Cybersecurity BootCamp

week 3

ZERNA Lesly

October 2017

1 Introduction

Cybersecurity BootCamp by CyberWayFinder SPRL consists in a four week training in topics to be introduced into the cybersecurity field. This training is mainly focused on women inclusion into this field.

This document contains my personal notes and extra info about the topics involved during the 4 weeks.

2 Class 7 - Risk management 2 (Practice)

Study case with:

- multi-tiered system architecture
- private cloud solutions (Ex. Amazon)

What do you need to assess?

- threats
  - potential danger
- inherent risk (raw)
  - cost is on a yearly basis
  - probability * impact = risk
  - impact:
    - significant
    - major
    - catastrophic, financial loss > 1 million EUR
- type:
  - strategic (out of scope)

  - operations
  - architecture
  - access control

What to do?

- excel sheet
- network diagram
- mr-know-it-all (ask?!)

We analyzed two networks. The first one is typical with a firewall in front of the Data Bases, Middleware and a Front. The second network involved Virtual Machines and only a firewall at the input of the local network.

Differences between: system stand-by vs. loading balance **loading balance** preferable when you have multiples data centers. Normally 1 data center should not have more than 40% of work or load.
When using the excel for analysis:

- threat agents
- **threat scenario** -> when you make presentation to CEO, board, etc
- **exploitable**
- **impact** -> board might know about this impact
- **control**
  - detect
  - prevent
  - reactive

Example study case in excel! <ask excel example>

Acronyms

- **SOC** -> Security Operation Center
- **SIEM** -> Security Incident Event Management
- **BCP** -> Business Continuity Plan
- **DRP** -> Disaster Recovery Plan
- **IPS** -> Intrusion Prevention System
- **IR** -> Incident response
- **ISO27000** -> standard

- **UPS** -> Uninterrupted Power Supplies
- **RPO** -> Recovering Point Objective
- **RTO** -> Recovering Type Objective
- **PUAM** -> Privilege User Access Management
- **HSM** -> Hardware Security Module
- **RAP** -> Remote Access Protocol

3 Class 8 - NVISO IOT & Threat Hunting

Link Event -> https://www.eventbrite.com/e/sans-community-night-october-2017-nviso-tickets-38329670075

Topics of the evening:

- **Defeating Advanced Adversaries - Leveraging the Kill Chain** (Erik Van Buggenhout): How advanced adversaries are attempting to penetrate the environment and how to stop or detect them. Examples of adversaries:
  - Shamoon - destructive attacks in middle East
  - Black Energy - lights out in Ukraine
  - Stuxnet - the world's first digital weapon
- **How I hacked your home - IoT security nightmares...** Cédric Bassem): review of key concepts design choices when devices are being connected to the Internet. As a key example, the use of a home security system to illustrate a number of things that could go wrong.

4 Class 9 - Application Security (Enterprise Architecture)

It is important to define the security architecture as early as possible. Because, this will reduce the complexity of communicating business with IT, and will help to reduce the operational risks. The architecture should be transparent, this means that users should not notice changes in the functionality.
Possible tools to approach the architecture -> TOGAF v9.1 (Open Group Architecture Framework)
When doing the architecture we need to think about the relationship among: Business Architecture <-> Information Systems Arch <-> Technology Architecture. As well as, the balance for risks and controls according to the business objectives.
Resource -> **SABSA** (SABSA matrix, slides in pag. 26)
Important to keep a business driven security mindset.

Software Developer Security

Aspects to consider in software development:

- Securing coding and testing
- The product and process quality
- Use design patterns and add security
- Analyze possible type of attacks
- Review the requirements between the planning and design phase
- Try penetration tests

Nowadays "Agile methodologies" are used for software development. “Agile software development describes a set of values and principles for software development under which requirements and solutions evolve through the collaborative effort of self-organizing cross-functional teams.[1] It advocates adaptive planning, evolutionary development, early delivery, and continuous improvement, and it encourages rapid and flexible response to change. These principles support the definition and continuing evolution of many software development methods.”[wikipedia]

"Waterfall methodologies" for software development were very common some years ago. It has the following structure:

- Requirements (definition of)
- Design
- Development/Implementation
- Testing
- Deployment

It is important to define the "security" for every step of the development:

- Development (think about the infrastructure security)
- Testing (how to secure the QA (Quality Assurance)?)
- Stage of QA (Penetration Test)
- Production (assure "high standard security")

Note: Don’t forget the "onion layers". The "data" (our precious asset) is in the very inner circle.

Another example in class: "Describe requirements for Online Banking and define the security it should have"

References

Agile software development, https://en.wikipedia.org/wiki/Agile_software_development [cited November 2017].