RSAConference2022

San Francisco & Digital | June 6 – 9

TRANSFORM

SESSION ID: **STR-T02**

Electrons to Clouds--One SDL to Rule the World from Hardware to the Cloud

Mohit Arora

Software Security Architect
Dell
@NonceNinja

Richard Tonry

Firmware Security Architect
Dell
@RMTonry

Disclaimer

Presentations are intended for educational purposes only and do not replace independent professional judgment. Statements of fact and opinions expressed are those of the presenters individually and, unless expressly stated to the contrary, are not the opinion or position of RSA® Conference, RSA Security LLC or any other co-sponsors. RSA Conference does not endorse or approve, and assumes no responsibility for, the content, accuracy or completeness of the information presented.

Attendees should note that sessions may be audio- or video-recorded and may be published in various media, including print, audio and video formats without further notice. The presentation template and any media capture are subject to copyright protection.

RSAConference2022

Rhetorical Question

RSAConference2022

Rhetorical Question

Does one SDL rule the world?

RSAConference2022

Rhetorical Question

Does one SDL rule the world?

What is this nonsense about Electrons to Clouds?

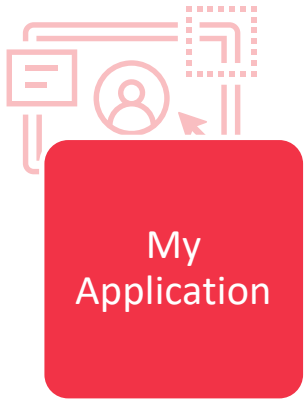
Objectives

Describe Full-Stack SDL

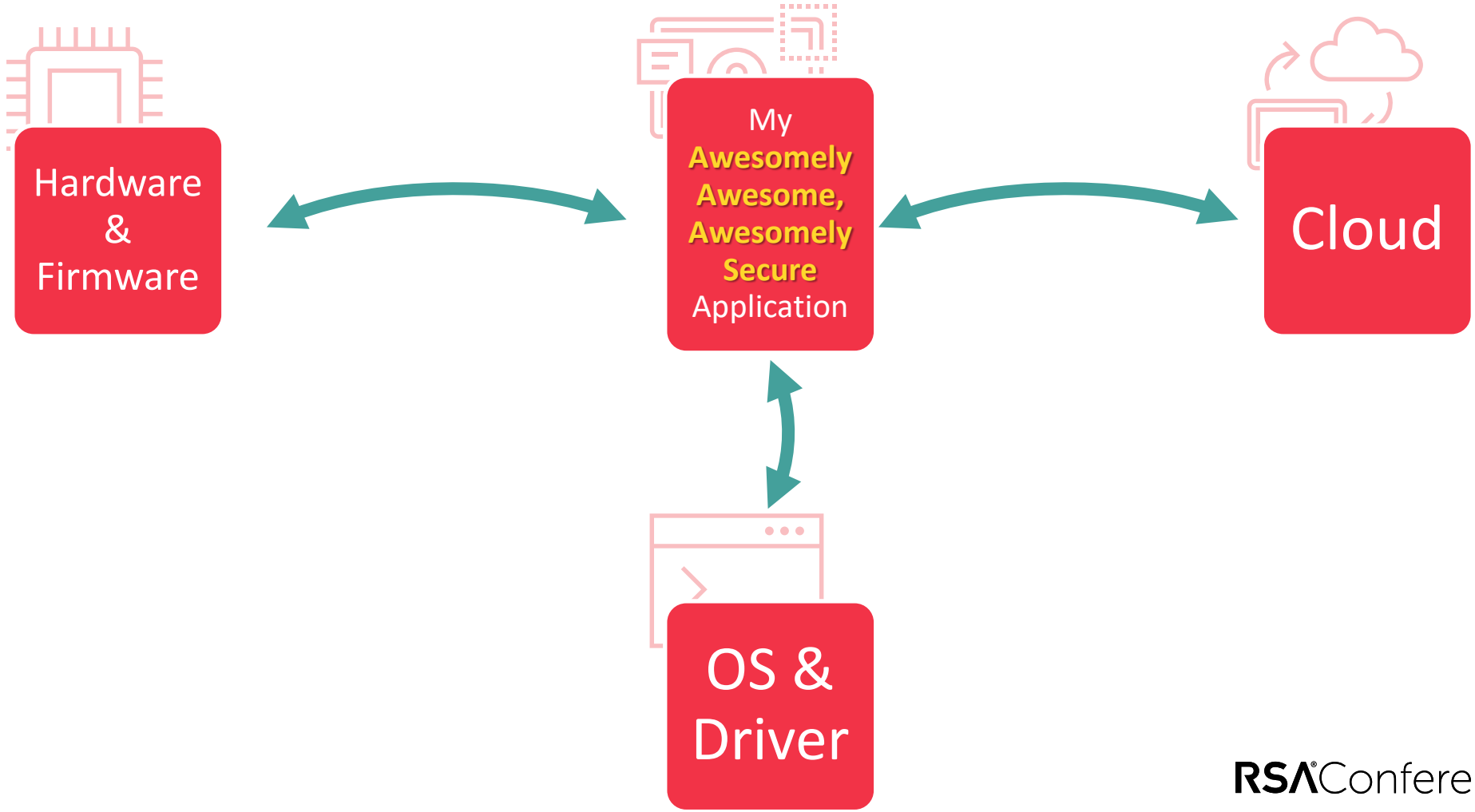
Review an Approach

Discuss What's Next

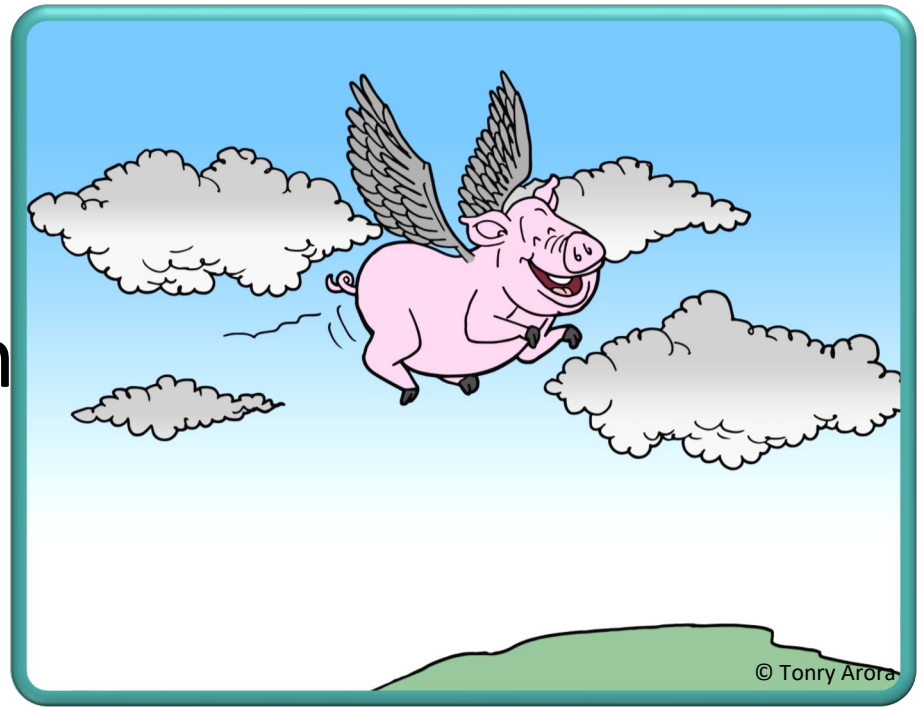
Does one SDL rule the world?



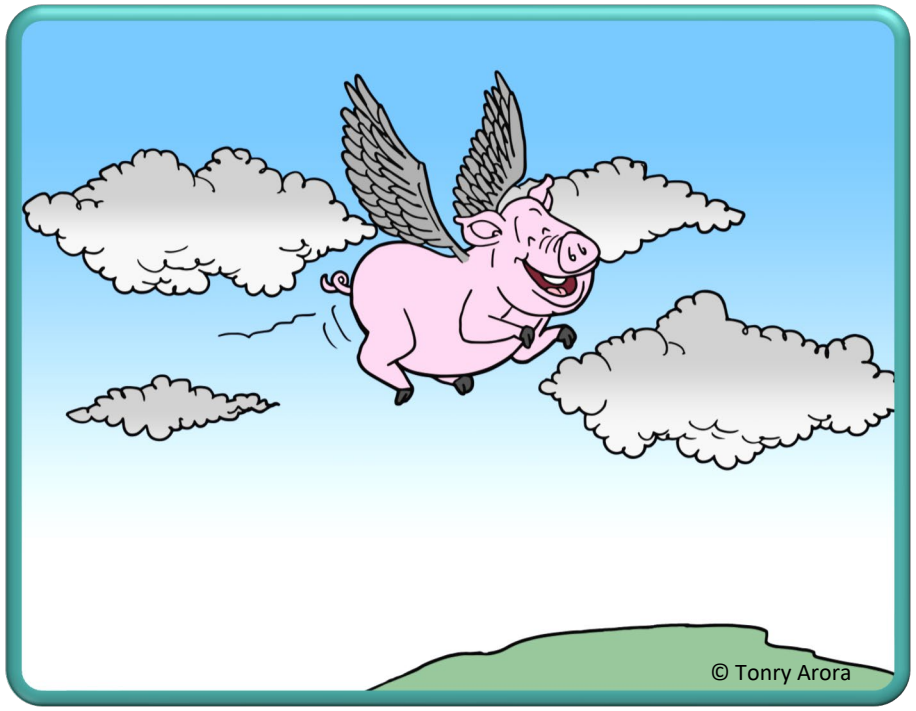
Does one SDL rule the world?



Does one SDL rule the world?



Does one SDL rule the world?



Does one SDL rule the world?

Does one SDL rule the

RSAConference2022

Rhetorical Question

Does one SDL rule the world?

It Depends

RSAConference2022

Rhetorical Question

Does one SDL rule the world?

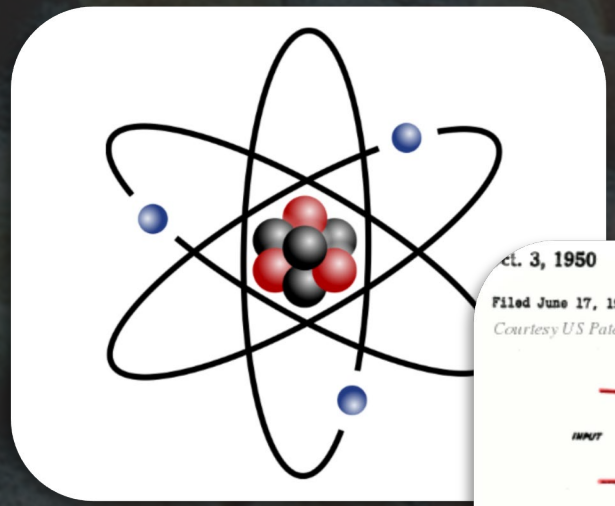
What is this nonsense about Electrons to Clouds?

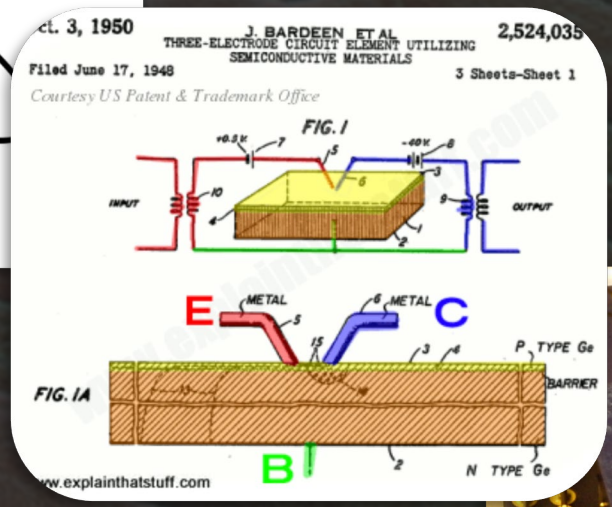
SDL layers of the technology stack

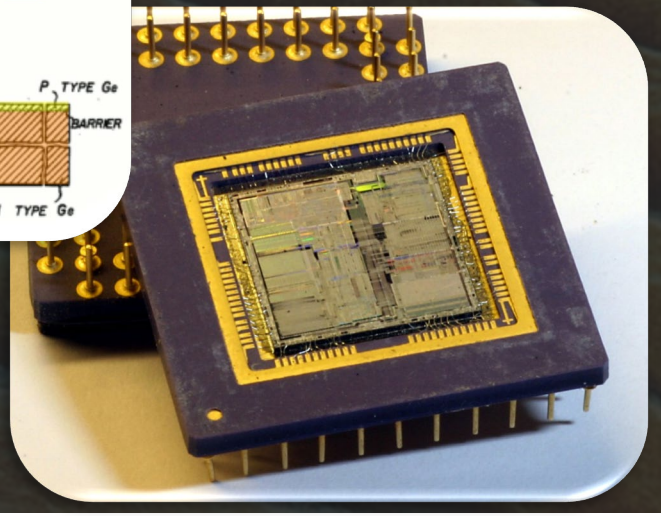
Hardware Development
- Component Production

Device Selection
- Motherboard Integration

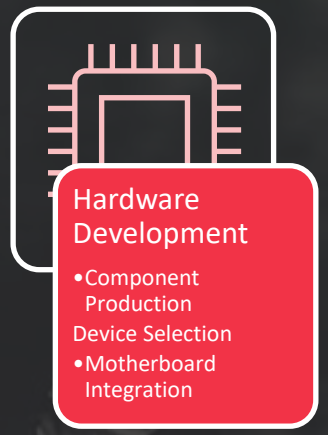
- Electrons
- Transistors
- Silicon
- Chipset





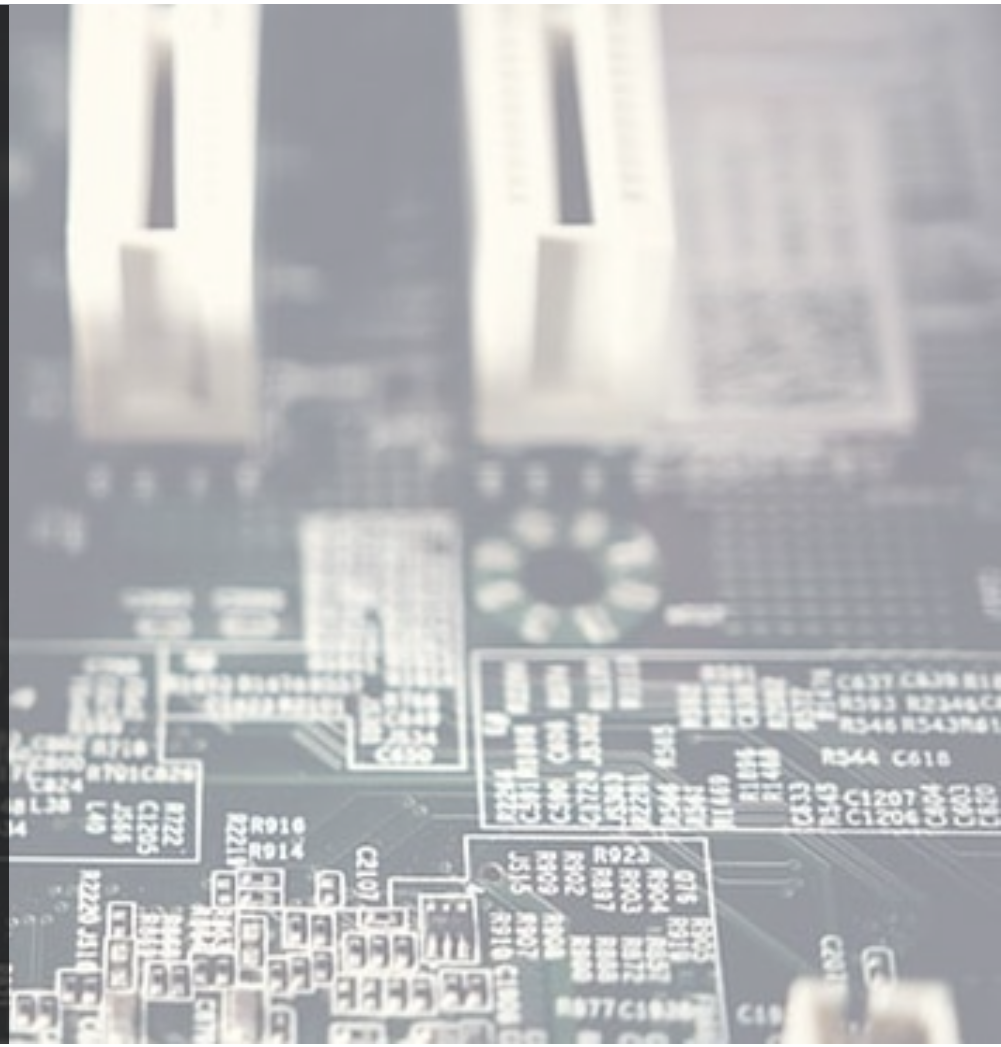


SDL layers of the technology stack

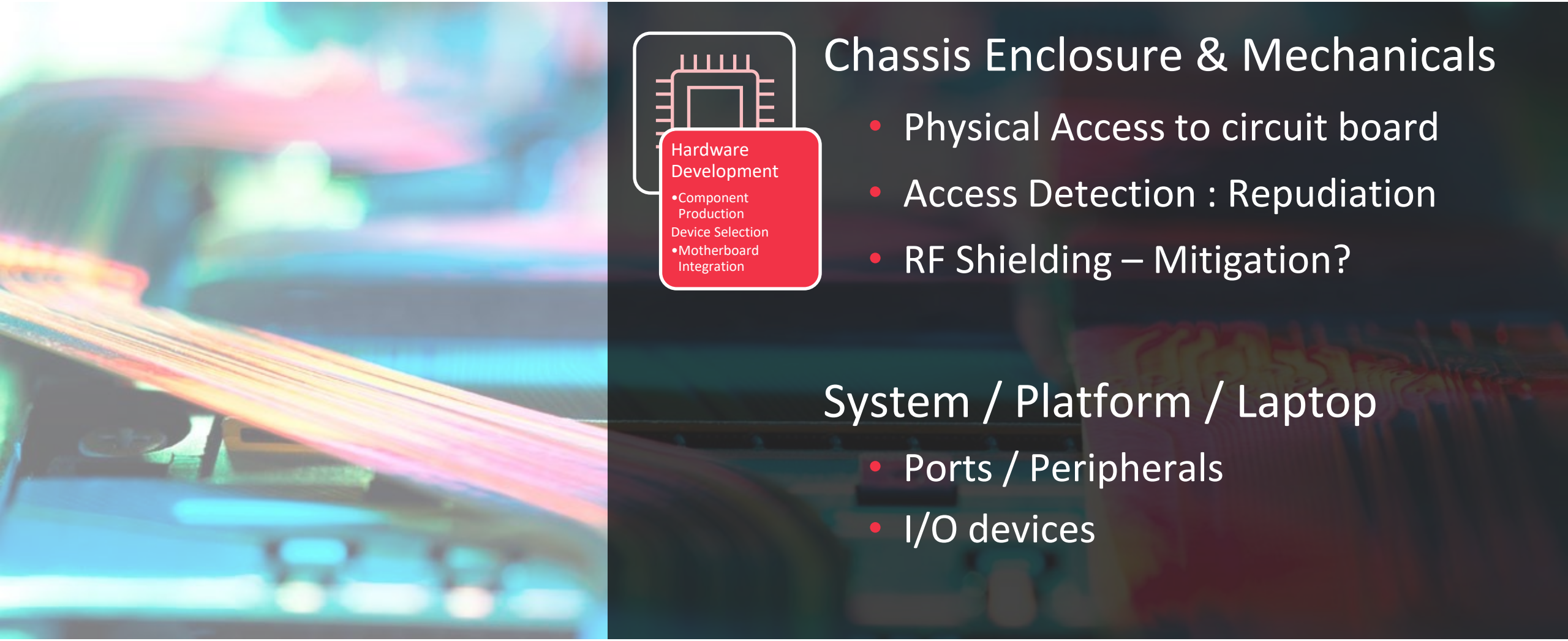


Circuit Boards

- Power analysis
- Access to interfaces, buses, debug ports
- Unauthorized components (Supply Chain)



SDL layers of the technology stack



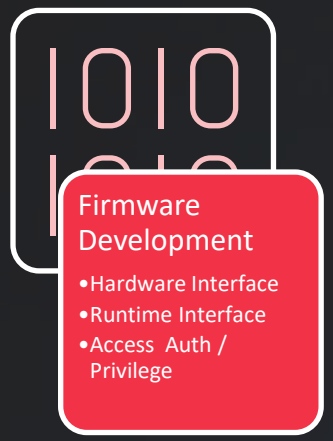
Chassis Enclosure & Mechanicals

- Physical Access to circuit board
- Access Detection : Repudiation
- RF Shielding – Mitigation?

System / Platform / Laptop

- Ports / Peripherals
- I/O devices

SDL layers of the technology stack



Firmware

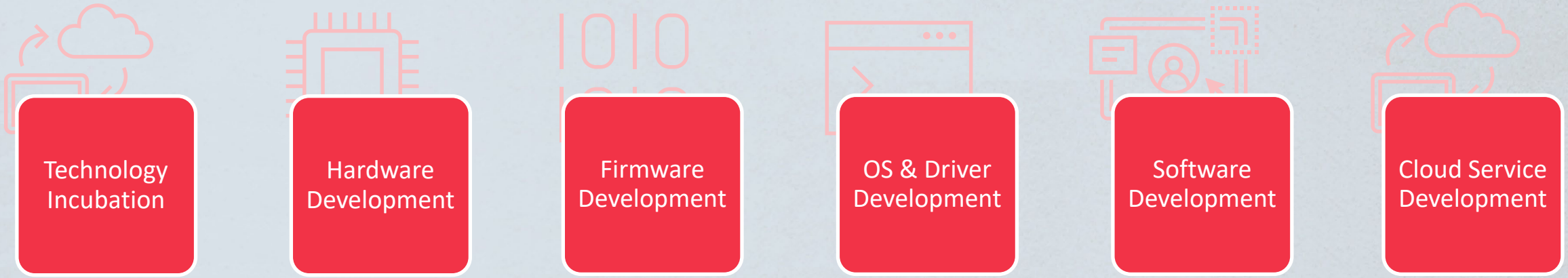
- Root-of-Trust, Attestation
- Identity
- Configuration
- Resiliency
- API's / Interfaces

SDL layers of the technology stack

- OS
- Drivers
- Software
- Network Services
- Cloud Services

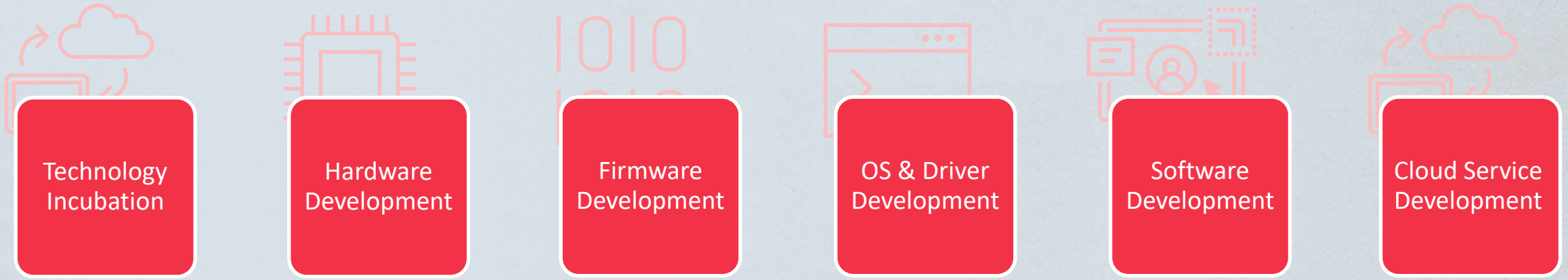
Full-Stack SDL

Electrons to Cloud (E2C)

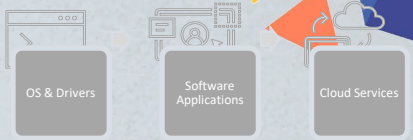


Full-Stack SDL

Electrons to Cloud (E2C)



Full-Stack SDL



Technology Incubation

- Problem:
  - Least agile layers require more lead time
  - Hardware is immutable
  - Product support duration
  - ‘You gotta live with what you got’
- Solution:
  - Early SDL engagement
  - Shift SDL left
  - SDL across the stack (E2C)
- SDL Goal:
  - Apply full-stack SDL during incubation
  - Security risk assessment
  - Threat modeling during incubation
  - Don’t forget about emerging threats

Full-Stack SDL

Hardware Development

- Problem:
  - Hardware is immutable (you gotta live with what you got)
  - Component SDL
  - Hardware based threats
    - Access to critical signals, power analysis / emissions
- Solution:
  - Hardware development SDL
  - Shift SDL left
  - SDL across the stack (E2C)
- SDL Goal:
  - Develop hardware security specification
    - Component selection, Layout rules, Tamper detection, Resilience / recovery, Emission suppression
  - Apply SDL during hardware design
    - Include mitigations identified from higher level stack threat models

Full-Stack SDL

Firmware Development

- Problem:
  - Highly privileged environment that is mutable
  - Interfaces to hardware, OS, software, cloud layers
  - Persistence of attack
  - Attack detection
- Solution:
  - Firmware SDL
  - Hardware SDL
  - SDL across the stack (E2C)
- SDL Goal:
  - Develop firmware specific SDL standards
  - Firmware architecture phase SDL (pre-development phase)
  - Hardware SDL / Firmware SDL continuity
    - Include mitigations identified from higher level stack threat models

Full-Stack SDL

OS & Driver Development / Integration

- Problem:
  - OS / HW / FW interfaces
  - Hardware Management / Configuration
  - Security Patch Management
- Solution:
  - Security Specifications
  - SDL across the stack (E2C)
- SDL Goal:
  - Establish OS Security Standards
    - Extend 3rd Party Component Security Standards
  - Establish OS, Firmware and Hardware Requirements

Full-Stack SDL

Software Development

- Problem:
  - Security is dependent on HW, FW, OS, Cloud
  - Broad attack surface (local/remote)
  - A vector for hardware and firmware attacks
- Solution:
  - Security Specifications
  - SDL across the stack (E2C)
- SDL Goal:
  - Perform defense in depth across the stack (E2C)
  - Continuity of E2C SDL
  - Develop and integrate Security Specifications in SDL

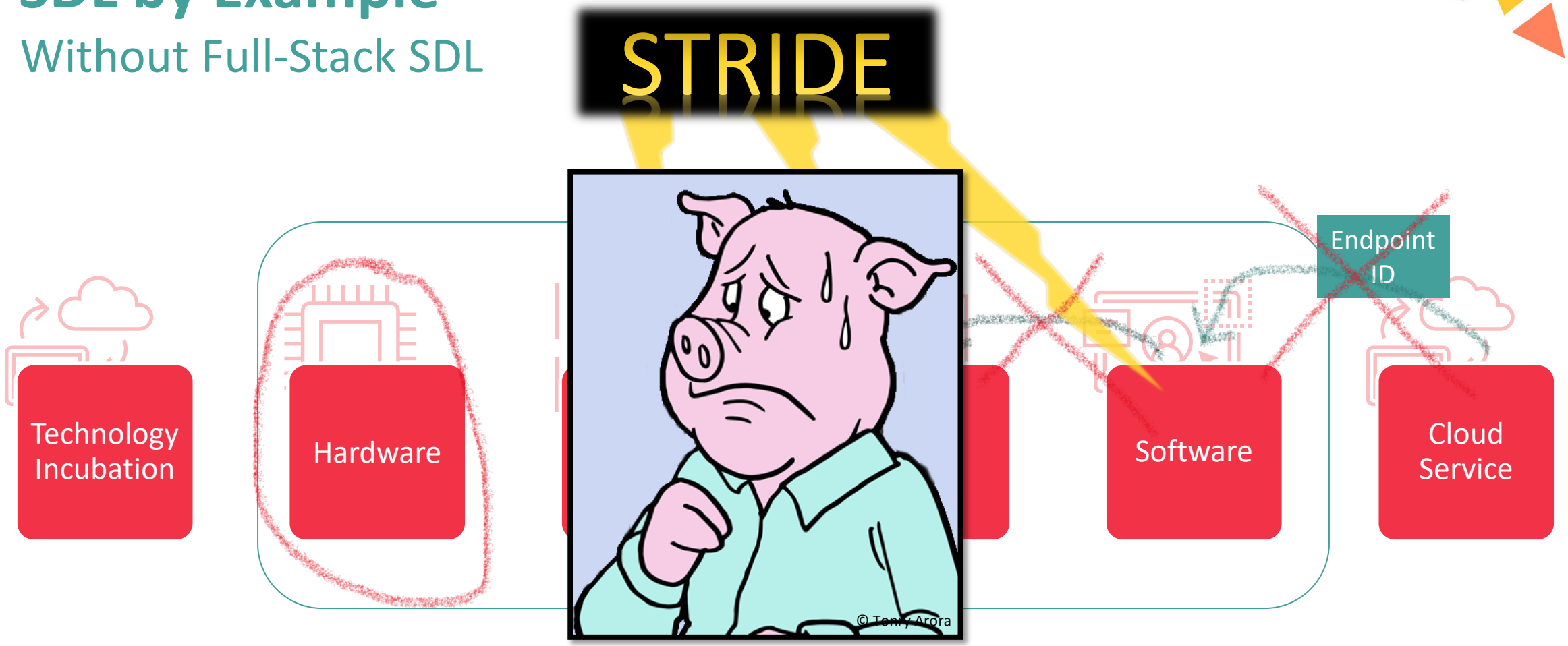
Full-Stack SDL

Cloud Services

- Problem:
  - Trust anchoring vs zero trust model
  - All the other cloud security challenges (not for this talk)
- Solution:
  - Security Specifications
  - SDL across the stack (E2C)
- SDL Goal:
  - Establish cloud, firmware and hardware security requirements
  - Develop and integrate security specifications in SDL
  - Perform defense in depth across the stack (E2C)
  - Continuity of E2C SDL

SDL by Example

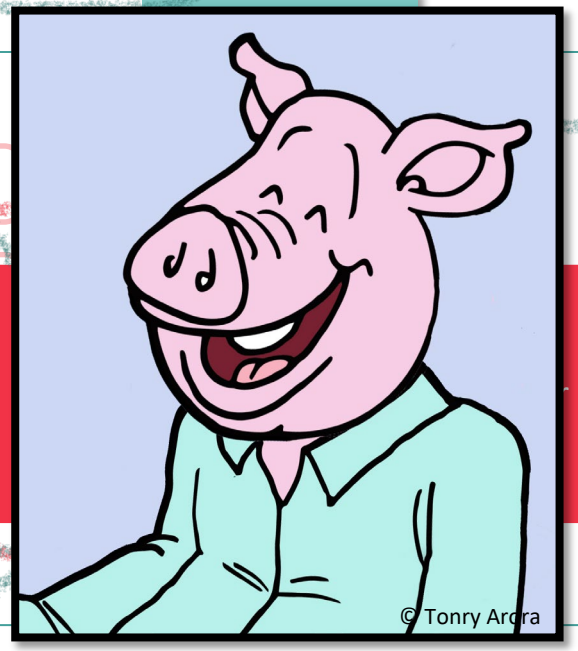
Without Full-Stack SDL



SDL by Example

Electrons to Cloud (E2C)

STRIDE



Endpoint ID

Technology Incubation

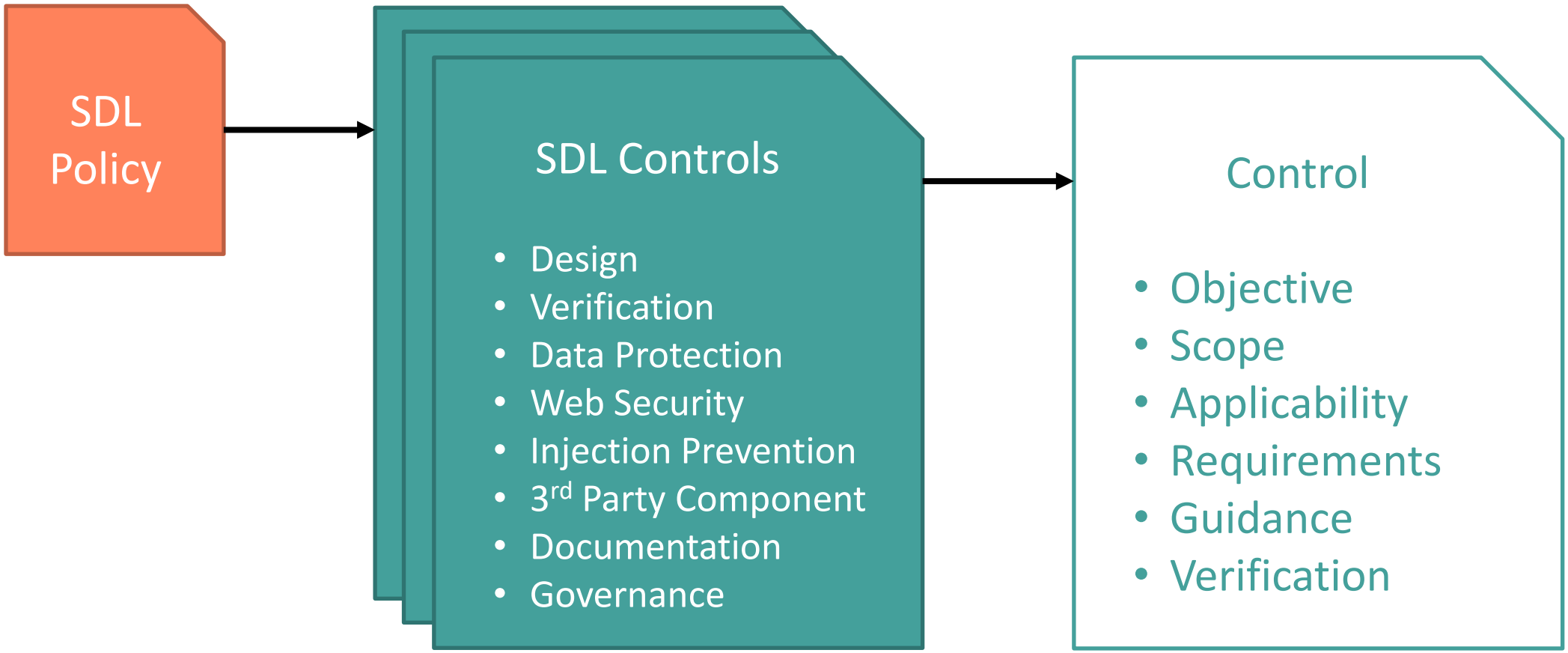
Hardware

HW ID

Software

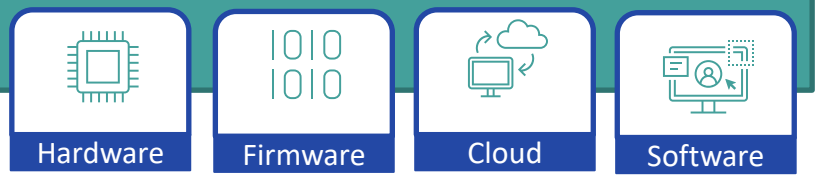
Cloud Service

Our Approach for One SDL

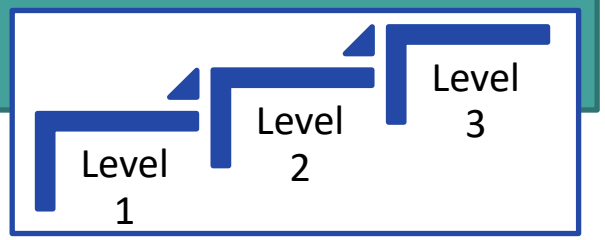


Our Approach for One SDL

Each control provides technology stack specific Applicability, Requirements, Guidance and Verification Activities



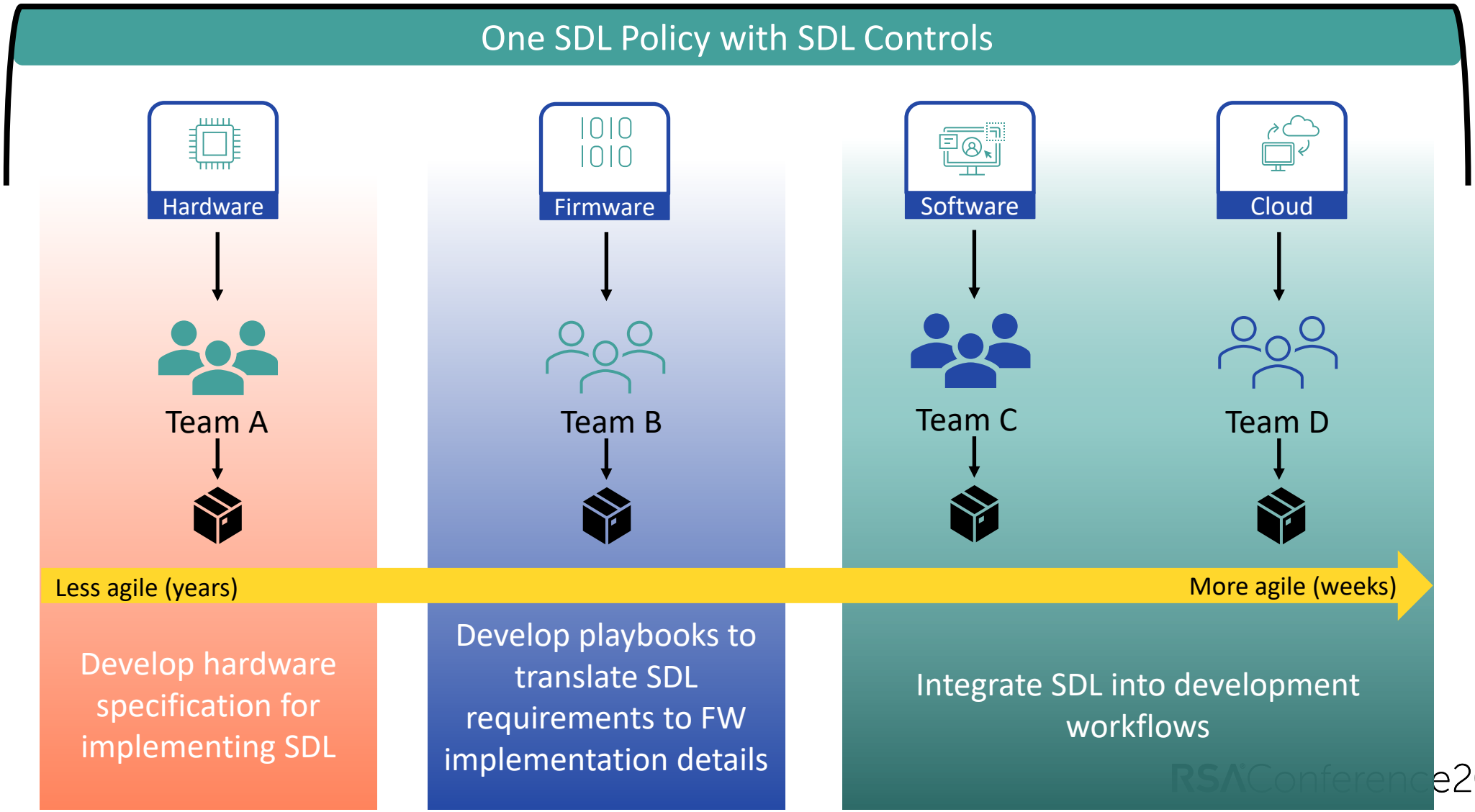
Each control also provides Requirements and Verification Activities based on SDL Maturity levels



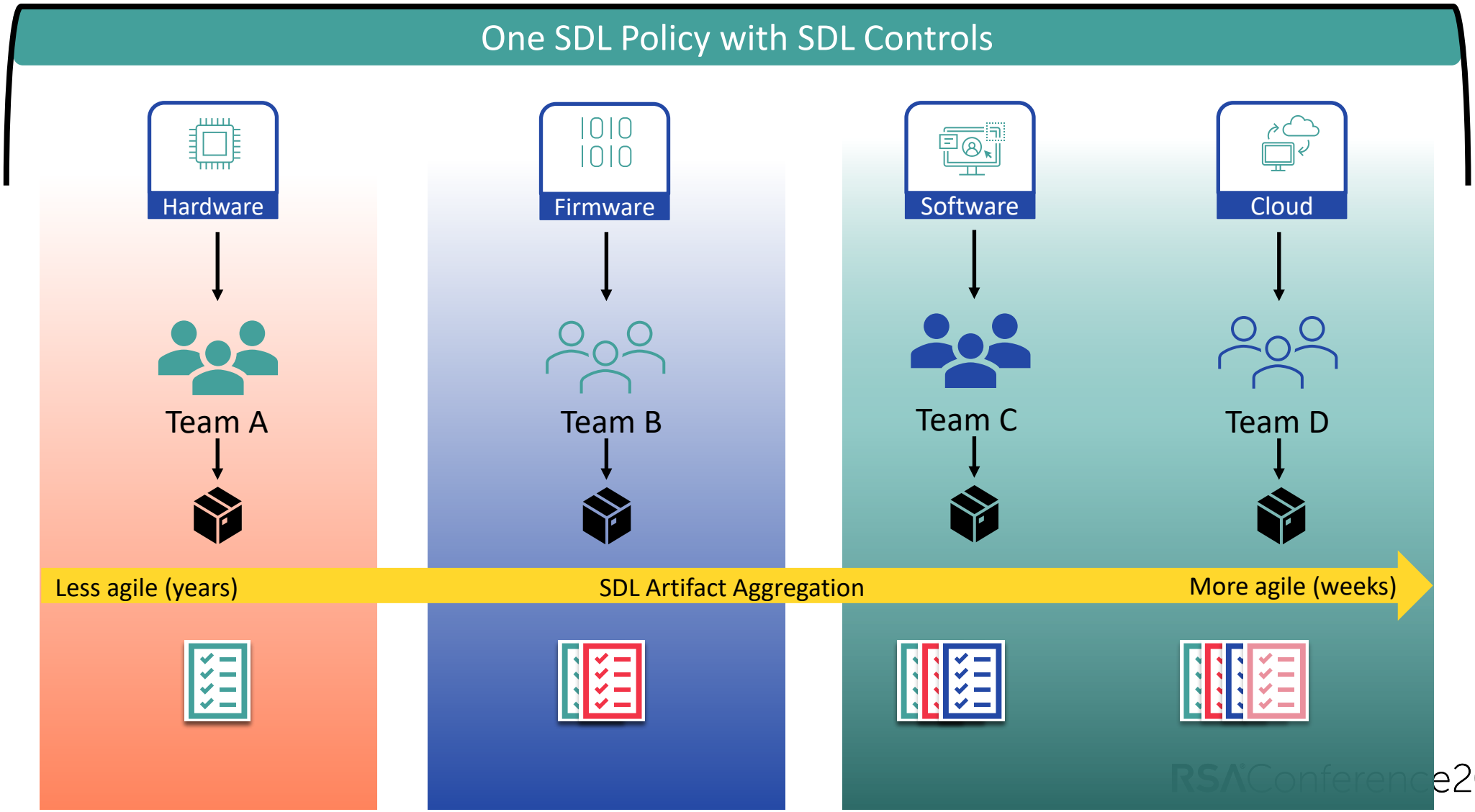
Control

- Objective
- Scope
- Applicability
- Requirements
- Guidance
- Verification

Our Approach for One SDL



Our Approach for One SDL



Supply Chain and SDL

- We can't talk about full-stack SDL without supply chain security
- The majority of organizations don't own the full-stack
  - Even if you think you do, you are still impacted by supply chain SDL.
- We must expand SDL scope to suppliers and vendors
  - Today, the industry relies on a contract-based approach.
  - We can benefit as an industry with a baseline full-stack SDL.
  - There are many industry SDL frameworks, but they tend to have a focused scope. None really provide a comprehensive full-stack approach.

Human Element – Culture Shift

- The same approach for shifting security culture works across software, firmware, hardware and technology incubation teams.
- A key aspect is helping teams recognize and understand SDL impacts in other areas.

Our previous RSAC talk discusses SDL culture shift and how to mature the security culture of an organization.

Key Takeaways

- Can one SDL rule the world?
  - Yes, with Full-Stack SDL
  - But it involves more than just one team, product, organization
- Why Full-Stack SDL?
  - The vulnerabilities will happen at the weakest SDL link
    - Lower layers (foundational security) secure the higher layers
    - Lower layers are less agile
  - Mind the Gap
    - Your SDL should overlap where different technologies overlap
    - Gaps between the different SDL layers need to be addressed with a full-stack SDL approach

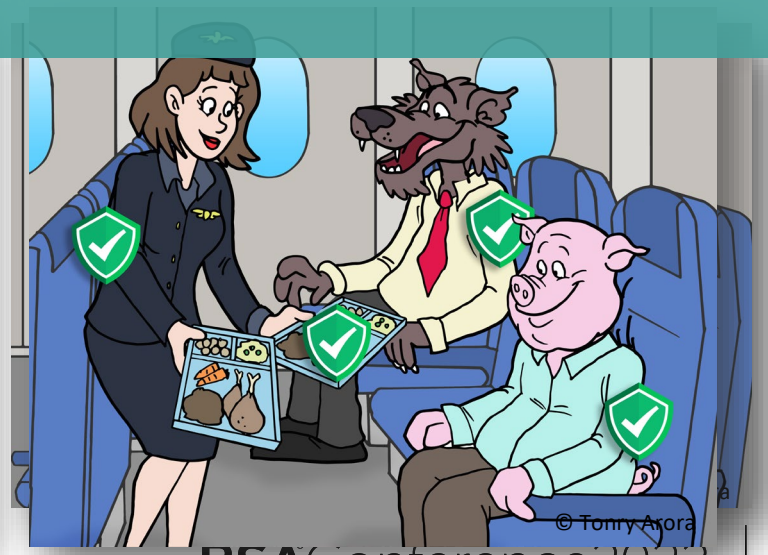
Key Takeaways

- What worked in the past may not work in the future
  - SDL has been around for many years
  - IoT, Smart Things, smart cars, AI/ML

Key Takeaways

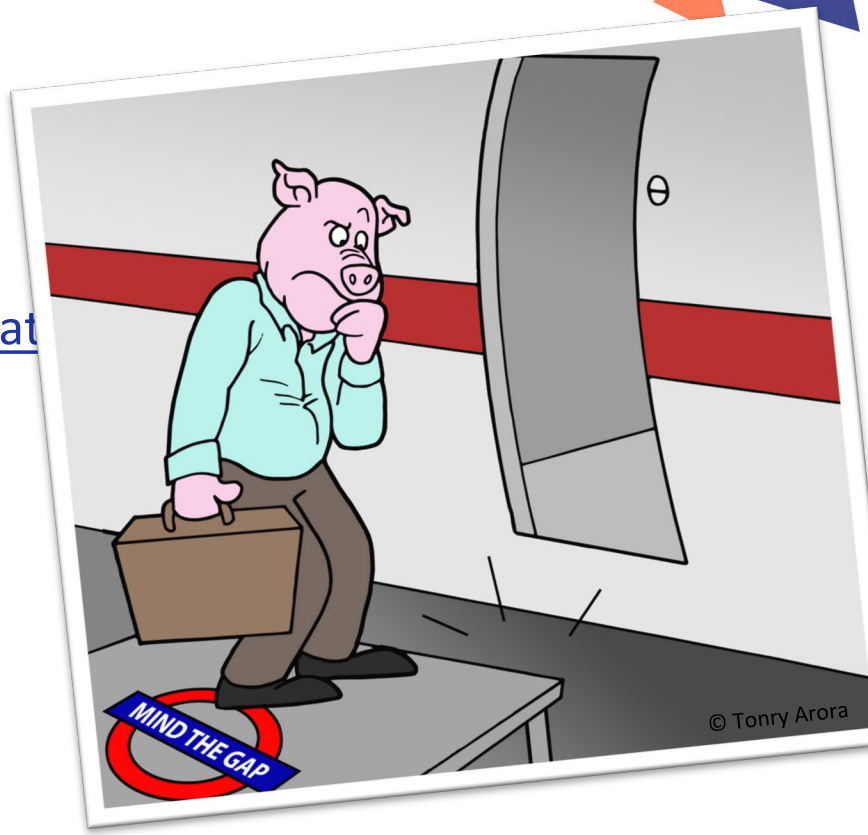
- What worked in the past may not work in the future
  - SDL has been around for many years
  - IoT, Smart Things, smart cars, AI/ML
- It's time to transform the industry
  - We are in the middle of the transformation

THE TIME IS NOW!



Applying To Your Organization – Immediate Steps

- Got SDL?
  - Take a look at our previous RSA talk:
  - Dude, You're Getting a Dell: Organizational Culture Shift to SDL Mat
- What does SDL mean for your organization
  - Evaluate your organization's needs
  - Assess the gaps
- What is your product portfolio
  - Does your organization span the technology stacks (SW, FW, HW)?
  - Does your organization rely on 3rd party components across these layers?



Applying To Your Organization – Short Term

- Establish or expand your organization's SDL requirements based on the needs and gaps identified.
- Pull together all development, architecture and technology incubation teams for full-stack SDL coverage
  - It needs to be a collaborative effort because different parts of the organization understand their specific business needs, product risks, security profile and risk tolerance.
  - It is vital to have these representatives participating to build a comprehensive full-stack SDL.
- Include procurement / supplier management to help expand SDL to your 3rd party suppliers, vendors and strategic partners.

Applying To Your Organization – Long Term

- Define and implement processes and controls to execute SDL activities based on your defined standards.
- Don’t forget to include the deployment and training needed to enable your teams.
- This is a journey
  - Developing a full-stack SDL process takes time.
- Review evolving needs / requirements
  - Continue iterating on SDL on a regular cadence
  - Once full-stack SDL is established and rolled out keep up with emerging industry threats, business needs and new technology stacks.

Applying To The Industry – Transforming SDL

- Time to transform our industry with a Full-Stack SDL
  - Consider industry standards
- Start the conversation
- Get involved
- Talk to suppliers and partners
- Incorporate standardized SDL
  - Your Org
  - Partners
  - Suppliers

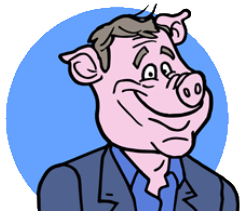
01010
0100000
010001
01000
0100001 SAFECode

SAFECode is a global nonprofit organization that brings business leaders and technical experts together to exchange insights and ideas on creating, improving and promoting scalable and effective software security programs.

Currently has multiple working groups across security domains.

safecode.org

Questions?



Richard Tonry
Firmware Security Architect
Dell
@RMTonry
Richard.Tonry@Dell.com

Mohit Arora
Software Security Architect
Dell
@NonceNinja
Mohit.A@Dell.com