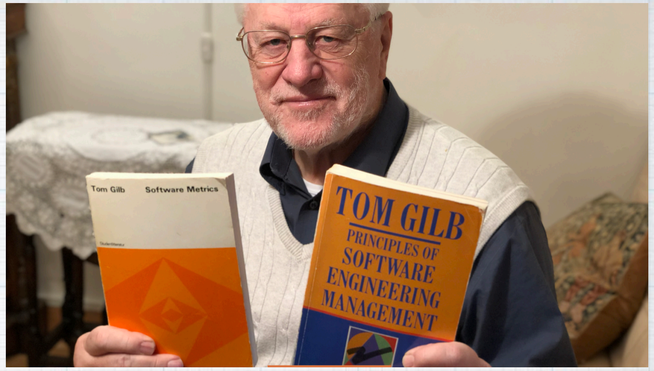
Wednesday 1st FEBRUARY 2023
1 HOUR WITH QUESTIONS
Videoed

Slides Folder



Quantifying Software Security-Engineering Cyber-Security

BY TOM GILB

Software Excellence Academy
Host
Linda Westfall

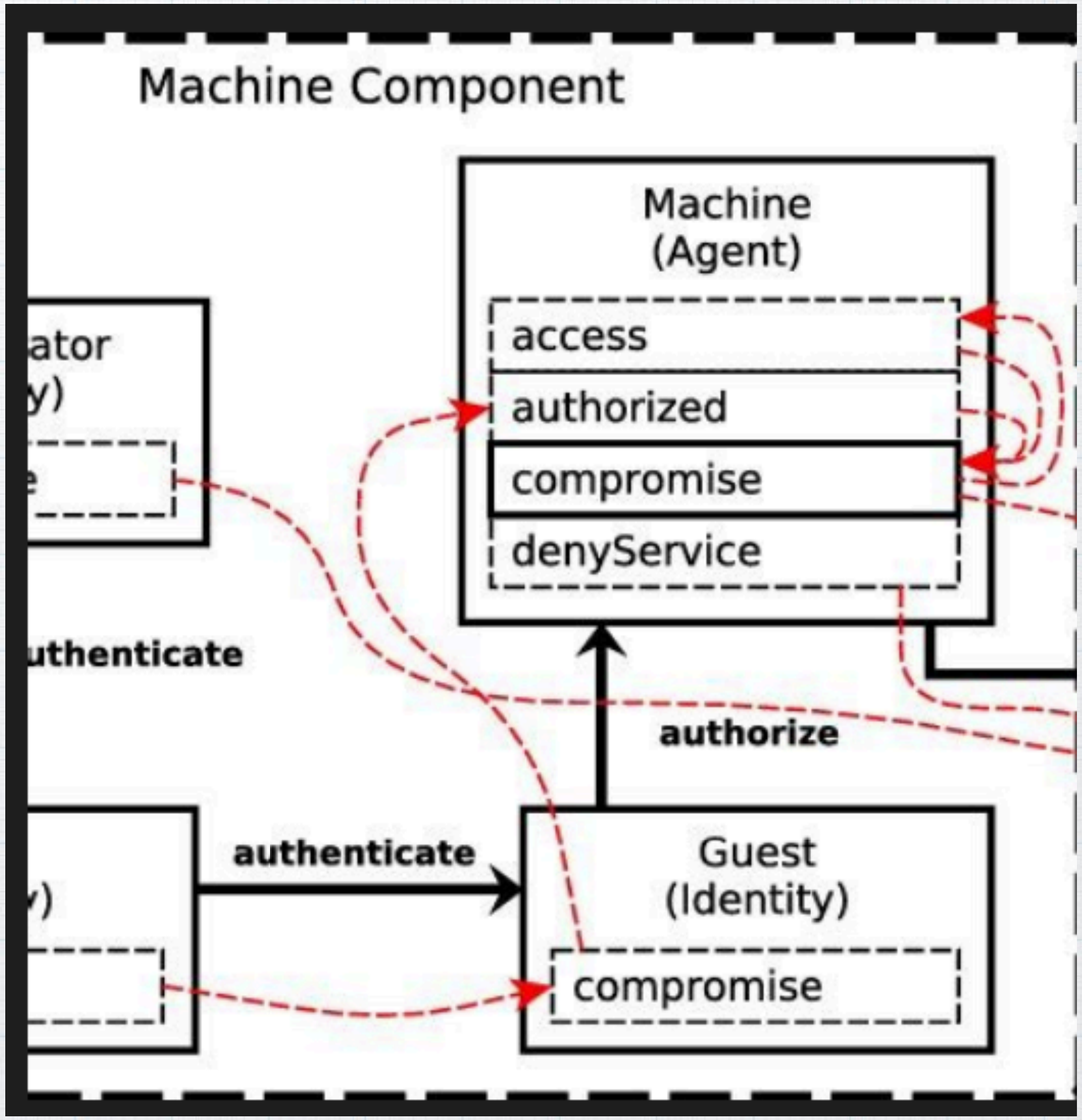
These slides are at:https://www.dropbox.com/sh/mbnkowk9um11etz/AAD56kdXp5YXgqOuZQKFR7U-a?dl=0

Tom@Gilb.com, gilb.com, +47 92066705, twitter.com/ImTomGilb, http://www.linkedin.com/in/tomgilb

Security and Engineering

- I got to know a Cyber Security professor from a first class nordic engineering university
- And noticed none of his papers made any attempt to quantify security
- So I asserted that it should be obvious we need to quantify security in order to engineer it
- And offered to teach him how to do so
- He replied: maybe- but I do not have time to do that now because of all my academic duties.
- I was shocked
- **If you believe security is a serious and complicated systems engineering subject then, never trust any source of security ideas, who cannot, or will not quantify security**
  - **They are not 'secure' sources (fake news)**
  - **Real experts can quantify their subjects**

What do the arrows mean?



Do you see any hint of quantification or engineering here?
This is from a Cyber Security Professor

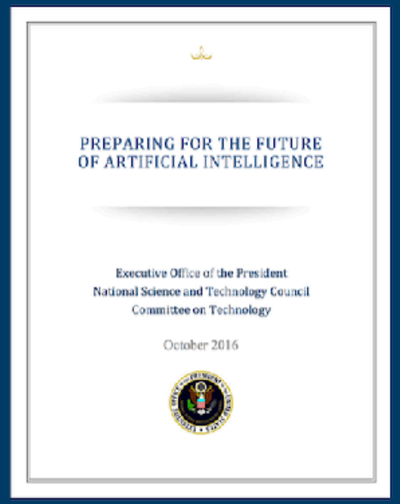
The best USA Universities with AI Security not quantified: for the President

With the *usual lack of quantified definition*

How much security does a government AI system require in 2023?

Oct. 2016 : "Preparing for the future of AI" from USA

IBM


PREPARING FOR THE FUTURE
OF ARTIFICIAL INTELLIGENCE

Executive Office of the President
National Science and Technology Council
Committee on Technology

October 2016

Distributed material

Proposal of Discussion toward Formulation of AI R&D Guideline

Referring OECD guidelines governing privacy, security, and so on, **it is necessary to begin discussions and considerations toward formulating an international guideline consisting of principles governing R&D of AI to be networked** ("**AI R&D Guideline**") as framework taken into account of in R&D of AI to be networked.

Proposed Principles in "AI R&D Guideline"

1. **Principle of Transparency**
   Ensuring the abilities to explain and verify the behaviors of the AI network system
2. **Principle of User Assistance**
   Giving consideration so that the AI network system can assist users and appropriately provide users with opportunities to make choices
3. **Principle of Controllability**
   Ensuring controllability of the AI network system by humans
4. **Principle of Security**
   Ensuring the robustness and dependability of the AI network system
5. **Principle of Safety**
   Giving consideration so that the AI network system will not cause danger to the lives/bodies of users and third parties
6. **Principle of Privacy**
   Giving consideration so that the AI network system will not infringe the privacy of users and third parties
7. **Principle of Ethics**
   Respecting human dignity and individuals' autonomy in conducting research and development of AI to be networked
8. **Principle of Accountability**
   Accomplishing accountability to related stakeholders such as users by researchers/developers of AI to be networked

17

4. Principle of Security

Ensuring the robustness and dependability of the AI network system.

Are "Robustness" and "Dependability" the only aspects of security?
How can we test that the security is "ensured" ?

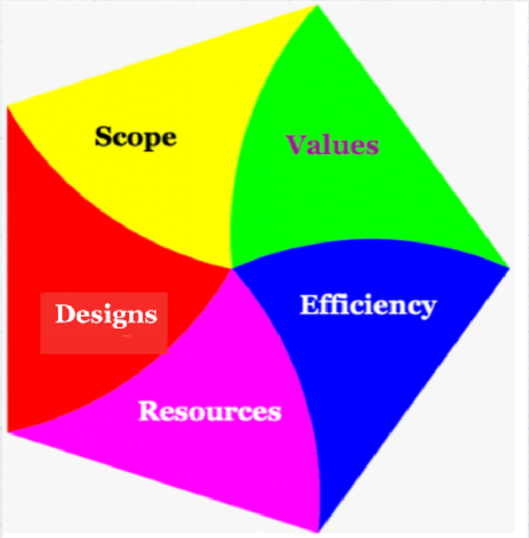
XAI Explaining AI

http://concepts.gilb.com/dl958

Security is a Stakeholder Value Requirement
What are 'value' Requirements?

- **Value Requirements are the <u>most important requirements</u> for any project.**
- **They are the <u>main purpose</u>, and main justification, for a project.**
- **Value requirements start life as value 'attributes' needed by 'stakeholders'. Like qualities, security, usability, low tech debt.**
- **No project can deliver all 'desired' values, by a deadline.**
- **No project will find all desired stakeholder values to be worth delivering.**
- **So all value requirements start life by being acknowledged as possible delivery candidates. We call them 'Wish Level' statements.**
- **Then we *potentially* reclassify them as committed project requirements, which we call a 'Goal' level value requirements**

See next slide for detail



Penta Model

A simple framework for organizing Security ideas

‘Penta' Definitions: The Agile Penta Conflict Balance

- The PENTA is a simplified model of 5 basic *conflicting forces* in any system, which can be adjusted to give a more optimum balance.
- The **PENTA Forces** are: Scope, Values, Efficiency, Resources, and Designs. ‘SVERD’ (Norwegian for ‘Sword”).

1. **Scope**: is the specified set of stakeholder- and system- *functions* (what it must **do**) and *constraints* (what it must ***not*** do). Scope draws a border around a given system.

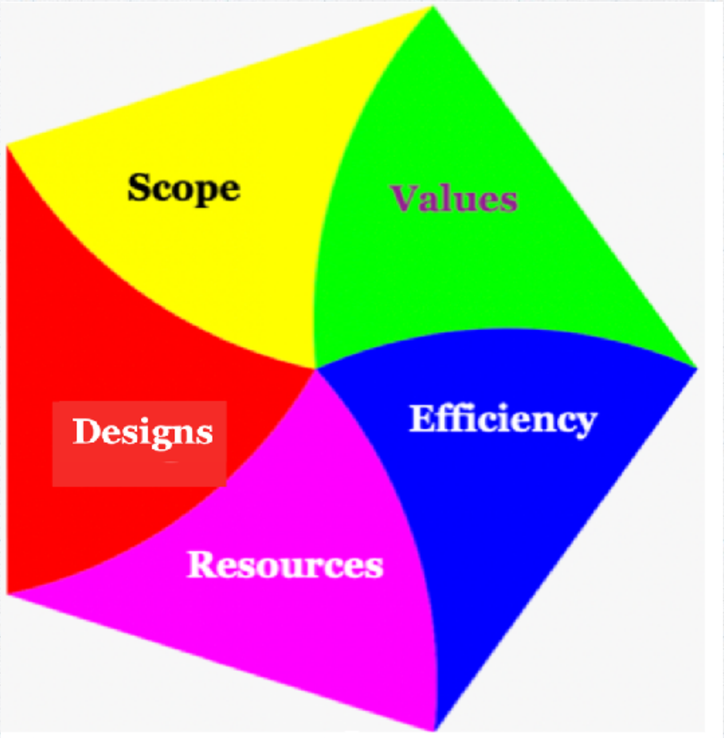
2. **Values**: is the specified set of *stakeholder* values (‘wants’, ‘needs’, ‘wishes’, ‘visions’) and *system* qualities, including system performance attributes (‘*potential* values’ for stakeholders).

3. **Efficiency:** is **‘effectiveness-to-costs ratio’**. *Effectiveness* includes all stakeholder-values actually *delivered*. The costs are life-cycle costs, not just ‘capital’ costs. This is a view outside the black box of Designs.

4. **Resources:** are any *critical* and *prioritized*, set of *limited* resources, such as time, money, people, space.

5. **Designs**: are any types of ‘implementable ideas’ (designs, strategies, architecture, solutions) which we use, in order to deliver a ‘best available’ balanced delivery of Values, Efficiency, Resources, and Scope. The other 4 Quints.

- **Imperfect**: The Penta model is never complete, updated or fully detailed. It can be simplified and summarized. It can view selected components, that are *useful* for consideration.
- **Planguage:** Planguage [CE] can be used to define concepts, and specify details, as well as to evaluate balance (Impact Estimation Tables).
- **Freeware**: The Penta ideas are Creative Commons for free non-exclusive exploitation for everyone.



The Penta Model Paper Alone August 2022
https://tinyurl.com/PentaPaper
URL: https://tinyurl.com/SIMPLEGilb

[CE] Tom Gilb, Competitive Engineering: A Handbook For Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage (paper or digital 2005). The definition of the Planguage, https://www.gilb.com/p/competitive-engineering (free pdf)

Basic Principles of Security Engineering.

This is the main talk outline

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

1.Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.

2.You
3.A Sy
4.An e
varia
5.Syst
cycle
6.The
mea
7.A se
cons
8.Secu
Leve
arch
9.Secu
exan
10.Sec
threa

Balance.
her
and life-
sign,
h the least
Target
ems
ty.
evels, for
security

2.Stakeholder Level

Create...

Stakeholders

- Academic Students
- App Extension Instances
- C Level Managers
- Chief Security Officer
- Gilbs Method Creators
- Government Planners
- Individual Self-Learner
- Management Consultants
- Method Teachers
- Planners
- Planning Methods Experts
- Potential App Buyers
- Private Organizations
- Professional Students
- Public Institutions
- Quality Controllers Auditors
- Trainers Coaches
- University Teachers
- VALUE REQUIREMENTS

Values and Resources

- Confidentiality
- Extendability
- Independence
- Intelligibility
- Interoperability
- Language Capability
- Management Attractiveness
- Method Consistency
- Responsibility Control
- Reuse Capability
- Scalability
- **Security**
- Selectiveness
- Sharing Capability
- Simplicity
- Spreading Capabiity
- Tailorability
- Usage Tenacity
- Value Focus
- Visualization
- Warning Capability

Security

Level: Stakeholder, **Status:** Not Determined **Type:** Value, **Labels:** no labels Edit

Is Part Of: VALUE REQUIREMENTS

Status 10 — Wish 1

Wish [Attack Sources = **Professional Hackers**, Attack Consequences = **Sensitive Information**, Coped With = **Detected**, Attacks = **Threats**] @ 16 Oct 2017 : **1** % Attacked Coped with <- *tg*

Ambition Level: enterprise and government levels of security for any threats, attacks or accidents: I...

Scale: % probability of of [Attacks] from [Attack Sources] with [Attack Consequences] being [Coped...

Stakeholders: App Sales Channels, C Level Managers, Chief Security Officer, DSS Norwegian Gove...

Status: 10 % Attacked Coped with [Attacks = **Threats**, Attack Sources = **Professional Hackers**, Attack Cons...

Wish: 1 % Attacked Coped with [Attack Sources = **Professional Hackers**, Attack Consequences = **Sensitive I...**

Implementation Plan: see Design 'Security Confidentiality Independence'1.Protection: The protecti...

Test: Security testing will be done by independent specialist Security Expert Contractors.1. They will...

Due: **Planned** (by end of): ?

1.Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.

2.You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.

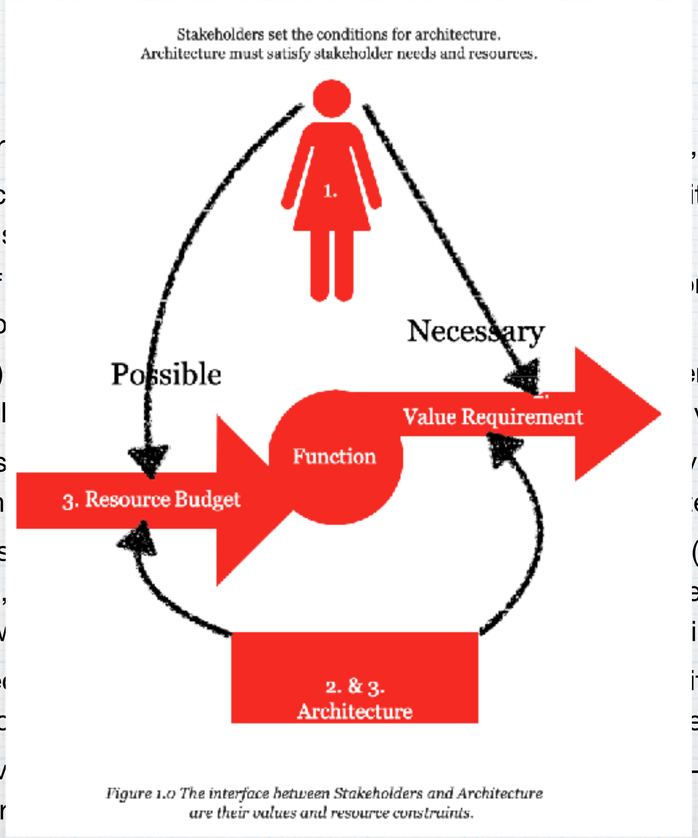


Figure 1.0 The interface between Stakeholders and Architecture are their values and resource constraints.

3.A Systems Architect is one name for the instance that co-or , including security.

4.An engineering approach is necessary, to model large and c ity. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the s

5.Systems and Security Engineering must include a means of n all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Tab

6.The safest proven approach (See IBM Cleanroom, Quinnan) r small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail vered. This is Agile Security Engineering.

7.A security design can be absolutely anything, not violating s impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything mean em.

8.Security requirements can state minimum levels (constraints (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, ese 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example v imension loses all current priority.

9.Security engineers need to co-operatively recognize that se ities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustwo erability and many others.

10.Security engineering and maintenance of good security lev -front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime oper

Figure 1.5 B. 'Value Side-effect's and costs This is your first peek at a major architectural tool, an Impact Estimation Table (IET). In this case 4 architectures (strategies) are rated (estimated) for potential impact on the 9 UN Sustainability Goals. More later about this method. But I pulled it out to show the idea of side effects, and costs. Your architecture impacts it all, and you had better keep track.

Requirements	S1 End Poverty	S2 End Hunger Str...	S3 Healthy Lives	S4 Quality Educatio
G1. Poverty (Decomposed) Status: 0 → Goal: 100 % of sub-g Δ%:	95 / 95 % / 95%	???? / 0 % / ????	???? / 0 % / ????	???? / 0 % / ????
G2 End Hunger Δ: Status: 0 → Goal: 100 % of sub-g Δ%:	42 / 42 % / 42%	96 / 96 % / 96%	???? / 0 % / ????	Δ: ???? / Δ%: 0 % / ????
G3 Healthy Lives Δ: Status: 0 → Goal: 100 % of sub-g Δ%:	-42 / -42 % / -42%	23 / 23 % / 23%	???? / 0 % / ????	???? / 0 % / ????
G4 Quality Education Δ: Status: 0 → Goal: 100 % of sub-g Δ%:	5 / 5 % / 5%	-12 / -12 % / -12%	???? / 0 % / ????	???? / 0 % / ????
G5 Gender Equality Δ: Status: 0 → Wish: 100 % of sub-g Δ%:	-5 / -5 % / -5%	0 / 0 % / 0%	???? / 0 % / ????	???? / 0 % / ????
G6 Water And Sanitation Δ: Status: 0 → Wish: 100 % of sub-g Δ%:	???? / 0 % / ????	42 / 42 % / 42%	???? / 0 % / ????	???? / 0 % / ????
G7 Energy Access Δ: Status: 0 → Goal: 100 % of sub-g Δ%:	0 / 0 % / 0%	3 / 3 % / 3%	???? / 0 % / ????	???? / 0 % / ????
G8 Employment And Growth Status: 0 → Goal: 100 % of sub-g Δ%:	33 / 33 % / 33%	-12 / -12 % / -12%	???? / 0 % / ????	???? / 0 % / ????
Sum Of Values: Σ%:	128 %	140 %	0 %	0 %
Annual Cost Per Dwelling Δ: Status: 5 → Budget: 1 Cost for t... Δ%:	-1 / 25 % / 25%	-0.5 / 13 % / 13%	???? / 0 % / ????	???? / 0 % / ????
Years To Do Δ:	3	4	????	????

1.Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.

2.You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.

3.A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.

4.An engineering approach is ... nts and other variable stakeholder values,

5.Systems and Security Engi ... er values) and life-cycle system-resource cons

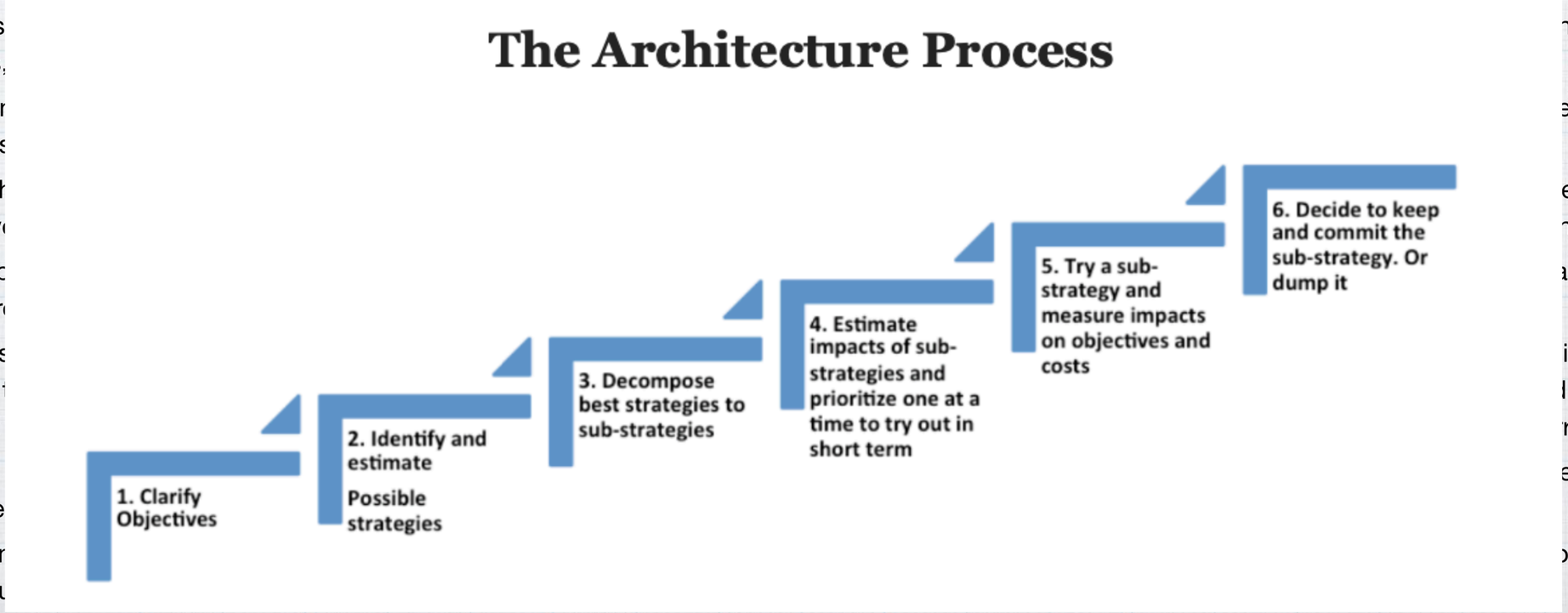
6.The safest proven approach ... ecurity design, measure actual security lev ... neering.

7.A security design can be ab ... argets, with the least consumption of budgeted r

8.Security requirements can s ... ifference (Target Level - Constraint Level) in ... l the systems architect determine current ... rent priority.

9.Security engineers need to ... eresting levels, for example usability, safety, re

10.Security engineering and r ... onitor the security threats, and necessary secu



Architecture Organization and Responsibility

So, how should Enterprise Architecture organize itself? And how should EA relate to the Enterprise, and its external stakeholders?

FACTUAL ARCHITECTURE: Architecture itself will focus on a **fact-based, evidence-based analysis and presentation, and real implementation,** of architecture.

<u>Basic architecture objective:</u>

Do anything that in fact

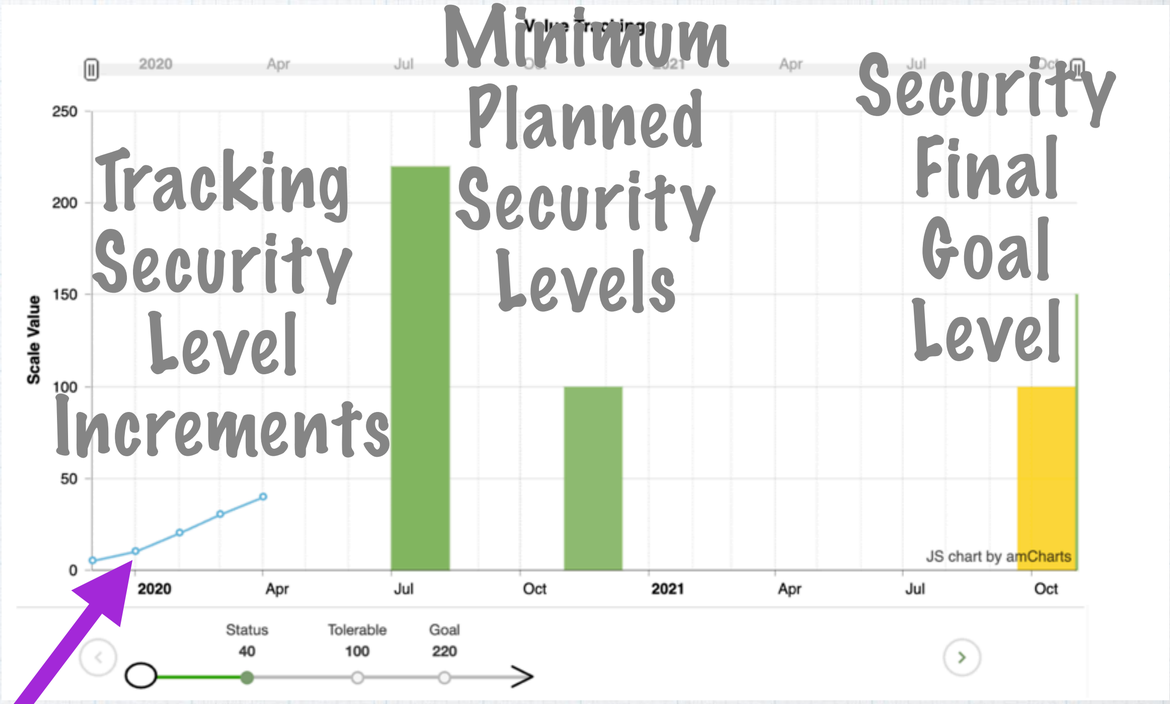
<u>measurably works,</u> to help the Enterprise deliver the targeted Enterprise values, at lowest costs.

MAIN IDEAS FOR SECURITY AND ALL THE OTHER PRIORITIES

1. **VALUE CLARITY: Serious focus on all levels of Architecture requirements, *not just Architecture level, but at Project level, for all design,* too, especially the non-financial Values and Qualities. KEEP FOCUS ON VALUES, NOT ON TECHNOLOGY ITSELF**
2. **SELL CLARITY TO ALL: Architecture must point out the need for, and demand, clear requirements (especially values and qualities)**, so we can do good architecture.
3. ***AGILE ARCHITECTURE:*** **Architecture will apply an agile model, with sub-architecture, measurement and prioritization.**

Security Designers will prove that their architecture actually works, in delivery of priority objectives and values.

Set an example that impresses with visible results. Managers like that stuff.



Numeric feedback tracking of Security levels
Superimposed on the
Numeric levels of planned security levels

https://tinyurl.com/VIEbooklet

1.Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.

2.You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.

3.A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.

4. An *engineering approach* is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.

5.Systems and Security Engineering must include a means of both estimating, and measuring the multi

6.The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will atte
This is Agile Security Engineering.

7.A security design can be absolutely anything, not violating stated system constraints, which gives the

8.Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-d
help the security engineer and the systems architect determine current priorities as system developme

9.Security engineers need to co-operatively recognize that security itself is ultimately dependent on ma
and many others.

10.Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a

This Impact Estimation Table is symbolic of the 'engineering' approach

Corona Virus Management Norway Total Architecture

From Level: Stakeholder To Level: Stakeholder

Settings... | + Add | Help me! | Show VDT Sidebar

Requirements	Health Architecture	Transport Archite...	Tore Architecture	Workplace Archite...	Sum
Collect Information Status: 50 → Wish: 90 % [Relevan...	5 13 %	20 50 %	25 63 %	30 75 %	✔ ΣΔ%: 201 %
Education Status: 42 → Wish: 95 % [Student...	3 6 %	30 57 %	30 57 %	20 38 %	⚠ ΣΔ%: 158 %
Get People Where They Need ... Status: 42 → Wish: 99 [Important...	???? ????	30 53 %	2 4 %	???? ????	⚠ ΣΔ%: 57 %
Healthy Employees Status: 70 → Wish: 99 [Work Acti...	???? ????	10 34 %	20 69 %	25 86 %	⚠ ΣΔ%: 189 %
Stay Healthy Status: 30 → Wish: 90 % [Capacit...	10 17 %	30 50 %	25 42 %	25 42 %	⚠ ΣΔ%: 151 %
Sum Of Values: Σ%:	36 %	244 %	235 %	241 %	
Days To Implement Status: 0 → Budget: 1k Days Neede...	30 3 %	10 1 %	15 2 %	10 1 %	✔ ΣΔ%: 7 %
Capital Cost In Million NOK Status: 0 → Budget: 1k Million No...	0 0 %	50 5 %	40 4 %	100 10 %	✔ ΣΔ%: 19 %
Sum Of Development Resources: Σ%:	3 %	6 %	6 %	11 %	
Value To Cost:	12.00	40.70	39.20	21.90	
Ratio (Worst Case)	7.30	21.60	25.80	18.80	
Ratio (Cred. - adjusted)	0.70	13.50	14.50	5.00	
Ratio (Worst Case Cred. - adjusted)	2.80	22.50	113.40	292.00	

Figure 12.3.0.9 Source Oslo Sw. Arch. OSWA. Workshop March 2020 Virus Control in Norway: Exercise. ValPlan

SECURITY Architecture

Organization and Responsibility: Some Principles

1. **VALUE RESPONSIBILITY:** Each individual professional in the Enterprise, and its environment, is personally and as a team, responsible for delivering their assigned level of planned stakeholder value improvements.

2. **ARCHITECTURE SPEC INTELLIGIBILITY:** Each professional is responsible for understanding the design, or strategy, or architecture, correctly; for asking for clarification to be sure, and for documenting any issues or concerns, with the requirements and architecture specification

3. **SIDE EFFECT CONSCIOUS**: Each professional is responsible for being aware of both *side-effects*, planned or not, and *resource consumption*, planned or not: and taking action to minimize undesired effects. By design, by design to cost.

4. **LOCAL OPTIMIZATION:** Each professional has the **right** to optimize a design, architecture, or strategy, so that it works more effectively, in *local* conditions. They can add or modify the design, but they must *document* their additions, and the *reasons* for them, and identify the responsible parties for modifications. They will preferably add this modification documentation, immediately when planned, to the relevant global architectures specifications object, or at least transmit it to the Specification Owner.

5. **EXTREME INTELLIGIBILITY:** The architecture level, or any higher-level of design, strategy or engineering, will adhere to necessary quality of Rule-based standards of clear specification, so that misinterpretation is not possible. *"Unambiguous"*

6. **RIGOROUS QC:** Any architecture (or design, or strategy) level will themselves, carry out suitable quality control (SQC using Rules, and Exit levels [CE]), as well as Architecture Reviews to approve the bigger picture, before releasing their designs officially. The level of QC and Review passed or not, will be annotated on each specification object.

8. **CTO RESPONSIBILITY**: The Chief Technical Officer is responsible for all QC and Review methods, for their creations, maintenance and continuous evaluation, for all architecture and similar, or related (Objectives, Constraints, Policies, Contracts), planning.

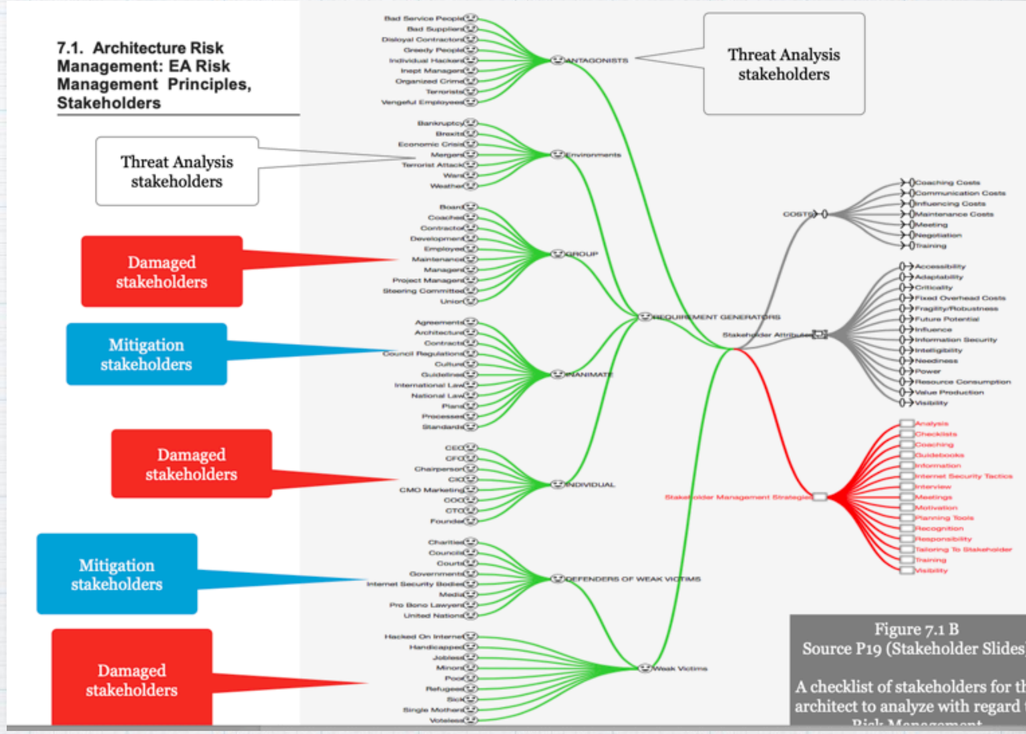
9. **ARCHITECTURE INTELLIGENCE (AI)**: The architecture specification documentation will be digitized, with suitable links to all related entities, and will deliver a maximum of programmed logic, to help follow good-practice standards, to specify clearly, to analyze, and to report possible missing specifications, or inconsistent specifications.

10. **OVERALL OPTIMIZATION:** each higher level of enterprise architecture, and similar planning, are responsible for enabling all related planning levels below, to the side, and even 'above' to co- ordinate their efforts continuously; so that they do not inadvertently sub-optimize, and so that they can sacrifice local benefit consciously, for the greater value of the whole. *In practice this will include such tactics as Impact Estimation Tables, with Landing Zone* flexible objectives, which show the side-effects, and costs of the local plans, in relation to the bigger-picture objectives and constraints.*

10. ***AGILE ARCHITECTURE:*** The Enterprise Architecture will, themselves - or through more-local professionals, be in 'continuous sensing mode' regarding everything: stakeholders, values, technologies, environments, competition, delivery step measures: and directly, or via other colleagues, be prepared to intervene when sensing potentially threatening or opportune data, to change any plans for the better of the larger enterprise.

*

Keeping track of multiple architecture factors
In a complex architecture process



This Systems model is automatically produced by the app ValPlan (the right shaded part)

Notice the Security classifications of Stakeholders

Organizing the Security-Value Architecture-process. Some specialities to train people for.

Who are the players to make system-enterprise architecture-value-delivery really happen? What responsibilities and skills do they need?

Here are some suggestions for specialist roles.

These are mainly responsibilities, rather than full time jobs. But they do require training and knowledge.

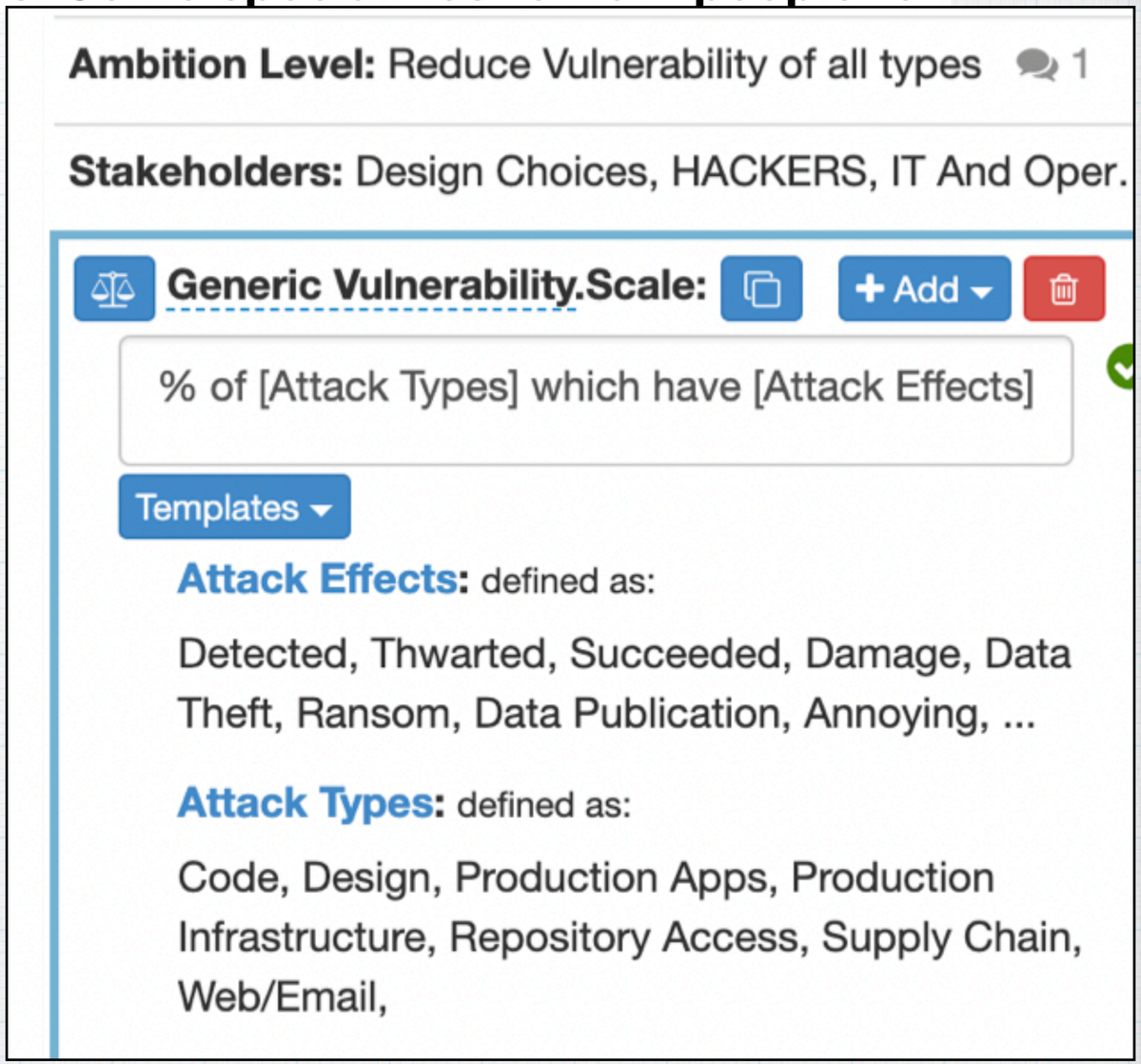
SECURITY **Value Analyst:** analyzes stakeholder needs, and priorities, and selects critical, or possibly critical, needs and specifies them as requirements, at least at the 'Wish' level (potential Goal requirement).

SECURITY **Specification Owner**: a person (or group) which has undertaken responsibility, by name, for the update and maintenance of a specification object, such as an objective, an architecture, or an architecture estimation table..

SECURITY **Implementation Responsible**: a person (or group) which has taken named responsibility, as specified in the specification object, for actual practical implementation of a design object. This can be for an objective level (reach the Goal), or for an architecture (deliver the sub-architecture and try to get the maximum value from it).

SECURITY **Value Designer**: a generic (all possible design areas) designer (or team) who undertakes to identify possible design components to reach a Value Requirement level, on time. To research them as to all side-effects and costs, documenting such facts in the design object and corresponding Value Tables. The Value Designer might hand over exploration of a design idea to a Specialist.

SECURITY **Value Engineering Specialist**: a designer with a narrow speciality (usability, **security**, performance, organizational improvement, AI) who is updated on the state of the art, and has a good international network of people and sources to find good specialist designs.



Specification of 'Vulnerability' Scale of measure, using Planguage and two [Scale Parameters]

This is an engineering specification and requires training in Planguage (at Intel, it is a 2-day course) [Terzakis, Simmons]

Clear Targets
Begin with clear Scales —->

For example: if the Stakeholder says:

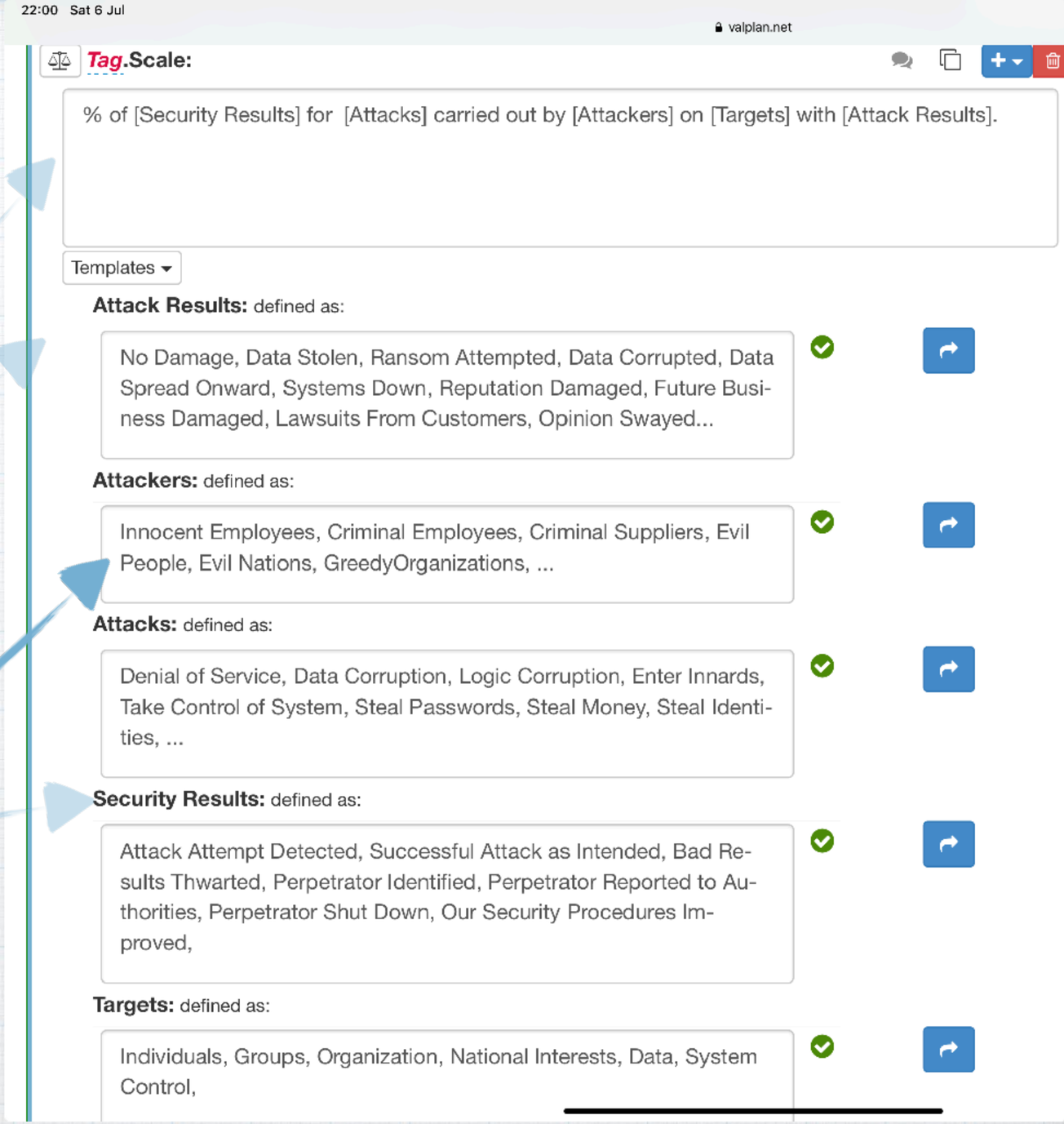
Ambition: ***I want the best security, to fight hackers, and protect my customers and company.***

Or

User Story: ***As a User I want good security, to fight bad guys.***

These are simply *unacceptable* statements

- **No defined scales**
- **No definitions**
- **No conditions [Scale Parameters]**
- **No levels (benchmarks, constraints, targets)**



1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. **Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption.**
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

US DoD. Persinscom **Impact EstimationTable:**

Designs

Requirements

Estimated Impact of Design -> Requirements

Design Ideas ->	*Technology Investment*	*Business Practices*	*People*	*Empowerment*	*Principles of IMA Management*	*Business Process Re-engineering*	*Sum Requirements*
	50%	[illegible]	5%	5%	5%	60%	185%
Availability 90% <-> 99.5% Up time	50%		5–10%	0%	0%	200%	265%
Usability 200 <-> 60 Requests by Users	[illegible]	[illegible]	5–10%	50%	0%	10%	130%
Responsiveness 70% <-> ECP's on time	50%	10%	90%	25%	5%	50%	180%
Productivity 3:1 Return on Investment	45% 50%						303% 251%
Morale 72 <-> 60 per month on Sick Leave							
Data Integrity 88% <-> 97% Data Error %	42%						177%
Technology Adaptability 75% Adapt Technology	5%						160%
Requirement Adaptability ? <-> 2.6% Adapt to Change	80%						260%
Resource Adaptability 2.1M <-> ? Resource Change	10%	80%	5%	50%	50%	75%	270%
Cost Reduction FADS <-> 30% Total Funding	50%	40%	10%	40%	50%	50%	240%
Sum of Performance	*482%*	*280%*	*305%*	*390%*	*315%*	*649%*	
Money % of total budget	15%	4%	3%	4%	6%	4%	36%
Time % total work months/year	15%	15%	20%	10%	20%	18%	98%
Sum of Costs	*30*	*19*	*23*	*14*	*26*	*22*	
Performance to Cost Ratio	*16:1*	*14:7*	*13:3*	*27:9*	*12:1*	**29.5 :1**	

Figure 6.1.3 C. Source: An Agile Project Startup Week MASTER.
Also 111111 Unity Method of Decomposition into weekly increments of value delivery.
Case Study US Dept. of Defence. (10 min slides). http://www.gilb.com/DL451

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. **The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.**
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

Mills on Design to Cost

- "To meet cost/schedule commitments based on imperfect estimation techniques, a software engineering manager must adopt a manage-and-design-to-cost/schedule process.
- That process requires a continuous and relentless rectification of design objectives with the cost/schedule needed to achieve those objectives."
- in IBM sj 4 80 p.420

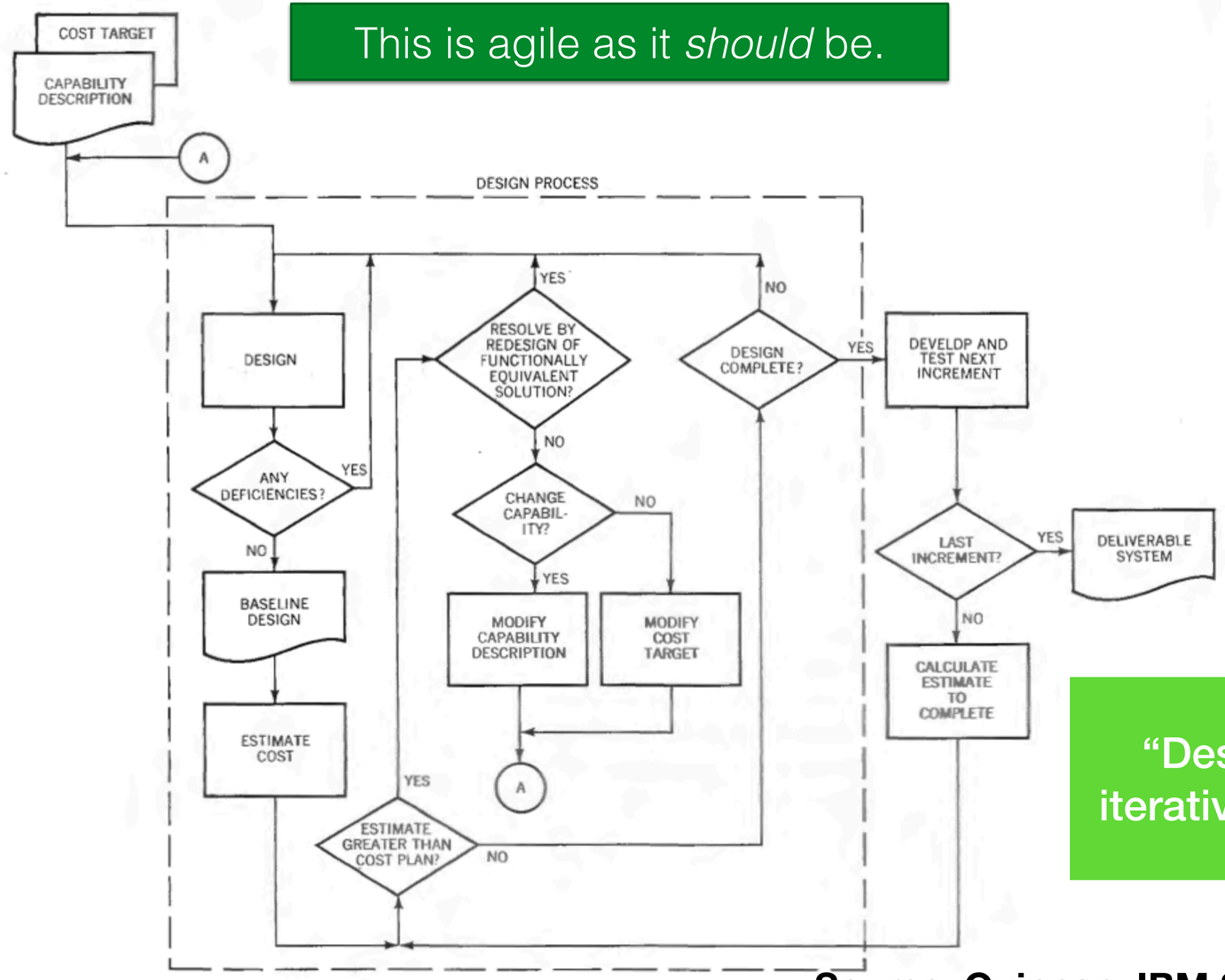
MILLS AND QUINNAN IBM CLEANROOM CASE IN GILB, BCS SPA 'VALUE DESIGN' 2 HOUR COURSE.
Video URL= https://www.youtube.com/playlist?list=PLKBhokJ0qd3_wlvr0j85YhmNfNj8ZJ8M-
Slide Location: = http://concepts.gilb.com/dl972

Dynamic Design to Efficiency (Value/Cost): The Architect in the Agile Loop (IBM Cleanroom, Evo)

* Does your 'Enterprise Architect'. Haha :)
* Redesign things
* If necessary
* For better cost or quality
* At every 'sprint' ?
* And achieve on-time, under budget, high quality in defence and space software?

Figure 3 Design-to-cost

This is agile as it *should* be.



Quinnan

"Design is an iterative process"

Source: Quinnan, IBM SJ, page 473

http://trace.tennessee.edu/cgi/viewcontent.cgi?article=1004&context=utk_harlan

And PoSEM page 105, Figure 7.10

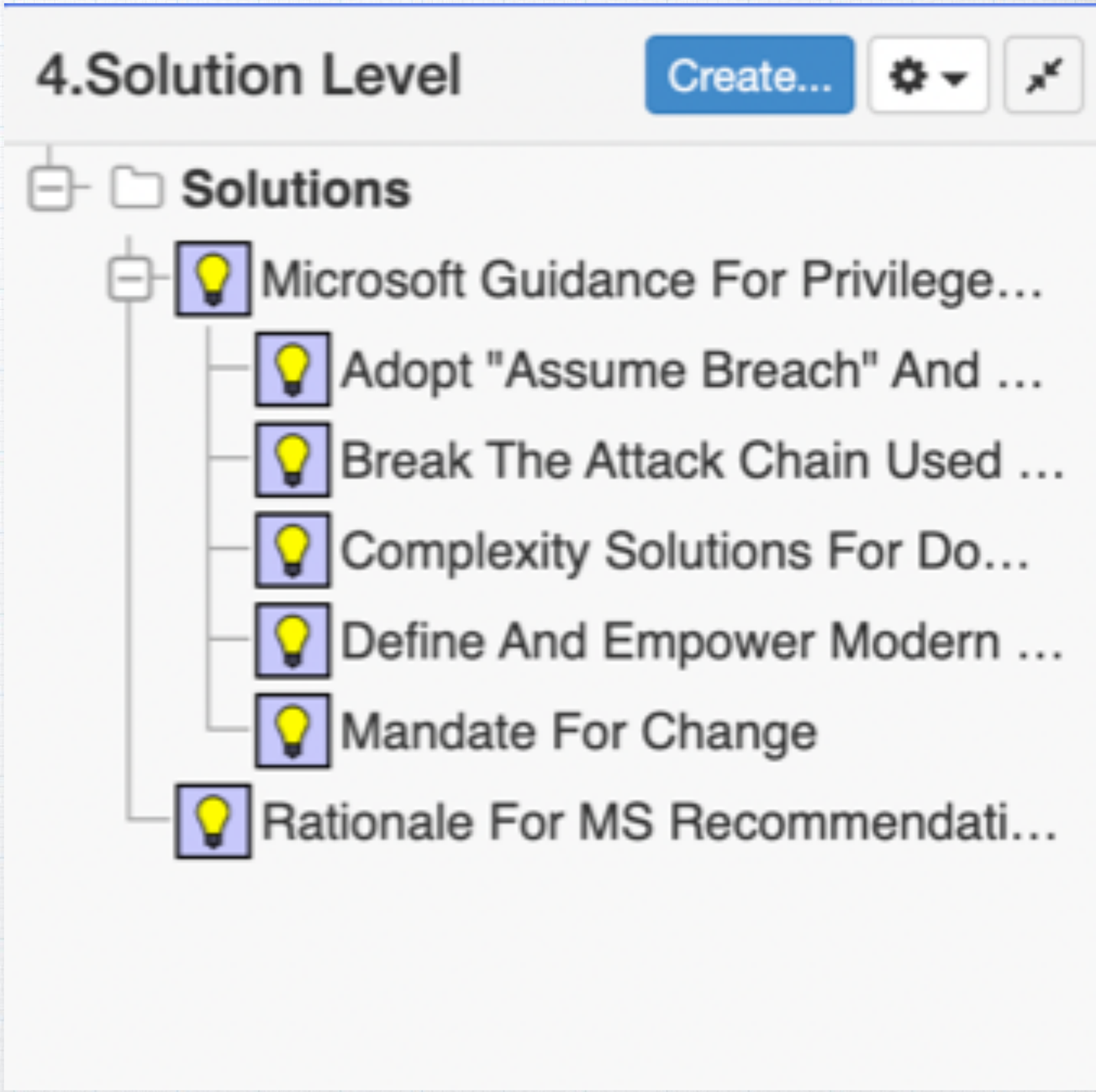
Value Agile book free

https://www.dropbox.com/sh/o2g7ib3z2g2uzfw/AAAypXlN0yA2WS4obwlDzZR3a?dl=0

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.

7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.

8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.



Some security designs from Ben Hanson, Microsoft 2022

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.

8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels).

We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels.

These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses.

For example when targets are reached the security or other quality dimension loses all current priority.

9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.
10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

Security:

Scale: % probability of detecting a hacker within 5 seconds.

Status: 10% last year.
(Benchmark level)

Tolerable: 80% by End this year.
(Constraint Level)

Wish: 98% by End Next Year.
(Target Level)

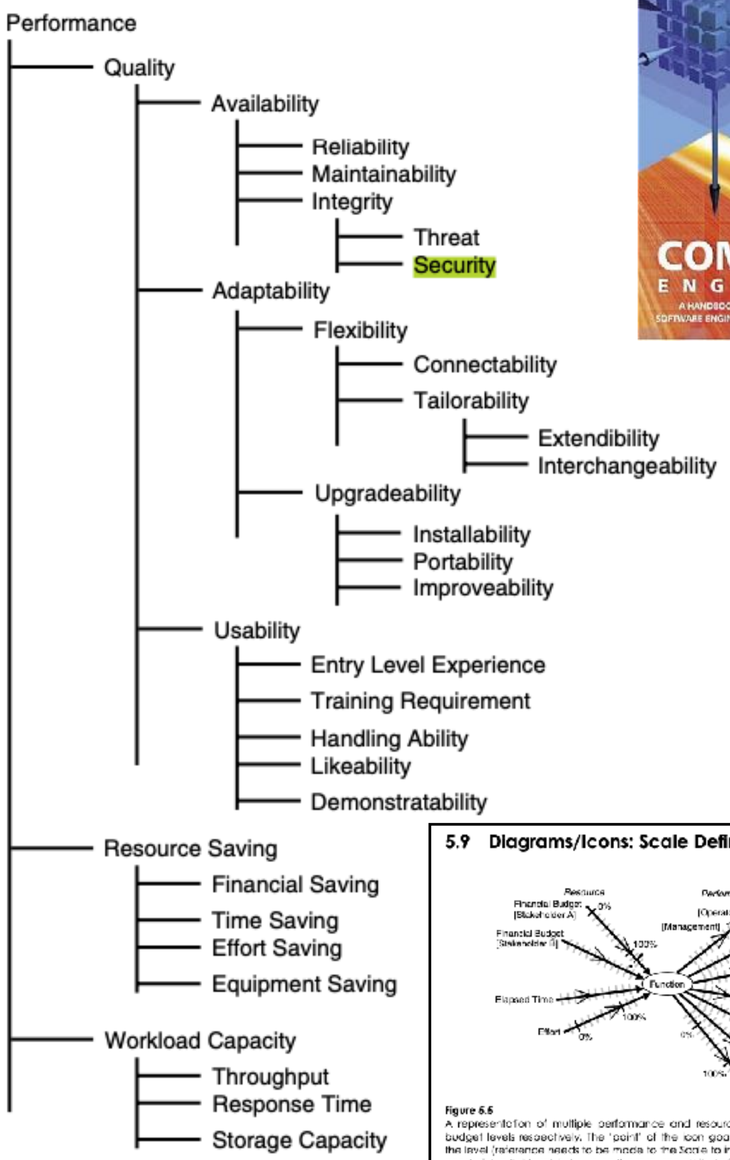
Security Template 2005

1.1.3 **Integrity**: 'The ability of the system to survive attack.'
Gist: Integrity is a measure of the confidence that the system has suffered no harm: its security has not been breached and, its use has resulted in no 'corruption' or impairment to it. An attack on the Integrity of a system can be accidental or intentional. The Integrity of a system depends on the frequency of *threat* to it and the effectiveness of its *security*.

Integrity: Type: Elementary Quality Requirement.
Scale: Probability for a defined [System] to achieve defined [Coping Action] with defined [Attack] under defined [Conditions].
Coping Action: {detect, prevent, capture}.
Integrity: Type: Complex Quality Requirement.
Includes: {Threat, Security}.

Chapter 5,Scales of measure. Gilb: 'Competitive Engineering', 2005
http://www.gilb.com/DL26

154 Competitive Engineering



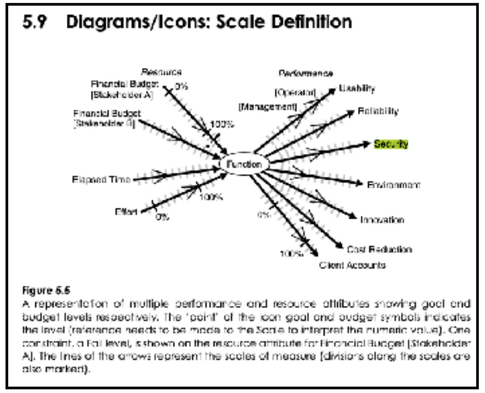


Figure 5.3
One decomposition possibility for performance attributes with emphasis on the detail of the quality attributes.

1976-7 Software Metrics book

SYSTEM INTEGRITY MEASURES

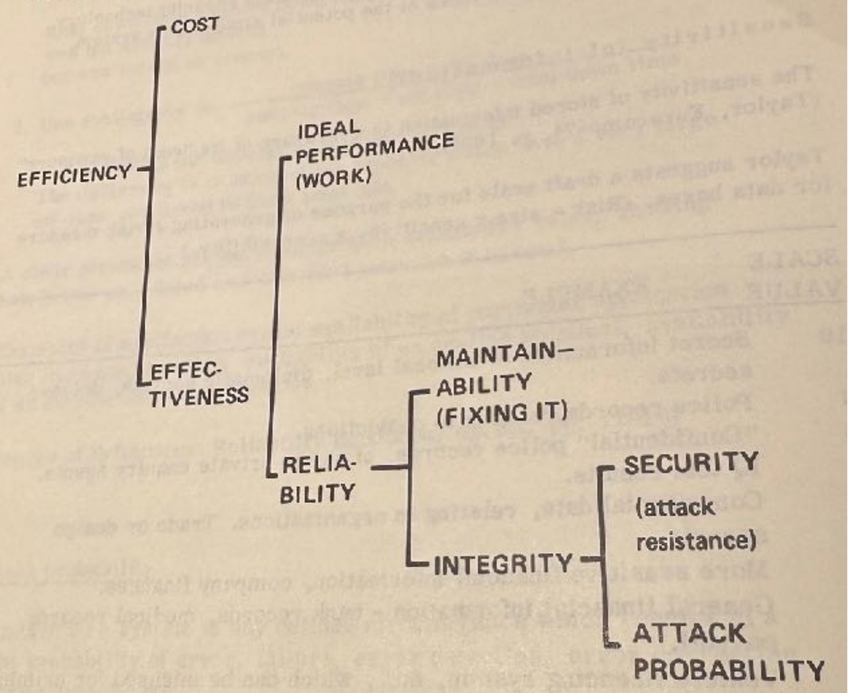


Fig 77.

Security probability: attack repulsion probability

Pr. $S_s(a)$ = the probability of successful attack (type a) rejection in system s, at any time.

RESPONSI-BILITY	SYMBOL	DEFINITION
Attack		
SYSTEM ANALYST	$At_s(a, t)$	= PROBABILITY of ATTACK of TYPE "a", DURING TIME "t" FOR SYSTEM "s"
Security		
SYSTEM DESIGNER	$S_s(a)$	= PROBABILITY of SUCCESSFULL RESISTANCE to ATTACK TYPE "a"
Integrity		
FINAL SYSTEM CONTRACTOR or USER	$Ig_s(a, t)$	= PROBABILITY of SYSTEM (S) SURVIVAL (= integrity, no undetected attacks) WHEN SUBJECTED to ATTACK "a" DURING INTERVAL "t"

$Ig = 1 - (At(1 - S))$

The SYSTEM INTEGRITY is RELATED DIRECTLY to the ATTACK FREQUENCY and the STRENGTH of SECURITY DESIGN for THOSE PARTICULAR ATTACKS

Example: ATTACK FREQUENCY 2% (.02), SECURITY DEGREE FOR THIS ATTACK 90% (.90)

$Ig = 1 - (.02(1 - .90))$

$= .998$ or 99.8%

Note: THE TOTAL SYSTEM INTEGRITY IS GIVEN BY THE PRODUCT OF THE INTEGRITY VALUE FOR EACH ATTACK !

Fig 78. System integrity measures

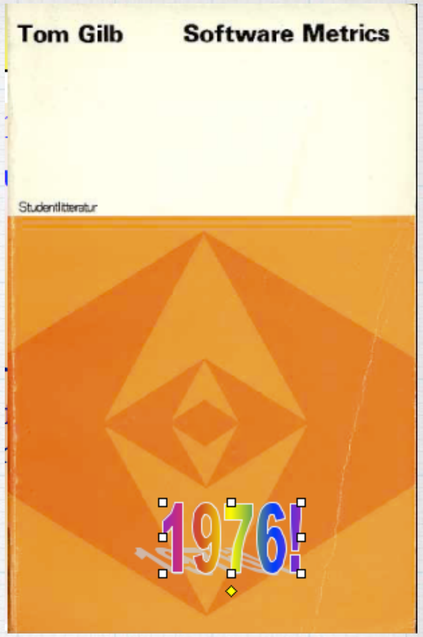
Attack probability

An attack on a system is any defined circumstance which results in a given probability of error, failure, error detection, error correction, security breach, etc.

Pr. $At_s(a, t)$ = the probability of an attack of type a on system s during time interval t.

This is an expression of the frequency with which latent problems occur. The success of these latent problems is then determined by the effectiveness of the error detection, error correction and security devices which are able to handle that type of attack.

Examples of types of attacks:



1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.

9. **Security engineers need to co-operatively recognize that <u>security itself</u> is ultimately dependent on many <u>*other qualities*</u> of the system *also* being attained, at high interesting levels,** *for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.*

10. Security engineering and maintenance of good security levels is a never-ending lifetime battle, not a one -time up-front design effort; so persistent resources to monitor the security threats, and necessary security levels, must be a part of the lifetime operational costs, of any large and complex system.

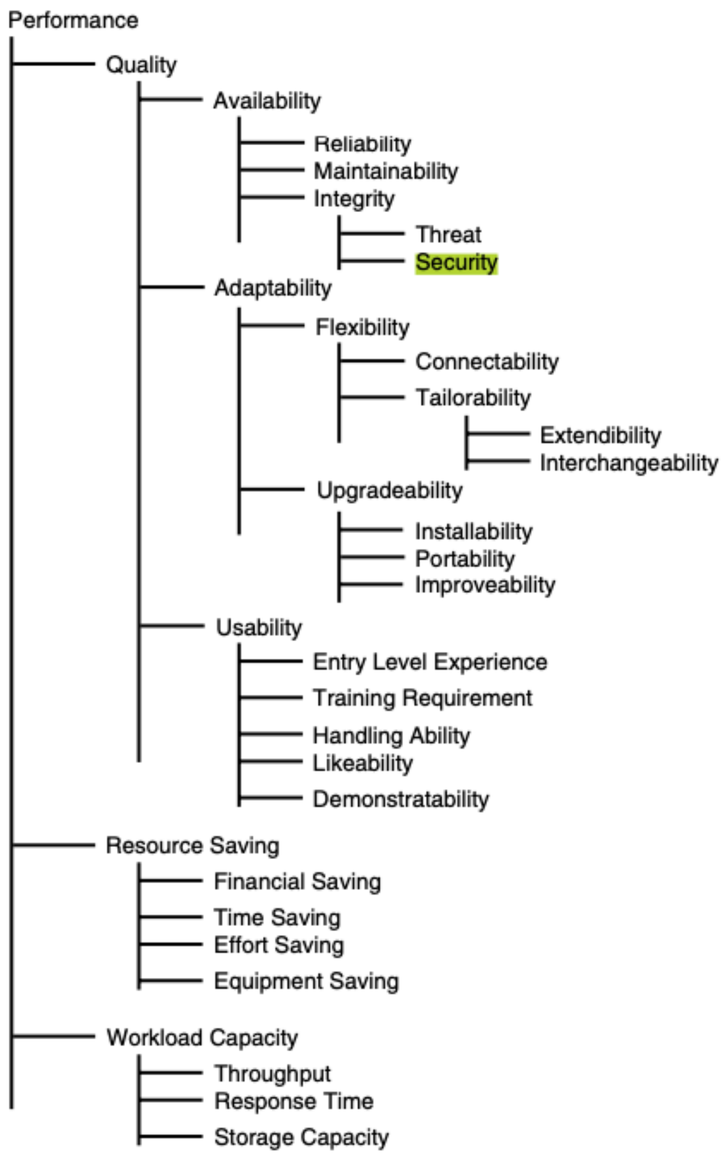
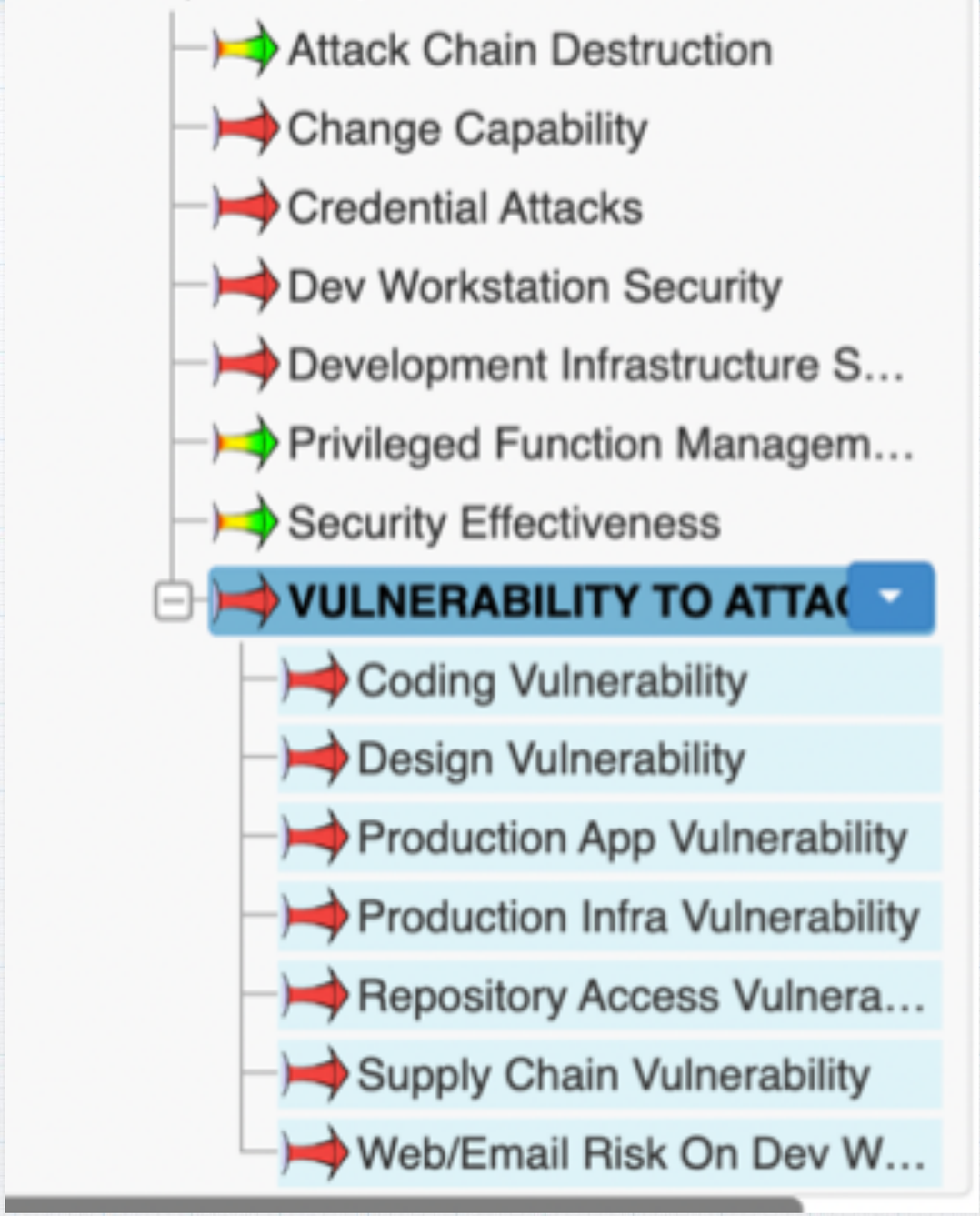


Figure 5.3
One decomposition possibility for performance attributes with emphasis on the detail of the quality attributes.

1. Security is only one of many critical stakeholder requirements of your system. Security needs to balance its needs with other stakeholder priorities.
2. You can only understand how much Security you can realistically plan to deliver to a system, by knowing rather specifically, about the levels of *all other* stakeholder priorities: Balance.
3. A Systems Architect is one name for the instance that co-ordinates and balances all competing stakeholder needs, including security.
4. An engineering approach is necessary, to model large and complex systems, and to find a good balance for Security. This includes quantifying all quality requirements and other variable stakeholder values, and limited resources: in the short term and for the system lifetime.
5. Systems and Security Engineering must include a means of both estimating, and measuring the multiple impacts on all critical stakeholder values (qualities and other values) and life-cycle system-resource consumption. (Hint see Gilb Impact Estimation Table, book Competitive Engineering 2005)
6. The safest proven approach (See IBM Cleanroom, Quinnan) for complex systems engineering will attempt to deliver small (2% of budget) incremental steps of the security design, measure actual security levels attained, change design when increments fail to deliver enough, and stop when target levels are delivered. This is Agile Security Engineering.
7. A security design can be absolutely anything, not violating stated system constraints, which gives the best security impact in the direction of our numeric security targets, with the least consumption of budgeted resources. 'Security Efficiency'. Anything means, any design, from any effective discipline, for the system.
8. Security requirements can state minimum levels (constraints, worst case), and more-valuable, more-desired levels (target levels). We should be able to explain the difference (Target Level - Constraint Level) in terms of consequential loss dimensions, such as costs, if we do not attain the target levels. These 2 levels help the security engineer and the systems architect determine current priorities as system development progresses. For example when targets are reached the security or other quality dimension loses all current priority.
9. Security engineers need to co-operatively recognize that security itself is ultimately dependent on many other qualities of the system also being attained, at high interesting levels, for example usability, safety, reliability, availability, work capacity, trustworthiness, adaptability, portability, maintainability, recoverability and many others.

10. Security engineering and maintenance of good security levels, is a *<u>never-ending lifetime battle</u>*; not a one-time up-front design effort.

So *persistent resources*, to monitor the security threats, and necessary security levels; must be a part of the lifetime operational costs, of any large and complex system.



Multiple Security Requirements derived from Ben Hanson, 2022, BCS Lecture, Microsoft Methods

Building a model of Microsoft Security methods, using Planguage and ValPlan

Ben Hanson's Core Security Principles
https://thebenhanson.com/2019/11/22/ccp/

VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

* This is my exercise in reading the detailed slides from Ben Hanson, presenting a Microsoft security set of ideas.
* One point is that any company can organise and relate their Security efforts in this 'systems engineering' pattern.
  * They can then detail and update specs like objectives, and ratings of security strategies as they mature, improve and learn more.
  * Sort of a 'digital twin' of your security effort

Ben Hanson, Microsoft
Cybersecurity & transformation thought leader. Believer in people. Leader in Microsoft's global cybersecurity community.
https://www.linkedin.com/in/the-ben-hanson/
SecurIT: Churchill, Cloud Security, and You
Monday, 20 June 2022 from 18:30 to 20:00 (BST)
BCS, The Chartered Institute for IT
https://thebenhanson.com/2019/11/22/ccp/

ValPlan Info
gilb.com/valplan

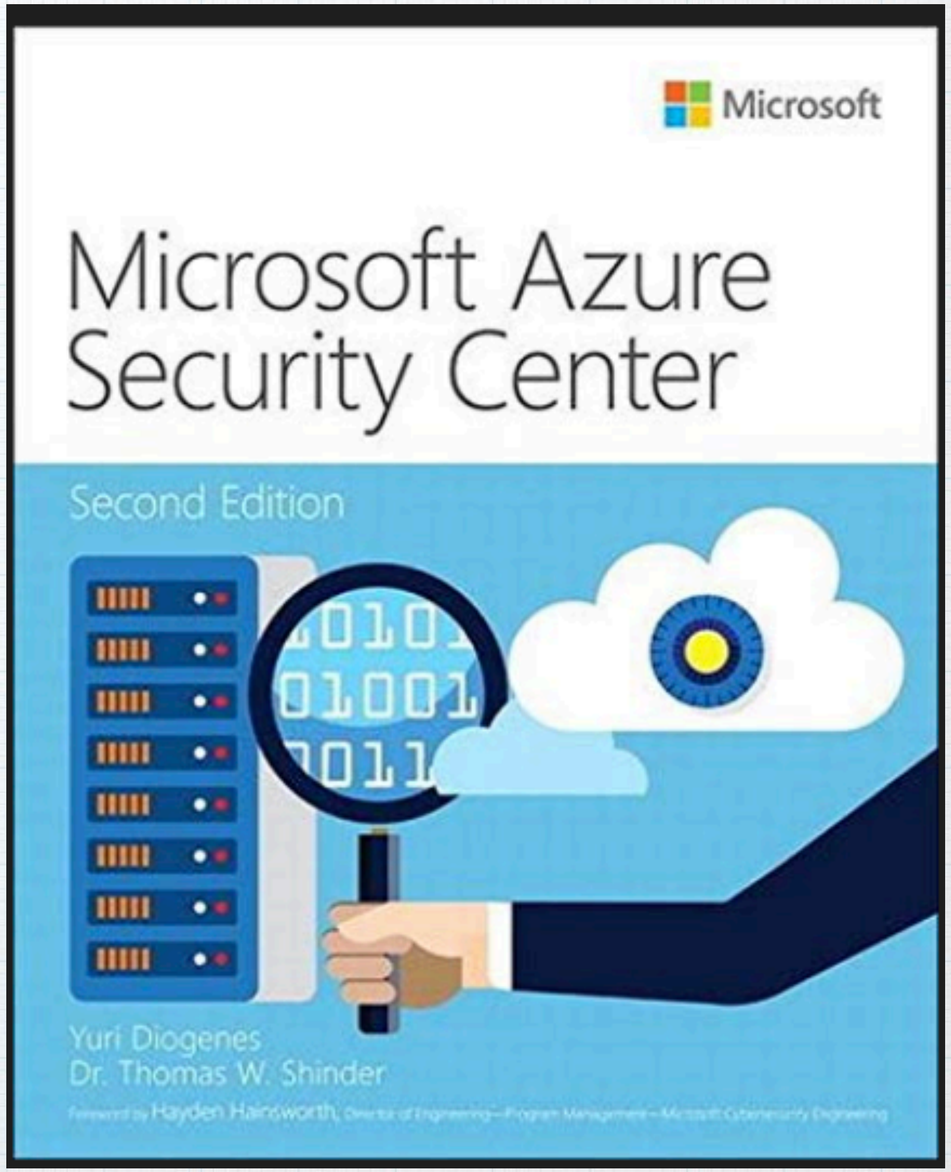
Why did I make this?

The presentation and slides was a series of ideas with 'names' connected to other 'names' of ideas.

But I could not understand the effectiveness, the cost-effectiveness or the competitiveness of the ideas.

So I studied the slides and 'names' and diagrams in detail and tried to organise the ideas into a Security Model.
This sets the stage for trying (or failing) to see if I or anybody (Hanson, Microsoft, Security Professors) can give us some information about the cost-effectiveness.

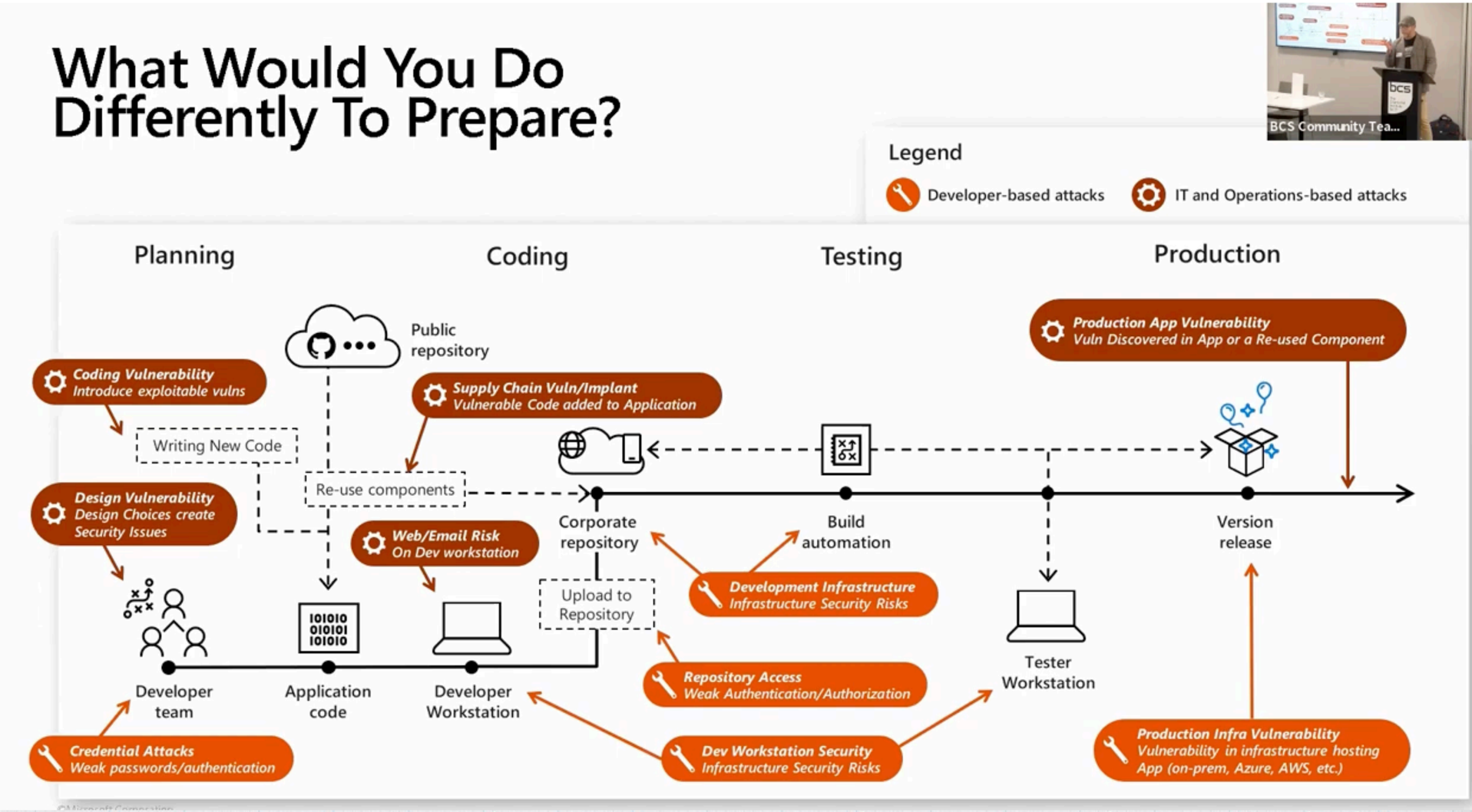
So, that, we have a fair chance to evaluate, Microsoft Security Ideas, and all other competitive Security ideas, in an objective way - as opposed to just believing that this stuff is 'good'.



BEN HANSONS SLIDE EXAMPLE

Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)

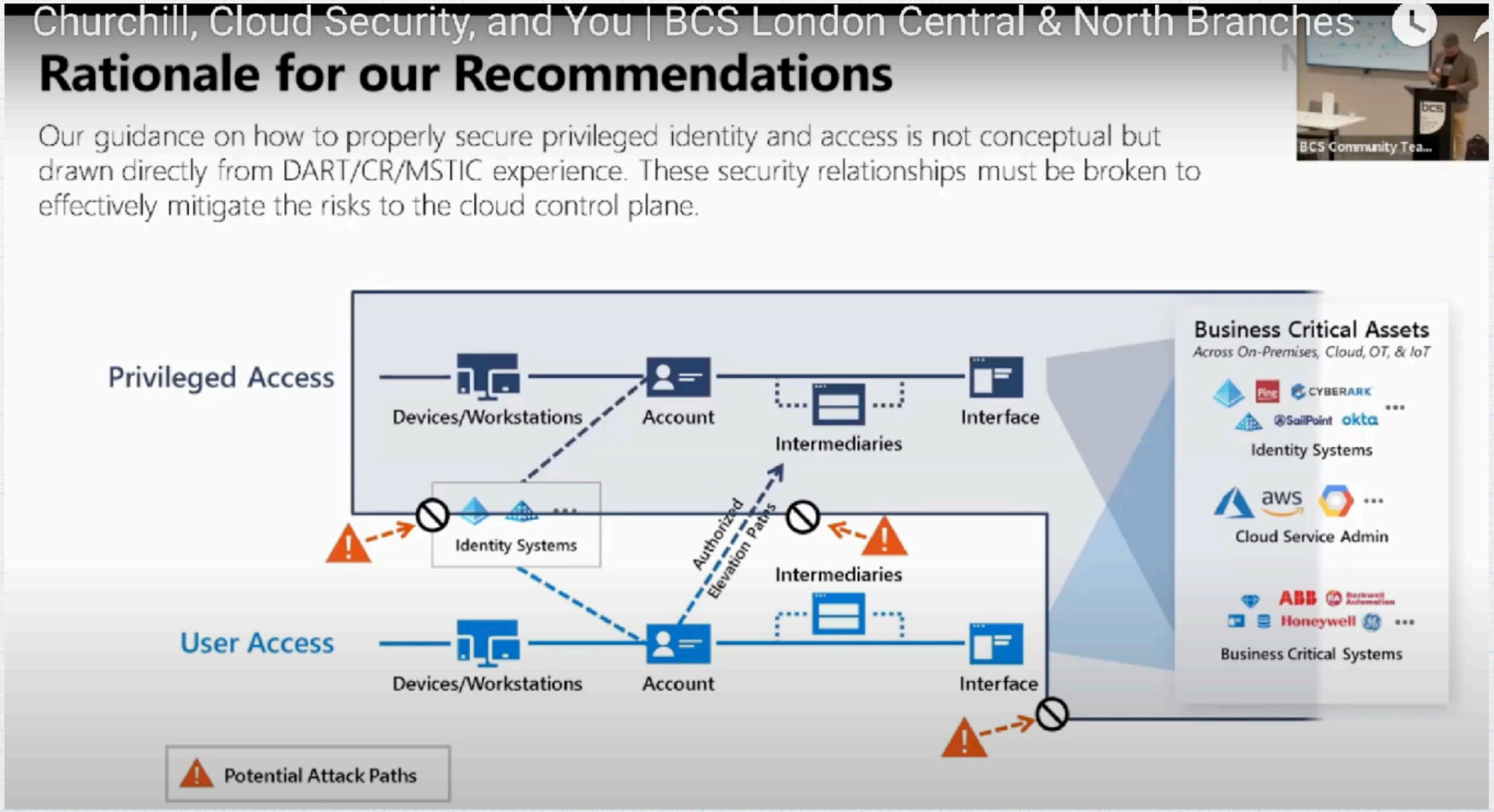


VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

BEN HANSONS SLIDE EXAMPLE

Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)

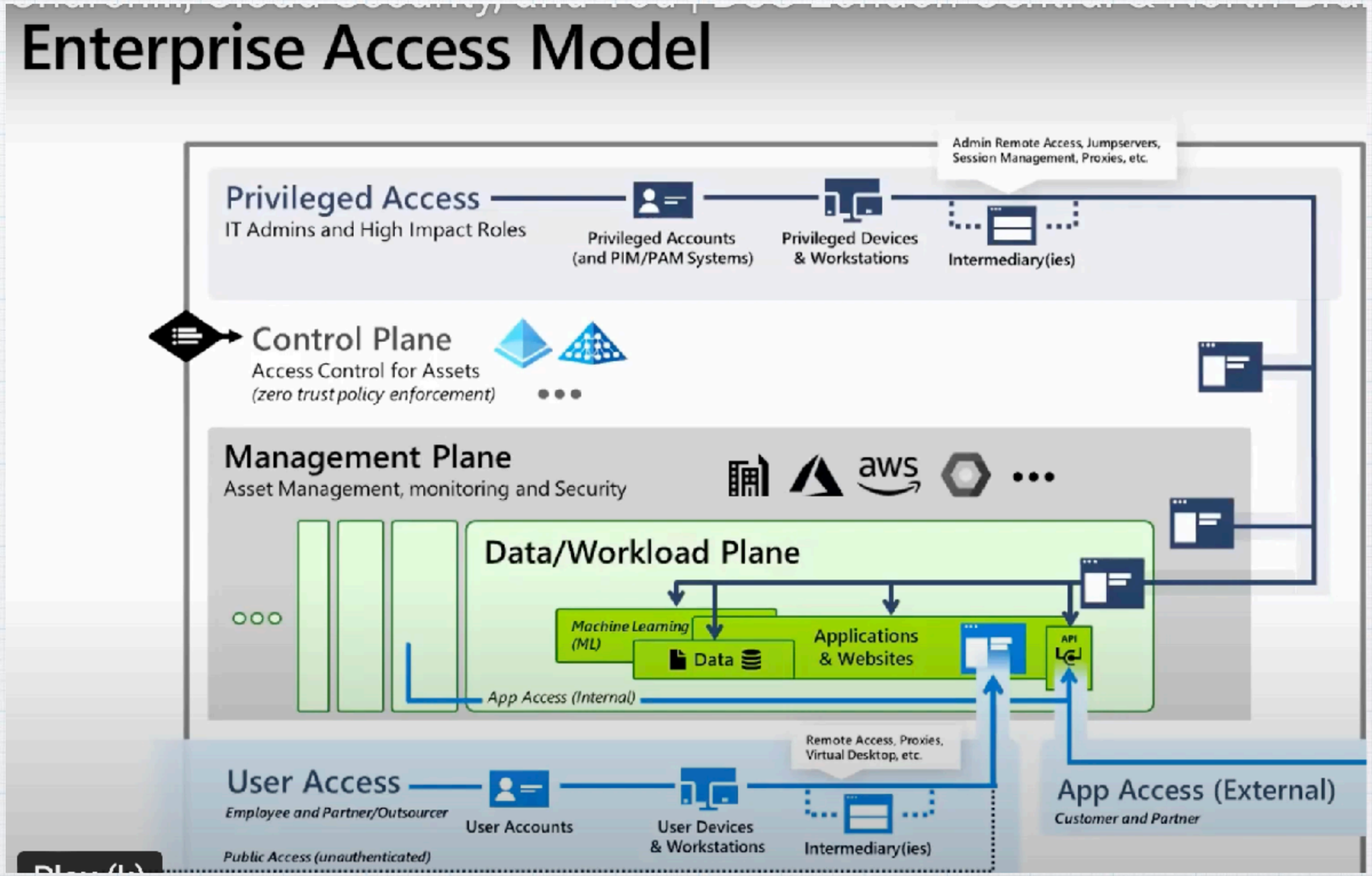


VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

BEN HANSONS SLIDE EXAMPLE

Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)

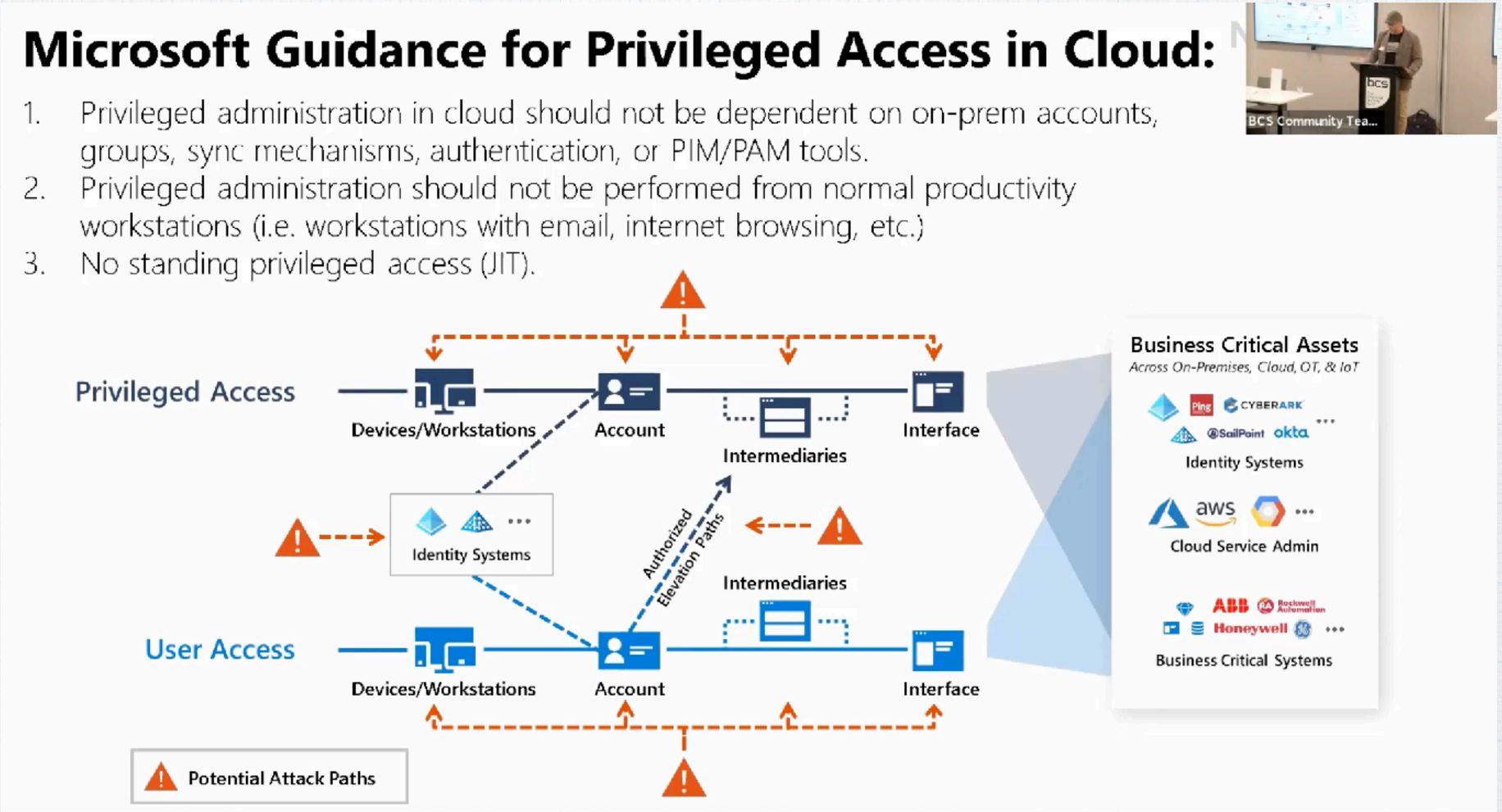


VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

BEN HANSONS SLIDE EXAMPLE

Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)



VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

BEN HANSONS SLIDE EXAMPLE: "revolutionise... effectiveness"

Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)

Closing Thoughts

Collision of Cultures ①

- You can't become a butterfly by trying to be a better caterpillar
- Have you established a mandate for change? Can you drive change through your organization? If not, what are you going to do about it?
- What about you personally? Are you learned or a learner?

Form and Function ②

- Structure impacts effectiveness more than we like to admit
- What are the downstream implications of complexity?
- Have you defined and empowered modern security roles and functions?

When, Not If ③

- Adopt "Assume Breach" and embed it at all levels
- It will revolutionize your approach to security, and your effectiveness
- It is the principal enabler for Zero Trust

Not This Again! ④

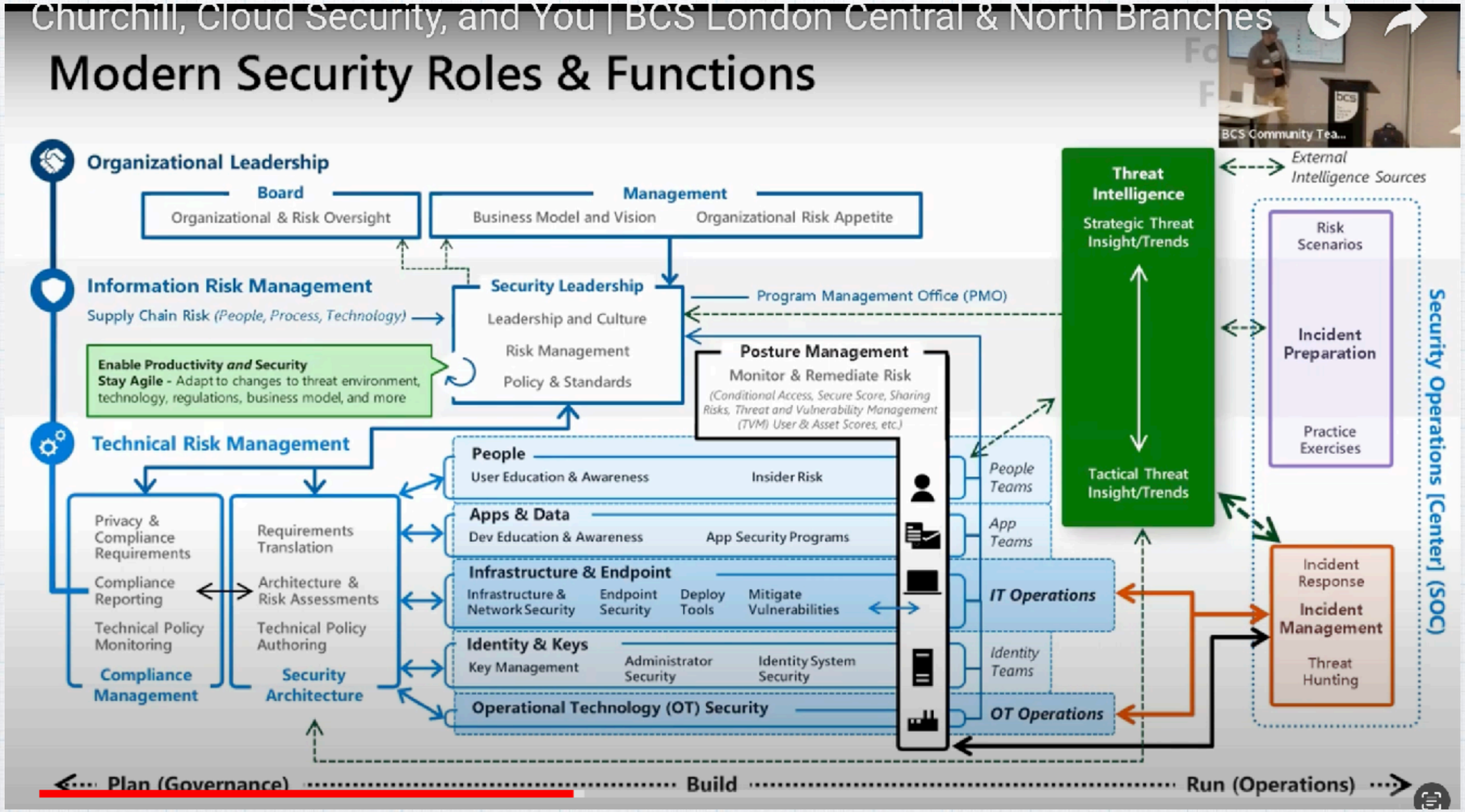
- You must break the attack chain used to pivot on-prem → cloud
- Tools used to manage privileged functions in cloud are *supposed* to be different
- For privileged functions, risk > operational overhead

VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

BEN HANSONS SLIDE EXAMPLE

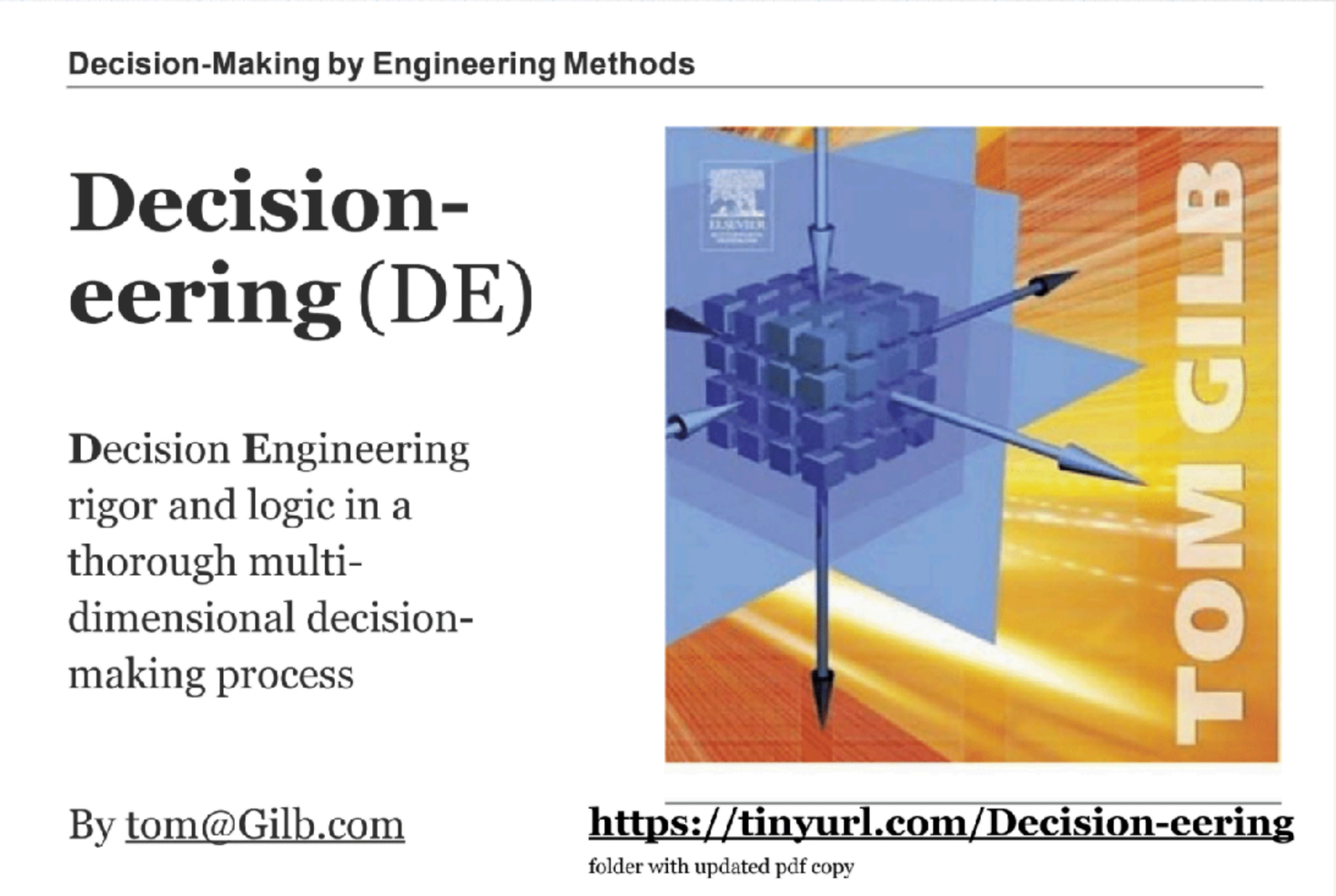
Is there any information (answer NO, nowhere. A Known unknown)
on costs, or on
these technique's effects - on all critical security qualities, and *other* quality side effects (example Usability, Maintainability)

If we do not have <u>facts</u> about the impacts of each technique - on our *many* security quality requirements, our many side-effect requirements, and all critical costs, then we do not have a logical engineering basis for adopting these ideas. (Except blind faith in the supplier.)



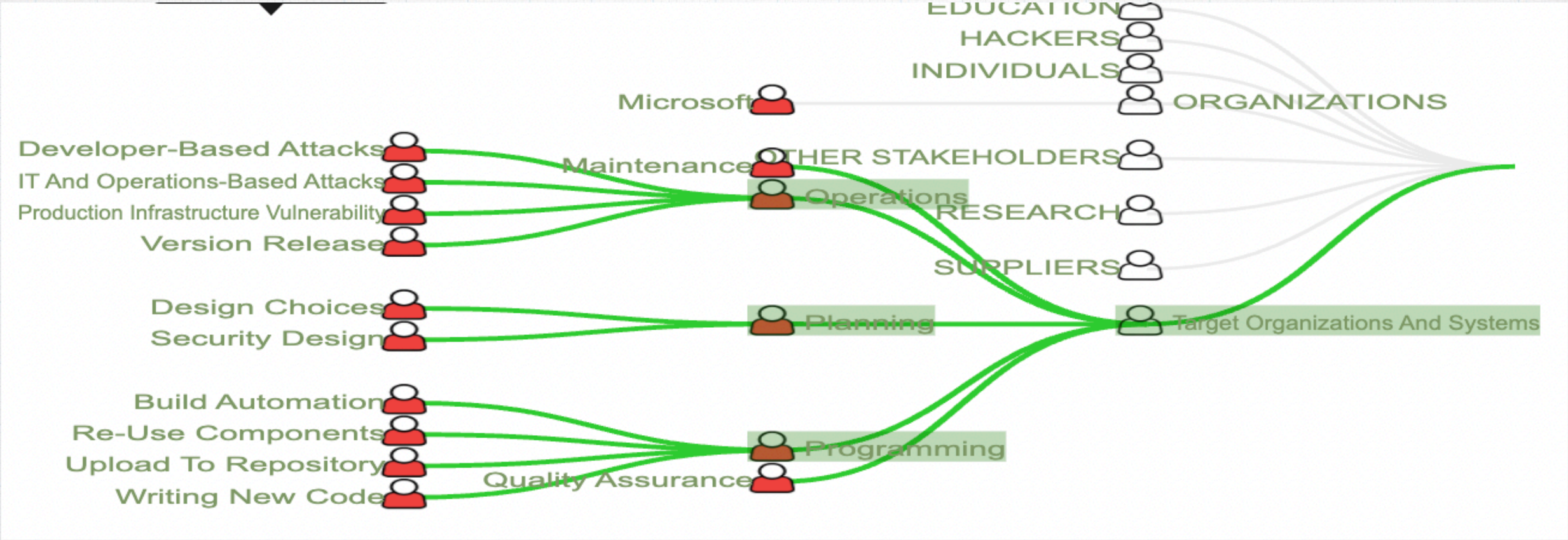
VIDEO: https://www.youtube.com/watch?v=sLiaub4boWI

* There is no time in this lecture to study and explain the detail in each slide in the following.
* You can do that after the lecture if you want.
* Relax and note my main message
  * THERE IS A SYSTEMATIC quantified 'engineering' WAY (PLANGUAGE) OF ORGANIZING SECURITY KNOWLEDGE
    * SO THAT WE CAN MAKE BETTER DECISIONS ON IT



Tom Gilb, Decision-eering. https://tinyurl.com/Decision-eering, booklet, pdf, 2022-3

MS Security Stakeholder Map



MS Security: Stakeholders, Functions, Solutions

2.Stakeholder Level

Create...

- Stakeholders
  - EDUCATION
  - HACKERS
  - INDIVIDUALS
    - Ben Hanson Trusted Security …
  - ORGANIZATIONS
    - Microsoft
      - CR
      - DART
      - Microsoft Threat Intelligen…
      - **Microsoft Trusted Secu**
    - MITRE
    - UK NCSC
  - OTHER STAKEHOLDERS
  - RESEARCH
  - Security Models
  - SUPPLIERS

- Functions
  - Control Plane
    - High Impact Roles
    - IT Admins
  - Management Plane
    - Data/Workload Plane
      - App Access External
      - Applications
      - Data
      - ML
      - Websites
  - Privileged Access
  - User Access
    - Employee Access
    - Intermediaries
      - Proxies
      - Remote Access

3.Product Level

4.Solution Level

Create...

- Solutions
  - Microsoft Guidance For Privilege…
    - Adopt "Assume Breach" And …
    - Break The Attack Chain Used …
    - Complexity Solutions For Do…
    - Define And Empower Modern …
    - Mandate For Change
  - Rationale For MS Recommendati…

MS Security: Top Level Values, Stakeholders, Functions

Contains 62 specifications

1.Business Level Create...

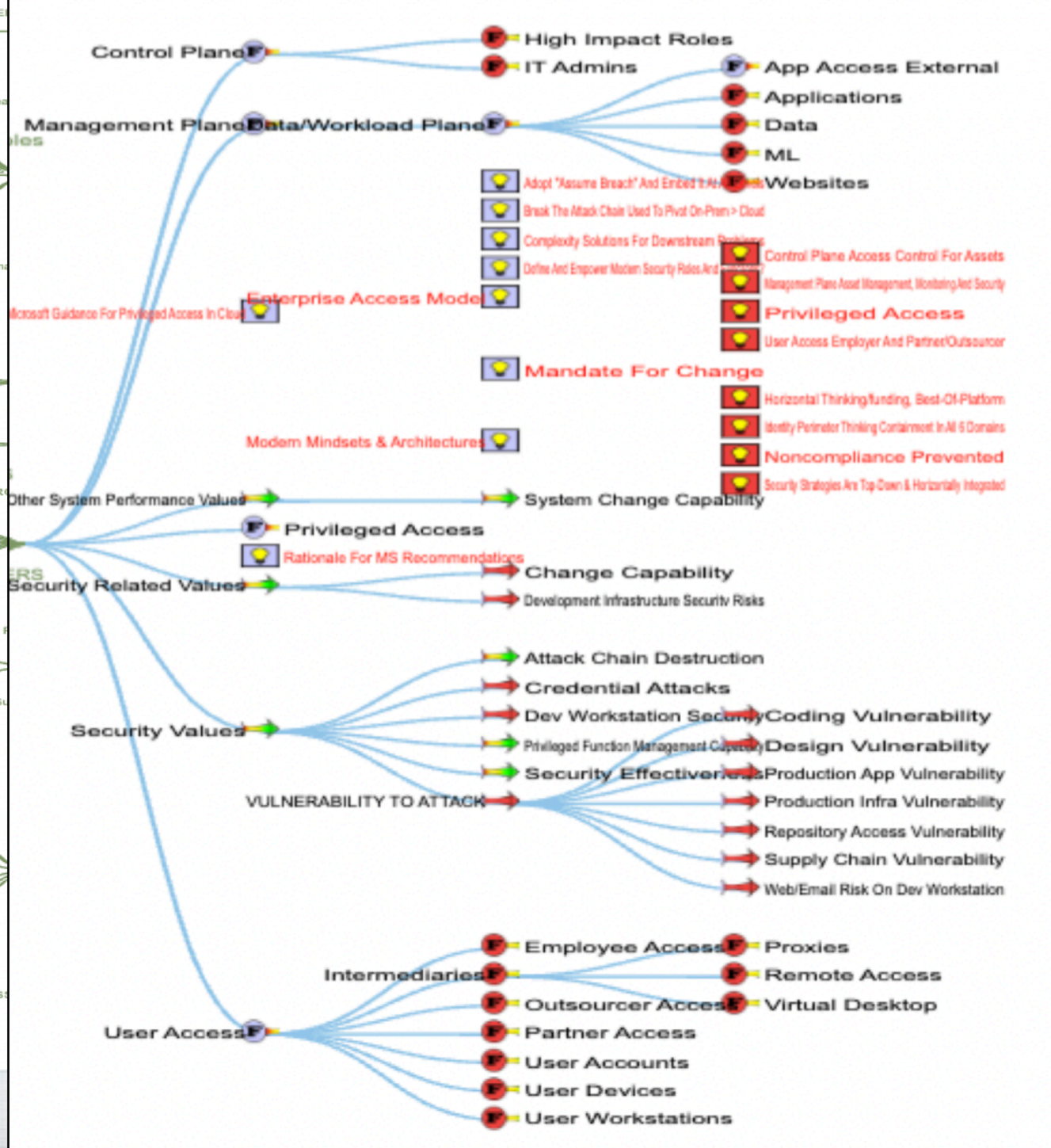
- Stakeholders
- Values and Resources
  - Other System Performance Values
    - Effectivness
  - Security Related Values
    - Change Capability
  - Security Values

2.Stakeholder Level Create...

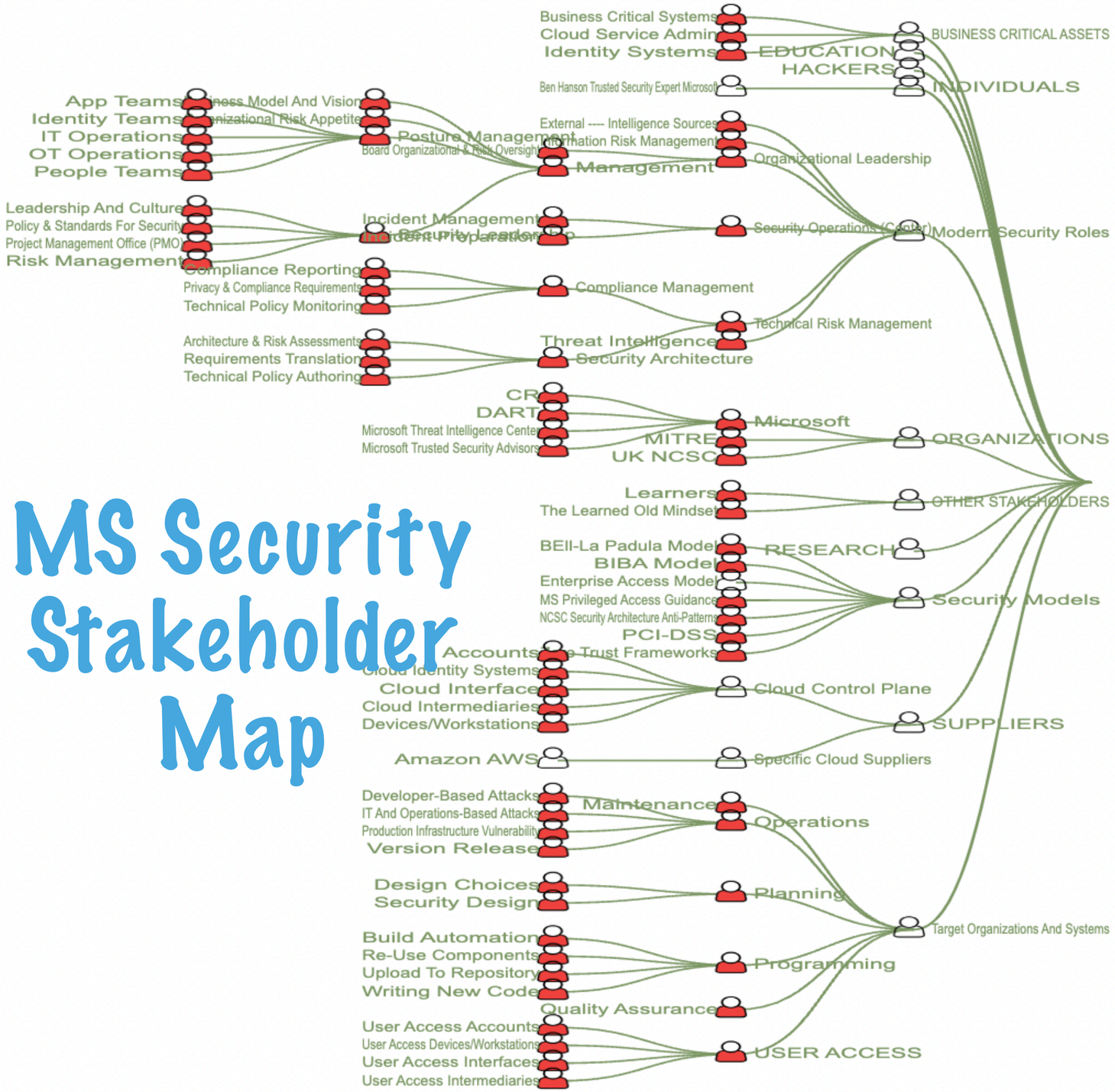
- Stakeholders
  - EDUCATION
  - HACKERS
  - INDIVIDUALS
    - Ben Hanson Trusted Security …
  - ORGANIZATIONS
    - Microsoft
      - CR
      - DART
      - Microsoft Threat Intelligen…
      - Microsoft Trusted Secu
    - MITRE
    - UK NCSC
  - OTHER STAKEHOLDERS
  - RESEARCH
  - Security Models
- Functions
  - Control Plane
    - High Impact Roles
    - IT Admins
  - Management Plane
    - Data/Workload Plane
      - App Access External
      - Applications
      - Data
      - ML
      - Websites
  - Privileged Access
  - User Access
    - Employee Access
    - Intermediaries
      - Proxies

MS Security Diagram

Security Stakeholders



MS Security Stakeholder Map

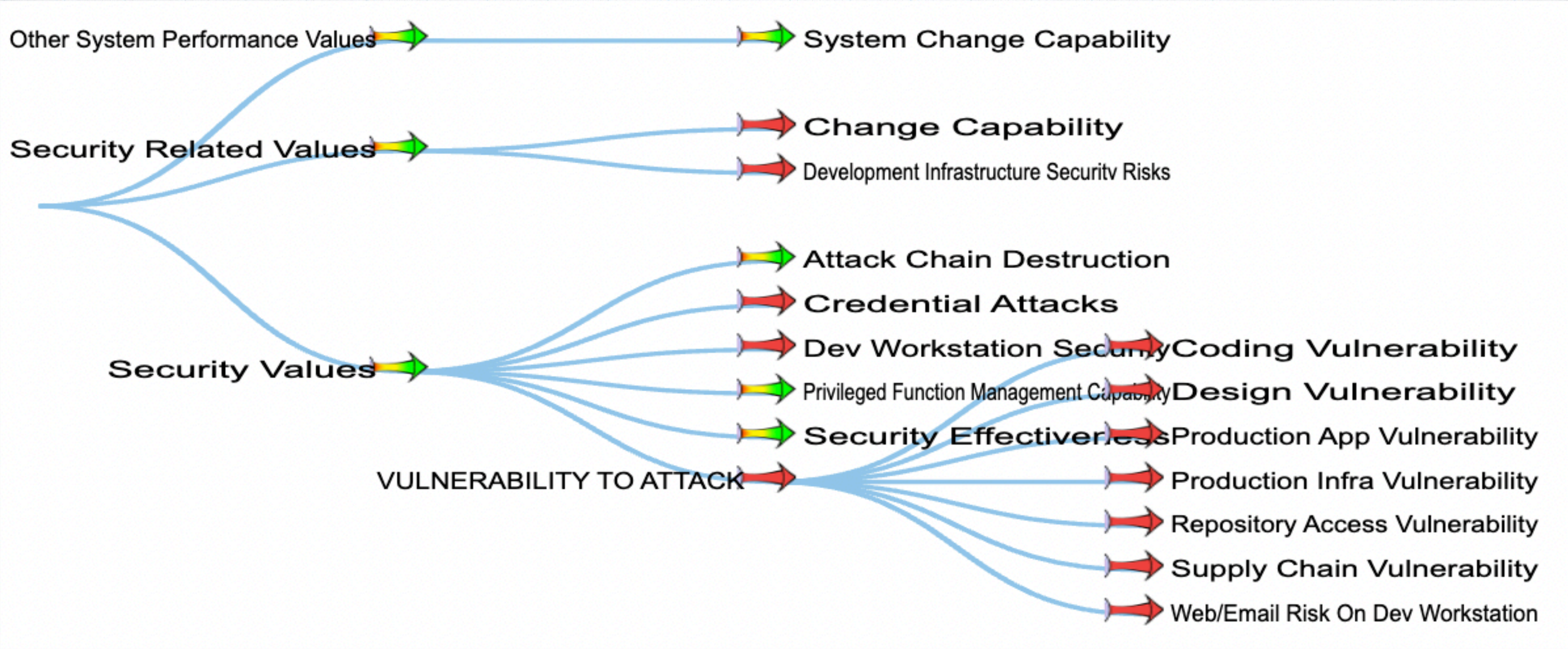


Security Stakeholders By definition

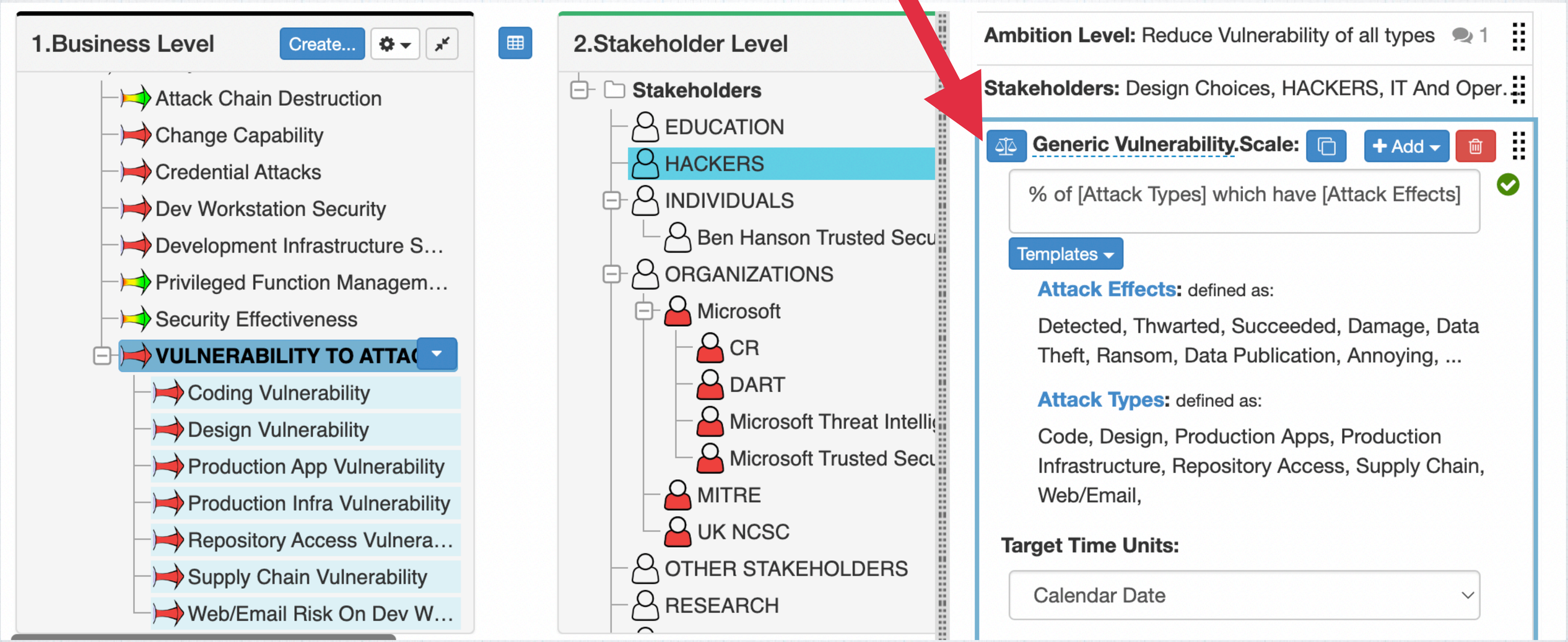
Have 'Requirements' for your security system

Stakeholder Engineering. By Tom Gilb Leanpub.com/StakeholderEngineering Released 27 July 2021, Leanpub, 177 pages.

MS Security: The hierarchy of Security Attributes (requirements)



MS Security: Vulnerability Scale of measure by TG

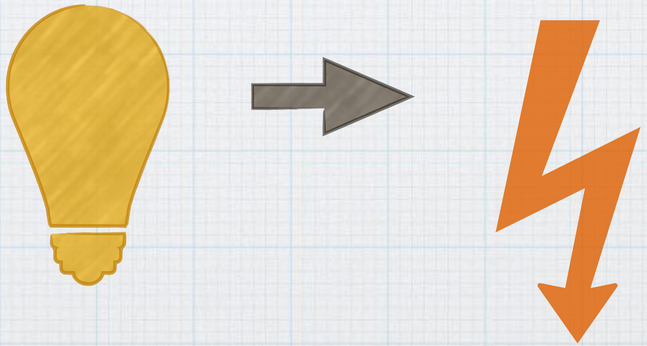


And now we can look for known and unknown effectiveness and costs.

and for

Relationships

between means and ends



Estimating the impact of 3 security techniques, on 3 value requirements, and 1 cost concept.

ValPlan Info
gilb.com/valplan

Here is the detail of the selected estimate

Sample MS Security Table

From Level: Level? To Level: Level?

Settings... | Add | Help me!

Requirements	Adopt "Assum...	Break The Attack ...	Control Plane Acc...
Attack Chain Destruction Status: 0 → Wish: 95 %AttOK	50 53 %	-30 -32 %	???? ????
Privileged Function Managem... Status: 95 → Wish: 100 % Correct ...	20 400 %	10 200 %	???? ????
Security Cost-Effectiveness Status: 3 → Wish: 17 ROI %	42 300 %	3 21 %	???? ????
Sum Of Values:	753 %	189 %	0 %
Future Tech Debt Status: 0 → Budget: 100 Financial ...	42 42 %	5 5 %	???? ????
Sum Of Development Resources:	42 %	5 %	0 %
Value To Cost:	17.90	37.80	Infinity
Ratio (Worst Case)	17.40	37.80	Infinity
Ratio (Cred. - adjusted)	0.30	0.00	0.00
Ratio (Worst Case Cred. - adjusted)	12.80	0.00	0.00

Comments:

Add Comment...

+Specification

−Impact Cell

Tag.Value Impact:

%AttOK

Δ 50 ± 20

Actual: %AttOK

Δ scale value ± 0

Credibility: 0.4

Several measurements relevant for the use case (e.g. qualifier values)

Evidence:

academic studies USA 2023

Source: by tomgilb - Jan 30th 2023, 21:58

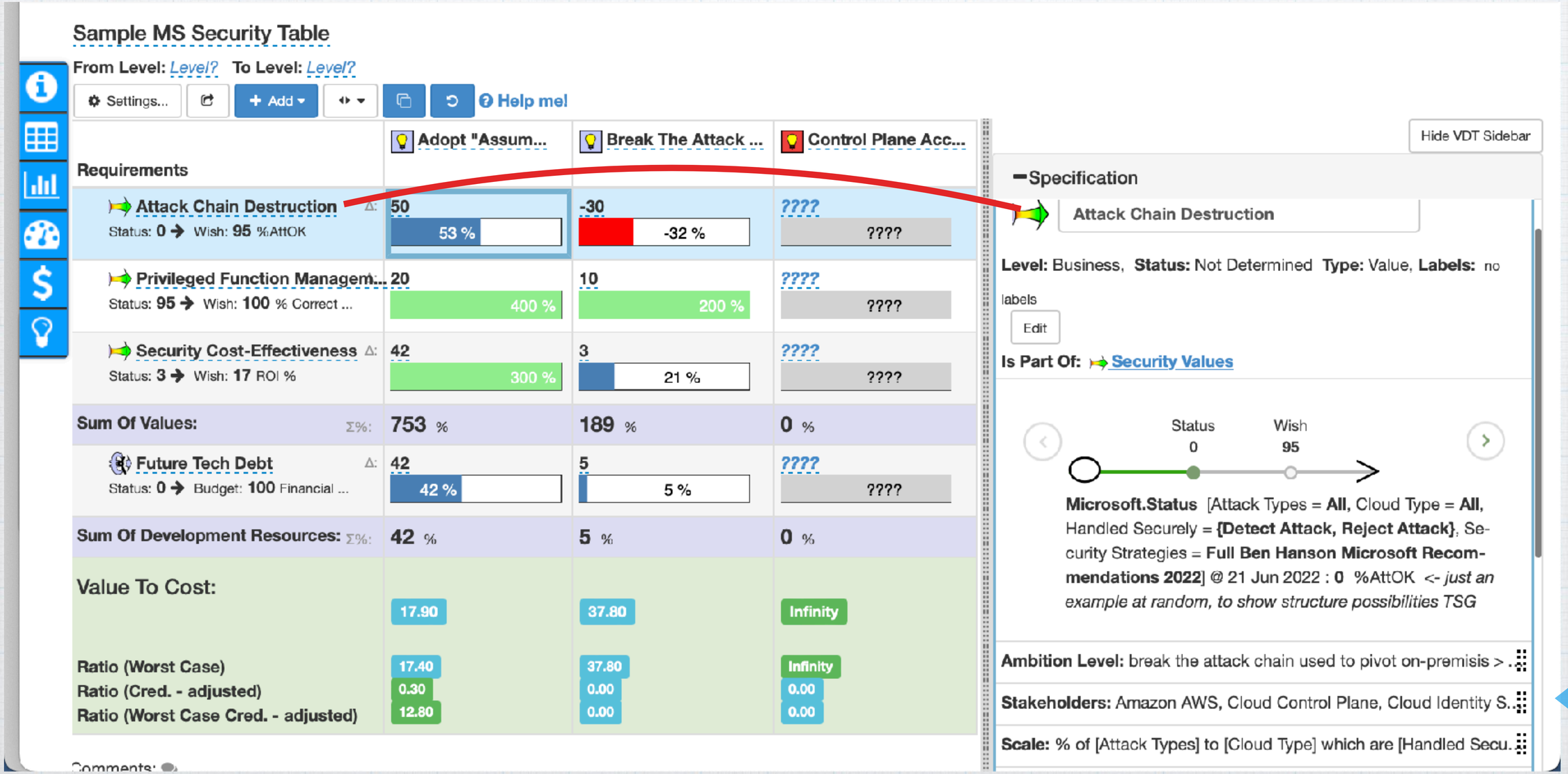
tom

⌘+e for formatting, to add qualifiers / terms / specs

Templates

3 Techniques impact on 3 Values and 1 cost

On the right is detail, for 1 of the quantified security requirements



A numeric relationship table for MS Security ideas.
On the right a click on the "Break,,," tag in the table, gives the detail of the security technique being evaluated.
The '????' Indicates a 'known unknown' relation. Ask an expert!

Sample MS Security Table

From Level: *Level?* To Level: *Level?*

Settings... | + Add | Help me!

Requirements	Adopt "Assum...	Break The Attack ...	Control Plane Acc...
Attack Chain Destruction Status: 0 → Wish: 95 %AttOK	50 53 %	-30 -32 %	???? ????
Privileged Function Managem... Status: 95 → Wish: 100 % Correct ...	20 400 %	10 200 %	???? ????
Security Cost-Effectiveness Status: 3 → Wish: 17 ROI %	42 300 %	3 21 %	???? ????
Sum Of Values: Σ%:	753 %	189 %	0 %
Future Tech Debt Status: 0 → Budget: 100 Financial ...	42 42 %	5 5 %	???? ????
Sum Of Development Resources: Σ%:	42 %	5 %	0 %
Value To Cost:	17.90	37.80	Infinity
Ratio (Worst Case)	17.40	37.80	Infinity
Ratio (Cred. - adjusted)	0.30	0.00	0.00
Ratio (Worst Case Cred. - adjusted)	12.80	0.00	0.00

Hide VDT Sidebar

Specification

Authoring

Break The Attack Chain Used To Pivot On

Level: Solution, **Status:** Not Determined **Type:** Solution Idea, **Labels:**

no labels

Edit

Is Part Of: Microsoft Guidance For Privileged Access In Cloud:

Summary: Tools used to manage privileged functions in cloud are ...

Tag.**Description:** + Add

1. Break chain from Account to Cloud identity systems to cloud account

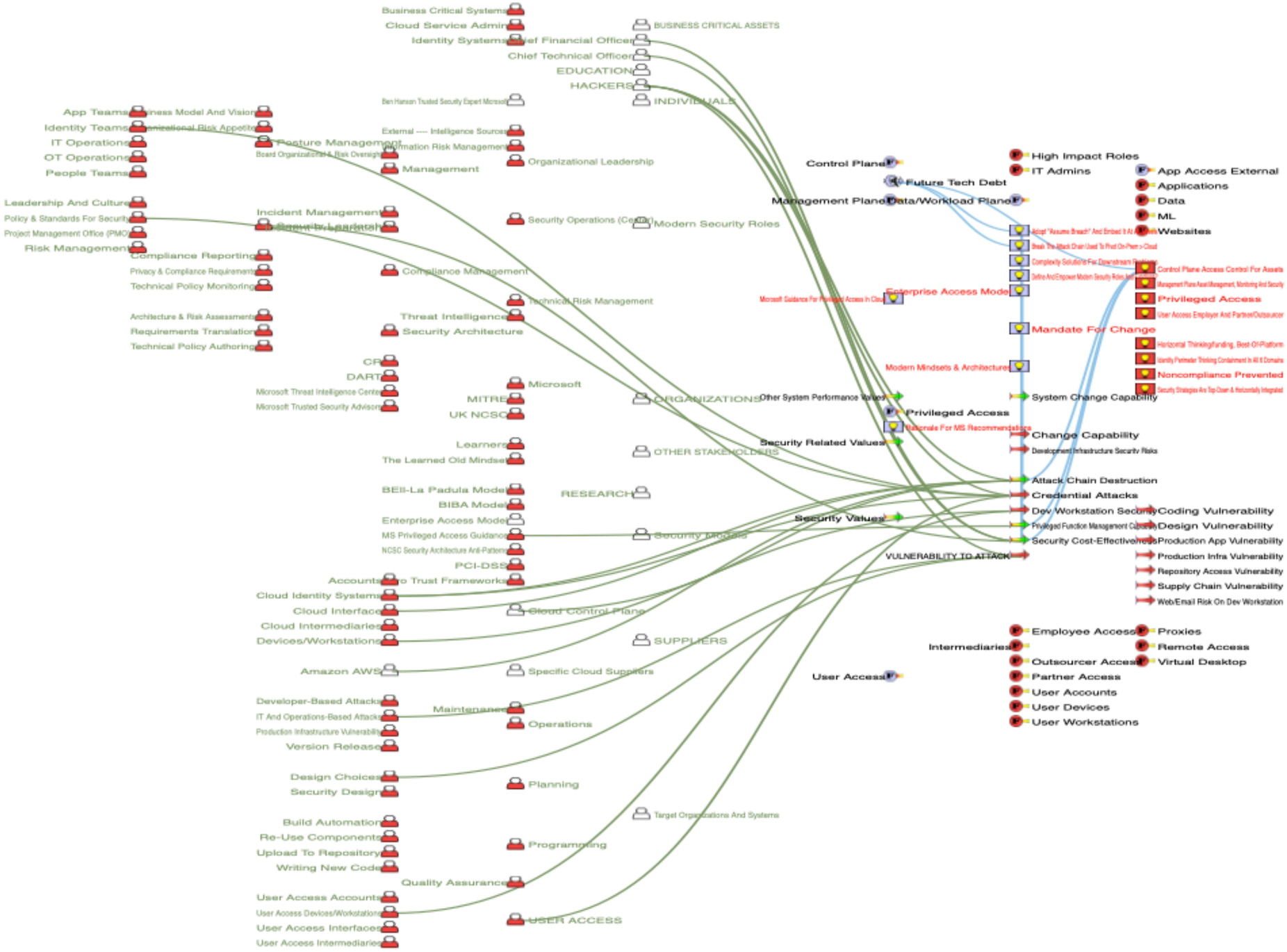
Break chain of intermediaries with user access into the cloud (Authorized Elevation Paths).

3. There is a danger triangle and stop sign at intersection of user interface and cloud. BUT I DO NOT KNOW WHAT IT MEANS. **SPEC DEFECT HERE.**

Automatic relationship links based on Table structure

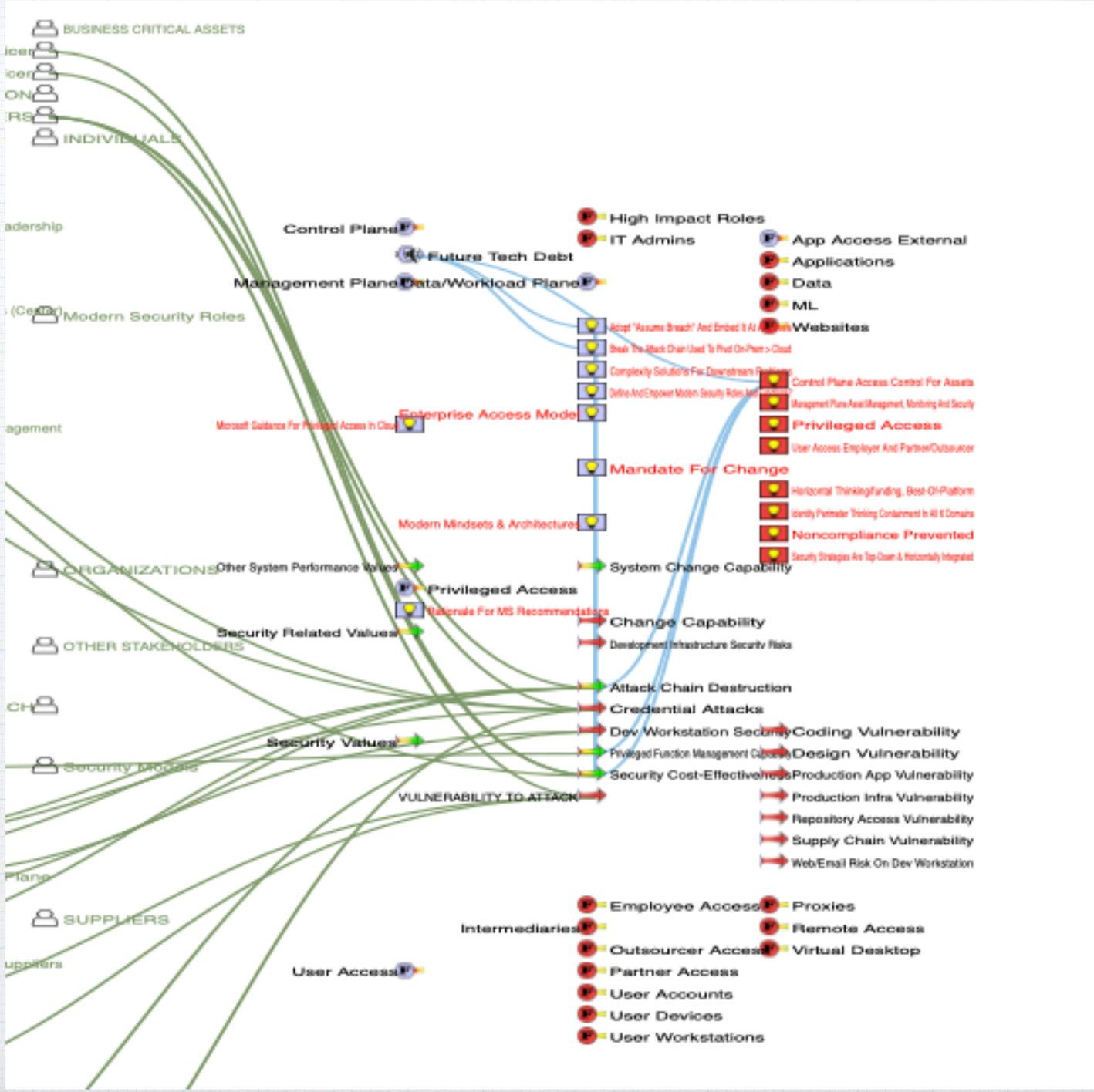
Seeing the big picture With Selected detail

Automatically updated



Automatic relationship links based on Table structure

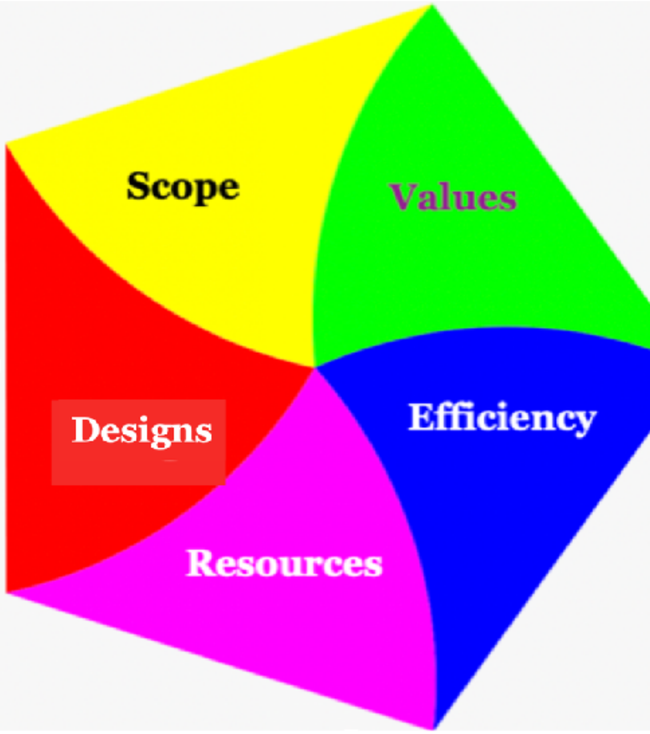
This is
* Planguage, a systems engineering language with well defined concepts
&
* Val Plan Digital app to present security model data in various useful views



Did you realize
(or begin to suspect)

that we can <u>build *digital* models</u>

of our security options and practices?



The Penta Model
(See earlier slides)

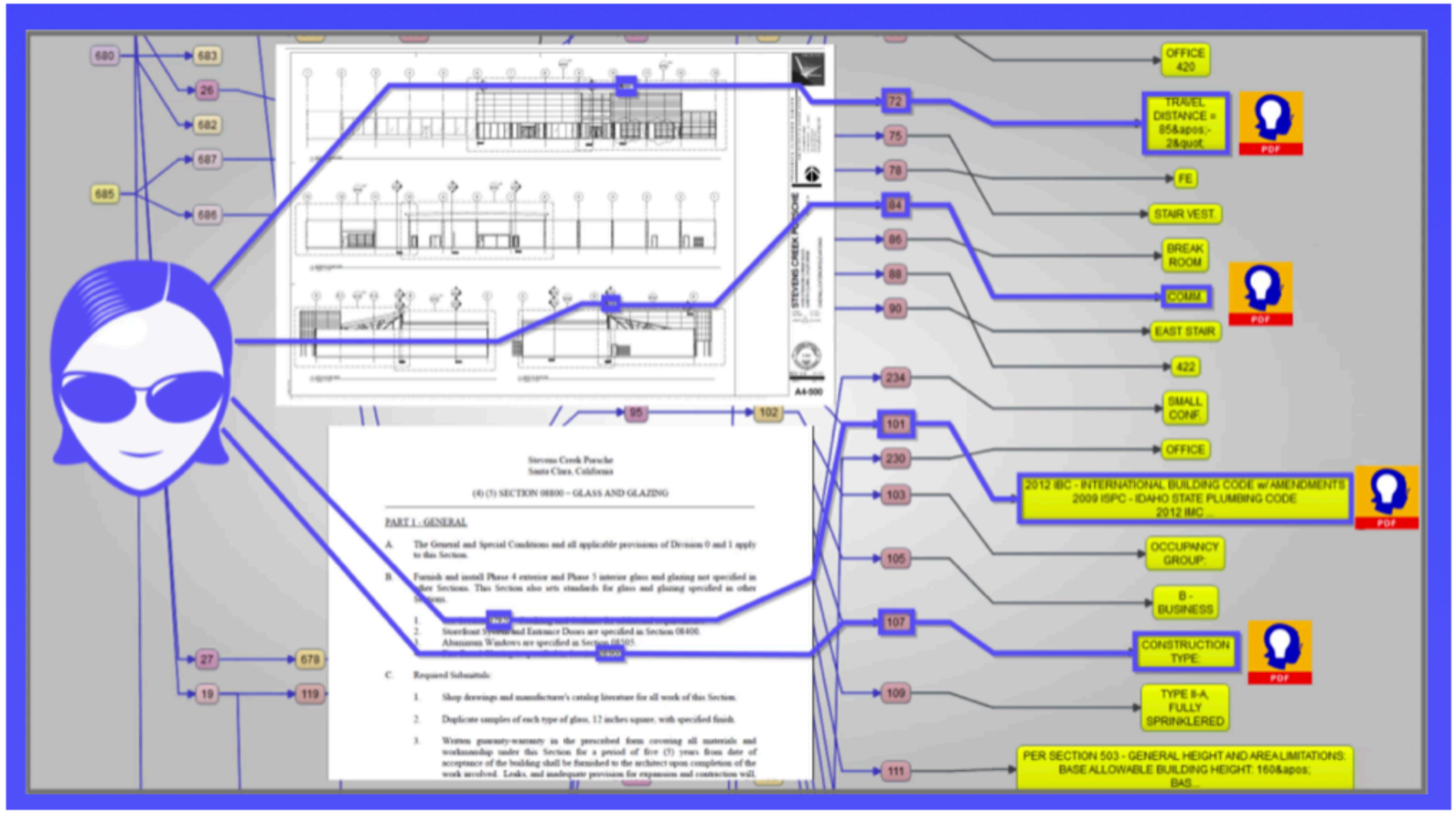
Can you build a model of *your* security practices, and keep it up to date, and quantify the relationships (the cost effectiveness) ?

A proposal for DNV

Cyberneering

The key to Cyber-Security Knowledge Management and engineering. From tom@Gilb.com, V=280922

Next Generation AI Search in Complex High-volume engineering documentation
(For example everything about Cyber Security)



Source: GraphMetrix.com

Cyberneering Folder

Word Format

https://www.dropbox.com/sh/klcj0rpm8vdgpda/AADkf7uPrE_hPXUaYGQsNs5Aa?dl=0

A Word copy of Cyberneering, Including references.

Link tested 310123

Summary: Security Engineering

* If you are serious about security, it **must be engineered quantitatively**
  * As one part of your system's engineering
* If you do not understand this, you are the **first threat** to your own system Security

Ambition Level: Reduce Vulnerability of all types 1

Stakeholders: Design Choices, HACKERS, IT And Oper..

Generic Vulnerability.Scale: + Add

% of [Attack Types] which have [Attack Effects]

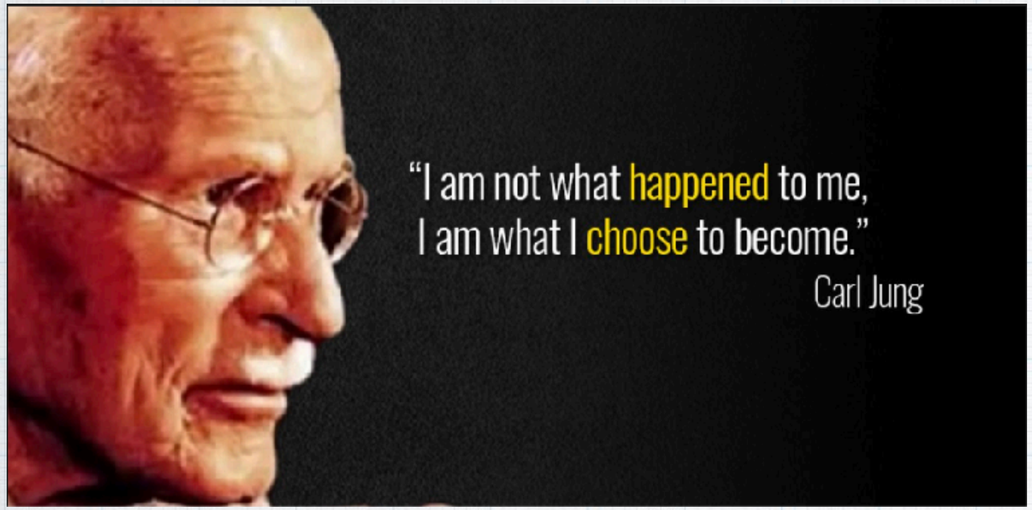
Templates

Attack Effects: defined as:

Detected, Thwarted, Succeeded, Damage, Data Theft, Ransom, Data Publication, Annoying, ...

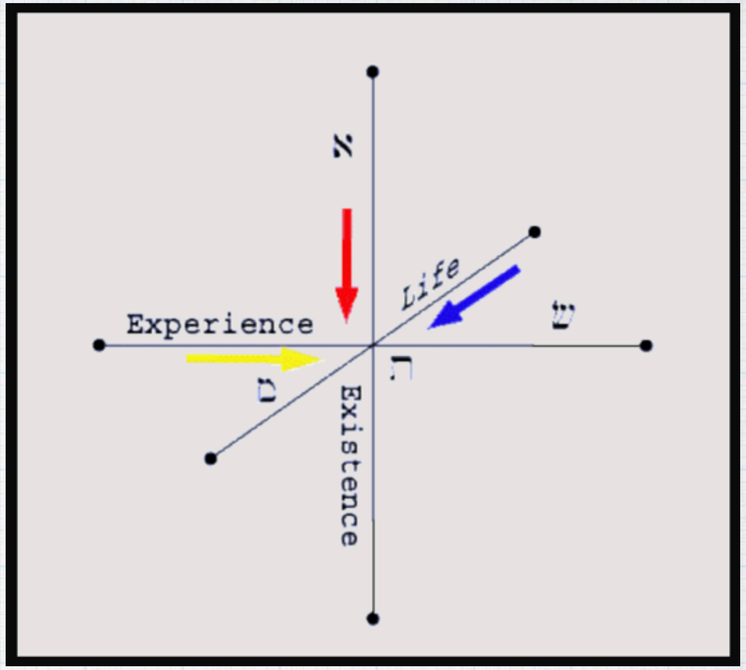
Attack Types: defined as:

Code, Design, Production Apps, Production Infrastructure, Repository Access, Supply Chain, Web/Email,



End Slide

Slides Folder



Carl Jung:

(born July 26, 1875, Kesswil, Switzerland—died June 6, 1961, Küsnacht), Swiss psychologist and psychiatrist who founded analytic psychology,

"Number helps more than anything else to bring order into chaos of appearances. It is the predestined instrument for creating order, or for apprehending an already existing, but still unknown, regular arrangement or 'orderedness'".

"Number" [sic]. that is how he used the term
https://aras.org/concordance/content/number

Cyberneering References are in the
https://www.dropbox.com/sh/klcj0rpm8vdgpda/AADkf7uPrE_hPXUaYGQsNs5Aa?dl=0

Cyberneering Folder

Very Last Slide

Quantifying Sw Security
Slides Folder