# *Managing Security Risks Inherent in the Use of* **Third-party Components**

Investigação do tópico de desenvolvimento seguro de software

**Grupo 5 -** Diogo Araújo A78485, Diogo Nogueira A78957 - **Engenharia de Segurança**

# ÍNDICE DE CONTEÚDO

TÓPICOS A ABORDAR AO LONGO DA APRESENTAÇÃO

# 1 THIRD-PARTY COMPONENTS

1.1. O QUE SÃO?
1.2. QUAL O SEU SIGNIFICADO NO PRODUTO FINAL?

# O QUE SÃO?

*"Third Party Components means, with respect to a product, all software that is embedded in, used in, incorporated into, combined with, linked with, distributed with, provided to any Person as a service with, provided via a network as a service or application with or made available with such product, in each case that is owned, in whole or in part, by a third party."*

Consiste num componente de terceiros produzido com o objetivo de ser reutilizável para que possa depois ser distribuído de forma livre ou até mesmo vendido posteriormente por uma entidade/empresa que não o fornecedor original do mesmo.

# QUAL O SEU SIGNIFICADO NO PRODUTO FINAL?

DE QUE FORMA OS TPC'S INFLUENCIAM O PRODUTO FINAL?

MENOR CICLO DE DESENVOLVIMENTO DE *SOFTWARE*

AVANÇO NO PROJETO FINAL

DIMINUIÇÃO CUSTOS A CURTO PRAZO

# QUAL O SEU SIGNIFICADO NO PRODUTO FINAL?

VANTAGENS E DESVANTAGENS DA INTEGRAÇÃO DE TPC'S

| Vantagens Longo Prazo | Desvantagens Longo Prazo |
|---|---|
| • Fornecem funcionalidades que são constantemente atualizadas e com os *standards* mais recentes;<br>• Possibilidade de manter o produto final mais seguro e funcional. | • Perda do controle total sobre os TPC's;<br>• TPC pode ser descontinuado;<br>• Necessidade de implementações alternativas;<br>• Atualizações Problemáticas com custos acrescidos;<br>• Impossibilidade de atualizar o TPC. |

# 2 DESAFIOS NO USO DE THIRD-PARTY COMPONENTS

2.1. CENÁRIO DA EMPRESA DO BOB
2.2. PROBLEMAS DO PRODUTO FINAL
2.3. RESPOSTA ÀS QUESTÕES LEVANTADAS PELA EMPRESA

# CENÁRIO DA EMPRESA DO BOB

QUAL O PROBLEMA DO PRODUTO FINAL?

PRODUTO INICIAL → INCLUSÃO/USO DE TPC'S → PRODUTO FINAL

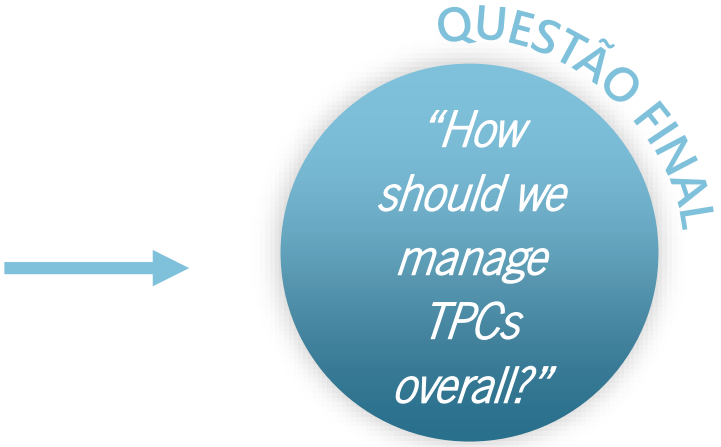
*Reports* Negativos relativamente ao produto final


Vulnerabilidade grave de segurança num *open source component*

# PROBLEMAS DO PRODUTO FINAL

QUAIS AS QUESTÕES QUE SURGEM PERANTE O PROBLEMA DESCOBERTO?

1. *"What Third-Party Components are included in my product?"*
   Quais os TPC's que se encontram incluídos no produto final?
2. *"Is the product affected by the CVE?"*
   O produto final é realmente afetado pelo CVE que está a afetar o produto?
3. *"What should we do to maintain the TPCs within our product?"*
   O que deve ser feito para manter esses TPC's não comprometendo o produto?
4. *"What TPCs should we use and what is the security risk associated with them?"*
   Quais os TPC's que devem ser usados e quais os seus riscos de segurança?



# RESPOSTA ÀS QUESTÕES LEVANTADAS

QUESTÃO 1 *"What Thirdy-Party Components are included in my product?"*

| Possível Resposta/Resolução | Possíveis Obstáculos |
|---|---|
| • Uso de uma *Bill of Materials* para obter uma lista dos componentes incluídos no produto. Facilita a localização de certos componentes afetados no caso de um produto final conciso. | • Uso de várias linguagens de programação distintas complica na interpretação dos TPC's incluídos;<br>• Vários nomes para o mesmo TPC;<br>• TPC pode ter subcomponentes. |

# RESPOSTA ÀS QUESTÕES LEVANTADAS

QUESTÃO 2 *"Is the product affected by the CVE?"*

| Possível Resposta/Resolução | Possíveis Obstáculos |
|---|---|
| • Determinar se o TPC afetado está incluído no produto final e se o produto em si está afetado pelo CVE em causa.<br><br>Não é incomum um TPC listado com CVE estar incluído no produto e o mesmo não se encontrar afetado. | • Dificuldade em interligar a lista de BOM com a lista de CVE's;<br>• Conteúdo do CVE pode não ser suficiente para determinar com 100% de certezas se o componente está afetado, podendo ser necessária uma análise mais profunda. |

# RESPOSTA ÀS QUESTÕES LEVANTADAS

QUESTÃO 3 *"What should we do to maintain the TPCs within our product?"*

| Possível Resposta/Resolução | Possíveis Obstáculos |
|---|---|
| • Selecionar os componentes que foram desenvolvidos já com a segurança em mente, não pensando apenas na sua funcionalidade futura. | • Tentar estabelecer uma relação minimamente ideal entre aquilo que é realmente necessário para o funcionamento do produto e o quão seguro o mesmo deve ser. |

# RESPOSTA ÀS QUESTÕES LEVANTADAS

QUESTÃO 4 *"What TPCs should we use and what is the security risk associated with them?"*

| Possível Resposta/Resolução | Possíveis Obstáculos |
|---|---|
| • Manter sempre uma ideia do risco de segurança a longo prazo de um determinado componente usado ou incorporado. | • Nem sempre é fácil responder às vulnerabilidades que vão surgindo, dado que é necessário manter um plano organizado daquilo que o produto realmente contém. |

# RESPOSTA ÀS QUESTÕES LEVANTADAS

## QUESTÃO PRINCIPAL DO PROBLEMA

*"How should we manage TPCs overall?"*

Afinal, como deve ser feita a gestão dos TPC's?

# 3 GESTÃO DE THIRD-PARTY COMPONENTS

3.1. *Software Development Life Cycle* (SDLC) vs *Third-party Components Life Cycle*

3.2. Ingredientes-chave para a Gestão de um TPC

# SDLC VS CICLO DE VIDA DE UM TPC

## DEFINIÇÃO DE AMBOS OS CICLOS

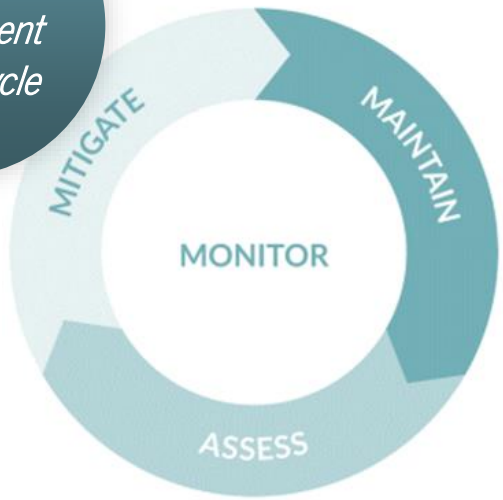
O processo de gestão que do ciclo de vida de um TPC deve começar o mais cedo possível no SDLC estando por isso integrado internamente no mesmo. Assim, as organizações devem definir e consequentemente adotar um modelo para controlar os riscos de segurança dos seus TPC's ao mesmo tempo que encaixam esse processo num SDLC já existente nas suas organizações.

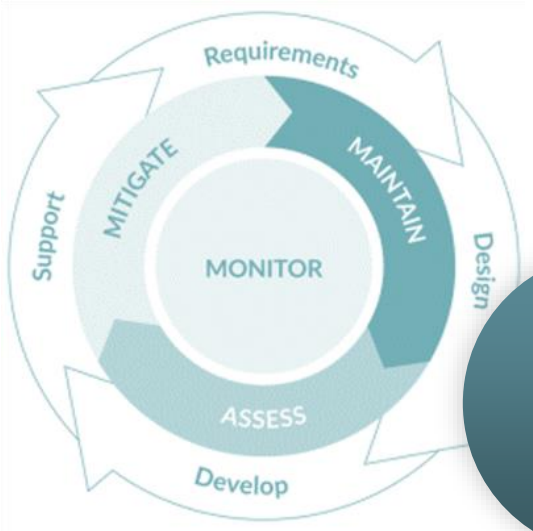
| *Software Development Life Cycle* | *Third-party Components Life Cycle* |
|---|---|
| Framework que define todo o processo usado pelas organizações de forma a desenvolver uma aplicação do início até ao fim da vida. | Conjunto de etapas *high level* essenciais definir para o ciclo de vida de um TPC. |

# SDLC VS CICLO DE VIDA DE UM TPC

ETAPAS DE AMBOS OS CICLOS

*Third-party Components Life Cycle*

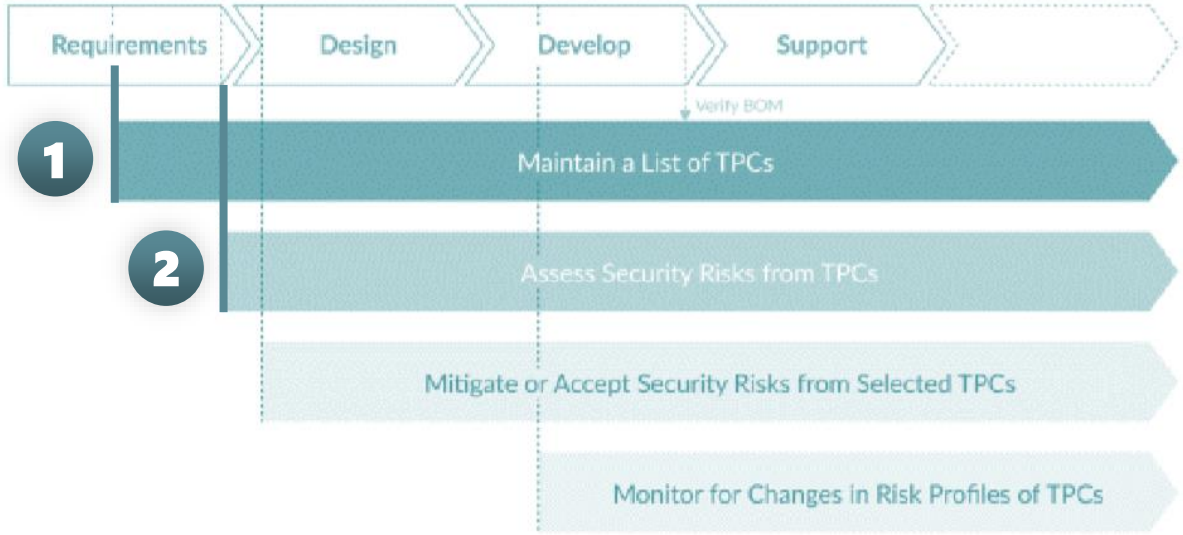



*Software Development Life Cycle*

# SDLC VS CICLO DE VIDA DE UM TPC

## RELAÇÃO EXISTENTE ENTRE AMBOS OS CICLOS

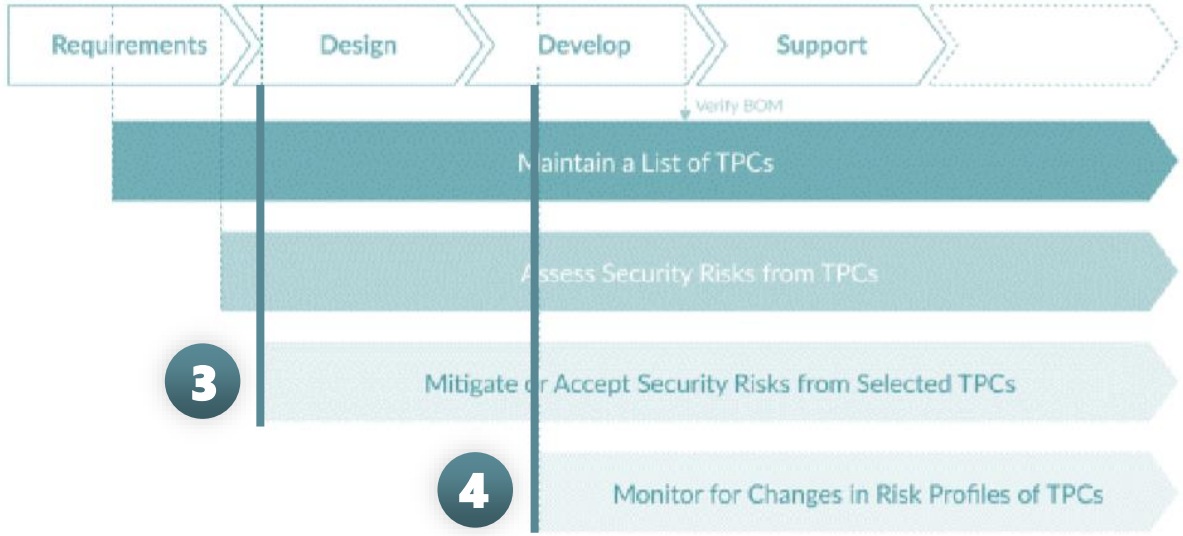



**1** Listar os TPC's no decorrer da fase *Requirements*. Os requisitos funcionais serão importantes para ditar o uso de TPC's específicos.

**2** A posterior avaliação de riscos inicia-se assim que um TPC candidato é identificado pela listagem anterior.

# SDLC VS CICLO DE VIDA DE UM TPC

## RELAÇÃO EXISTENTE ENTRE AMBOS OS CICLOS

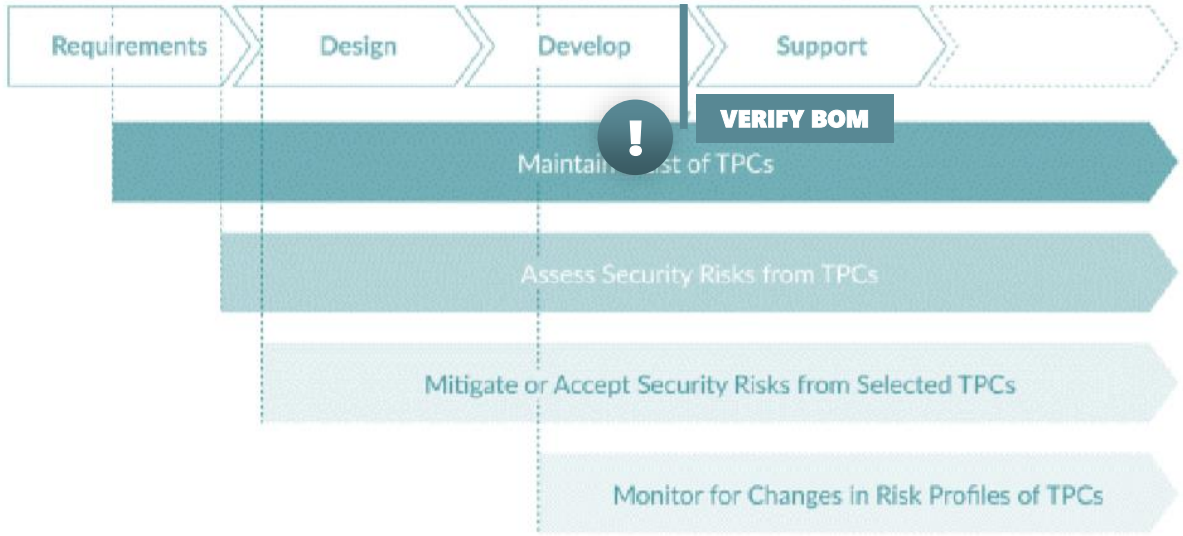



**3** A mitigação/aceitação destes riscos vai desde a fase de *Design* até ao fim do SDLC, dado que é necessário salvaguardar certos níveis de código para conseguir mitigar tais riscos.

**4** A monitorização visa acompanhar continuamente o TPC depois de tudo feito. Por isso deve ser feita na parte de *Develop* e antes de enviar o produto final.

# SDLC VS CICLO DE VIDA DE UM TPC

## RELAÇÃO EXISTENTE ENTRE AMBOS OS CICLOS




! Verificação da BOM feita antes da fase de *Support*, dado que pode ser preciso acionar uma nova avaliação de riscos antes de enviar o produto caso existam novos TPC's ou atualizações sobre os mesmos.

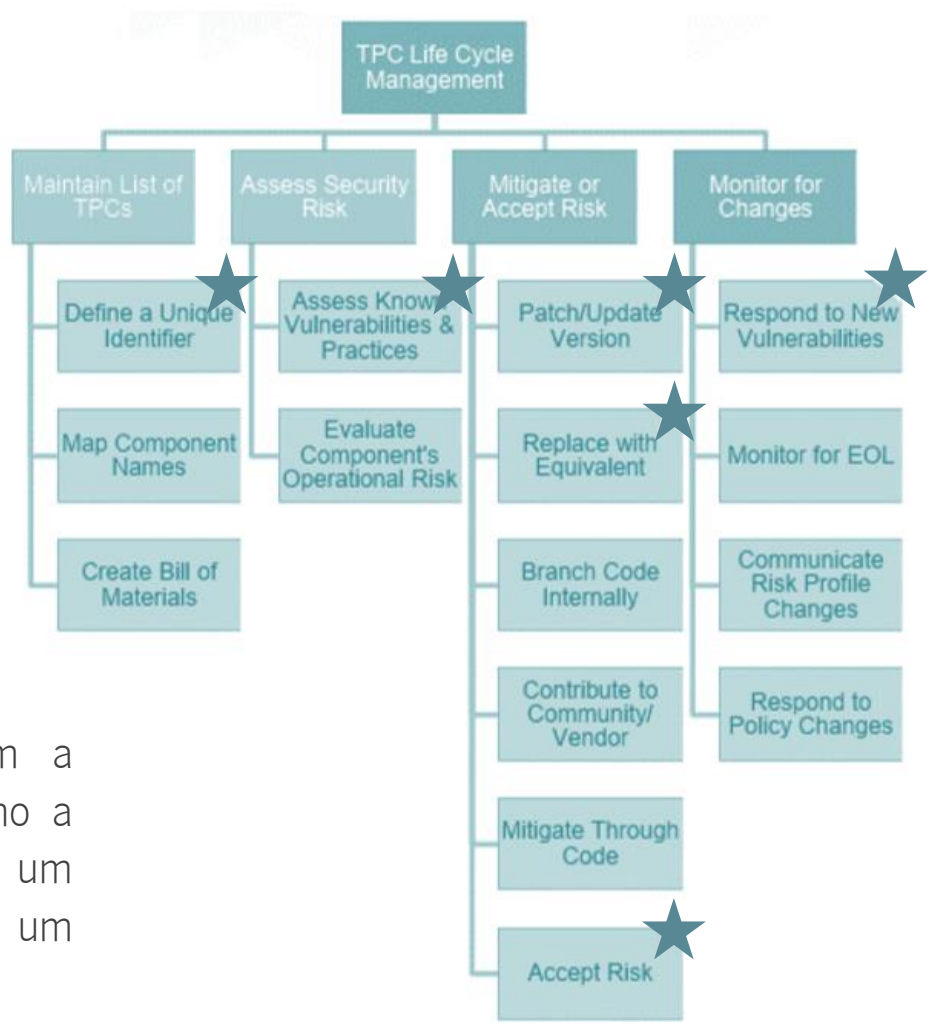
# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

## ETAPAS DE AMBOS OS CICLOS

Conforme temos visto ao longo de toda esta apresentação os quatro maiores passos/fases, ou seja, *Maintain*, *Assess*, *Mitigate* e *Monitor* têm uns ingredientes-chave para serem considerados como uma excelente gestão do ciclo de vida dum TPC.

Existem uma panóplia de ingredientes para as quatro fases essenciais para a gestão do ciclo de vida dos TPC. Alguns destes ingredientes são considerados o mínimo para cada fase de forma a existir um controlo essencial e basilar dos TPC.

# CAPÍTULO 3. GESTÃO DE *THIRD-PARTY COMPONENTS*
## 3.1. INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S




★ Os ingredientes assinalados com a estrela são considerados o mínimo a cada fase de forma a existir um controlo basilar dos TPCs de um produto

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*MAINTAIN* Manter uma lista de TPC's
COMO FAZÊ-LO?

- Identificadores únicos para os TPC's para ajudar na criação da BOM;
- Mapeamento correto dos TPC's e seus identificadores com toda a informação útil sobre cada TPC e suas possíveis vulnerabilidades, *patches*, etc.;
- Criação da BOM utilizando a abordagem mais correta para a empresa e os seus *softwares* com TPC.

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*MAINTAIN* Manter uma lista de TPC's

MÉTODOS PARA CRIAR E MANTER UMA LISTA DE TPC'S

| Método | Prós | Contras |
|---|---|---|
| Lista manual dos TPC's | Gratuito | Pouca certidão, principalmente nas sub-dependências |
| Ferramentas Automáticas | Muito preciso e eficiente | Preço e disponibilidade de múltiplas ferramentas para diferentes linguagens |
| Combinação das duas | Combinação dos prós falados acima | Combinação dos contras falados acima |

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*ASSESS* Avaliar o risco de segurança
COMO FAZÊ-LO?

- Avaliar as vulnerabilidades conhecidas e as práticas usadas;
- Detetar as vulnerabilidades ainda por remediar relativamente à ultima versão (ou a em uso) verificando se existem modos já disponíveis para mitigar as mesmas;
- Avaliar o risco operacional do componente;
- Manter o TPC estável e atualizado através de manutenções *short* ou *long term.*

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*MITIGATE* Mitigar ou aceitar o risco de segurança
COMO FAZÊ-LO?

- **Etapa mais importante e de maior risco;**
- Etapa onde são encontradas as vulnerabilidades nos TPC usados no(s) produto(s);
- Tomar uma medida de forma proporcional à gravidade do CVE em causa;
- Atualizar sempre o TPC pensando no trabalho adicional;
- Substituir um TPC que seja equivalente.

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*MITIGATE* Mitigar ou aceitar o risco de segurança
COMO FAZÊ-LO?

- Criar um novo *Branch* internamente e responsabilizar por remediar a vulnerabilidade;
- Mitigar através de novo código por cima de forma a atenuar o problema encontrado;
- Aceitar o risco envolvido quando o TPC está numa porção do produto que a equipa não está a usar ativamente ou então a vulnerabilidade tem um *score* baixo de perigo.

# INGREDIENTES-CHAVE PARA A GESTÃO DE TPC'S

*MONITOR* Monitorizar mudanças aos TPC's
COMO FAZÊ-LO?

- Responder a novas vulnerabilidades colocadas nas bases de dados de segurança;
- Monitorizar a *End of Life* dos TPC e verificar se o componente foi descontinuado ou apenas a versão usada;
- Comunicar as mudanças sobre o perfil de risco da empresa quando é descoberta uma vulnerabilidade na sua porção do produto à medida que se vai mitigando a mesma;
- Responder às mudanças de politica de segurança sobre os TPC.

# 4 PESQUISA ADICIONAL E CONSIDERAÇÕES FUTURAS

4.1. Pesquisa de outros artigos/fontes que abordam a área dos TPC's

4.2. Considerações a ter em conta para futuro trabalho

# PESQUISA ADICIONAL

LISTAGEM DE OUTROS ARTIGOS BIBLIOGRÁFICOS RELACIONADOS COM A ÁREA

| Título do Artigo | Autores e Ano do Artigo | Descrição do Artigo |
|---|---|---|
| *A New Detection Method for Stack Overflow Vulnerability Based on Component Binary Code for Third-Party Component* | (2018) | *"This paper designs a stack buffer overflow vulnerability algorithm SBOD for the COM component and implements a prototype system of detecting stack buffer overflow vulnerability."* |
| *Shipboard ECDIS Cyber Security: Third-Party Component Threats* | Boris Svilicic; Igor Rudan; Vlado Frančić; Mateo Doričić (2019) | *"The analysis of the cybers security is based on the cyber security testing of the shipboard ECDIS using an industry vulnerability scanner."* |

# PESQUISA ADICIONAL

LISTAGEM DE UM ARTIGO ONLINE RELACIONADO COM A ÁREA

| Título do Website | Link e Ano do Artigo | Descrição do Artigo |
|---|---|---|
| *The Hidden Risk in All IoT Devices: Third-Party Components* | (2018) | *"Creating and designing secure IoT devices is a mission of high complexity. IoT devices often contain many functionalities that require using third-party software components within the device. In this article we will review the hidden risks in third-party components – as these components are highly critical in IoT devices, and affect device security greatly."* |

# PESQUISA ADICIONAL

## PEQUENA ANÁLISE DO ARTIGO ONLINE, TÓPICOS QUE ABORDA

- **Para que são utilizados os TPC's?** – Neste ponto referem a importância em se usar TPC's em dispositivos IoT;
- **Porque é que os TPC's são realmente importantes?** – Fala-se sobre a importância dos TPC's e a sua ligação interna com o dispositivo IoT em si;
- **Ferramentas de segurança existentes são insuficientes –** Deixa-se evidente que as ferramentas de segurança comuns necessitam de uma solução altamente eficaz para proteger os TPC's;
- **Conhecimento por parte dos atacantes** – Ideia de que os atacantes têm um conhecimento prévio sobre os TPC's;
- **Importância de manter o dispositivo atualizado** – O uso de TPC's pode dificultar a ideia de manter um dispositivo o mais atualizado possível.

# PESQUISA ADICIONAL

## PEQUENA ANÁLISE DO ARTIGO ONLINE, CONCLUSÕES

- Uso de TPC's em dispositivos IoT é inevitável;
- Necessidade de um cuidado extra em termos de segurança;
- Necessidade de mitigar ao máximo o risco de vulnerabilidade desses componentes;
- Tema atual e para o qual as medidas falas continuam a ser necessárias.

# CONSIDERAÇÕES FUTURAS

## IDEIAS/MÉTODOS A TER EM CONTA PARA TRABALHO FUTURO

| Método | Descrição do Método |
|---|---|
| *Crowdsourcing of Naming and Name Mapping* | A ideia seria aprofundar a ideia de *unique names* abordada no subcapítulo ***Maintain*** - Manter uma lista de TPC's, onde as organizações atribuíssem um nome exclusivo para um dado TPC. O grande problema a lidar seria a grande escala de TPC's, pelo que se esperaria que as organizações colaborassem entre si de forma a conseguir criar um sistema de nomenclatura padrão. |
| *Crowdsourcing of an End-of-life Repository* | Criar uma *database* que providencie as datas de quando um TPC vai atingir o seu fim de ciclo de vida, ou seja, quando deixará de ser suportado pelo seu vendedor e quando as vulnerabilidades de segurança associadas a si estão sem qualquer tipo de resolução existente. |
| *Crowdsourcing of a Vulnerability Source Listing* | Criar uma lista das vulnerabilidades dos TPC's através das organizações em si como dos próprios fornecedores. |