we didn't use a kettle?
  - You might wonder why
  - Where should things be?
- Now we can add plans and teams
  - Two teams + Outsource (Grey)



Starting with Purpose

- Why do you exist?
  - This doesn't have to be extremely complicated as it will emerge and evolve over time. **It's not fixed.**
  - It's essential for backing from executives and higher-ups.

- Why does your work need to be secure?
  - For us, it's in our mission statement that we produce secure solutions
  - Ultimately, it was fraud reduction that our executives were looking to accomplish

- After some refining by the team our purpose is

We reduce fraud by reducing misuse of our systems

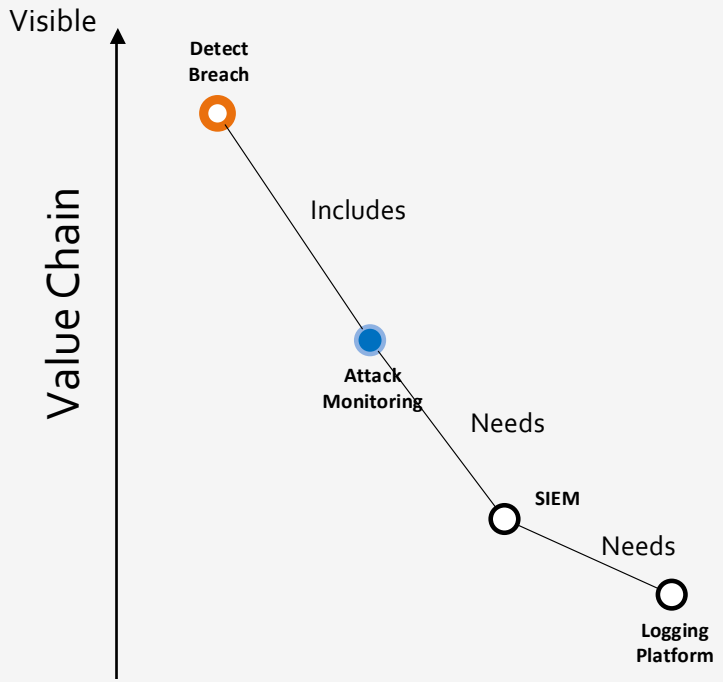
Landscape

Who are your users? What are their needs?

- We started with: Confidentiality, Integrity, Availability of data
  - Assure that through Secure Design and Operation
  - No – through Secure Design and Breach Management
  - No - through Breach Management demonstrated through Compliance
  - No – through Secure Design, Compliance and Breach Management
  - We're currently at Secure Design, Breach Management and moving to System Visibility

Next we develop the Value Chain

- Projects and Organizations operate within a context or **landscape**.
  - Wardley maps express this as a value chain
- So we plugged in our tasks as components (dots)
  - Start with the core user need
  - Add the task as a component (or service)
  - Tie it to the **dependencies**
  - Rank the visibility of each component / dependency
- **On it's own: Interesting but not very useful**



Value chain of Security Event Management

- **User need: Detect Breach**
- **Includes: Attack Monitoring (Service)**
- **Needs: SIEM Tool and Logging Platform**

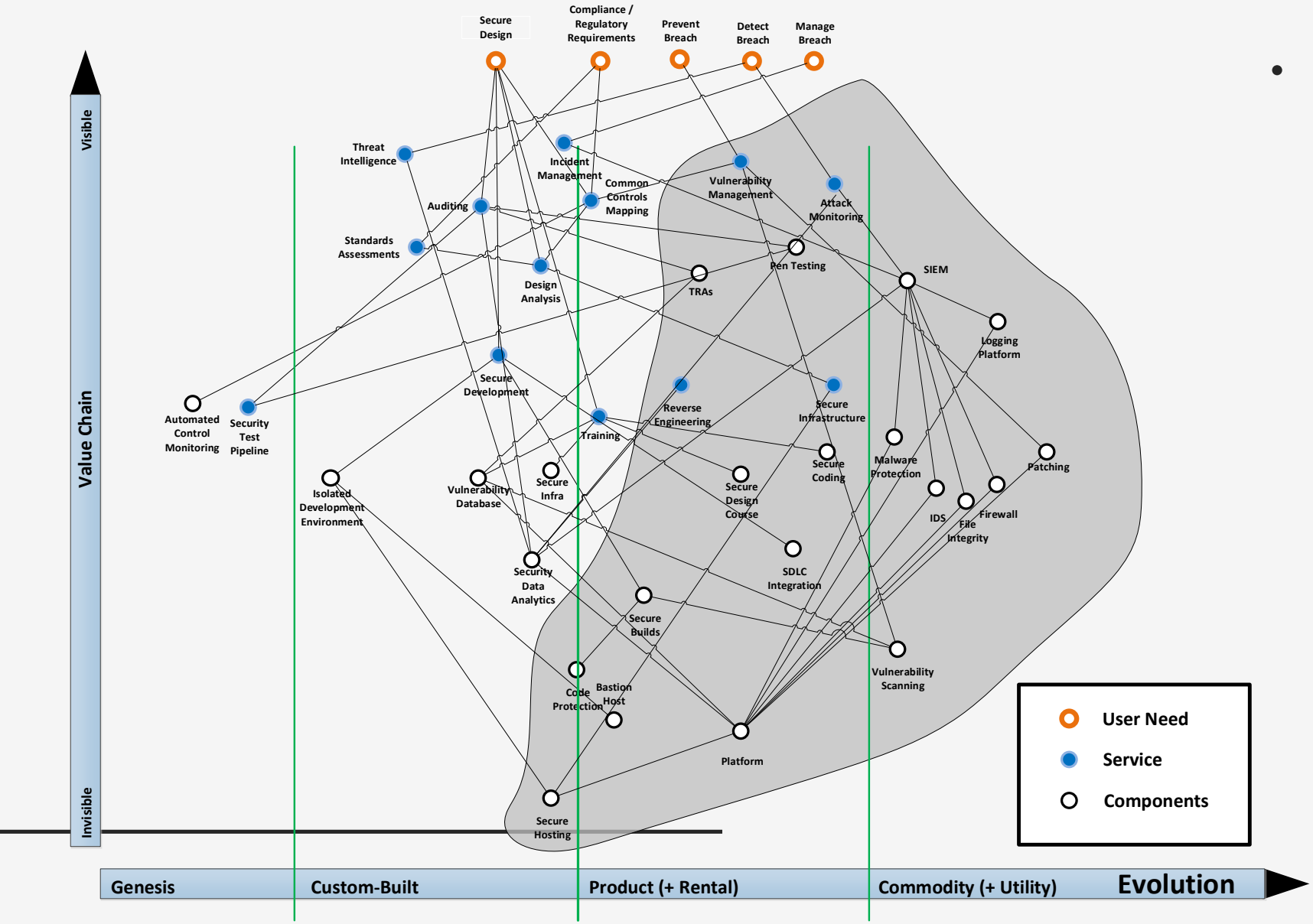
I	II	III	IV
Genesis	Custom	Product (+rental)	Commodity (+utility)

Uncharted	Industrialised
Chaotic	Ordered
Uncertain	Known
Unpredictable	Measured
Changing	Stable
Different	Standard
Exciting	Obvious
Future Worth	Low Margin
Unusual	Essential
Rare	Ubiquitous
Poorly Understood	Defined
Experimentation	Volume Operations
Differential	Operational Efficiency
Competitive Advantage	Cost of Doing Business

Our environment is the evolution of an offering

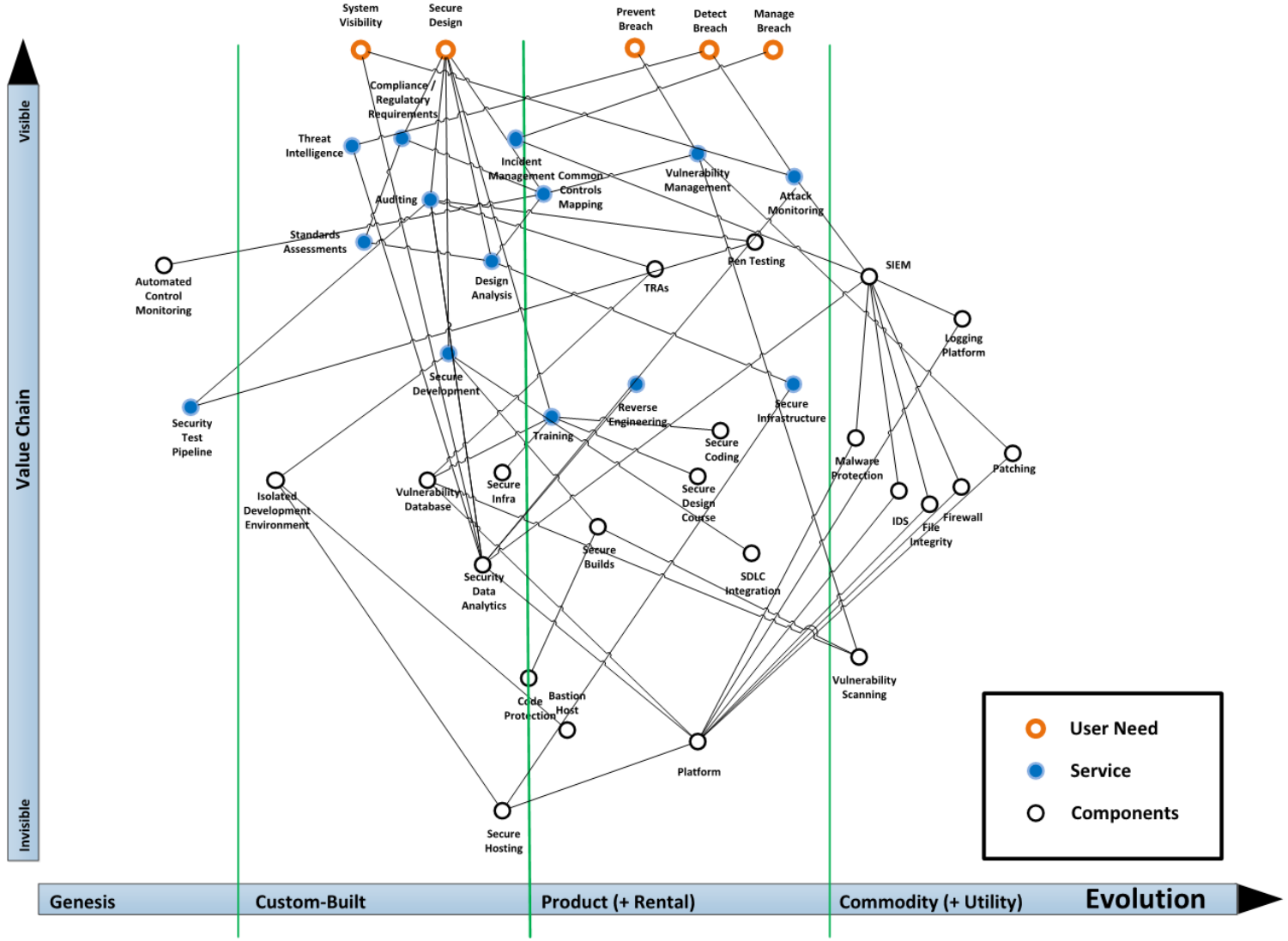
- Climate is core to Wardley Mapping
- Everything moves from Genesis to Commodity
  - The fact it will move is **out** of our control
  - How fast and when it does move is **in** our control
- Phases of an offering
  - Genesis: First time ever doing this (Predictive AI for Incident Response)
  - Custom: Built specifically for a particular use (Automated Incident Response)
  - Product: Generic enough to have multiple uses (SIEM and Alerting Tools)
  - Commodity: Well defined and highly available (Malware Detection)

Security Program: Iteration 1

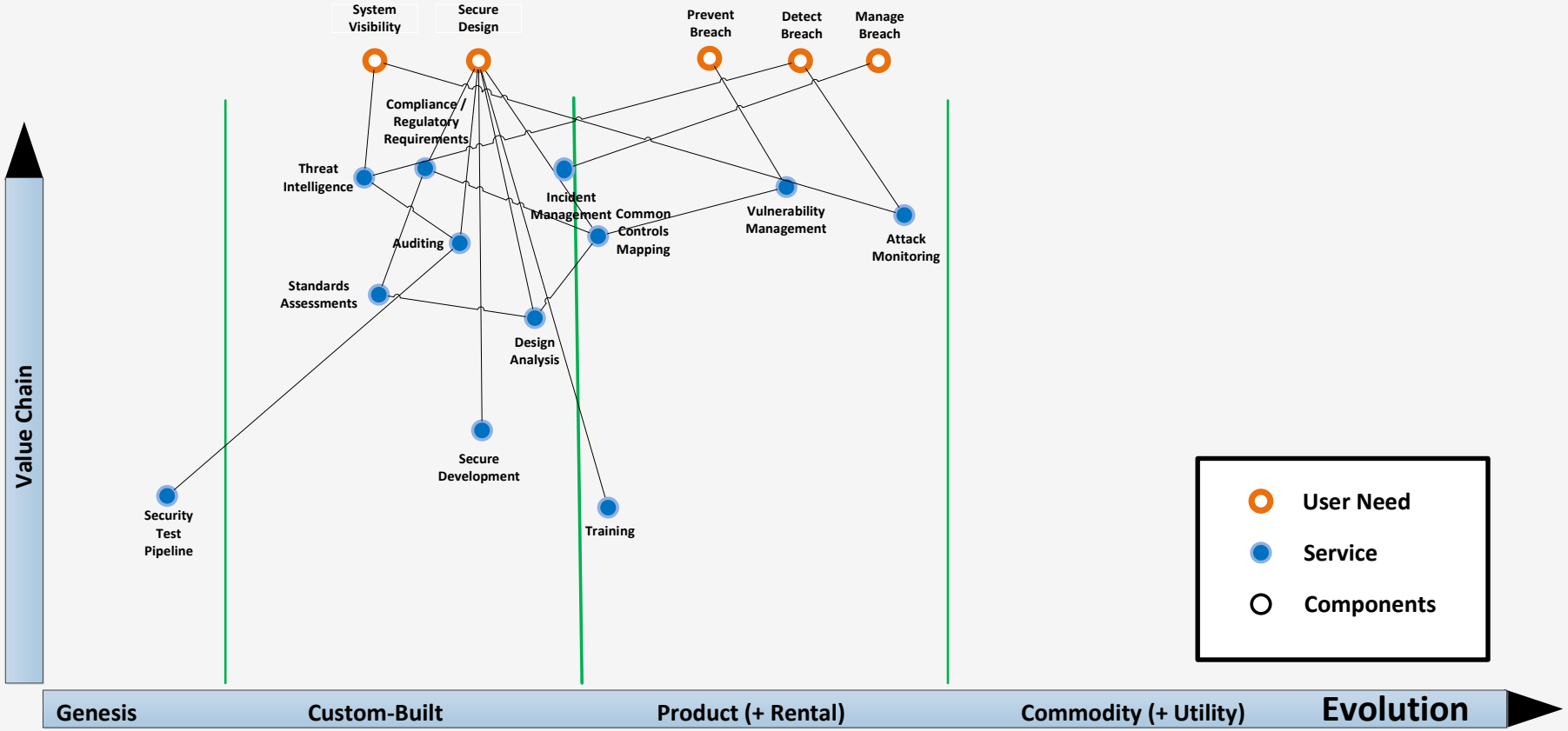


- I learned:
  - We offer services
    - Might be underpinned by execution
  - We offer services that we can buy:
    - Vuln Mgmt
    - Attack Monitoring
    - Reverse Engineering
  - Clear that secure hosting is going away
  - Stuff I'm working on should be pushed to infra teams

Security Program: Iteration N:



- I learned:
  - I'm in the data analytics game
    - Standardize Logging
    - Leverage your data analytics team
    - It's really deep but connects to a direct user need. What's missing?
  - I need a pipeline fast
    - Are you ready for the stream that's going to hit?
  - Why are we teaching a custom course on Secure Coding?
  - How much customization do I have to do to ?
    - Do I need custom Pen Testing & TRAs?



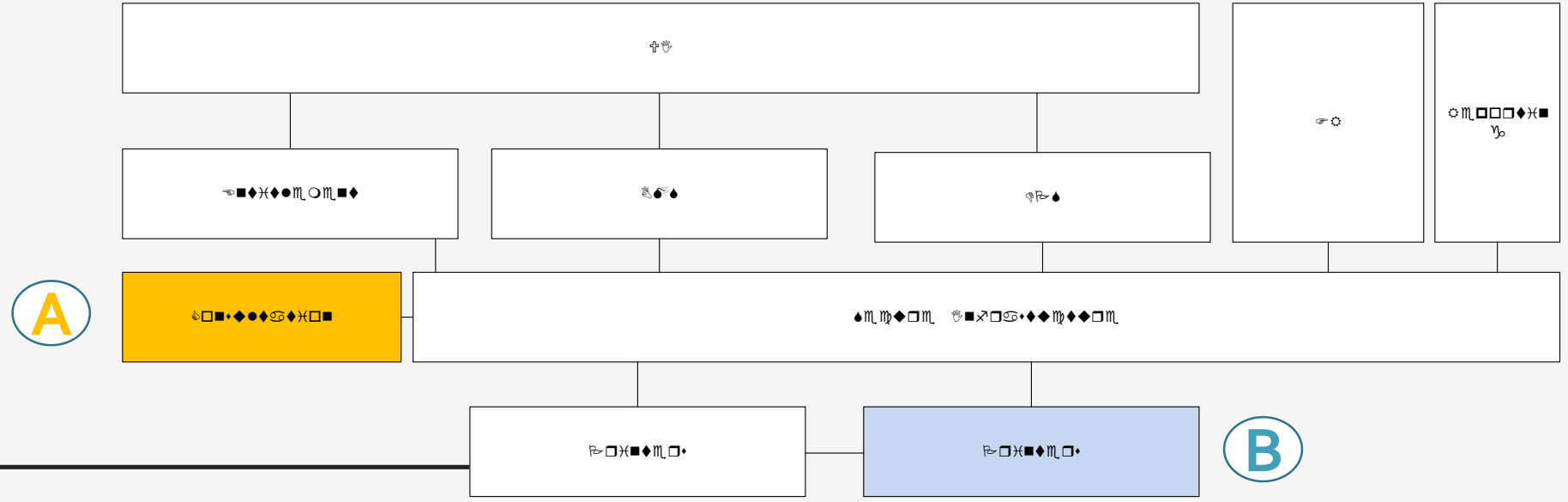
TOP-VALUE COMPONENTS: THE "GENERAL COUNSEL" MODEL

- Security is always about sharp technical skill
  - But it is shifting to becoming **a set of in-house services**
  - Lawyers and General Counsel's pull in other lawyers when changing jurisdictions or disciplines
    - You'll will probably have to do that too.
    - It's OK to not know everything – have your specialties – otherwise get experts and translate to your users

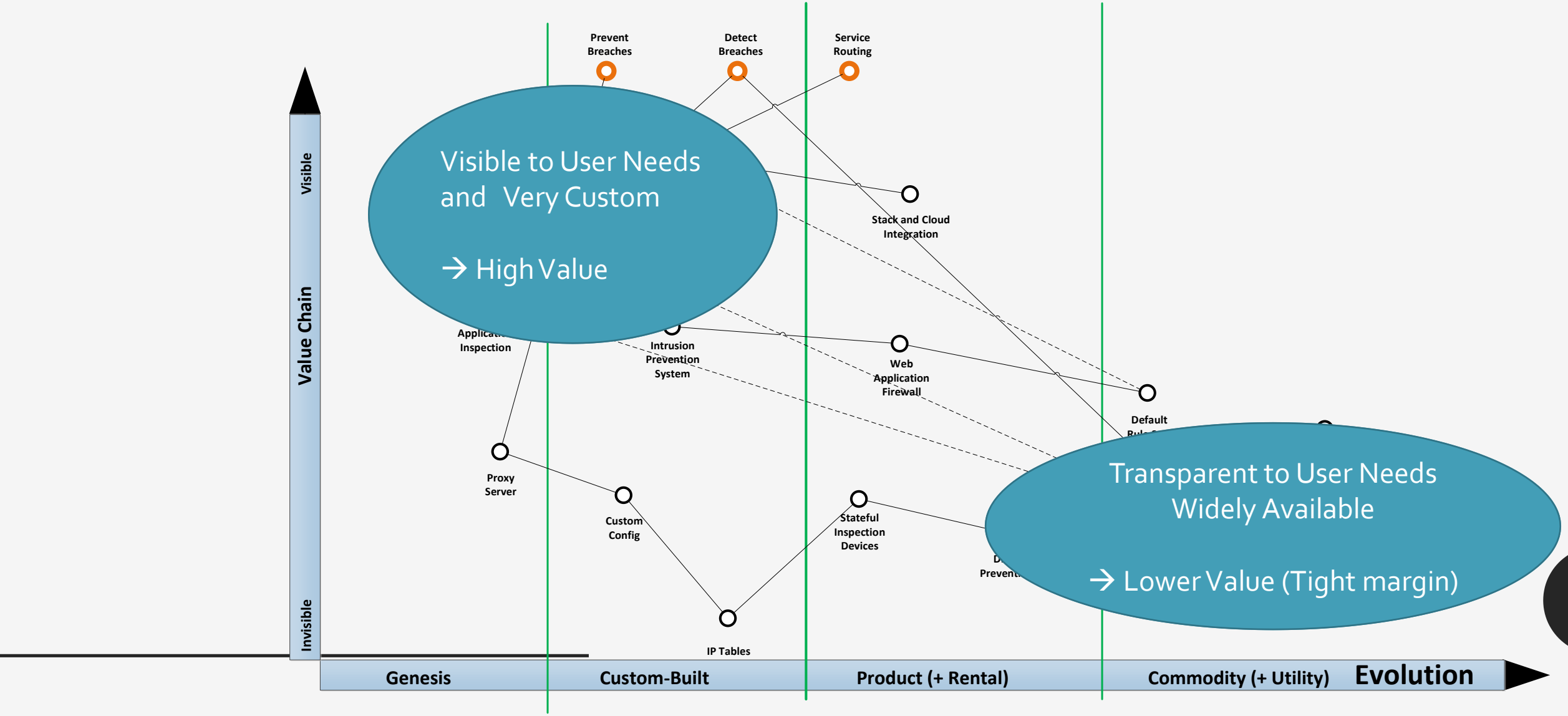
Wardley Maps help you determine what is worth focusing on

- What would you outsource? A or B ?
  - Imagine you're in a non-technical discipline. Do you build or buy these components below?

- This is from an actual system architecture diagram



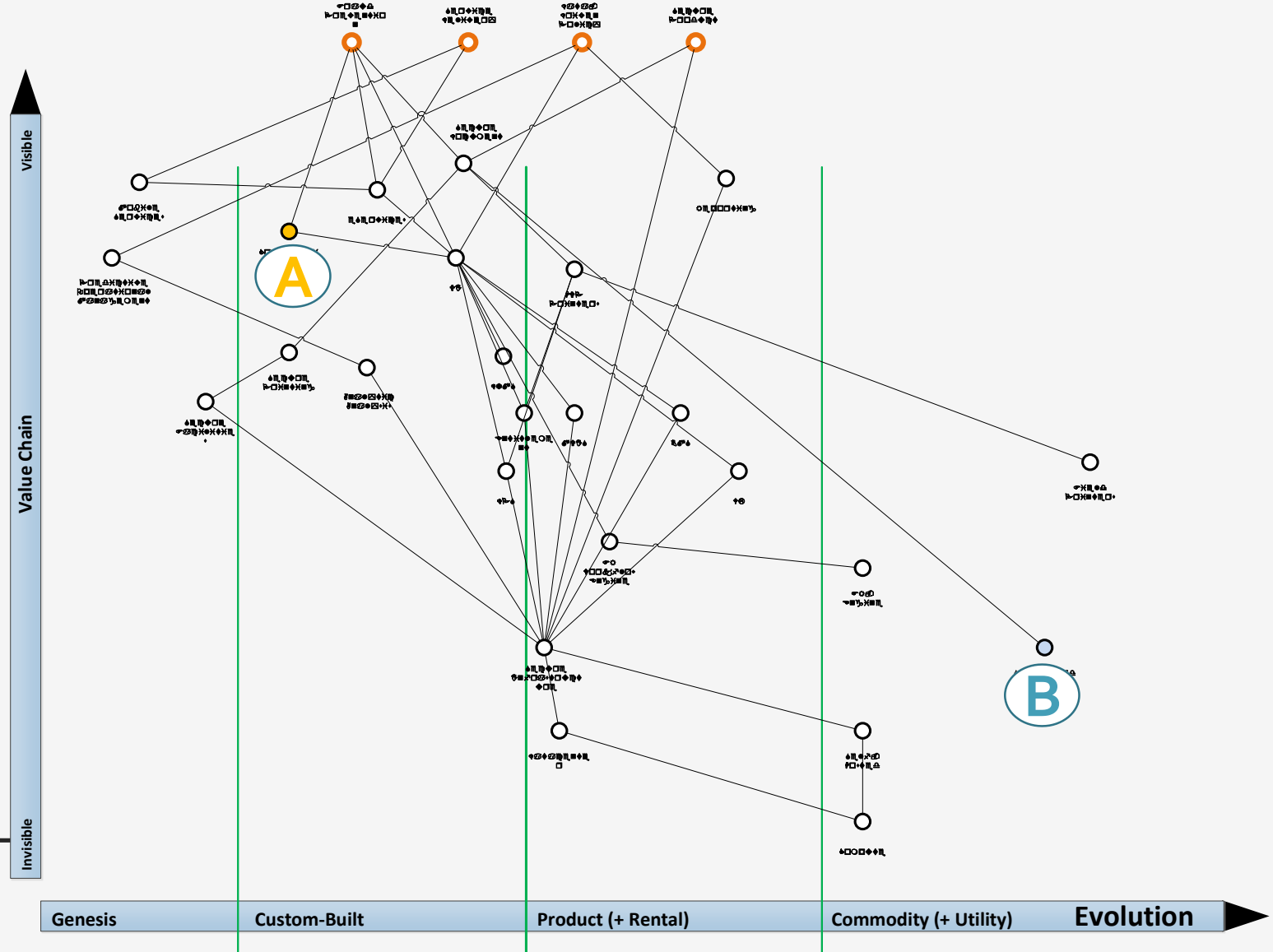
Value provided changes based on map location



Looking at the same diagram but slightly differently

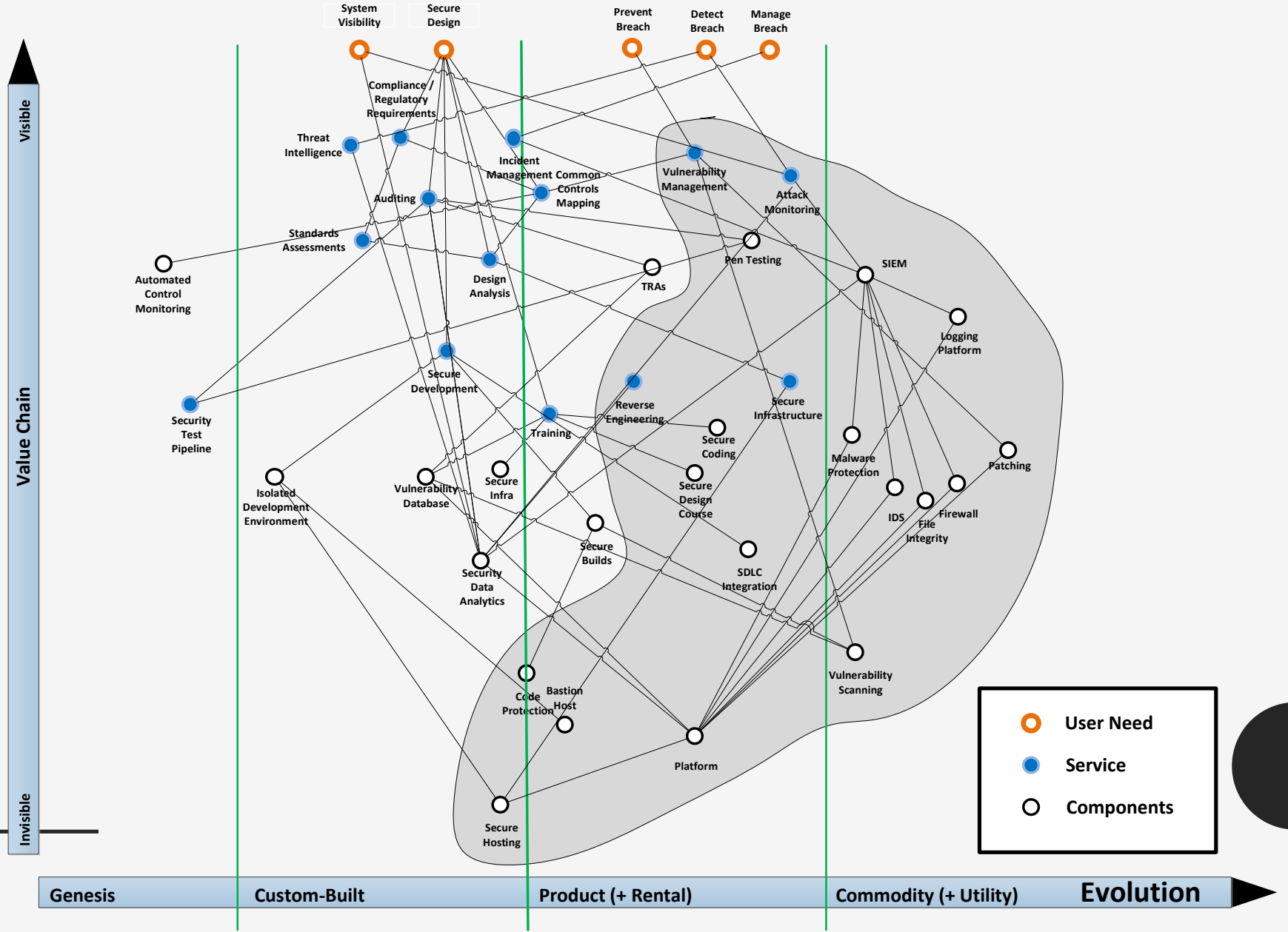
- Component A
  - Very visible to user
  - Custom built to their needs
  - Very expensive to build

- Component B
  - User expects it to be there
  - Readily available in the marketplace
  - Probably negotiating room on price



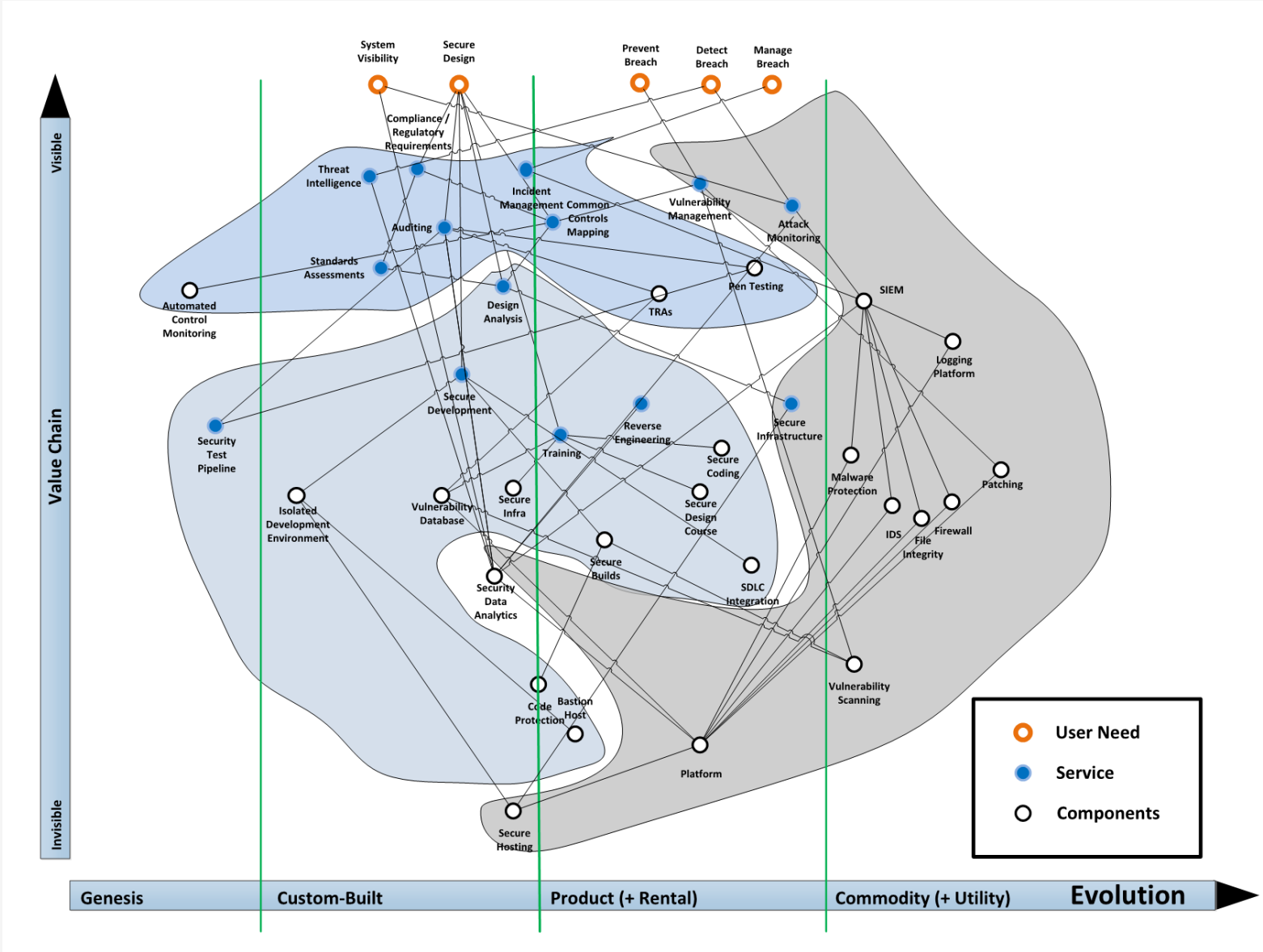
Coming back to the Security Program

- Security Engineering is **not** doing the grey bubble
  - SecEng will co-ordinate and oversee the implementation
  - Might still be done internally by other teams
- Other tasks are important
  - Focus on user needs
  - Jurisdictional Consulting, Architectural Analysis, New Tooling for Dev Models
  - Risk reporting and Analytic Delivery



What will the teams look like?

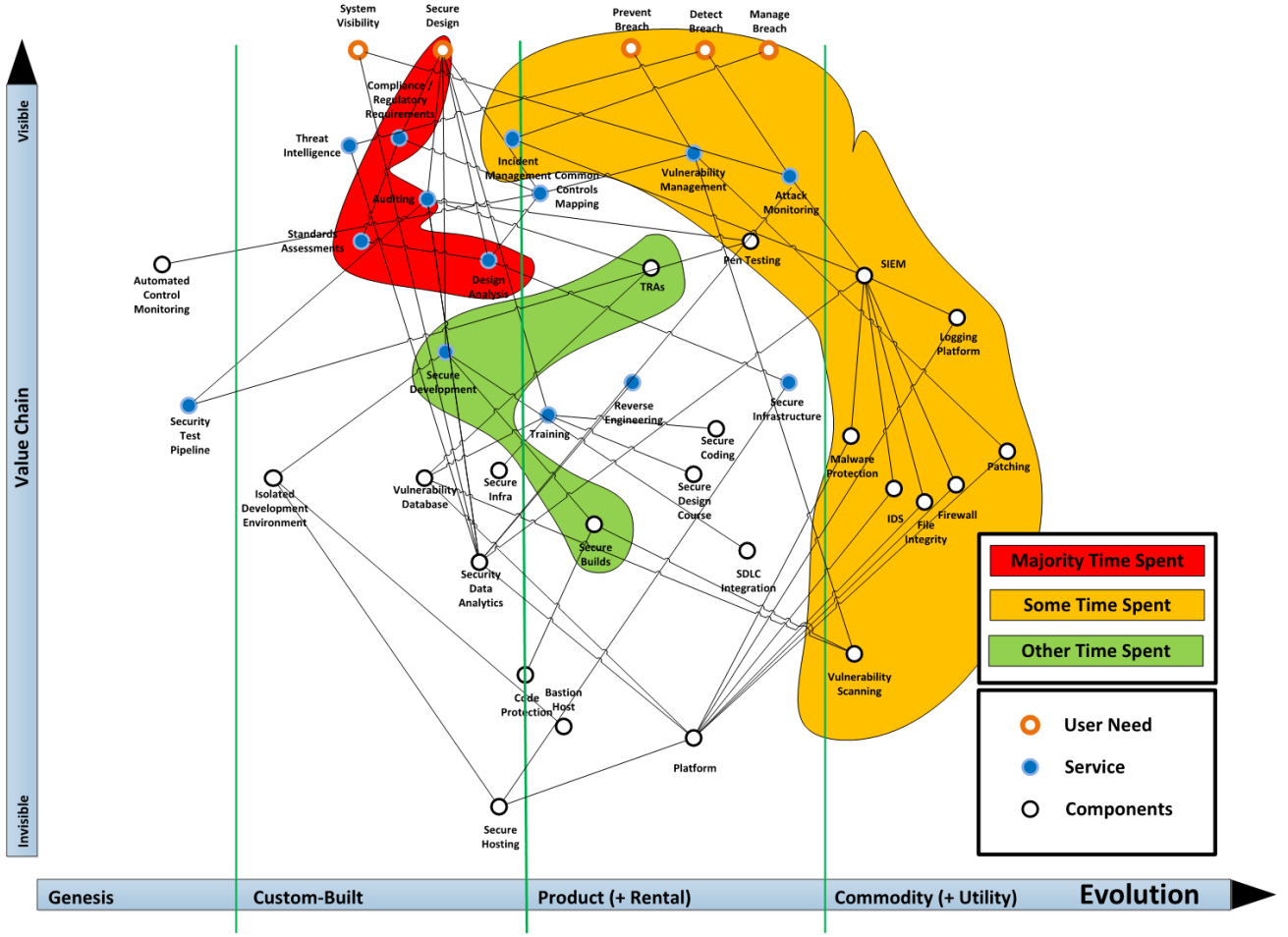
- We see these tasks handled by three teams.
- Design Team handles compliance, policy requirements and risk analysis
- Platform Team supports the development teams with tooling, training and controls
- Production Team performs implementation, remediation and analytic delivery



Experiment: Where was our time spent?

Findings

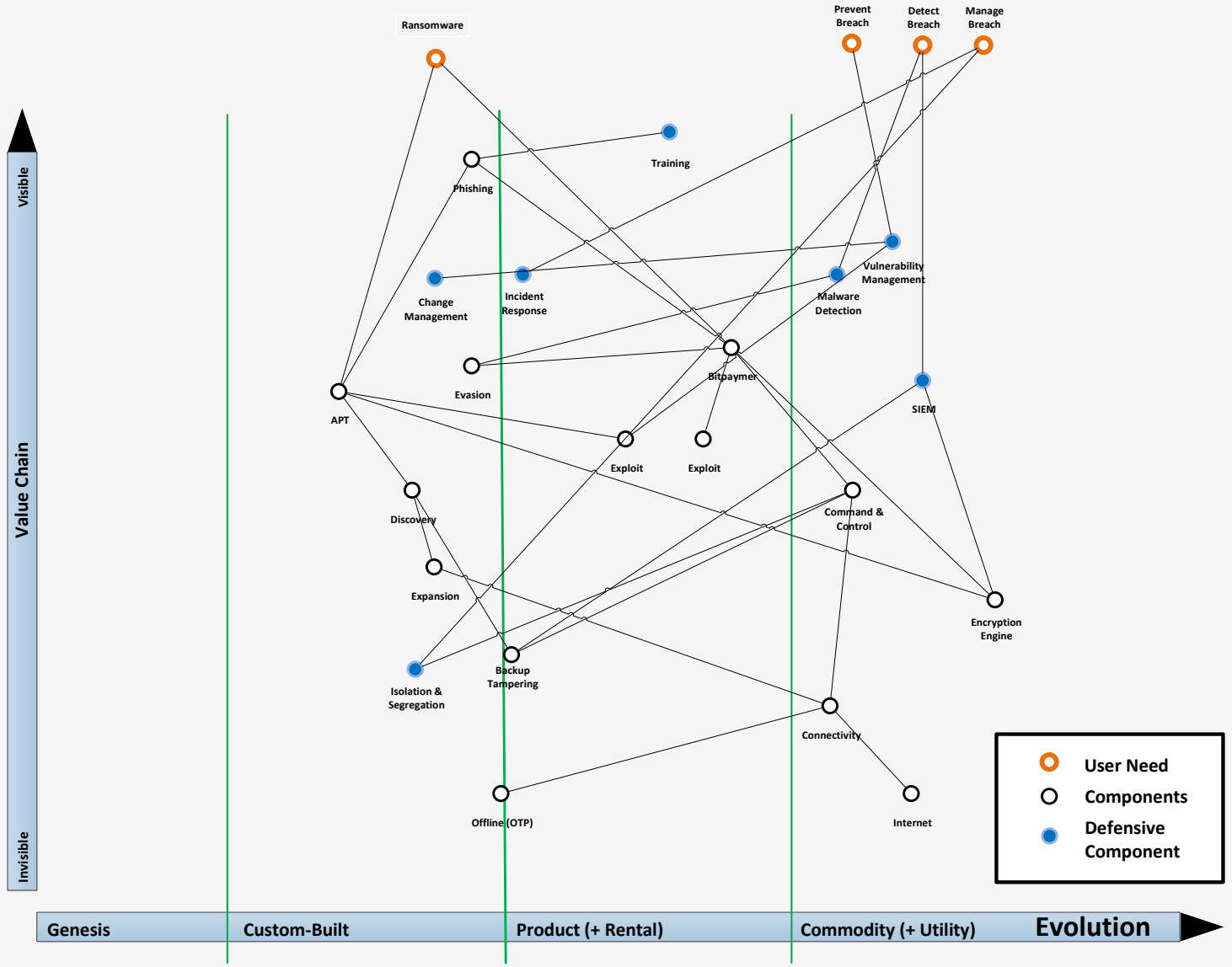
- Probably not ideal to not spend any time on core user need (Analytics)
- R&D work is minimal
- Spending bulk of time on design (red)
- Spending a good chunk of time on keeping the lights on (yellow)
  - Future vs Maintenance debate
- Interesting data point: TRA time vs Pen test time



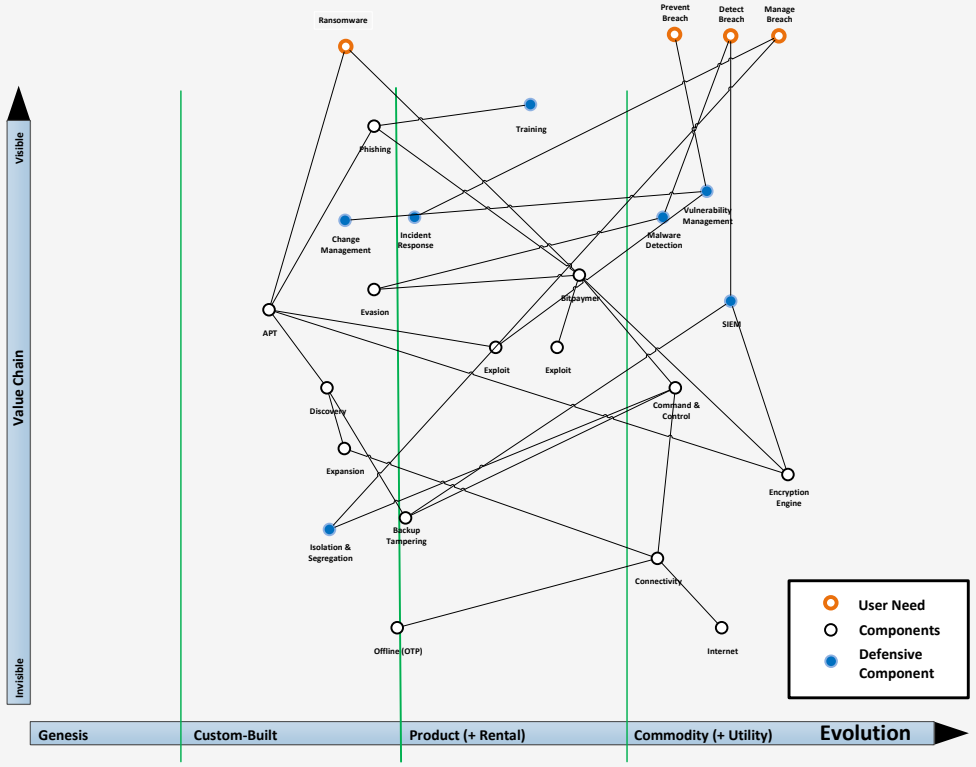
How do we handle CryptoLocker? Map it.

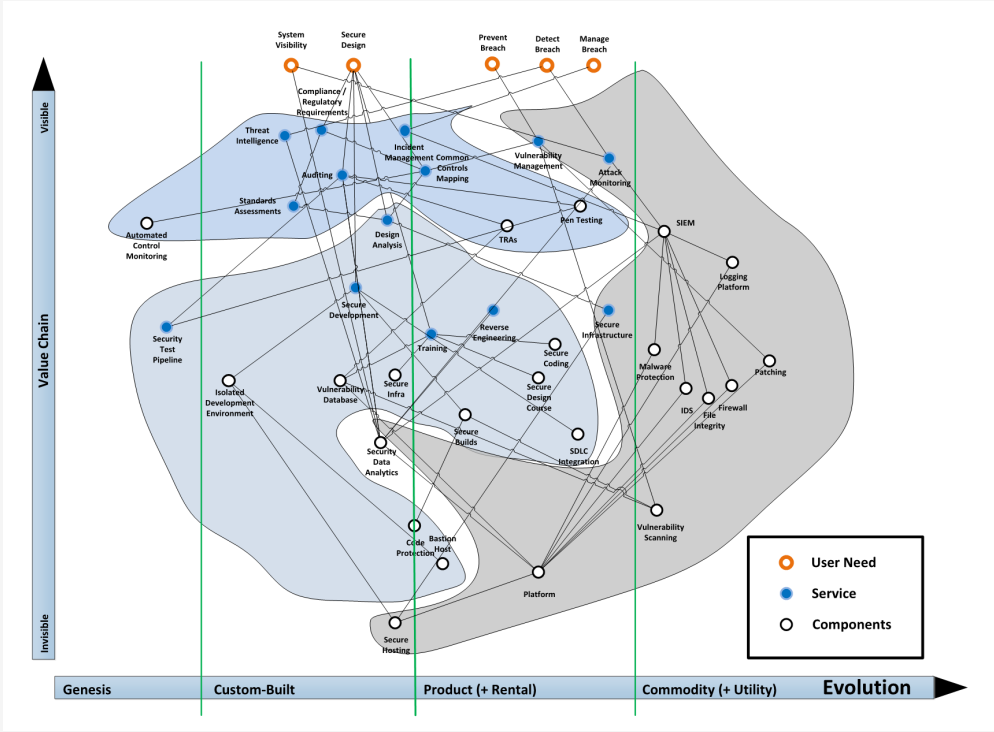
Observations

- Phishing is the key ingress point
  - Probably targeted to some degree
- APT is a completely separate path. Should it be?
- Blue dots are defensive components. Iterate over them.
- Patch team is dependent on Change Management
  - How do we accelerate CM team?



How does it fit into our program? OR "Called in on the red carpet"





- We run through the blue dots and describe what we're doing for each

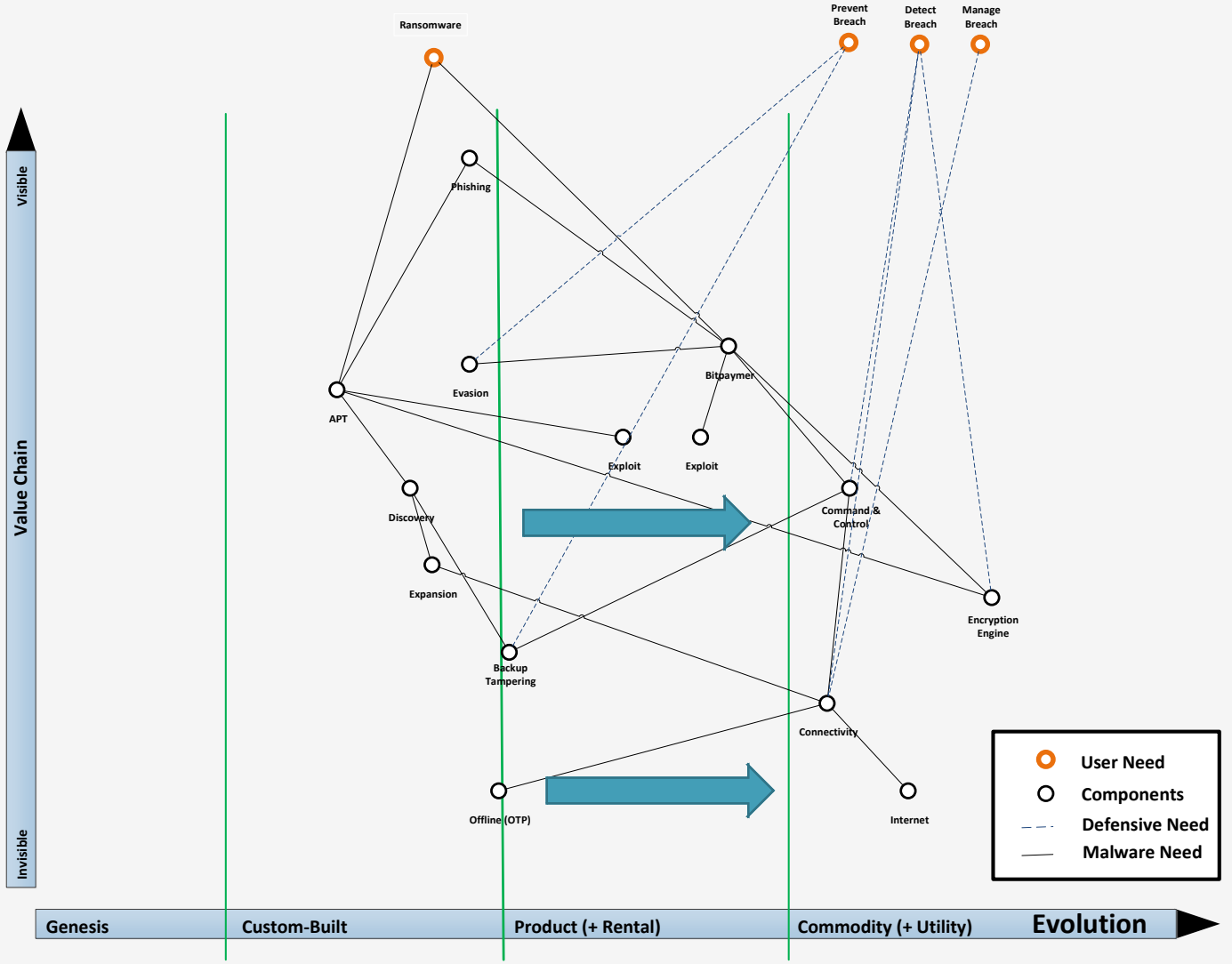
- Run through the overall map and priorities for each
  - Tie **everything** to the core-user needs

In fact, if I were them….

Priorities

- Full-time Phisher
- Buy exploits and Command and Control function. [Platform Team]
- One apt & one ops person
  - Thin custom exploit code and integrate into the common C&C
  - Ops person does discovery
- I'd want more analytics on backups and offline infections from Ops
  - I could see specializing in secure USB key delivery and/or specific backup methods

This could be wrong – but I can see the board



Doctrine

*What if standards and frameworks **are** the doctrine?*

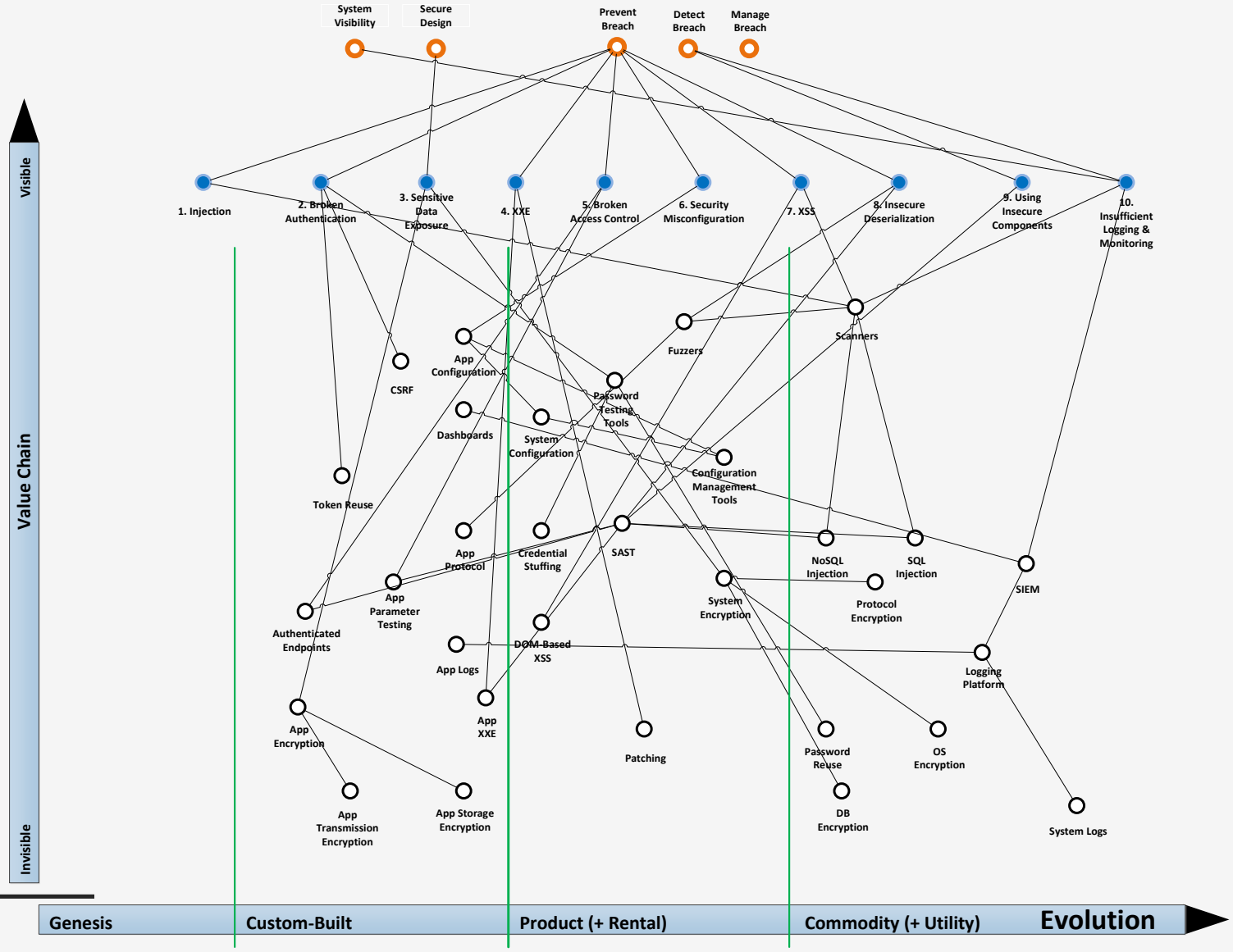
- Wardley has provided some detailed ideas on "doctrine" – the standard ways of operating
- To-date I've been mapping techniques and concepts
  - What if I mapped frameworks and standards
- Provides a nice-checkpoint on the security program

Category	**Wardley's Doctrine** (universally useful patterns that a user can apply regardless of context)			
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, "two pizza")*	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)*
Good Neutral / unknown Weak Warning	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			

Apply to OWASP Top 10

Findings

- Build-time Analysis is a primary defense for us
- We could have other teams test the infrastructure
  - App testing is manual
  - Corollary: lots of work to do for PaaS/Serverless
- Custom Configuration depends on Config Management tools like Puppet
  - In scope? ***Yes***

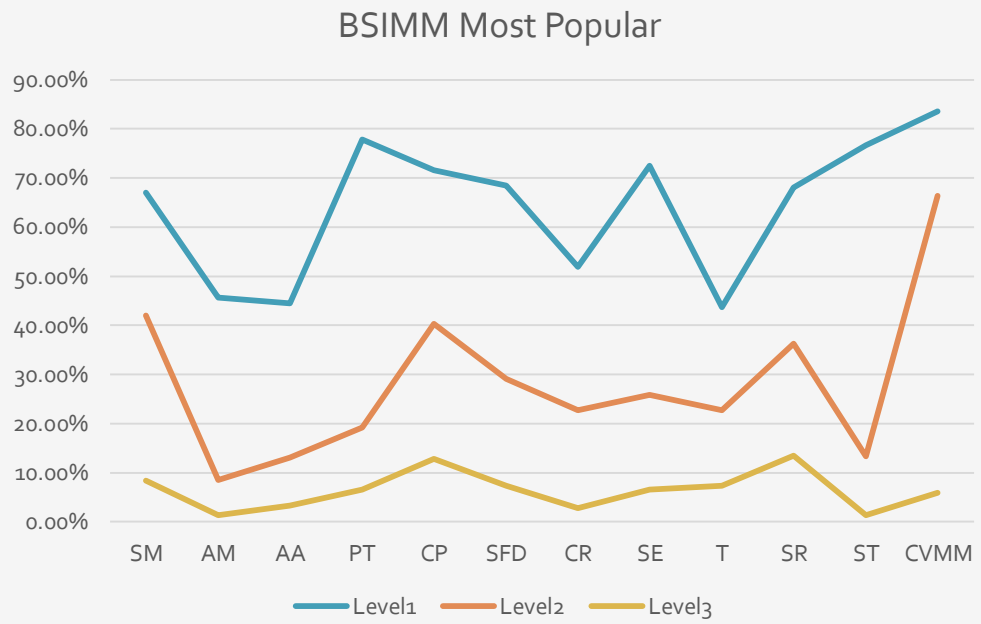


Wardley Mapping BSIMM

Observations

- Rated from Level 1-3 on major Categories of Governance, Intelligence, SSDL and Deployment
- They publish the percentage that implement each category and level

- Graphing it makes it look like three clear stratas with few overlaps
  - Approach would be to complete each level
  - Go from Governance across to Deployment

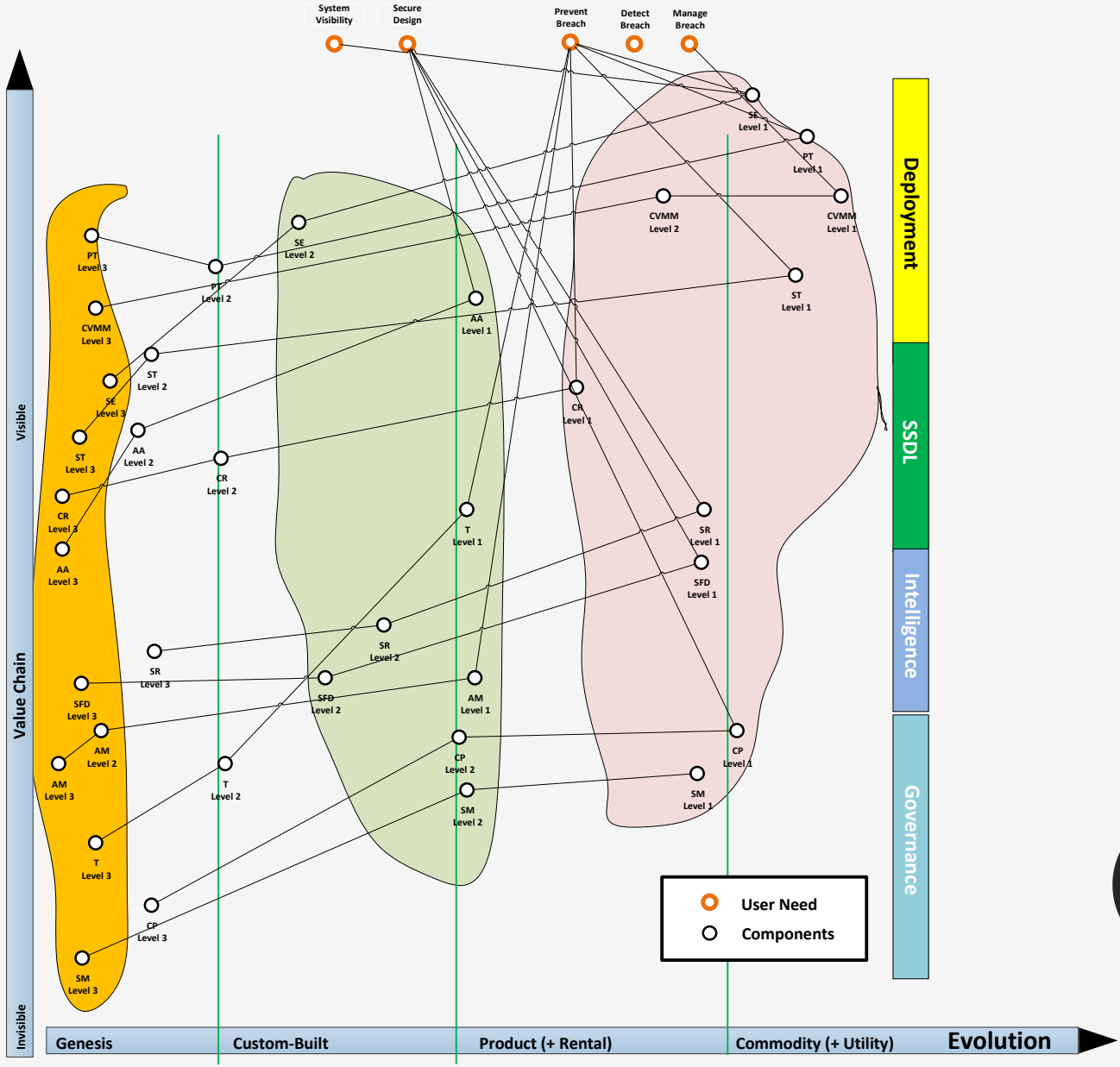


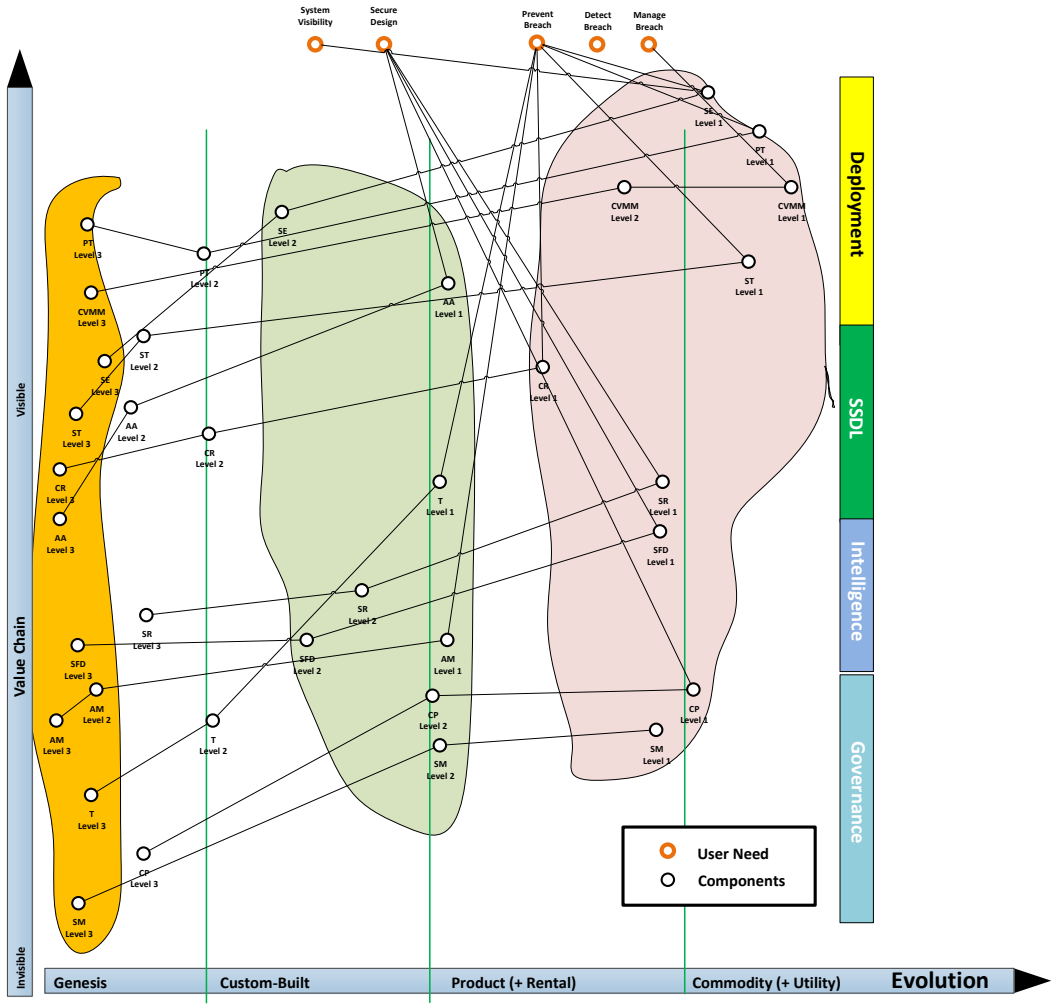
Wardley Mapping BSIMM

Observations

- Examining the map it actually breaks into two phases + possible add-on
  - Phase 1: 9 "Level 1" + 1 Level 2 (Vulnerability Management)
  - Phase 2: 3 Level 1 + 5 Level 2
  - Phase 3?: 5 Level2 + 1 Level 3

- I would argue that once Pink + Green are complete – increased depth in these areas is required to extend the lifespan of this model
  - New levels will necessarily be slower due to complexity
  - Hard to sustain momentum
  - Big question for BSIMM: Why do Level 3?





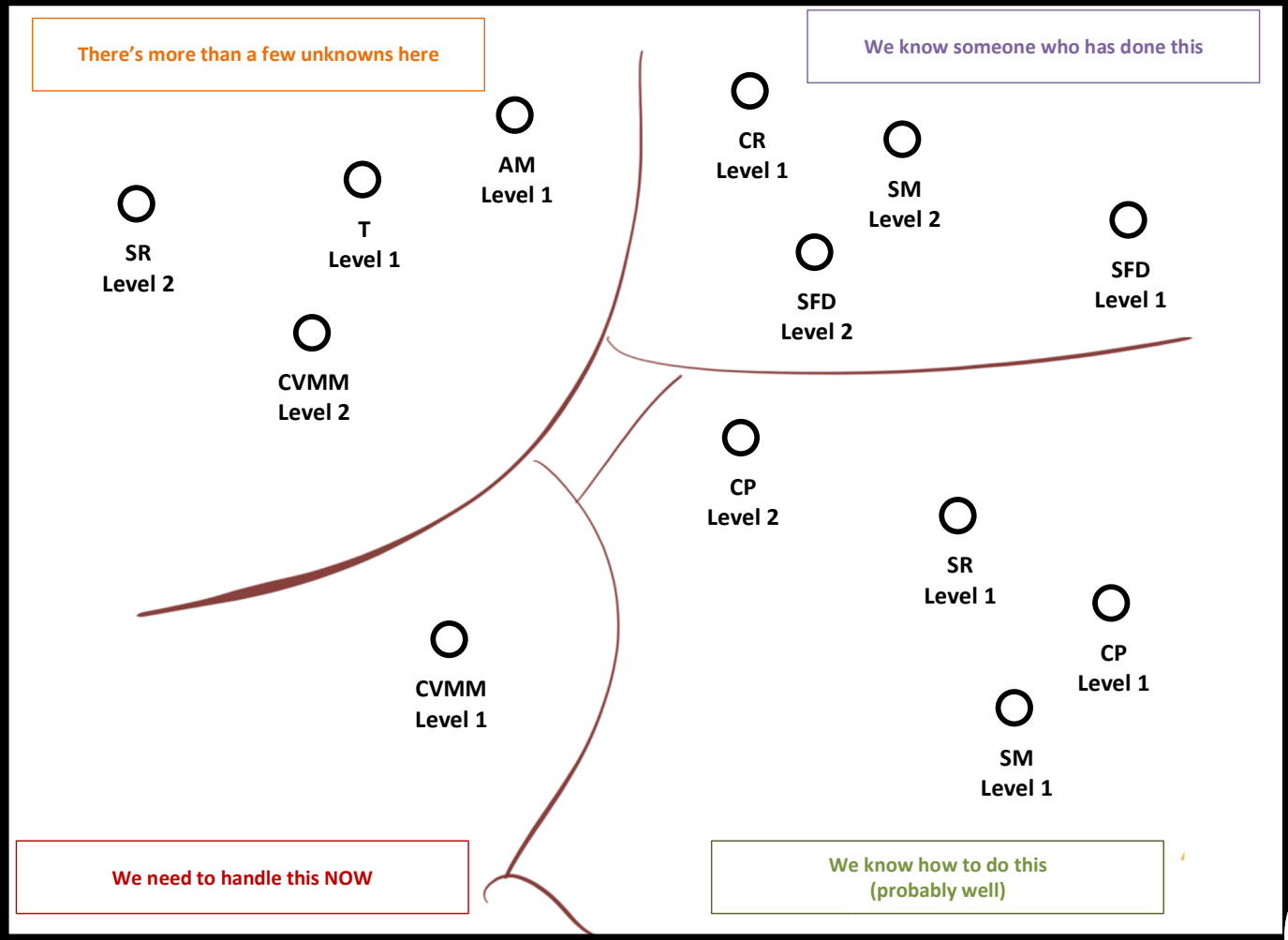
IT WOULD BE USEFUL TO APPLY THIS ELSEWHERE

- *ISO27000 SPECS*
- *NIST800-53*
- *NIST CYBERSEC*

Gameplay: work towards your strengths

- An interesting application is using the Cynefin framework

- This affects the edges between the components
  - We can add to the map, areas for improvement
  - Understanding certain areas of the map better than others may adjust focus
- When you start – everything will be in the lower right.
  - Make the teams map the dependencies



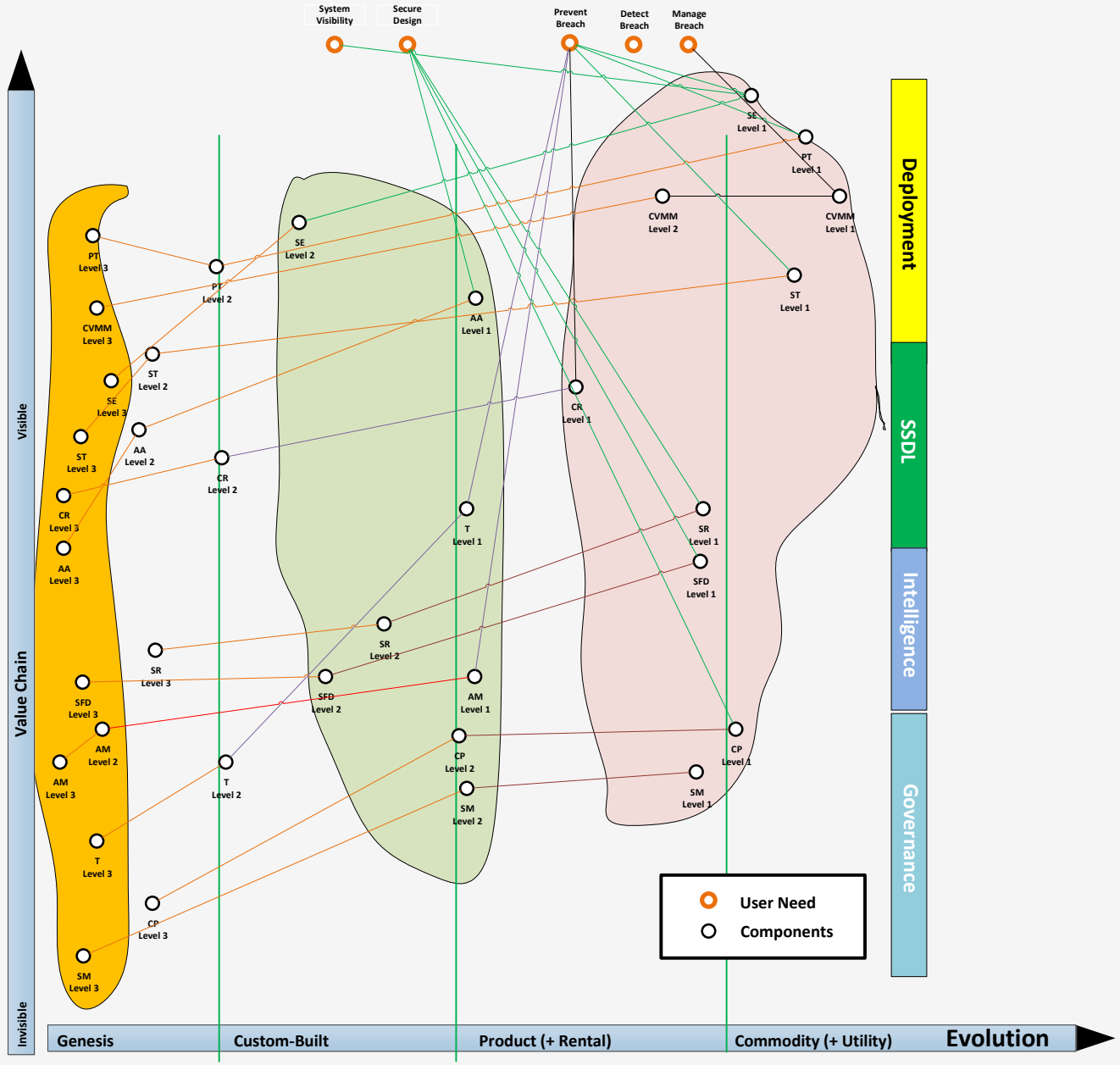
Apply the strengths to the map

- Chaotic won't appear on the map. Those points just need to be stabilized.

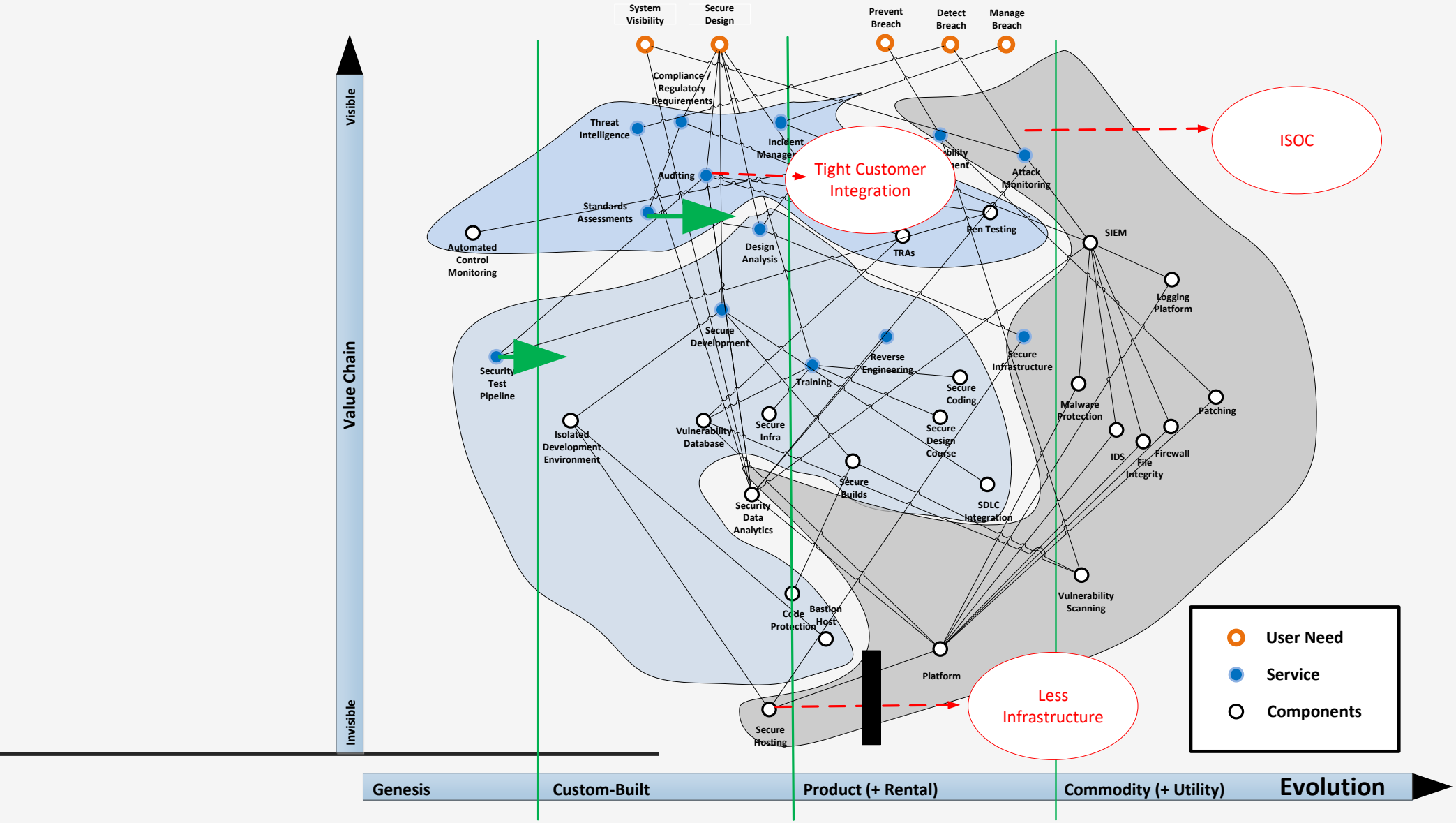
Observations

- Strong understanding of Phase 1
- Some gaps (red) between Phase 1 & 2. We should do training here or bring in outside help.
  - Do competitors have issues here?

- Also Gaps in understanding on top parts of Phase 2. Starting here would be bumpy.

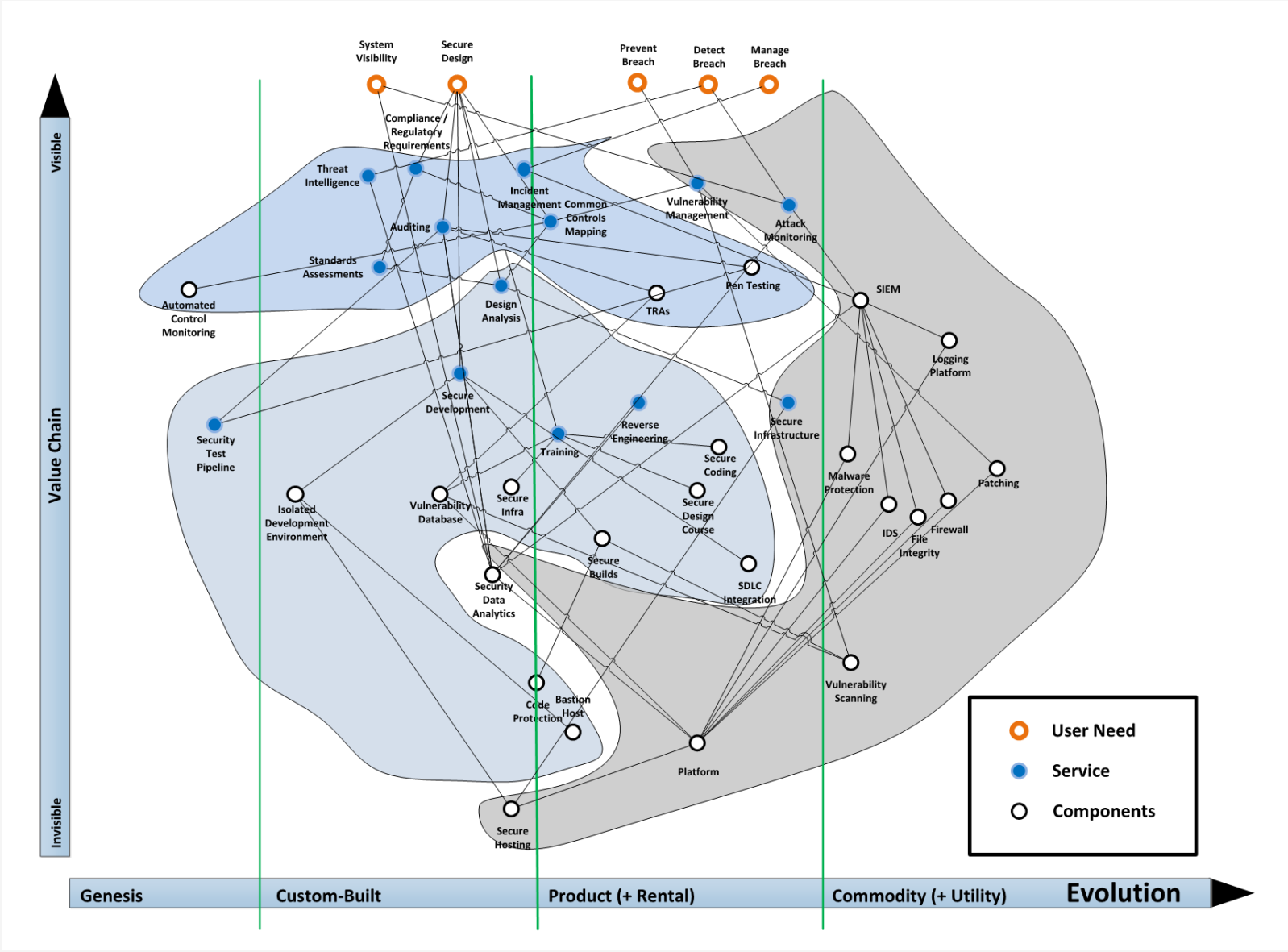


Gameplay outlines your strategy and vision



Using Wardley Maps to Lead a Security Program

- Use the product evolution scale help determine what should be done in-team, in-house and external
- Your priorities are linked to user needs by visibility
  - If you have disagreements, you've saved months/years of work
- New on-fire issues should be linked into your map.
  - But you can put CryptoLocker in **your** context
  - If not, you should wonder why



1. Do interviews and get a Risk Register
   - Use frameworks after you have your core priorities

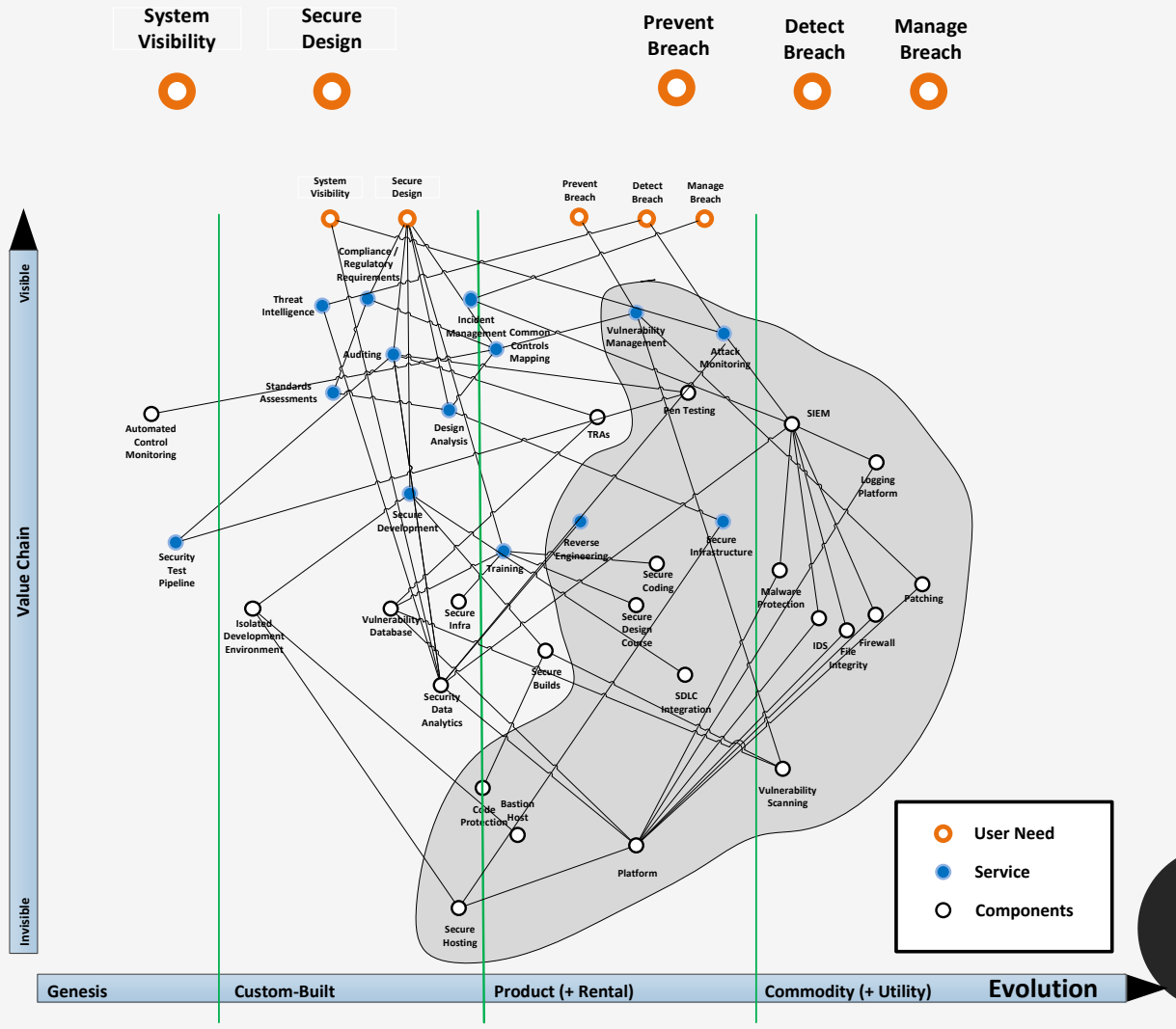
2. Establish User Needs

3. Put on the priorities (the dots)
   - What are their dependencies?
   - Where are they on the evolution scale?
   - How visible are they to the user?
   - (Try with all the new stuff you've learned this week)

Communicate it. Talk to everyone

- It will be wrong.
- The only accurate map of Canada is the size of Canada.

Date	Description	Person	Impact	Customer Example



HOW DO I GET STARTED?

John Duffy

Director, InfoSec, ID & Payment - CBN

john@duffy.dev

@jduffy

learnwardleymapping.com

Pragmatic Wardley Mapping

Ben Mosior

cbnco.com

YOU ARE HERE: BUT MAYBE YOU CAN BE THERE