Future proofing your PCI DSS program

PCI RESOURCES

GoSec – August 2019

Yves B. Desharnais, MBA, CISSP, PCIP

www.pciresources.com

Agenda

About Yves

1. PCI DSS & scoping: The PCI Resources Scoping Model and Approach
2. PCI DSS Controls: The PCI Resources PCI DSS requirements matrix
3. Combining both and ensuring maintenance of security controls

Q&A

About Yves

- IT/InfoSec expert generalist with experience in information security, development, Unix/Linux
- B.Eng. Computer Engineering, U. de Sherbrooke
- MBA, University of Notre Dame
- Worked with PCI since 2012 (2.0) & QSA in 2013-2014
- Author of books on PCI DSS in French and English
  - 5 releases since 2015 (www.pciresources.com)
- Author of NetBehave(.org) – a (NetFlow/IPFix) Network Behavioral Analysis Framework launched at Bsides Ottawa 2018

1. PCI DSS & scoping: The PCI Resources Scoping Model and Approach

The PCI SSC and the PCI DSS

PCI Security Standards Council™

The Payment Card brands created the Payment Card Industry Security Standards Council (PCI SSC) in 2006 to harmonize information security standards.

The PCI SSC (PCI council)

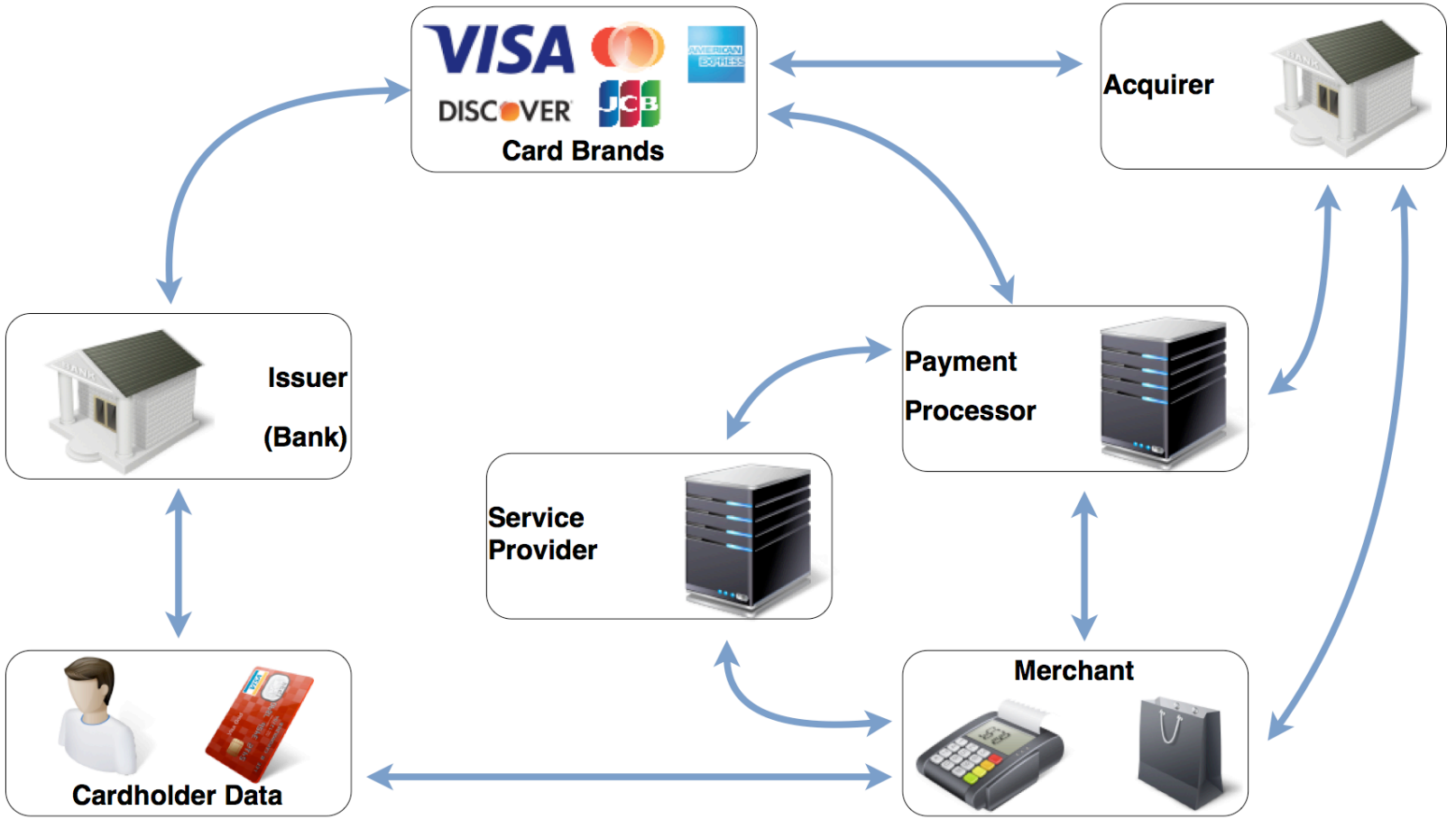
- defines Security Standards
  - PCI DSS to protect Cardholder Data (CHD)
  - PA-DSS (changing by 2021) for software
  - PCI PIN
  - PCI PIN PTS
  - Etc.
- And manages the firms that validate the standards
  - QSA & QSAC (QSA Companies) – PCI DSS
  - ISA – PCI DSS
  - PA-QSA – PA-DSS

Payment Card Industry (PCI) Payment Card Model

PCI DSS applies to all entities involved in payment card processing—including merchants, processors, acquirers, issuers, and service providers.

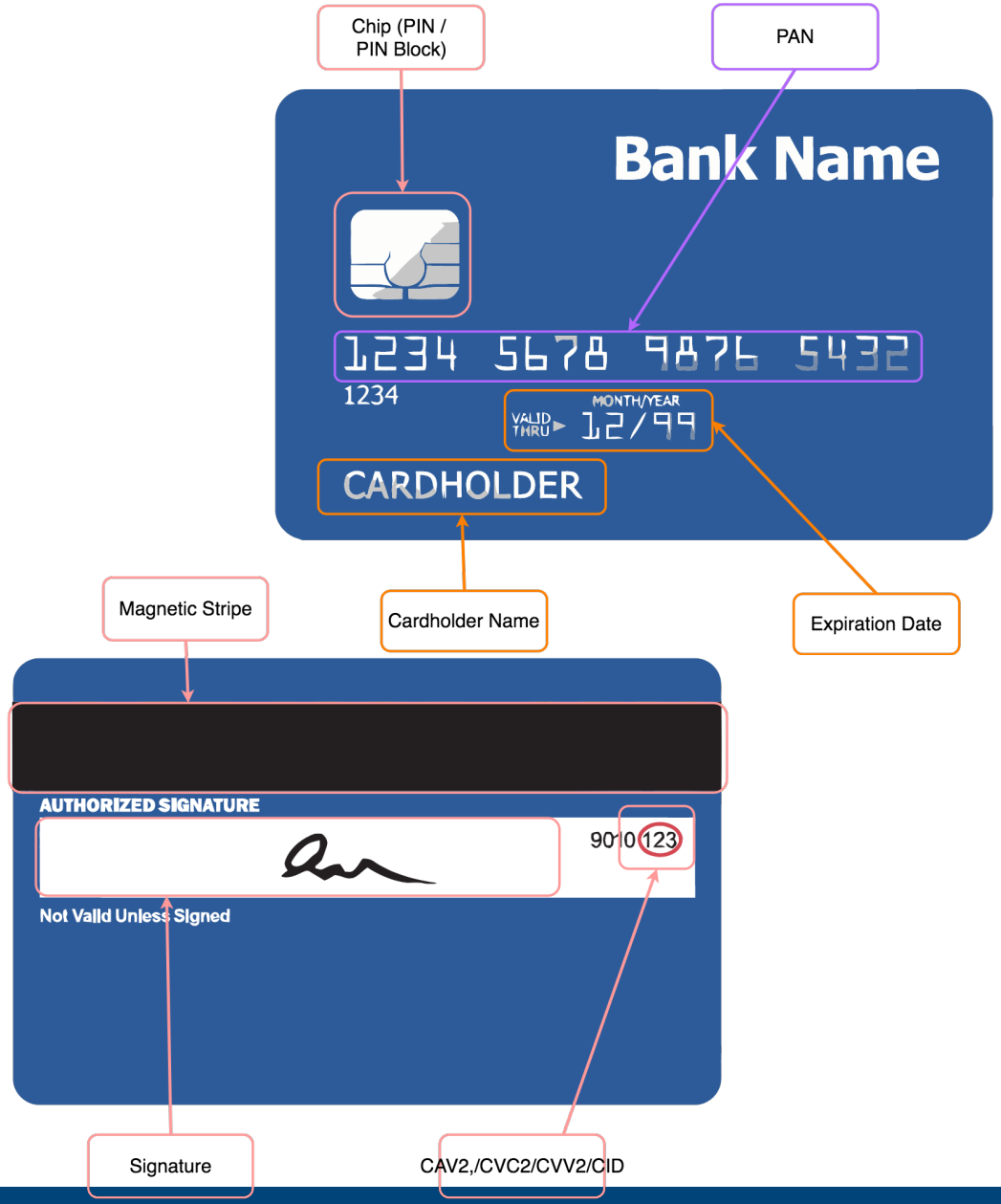
PCI DSS also applies to all other entities that store, process or transmit cardholder data (CHD) and/or sensitive authentication data (SAD).

Source: PCI DSS 3.2.1, p.5



GOAL: Protect Card Data
Data is of 2 main types

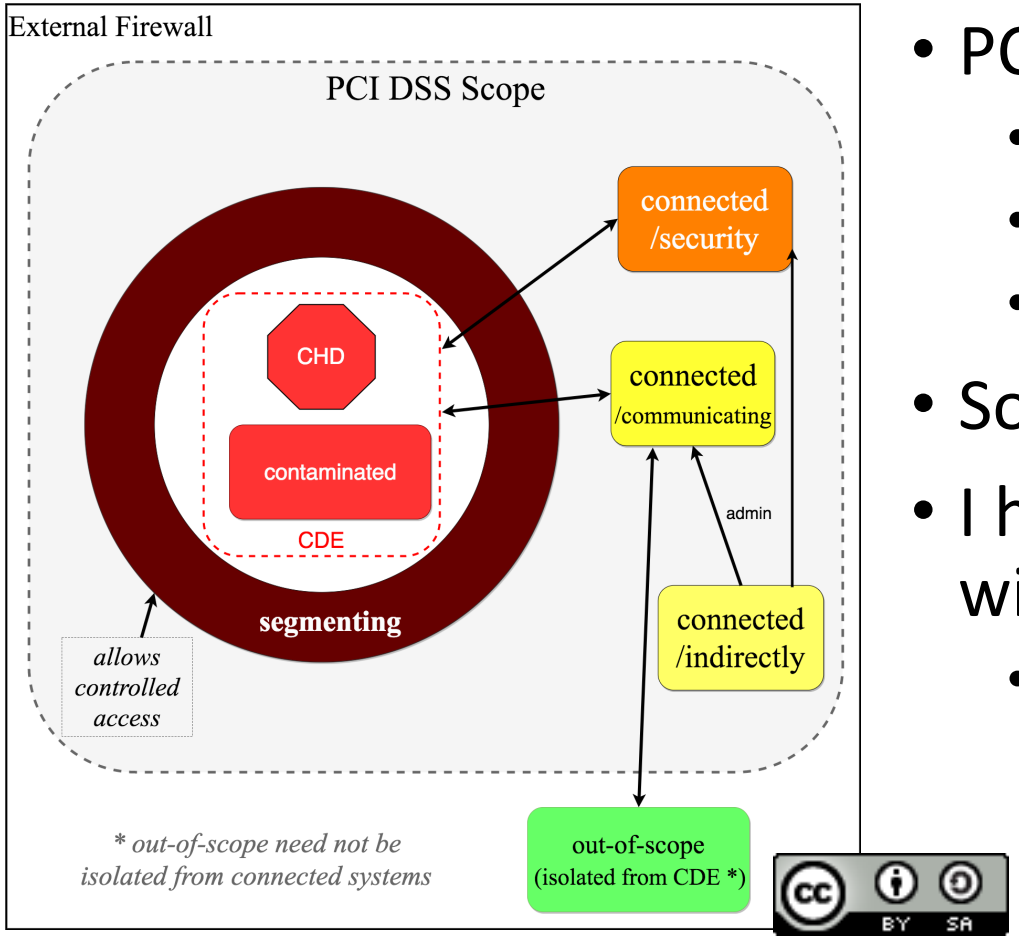
	Data Elements
Cardholder Data (CHD)	Primary Account Number (PAN)
	Cardholder Name
	Service Code
	Expiration date
Sensitive Authentication Data (SAD)	Full Magnetic Stripe Data
	CAV2/CVC2/CVV2/CID
	PIN / PIN Block



PCI DSS Scope includes

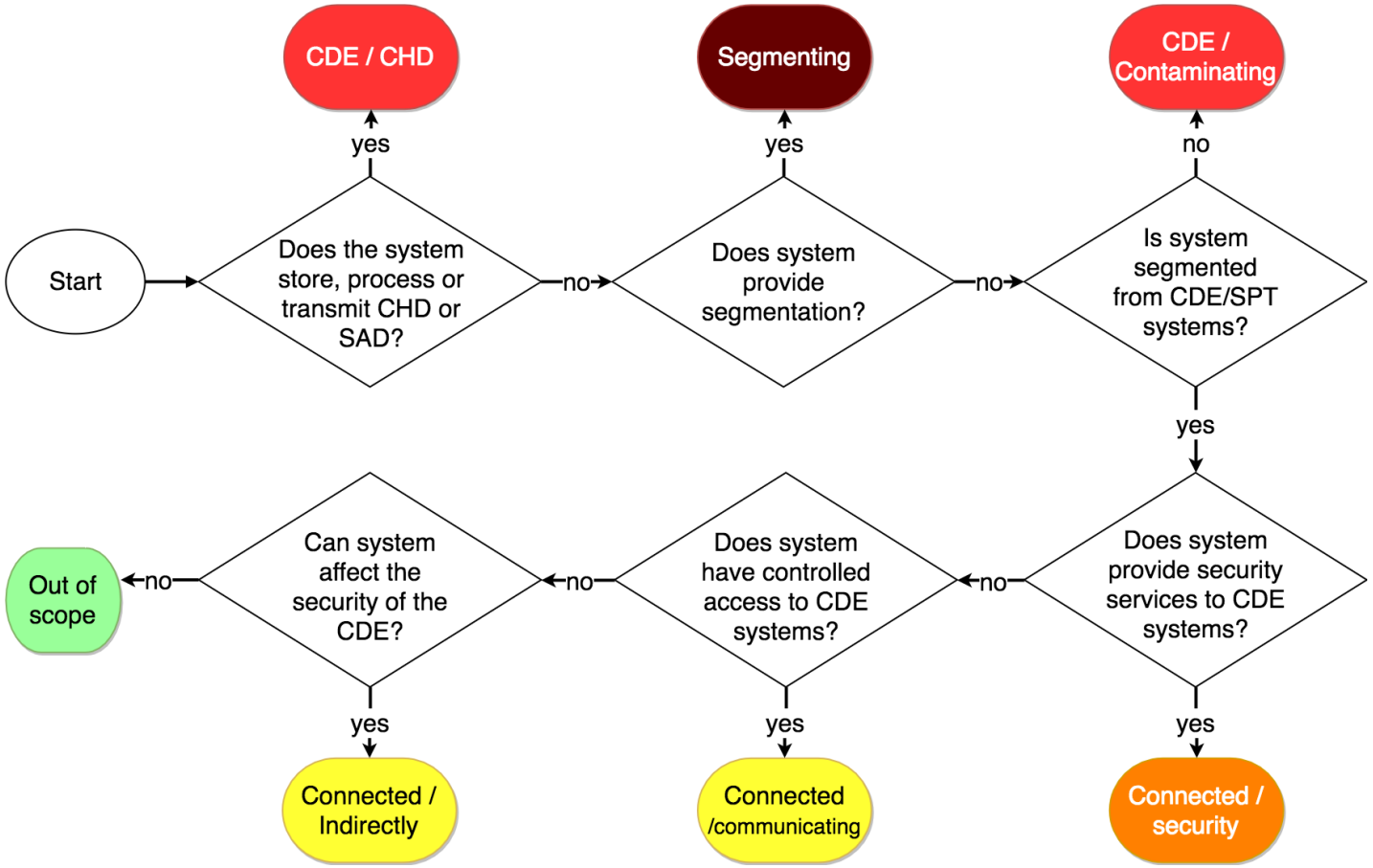
- The People (internal, external)
- performing Business Processes
- using Applications
- running on (physical and virtual) Systems
- and communicating over Networks
- in physical locations
- involved in the storage, processing or transmission (SPT) of card information (CHD/SAD)
- Or that could affect the security of card information (connected).

Overview of PCI Resources
PCI DSS Scoping Model and Approach



- PCI DSS only defines 3 types of systems:
  - CDE (Cardholder Data Environment)
  - Connected
  - Out-of-scope
- Some variations required to scope
- I have disagreements over segmentation with PCI SSC & OPST
  - In the absence of segmentation, everything is in scope and there are no connected systems

PCI DSS Scoping Model and Approach
PCI Scoping Type Decision tree



1. **CDE**
   - 1.1 CDE/CHD
   - 1.2 Segmentation
   - 1.3 CDE/Contaminated
   - 1.4 Validate with data discovery
2. **Connected**
   - 2.1 Use ACLs to identify communicating systems
   - 2.2 Connected/security
   - 2.3 3P connected systems (12.8.*)
   - 2.4 Connected/ communicating
   - 2.5 Connected/Indirectly

Network Segmentation - Overview

- Systems that provide the (generally network) segmentation and prevent "contamination" of CDE systems through "controlled access"
- Typically, these are firewall devices, but others are possible, generally at level 3 of the ISO model
- May be accomplished by a combination of devices and systems, but the more complex this gets, the better the documentation your assessor will require

- "If network segmentation is in place and being used to reduce the scope of the PCI DSS assessment, the assessor must verify that the segmentation is adequate to reduce the scope of the assessment." *PCI DSS 3.2.1, p.10*
- *Network segmentation testing is an annual (#11.3.4) or bi-annual (#11.3.4.1 for service providers) requirement*

2. PCI DSS Controls: The PCI Resources PCI DSS requirements matrix

PCI DSS 3.2.1

12 High-Level Requirements, > 250 requirements, > 400 tests

Goals	PCI DSS Requirements	Short Name
Build and Maintain a Secure Network	1. Install and maintain a firewall configuration to protect cardholder data	Firewall
	2. Do not use vendor-supplied defaults for system passwords and other security parameters	Hardening
Protect Cardholder Data	3. Protect stored data	Storage
	4. Encrypt transmission of cardholder data across open, public networks	Transmission
Maintain a Vulnerability Management Program	5. Use and regularly update anti-virus software	Antivirus
	6. Develop and maintain secure systems and applications	Secure Systems & Apps
Implement Strong Access Control Measures	7. Restrict access to cardholder data by business need-to-know	Need-to-know, RBAC
	8. Assign a unique ID to each person with computer access	Authentication
	9. Restrict physical access to cardholder data	Physical Security
Regularly Monitor and Test Networks	10. Track and monitor all access to network resources and cardholder data	Logging/Monitoring
	11. Regularly test security systems and processes	Testing
Maintain an Information Security Policy	12. Maintain a policy that addresses information security	Policy

PCI Resources approach to PCI DSS controls Matrix … by layer (stack) and function

	Scope Management	Access Control	Vulnerability Management	Logging and Monitoring
Governance				
Policy				
User				
Data				
App				
Operating System				
Network Architecture				
Physical				

* From PCI DSS made easy, section 3.15; note, no mapping is ever perfect, some overlap expected…

PCI Resources approach to PCI DSS controls Top layers: Governance and Policies

	Scope Management	Access Control	Vulnerability Management	Logging and Monitoring
Governance	• Infosec Responsibilities • Third-Party Management • Scope & Diagrams	• HR Background Checks	• Risk Assessments	• Detect failures of critical security controls (BaU) [1]
Policy	• Infosec & Acceptable Use Policy • Data Retention and Disposal Policy • Policy: No PAN via email, chat, etc.	• Security Awareness Training, including on passwords • Roles	• Pentests • Vuln. Management • Change Control • Change defaults settings • System Config Standards • Router/Firewall Config & Changes	• Logging • Monitoring • Incident Response
User		• User Identification & Authentication • No shared account		

[1] Service provider only in 3.2.1

PCI Resources approach to PCI DSS controls Middle layers: systems and applications

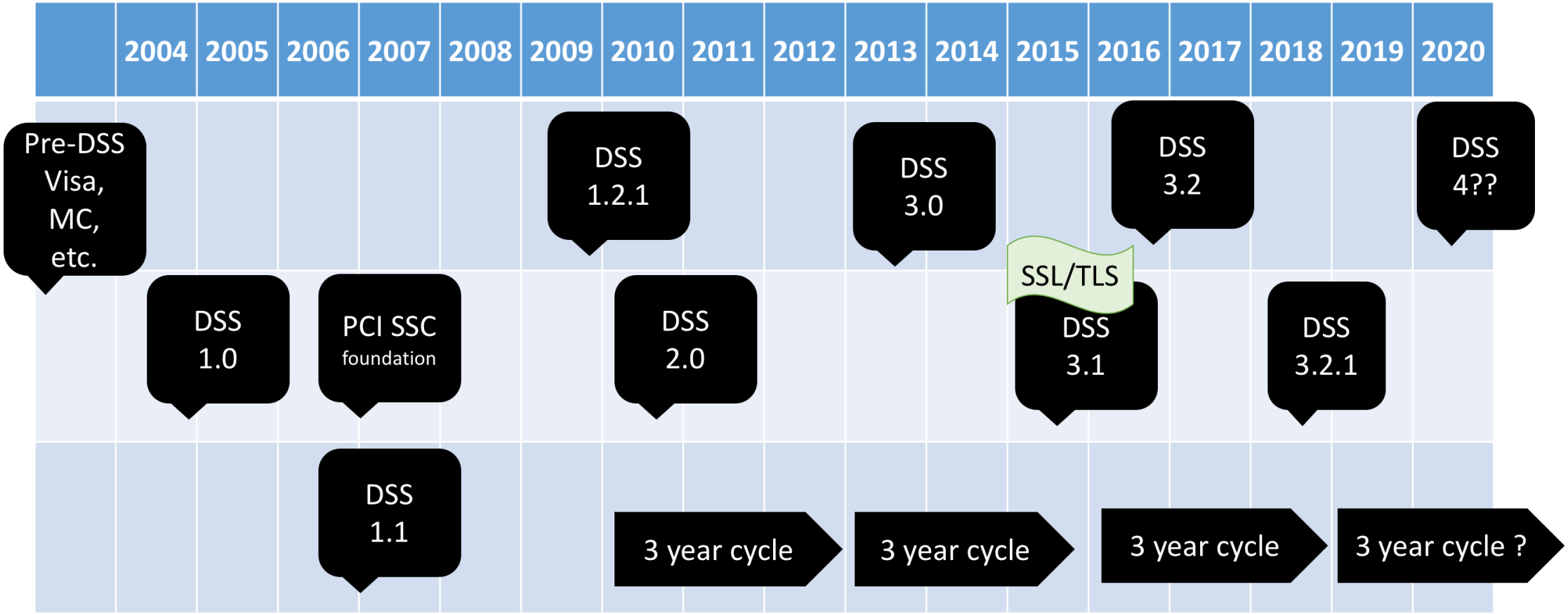
	Scope Management	Access Control	Vulnerability Management	Logging and Monitoring
Data	• Storage of: SAD, PAN • Cryptographic Key Management	• DB Separation of Duties (SoD) & programmatic methods		• All individual user accesses to cardholder data
App	• Secure transmission on open public networks (TLS, VPN) • Mask PAN when displayed	• RBAC	• Secure SDLC • Secure Application Development • Protect web apps	• Secure (centralize) logs • Log Retention
Operating System	• Inventory of system components	• Personal Firewall	• Antimalware • Patching • Vuln. scans	• Sync clocks (NTP) • Change Detection/File Integrity Monitoring

PCI Resources approach to PCI DSS controls Bottom layers: Physical and Network

	Scope Management	Access Control	Vulnerability Management	Logging and Monitoring
Network Architecture	• ACL documentation • ACL review (every 6 months) • Firewall between: • - CDE/untrusted • - CDE/internet • Secure wireless networks	• MFA (CDE & external) • Segmentation Pentest (CDE)	• Implicit in Risk Assessment	• Test or Monitor for Unauthorized wireless networks • IDS / IPS
Physical	• Implicit in RoC template	• Physical Access Control and Monitoring • Visitor management • Media Controls	• Implicit in some testing procedures	• POS Device Tampering checks

3. Combining both and ensuring maintenance of security controls

PCI DSS version history over time



* 4.0 expected late-2020 or later

Previous and expected major changes (called “Evolving Requirements” by PCI SSC)

- 2.0 (2010) to 3.0 (2013)
  - Scope clarifications:
    - Data flows: 1.1.3
    - Inventory: 2.4
  - Segmentation Testing (11.3.4)
  - Logging of more events (10.2.5,6)
  - Pentest Methodologies (11.3)
  - Responsibilities between entity and third-parties (12.8.2, 12.8.5, 12.9)
  - POS tampering (9.9.*)
  - Added an introduction to "BaU" (Business As Usual)
  - Report on Compliance (RoC) template
- Designated Entity Special Validation (DESV) for 3.1 in 2015 (maintenance of compliance), now Appendix A3
- 3.1 to 3.2
  - MFA for system administrators (8.3.1)
  - Consider PCI compliance maintenance in changes (6.4.6 "BaU")
  - "BaU" for service providers (10.8.*, 12.11.*)
- 4.0 (expected late-2020 or later)
  - Key priorities are:
    - Security (often based in attacks that are seen as successful)
    - Flexibility (for example, req.#11.5 was changed in 3.0 from "File Integrity Monitoring" to "Change Detection Mechanism")
  - See Software development standards (January 2019) that will replace PA-DSS

How to address PCI DSS?
Goal: protect card data

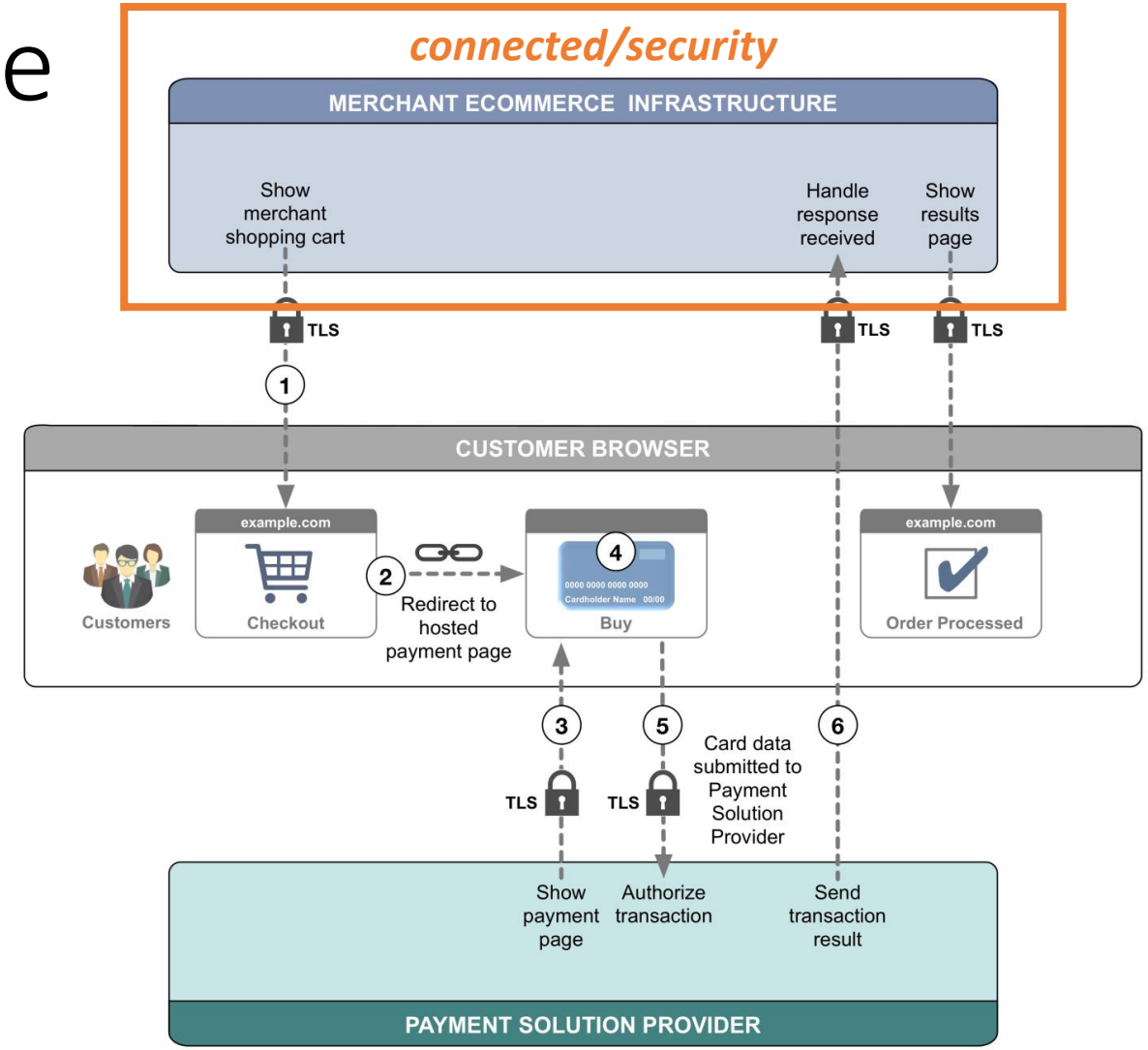
- Document (& reduce) scope
  - *Validate where information is stored*
  - You can't protect what you don't know you have
- Devalue data
  - *Limit data retention (privacy-by-design approach, also recommended by GDPR) : "Remember, if you don't need it, don't store it!" – PCI DSS 3.2.1, p.37*
  - *Eliminate, Truncate, Tokenize, Encrypt (in that order)*
- Use defense in depth
  - Attention to virtualization & cloud
    - Re-entry & all new attack methods: meltdown, specter, RAMBleed
  - Don't only consider M&M model: Consider east-west traffic (should 2 systems communicate) & micro-segmentation
- Use threat modeling
  - PCI DSS controls may not be sufficient
  - Evaluate new methods
- Train your people & fix processes (BaU)
  - Don't forget internal (insider) threats

Reduce Scope through third-parties Merchant eCommerce

Prefer:

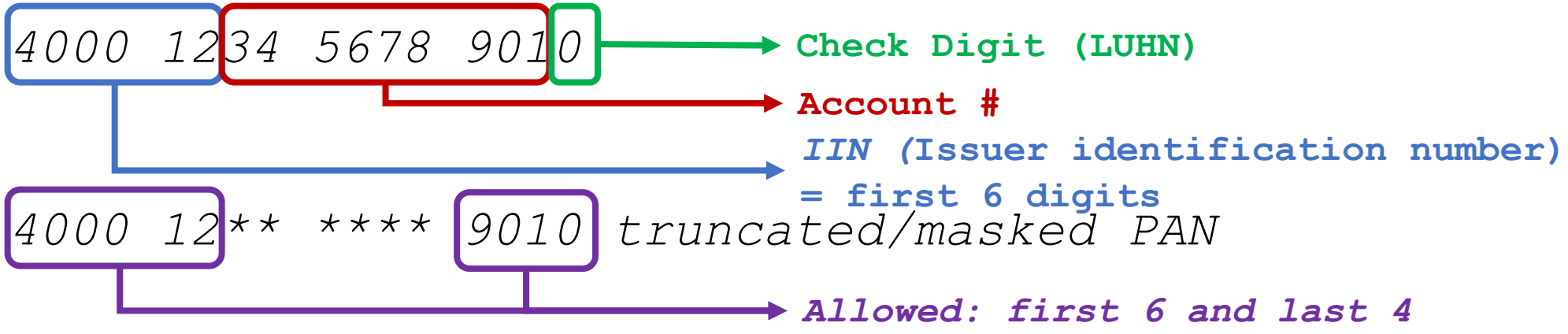
- URL Redirects
- iFrame
- Direct Post Method (DPM)
- JavaScript Form

Over Application Programming Interface (API)



Devalue data - PAN transformations
PAN Truncation / Masking

- PAN format
  - follows ISO/IEC 7812
    - supports up to 19 digits
    - most only 16 digits
    - American Express only use 15 digits.
- Difference between masking and truncation
  - Masking means only display is hidden (the system has all digits) (3.3)
  - Truncation removes data (the system performing truncation is in-scope), those receiving the truncated PAN are not contaminated if segmented



Devalue data - PAN transformations Tokenization

- one of the options of requirement 3.4 (PAN storage)
- "*An index token is a cryptographic token that replaces the PAN based on a given index for an unpredictable value.*" [1]
- Format Preserving Encryption (FPE) exists and keep the same format
- Could use 2 IIN (BIN) series reserved for internal use by Visa Europe: 468738 & 468739 [2]
- Simplest implementation is a database that maps IN to OUT (either at random or sequentially)
- Options using encryption exist, but they should be validated (and I am skeptical)

[1] PCI DSS 3.2, p.40

[2] https://www.visaeurope.com/media/images/12_using_the_visa_private_bin_range_-_best_practice_guide%20110615-73-24720.pdf

Devalue data - PAN transformations Encryption for scope reduction

Storage (data at-rest)

- Can reduce scope if entity does not have the decryption keys
- Key management is an issue
- Symmetric vs asymmetric cryptographic decisions

Transmission (data in-motion)

- Encryption required on non-entity networks (i.e. open/public)
- Transmission encryption can (validation required) remove devices from scope through encrypted tunnel
- P2PE reduces scope as the merchant does not have the keys (it is with acquirer/payment processor)

Use defense in depth

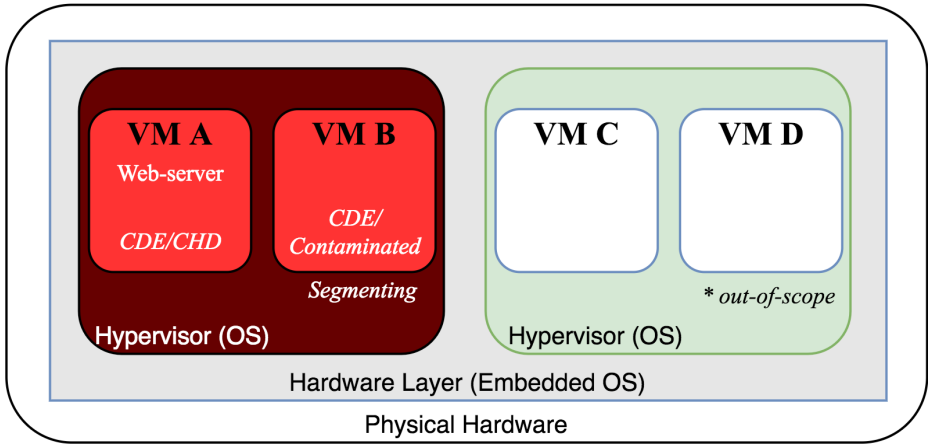
- The perimeter is broken:
  - The M&M model does not work
- Your data is everywhere:
  - End-user laptops
  - End-user phones (including BYOD)
  - Internal Servers
  - Cloud
  - At third-parties
  - (hopefully not) on the dark web…

- Know your environment and where your data is stored and transmitted
- Consider using:
  - Micro-segmentation (at each NIC) or at least multiple network zones
  - Zero-trust model (do not trust until you've identified and authorized the user)

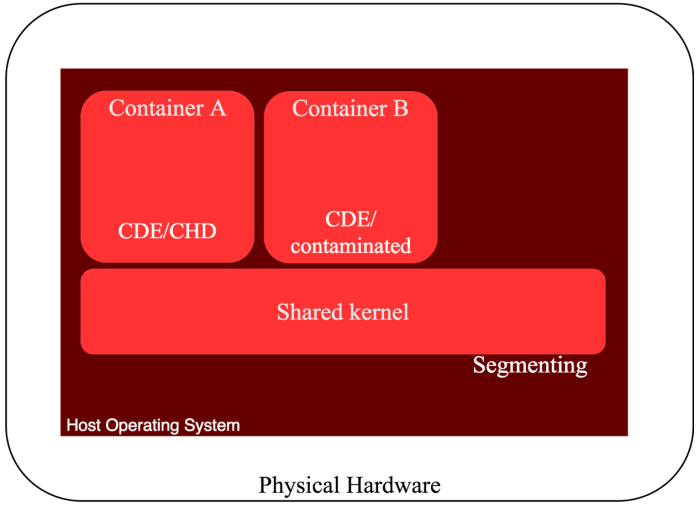
Virtualization & the cloud
VMs vs. Containers

- General virtualization
  - VM running on Hypervisor on Host
- Containers
  - A form of Software virtualization
  - Lighter than VMs
  - Examples:
    - Chroot (since 1982!)
    - FreeBSD jail
    - Solaris Containers
    - Docker [1]
- Problem for PCI DSS scope: No clear segmentation between VMs or containers
  - Segment at the hypervisor level

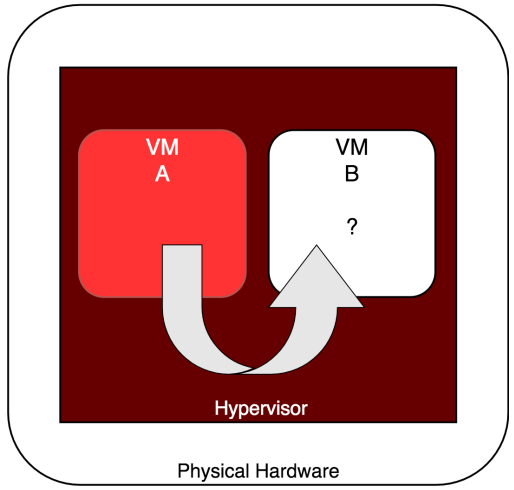
Virtualized web-server containing CDE isolated using hardware virtualization



Operating System Level Virtualization



Virtualization Re-entry



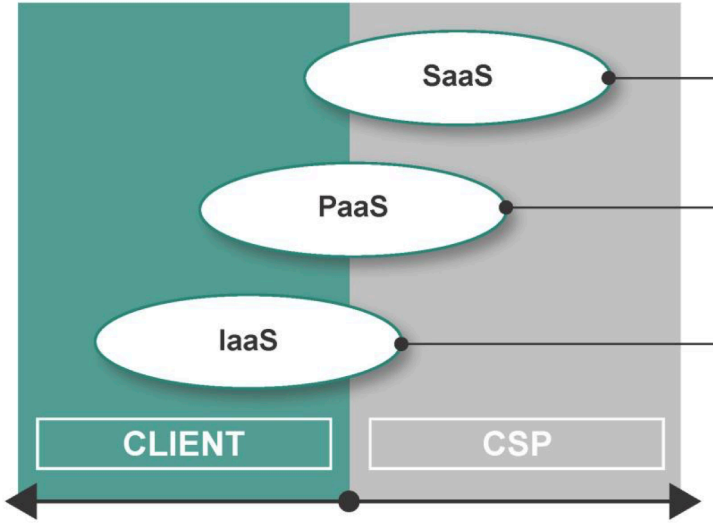
[1] Kubernetes allows management of docker containers programmatically

Virtualization & the cloud
Cloud computing: overview

- Cloud: product of virtualization and outsourcing
- Different models
  - IaaS : Infrastructure (aka VM)
  - PaaS : Platform
    - FaaS : Function (AWS Lambda, Google Functions, MS Azure Functions)
  - SaaS : Software
- Also
  - Private : Segmented to the entity
  - Public : Shared between entities
  - Hybrid : a mix of private/public

[1] aaS = as-a-service
[2] h

- Use in PCI DSS:
  - The boundaries for who is responsible for what are not clear.
  - Only use a pre-validated PCI DSS compliant version for CDE systems



Level of control/responsibility for client and CSP across different service models

Use Threat Modeling

- Threat modeling is
  - a practical discipline (many different methodologies [1])
  - is about using models to find security problems
  - Anyone can learn to threat model
  - everyone involved probably should.
- Goals with Threat Modeling?
  - catch a security problem before it starts. (at design stage), to anticipate the threats
  - The earlier you find problems, the easier it is to fix them => why? costs

- How do we do threat modeling?
- STRIDE focuses on four questions
  1. What are you building? (Step 1 – Model the systems)
  2. What can go wrong? (Step 2 – Finding threats)
  3. What should you do about those things that can go wrong? (Step 3 – Addressing threats)
  4. Did you do a decent job of analysis? (Step 4 – Validating your work)

[1] https://insights.sei.cmu.edu/sei_blog/2018/12/threat-modeling-12-available-methods.html

Maintaining compliance Business-as-Usual (BaU)

- "BaU" addresses maintenance of PCI compliance by integrating security controls in regular operating processes
  - Similar to moving from a type 1 audit (test of design) to a type 2 (test of operational effectiveness)
- Current requirements for service providers
  - 10.8.*: Processes for timely detection of failures of critical controls
  - 12.11.*: Quarterly review of operational processes (compliance)
- Current requirements of Appendix A3 for designated entities
  - A3.1: Compliance Program
  - A3.2: Scope documentation and validation
  - A3.3: Validate controls in BaU (controls are maintained)
  - A3.4: Control and manage logical access
  - A3.5: Identify and respond to suspicious events.

How to address BaU? Ensuring maintenance of compliance

- We need to prevent
  - Unauthorized changes (either malicious or bypassing processes)
  - Changes which can negatively affect compliance (that do not address the required controls)
- How can we detect:
  - Changes to scope
  - Changes to required controls
- Can you automate some of these? Some examples:

- Scope changes
  - Asset Management (CIS #1 & #2) including proper decommissioning: ping sweep
  - Applications: FIM solutions, port scans
  - Firewall rules changes: config diffs
  - Data flows : Traffic analysis
  - Perform periodic data discovery scans
- Changes to required controls
  - Logging & Monitoring: Statistics of logs collected over time vs Monitoring statistics
  - Change Control, Patching: FIM solutions, with reconciliation with tickets
  - Manual tasks: ensure by-product of evidence demonstrating what was done
  - Etc.

REMEMBER:

- Security is a journey
- Perfect is the enemy of the good
- No plan ever survived encounter with the enemy

Questions?

SO:

- Build secure foundations
- Simplify, standardize
- Start small, work incrementally
- Automate, orchestrate