Proposal of an innovative method to implement, measure, and validate the security level of a system based on system modelling.

Georges El Hajal

université de BORDEAUX

THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ de BORDEAUX

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE L'INGÉNIEUR

par

Georges EL HAJAL

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : PRODUCTIQUE

Proposal of an Innovative Method to Implement, Measure, and Validate the Security Level of a System based on System Modelling

---

Soutenue le 06 Décembre 2022

Après avis de :

Charles YAACOUB,	Professeur à l'Université Saint Esprit de Kaslik
Richard CHBEIR,	Professeur à l'Université de Pau et des Pays de l'Adour

Devant la Commission d'examen formée de :

Charles YAACOUB,	Professeur à l'Université Saint Esprit de Kaslik	*Rapporteurs*
Richard CHBEIR,	Professeur à l'Université de Pau et des Pays de l'Adour	
Mamadou Kaba TRAORE,	Professeur à l'Université de Bordeaux	*Président du jury*
Josef BOERCSOEK,	Professeur à Kassel University	*Examinateurs*
Jinane SAYAH,	Professeur à l'Université de Balamand	
Yves DUCQ,	Professeur à l'Université de Bordeaux	*Directeur de thèse*
Roy ABI ZEID DAOU,	Professeur associé à la Lebanese German University	*Co-directeur*

Abstract

The security of systems and networks has been the IT administrators' major issue. The frequency of the attacks is increasing and it is affecting the confidentiality, integrity, and availability of the data. Huge losses of resources and data were lost because of these attacks. For example, in 2017, the WannaCry ransomware paralyzed several medical entities in the United Kingdom. Cyber incidents have increased more after the COVID-19 pandemic because the attack surface has expanded due to the fact that most enterprises were pushed into the path of digital transformation.

The work in this thesis focused on creating a cyber security solution with the ability to protect, identify, classify, and manage risk priority with time efficiency. The main idea is to handle the threats and vulnerabilities based on the separation of the information system into three levels: Network level, device level, and human factor level. For each level, the system was modeled and a solution was presented and tested. And at the end, all of the solutions were applied to an information system to verify its usability and effectiveness.

To determine the solutions, a bibliographic study was made so we can specify what contribution can be offered. And the results were crossed with the idea of the protection for the three levels mentioned previously. From the study, we deduced that there was not a single work that tried to present a solution to protect an information system for the three levels. Most of the work was focused on a certain level only. Thus, our work in the thesis managed to create a solution for the three levels.

For the network level, a unidirectional network device A.K.A. data diode was used. Data diodes are a paradigm of cyber security which have not been studied extensively even though they can be used to overcome some of the limitations that exist today with current approaches to cyber security such as basic firewalls, and intrusion detection systems. The novelty of the work consists of on presenting a cost-effective solution with off-the-shelf components. We developed special software to use this data diode and we demonstrated its effectiveness especially in protecting the highly sensitive data due to its physical nature. Furthermore, we managed to demonstrate how its presence didn't affect the data flow yet provided the utmost security.

For the device level, we worked on protecting the flow of data. Of course, Artificial Intelligence (AI) algorithms and machine learning became an important pillar in the "Cyber security revolution". These technologies have already become part of everyday life and are being used by organizations for various purposes such as predictive maintenance, fraud detection etc. So, we collected a dataset of 54.000 packets from software that sends medical metric data to a server and used them inside Matlab R2018a. The dataset contained 35% cyber-attack traffic, 38% medical issue traffic, and 27 % normal traffic. After using 22

machine learning algorithms, K-Nearest Neighbor (KNN) algorithm, provided the most accuracy. Using this revelation, a python script created a model using the KNN algorithm and that model was used in the medical software to the label on the fly the received data.

Furthermore, to better protect the device level, we have developed software for managing cyber threats which includes methods and tools for risk assessment and attack classification. Several classifications of threats have been developed but the most renowned one is the CVSS (Common Vulnerability Scoring System). However, the scoring of the vulnerabilities is not unique. As an example, CVSS v2 has assigned 8330 threats with the highest score of 10 whereas, in CVSS v3, 193 threats have a score of 10 until the end of 2020. So, choosing which vulnerability must be remediated first will yield to a decision-taking problem. Additionally, manual prioritization can be achieved in small networks where the number of threats is limited, but, in large networks automation is a must to help security officers to take the right decisions. Consequently, our developed software will collect inputs from NVD (National Vulnerability Database), the Exploit Database (EDB), and Computer Incident Response Center Luxembourg (CIRCL) to apply them onto a weighted mathematical formula in order to provide a new priority list for all known vulnerabilities. All what the security officer has to do is cross reference it with the spotted vulnerabilities to have a priority list that he can follow for remediating his system.

As for the human level, cyber threat comes from two causes: ignorance or malignancy. We focused on the ignorance aspect to create a human firewall through cyber awareness. An AI-based conversational bot was created that provides information related to the company policies and procedures, information about cyber security and a test to evaluate the level of cyber awareness. This bot is used to make the interaction with the user appealing and friendly through the use of WhatsApp as a way of communication. The implementation and simplicity of the interaction with the bot were tested and evaluated.

Keywords: System Modeling; Secured network; Cyber threat; Cyber classification; Cyber risk management; Automation in Cyber security; Risk Assessment; Cyber Awareness; Human Firewall; Artificial Intelligence; Machine Learning;

Résumé

La sécurité des systèmes et des réseaux a été l'enjeu majeur des administrateurs informatiques. La fréquence des attaques augmente et affecte la confidentialité, l'intégrité et la disponibilité des données. D'énormes quantités de données ont été perdues à cause de ces attaques. Par exemple, en 2017, le rançongiciel WannaCry a paralysé plusieurs entités médicales au Royaume-Uni. Les cyberincidents ont augmenté davantage après la pandémie de COVID-19, car la surface d'attaque s'est étendue du fait que la plupart des entreprises ont été poussées sur la voie de la transformation numérique.

Le travail de cette thèse s'est concentré sur la création d'une solution de cybersécurité capable de protéger, d'identifier, de classer, de gérer la priorité des risques avec une efficacité temporelle. L'idée principale est de gérer les menaces et les vulnérabilités en se basant sur la séparation du système d'information en trois niveaux : niveau réseau, niveau appareil et niveau facteur humain. Pour chaque niveau, le système a été modélisé et une solution a été présentée et testée. Et finalement, l'ensemble des solutions a été appliqué sur un système d'information pour en vérifier l'utilisabilité et l'efficacité.

Travailler sur la cybersécurité nécessite la connaissance des menaces, des vulnérabilités, des acteurs de la menace et des contre-mesures connues. Une étude bibliographique a donc été réalisée pour approfondir nos connaissances sur celles-ci.

Une menace est un événement ou une condition qui entraîne la perte d'un actif et les conséquences ou l'impact indésirables de cette perte. La base de la perte d'actifs comprend toutes les formes d'événements intentionnels, non intentionnels, accidentels, de mauvaise utilisation, d'abus, d'erreur, de faiblesse, de défaut, de faute et/ou de défaillance et les conditions associées. De plus, une menace peut faire référence à des entités qui tentent d'accéder sans autorisation aux actifs de l'organisation à l'aide d'une communication de données. Cet accès peut être initialisé depuis l'intérieur d'une organisation par des utilisateurs de confiance ou depuis des emplacements distants par des entités inconnues utilisant Internet. Kaspersky classe les cybermenaces en trois catégories : la cybercriminalité qui est un acte qui cible les Systèmes de Technologies de l'Information (STI) à des fins financières ou qui provoquent des perturbations, la cyberattaque qui est un acte qui implique la collecte d'informations et le cyberterrorisme qui est un acte qui vise à déstabiliser les systèmes pour semer la panique ou la peur.

Les menaces de cybersécurité les plus courantes sont les logiciels malveillants, les virus, les chevaux de Troie, les logiciels espions, les rançongiciels, les logiciels publicitaires, les rootkits, les vers, les botnets, l'injection SQL, le phishing, les attaques de type Man-in-the-middle, le déni de service, le Cryptojacking.

Les vulnérabilités sont des faiblesses dans un système d'information, dans des procédures de sécurité du système, dans des contrôles internes ou dans une mise en œuvre qui pourraient être exploitées ou déclenchées par une source de menace. Les failles, les fonctionnalités ou les erreurs des utilisateurs sont souvent à l'origine de vulnérabilités.

Un acteur menaçant est une personne ou un groupe qui participe à une action ou à un processus hostile à l'aide d'ordinateurs, d'appareils, de systèmes ou de réseaux. Les acteurs de la menace sont classés en quatre groupes en fonction de leurs motivations et de leurs affiliations : les cybercriminels, les acteurs de l'État-nation, les hacktivistes, les organisations terroristes.

Les contre-mesures de sécurité sont les contrôles utilisés pour protéger la confidentialité, l'intégrité et la disponibilité des données et des systèmes d'information. Il existe un large éventail de contrôles de sécurité disponibles à chaque couche de la pile. La sécurité globale peut être grandement améliorée en ajoutant des mesures de sécurité supplémentaires, en supprimant les services inutiles, en renforçant les systèmes et en limitant l'accès. Certaines des méthodes de protection les plus utilisées sont énumérées ci-après (sans tenir compte de leur ordre d'importance) : système de détection d'intrusion (IDS), système de prévention d'intrusion (IPS), pares-feux, segmentation de réseau / Data Diode, Cryptage réseau, Antivirus, ...

Après avoir acquis des connaissances sur le domaine de la cybersécurité, la recherche s'est poursuivie sur 18 articles choisis parmi des centaines d'articles et d'articles lus pour avoir un aperçu du travail effectué dans notre domaine connexe. Ces articles nous ont guidé dans les conclusions décrites ci-après.

Pour mener à bien notre travail, nous devions connaître la partie modélisation en cyber sécurité. La modélisation des menaces est un moyen de planifier et d'optimiser les opérations de sécurité. Les équipes de sécurité définissent leurs objectifs, identifient les vulnérabilités et définissent des plans de défense pour prévenir et remédier aux menaces de cybersécurité. Il s'agit d'une approche structurée pour identifier, quantifier et traiter les menaces. Elle permet au personnel de sécurité du système de communiquer les dommages potentiels des failles de sécurité et de hiérarchiser les efforts de correction. Penser aux exigences de sécurité avec la modélisation des menaces peut conduire à des décisions architecturales proactives qui permettent de réduire les menaces dès le départ. La modélisation des menaces peut être définie comme un ensemble d'attentes concernant les actions malveillantes contre l'organisation. Nous avons étudié les méthodologies de classification de modélisation des menaces les plus utilisées appliquées dans le domaine du génie logiciel. Nous avons trouvé les éléments suivants : arbres d'attaque/menace, STRIDE, DREAD, diagrammes de flux de données, ATT&CK. Nous avons aussi déterminé les outils de modélisation des menaces les plus utilisés étaient : IriusRisk, l'outil gratuit de modélisation des menaces de Microsoft, ThreatModeler, PyTM, SecuriCAD, Tutamantic

"Automated Design Analysis", OWASP Threat Dragon Project, Mozilla SeaSponge, Draw.io pour la modélisation des menaces.

De plus, mesurer la performance d'une cybersécurité est essentiel pour améliorer son efficacité résilience. Des KPI bien pensés seront différents dans chaque organisation. Ils font ressortir les problèmes et les anomalies de manière exploitable et axée sur les solutions et peuvent donc aider à transformer une quantité significative mais abondante de données de sécurité en informations plus rapides à digérer pour la haute direction. Le suivi d'un trop grand nombre d'indicateurs de performance clés peut devenir un fardeau à la fois pour l'analyste et les décideurs qui doivent faire face à une surcharge de données et d'informations. Lors de la détermination des KPI qui doivent rester en haut de la liste, il faut vérifier s'ils cochent toujours les cases des critères SMART : Simple, Mesurable, Actionnable, Pertinent et Basé sur le temps.

Les KPI doivent refléter les priorités, les buts et les objectifs de chaque organisation, mais certains exemples de KPI de cybersécurité applicables à la plupart des organisations peuvent inspirer des idées lorsque l'administrateur informatique rédige les propres KPI de l'institution, tels que : le nombre d'appareils surveillés, le nombre total d'événements, nombre d'événements par périphérique ou hôte, temps moyen de détection, temps moyen de résolution.

Aussi de nombreuses normes et standards régissent la sécurité d'un système d'information. L'ISO / IEC 27001 est une norme officielle pour la sécurité de l'information des organisations. La norme ISO 27001 se concentre sur l'objectif de niveau supérieur consistant à s'assurer que les organisations disposent d'une structure (appelée système de gestion en langage ISO) qui garantit que l'organisation améliore la sécurité de l'information. Il se compose d'objectifs, de ressources, de politiques et de descriptions de processus.

Le cadre NIST est un guide volontaire, basé sur les normes, directives et pratiques existantes pour les organisations afin de mieux gérer et réduire les risques de cybersécurité. Le cadre de cybersécurité se compose de trois composants principaux : le noyau, les niveaux de mise en œuvre et les profils.

La norme de sécurité des données de l'industrie des cartes de paiement (PCi DSS) est un ensemble d'exigences visant à garantir que toutes les entreprises qui traitent, stockent ou transmettent des informations de carte de crédit maintiennent un environnement sécurisé. Il a été lancé le 7 septembre 2006 pour gérer les normes de sécurité PCI et améliorer la sécurité des comptes tout au long du processus de transaction. Organisme indépendant créé par Visa, MasterCard, American Express, Discover et JCB, le PCI Security Standards Council (PCi SSC) administre et gère le PCi DSS.

La détection des menaces consiste à analyser l'ensemble d'un écosystème de sécurité pour identifier toute activité malveillante susceptible de compromettre le réseau. Si une

menace est détectée, des efforts d'atténuation doivent être déployés pour neutraliser correctement la menace avant qu'elle ne puisse exploiter les vulnérabilités actuelles. Un programme de détection des menaces robuste doit utiliser : la technologie de détection des menaces d'événements de sécurité, la technologie de détection des menaces réseau, la technologie de détection des menaces des terminaux.

En utilisant une combinaison de ces méthodes défensives, l'administrateur informatique augmentera les chances de détecter et d'atténuer une menace rapidement et efficacement. La sécurité est un processus continu et rien n'est garanti. Il appartiendra à l'administrateur système, aux ressources déployées et aux processus mis en place de maintenir l'entreprise aussi sécurisée que possible.

De l'analyse de l'état de l'art présenté précédemment, on remarque l'absence d'ouvrages publiés traitant de la sécurité de l'ensemble du système. Ainsi, la nouveauté de ce travail est la décomposition de la solution aux trois niveaux (réseau, humain et appareil) et de les présenter en une seule solution cohérente. Le travail présentera des solutions par niveau, puis démontrera comment elles peuvent être utilisées ensemble pour fournir aux organisations une protection globale contre la plupart des menaces qui peuvent être décrites dans le top 10 de l'OWASP. Plus en détail, la solution proposée abordera trois problèmes principaux :

- Définir une solution physique en intégrant la data diode pour protéger la couche réseau ;

- Définir une solution logicielle en mettant à jour automatiquement la liste des principales vulnérabilités grâce à un processus de calcul actif ; ce nouveau processus de calcul vise à hiérarchiser les vulnérabilités qui doivent être traitées dans un premier temps afin d'augmenter la sûreté et la sécurité du système ;

- Définir une solution logicielle pour renforcer le facteur humain dans une organisation en fournissant un module d'auto-apprentissage et d'autotest simple et efficace pour mettre à jour la sensibilisation de l'employé à la cybersécurité.

La méthodologie mise en avant dans ce travail de recherche consiste à définir les principales menaces aux trois niveaux (c'est-à-dire réseau, appareil et humain), à modéliser leurs sources et à proposer des solutions techniques. De plus, l'objectif principal repose sur le développement d'une proposition sécurisée complète qui ne nécessite pas d'énormes ressources ni pour sa mise en œuvre ni pour son application.

La première étape de ce travail consiste à montrer la bonne solution pour chaque niveau. Cette solution sera testée seule pour démontrer son efficacité et mesurer quelques indicateurs de performance afin de la valider ou d'apporter des modifications pour l'améliorer.

Une fois la sécurité à chaque niveau confirmée, une solution complète sera proposée et testée sur un réseau où plusieurs types de menaces et de vulnérabilités seront injectées. Pour valider ce système, une modélisation sera d'abord présentée puis quelques mesures, notamment le temps de réponse et le niveau de sécurité, seront réalisées.

A la fin, des comparaisons techniques entre la méthode proposée et les solutions existantes seront réalisées. Une telle comparaison aidera à montrer la puissance du système proposé et la possibilité de réduire l'intervention humaine chaque fois que cette application sécurisée est en cours d'exécution.

Au chapitre trois, la couche réseau a été traitée. La couche réseau représente les appareils avec leurs périphériques (au sein du réseau lui-même et avec d'autres réseaux). C'est le premier vecteur d'attaque où des menaces peuvent être injectées dans le système. Ainsi, la principale contribution de ce chapitre est de définir les menaces essentielles, leur origine et la manière de sécuriser les données contre le vol, l'altération ou l'injection. En fonction de ces menaces, des dispositifs matériels et des configurations seront proposés afin d'assurer la sécurité maximale du réseau vis-à-vis des interfaces externes.

La sécurité du réseau est une préoccupation majeure pour les parties prenantes qui ont des données sensibles stockées ou transmises entre les appareils. L'incapacité à concevoir des réseaux sécurisés entraîne la création d'une brèche qui peut permettre aux pirates de s'infiltrer dans le réseau et de mettre en péril l'intégrité des données stockées. La nature du préjudice peut être la destruction, la modification ou l'exposition de données. Cela oblige toutes les entreprises à investir dans la sécurisation de leurs réseaux, principalement lorsqu'elles ont des terminaux exposés à Internet, afin de minimiser le risque de perte de données.

Les principales menaces rencontrées au cours de la dernière décennie sur la couche réseau sont causées par les menaces persistantes avancées (APT) qui appliquent une grande variété d'attaques telles que l'attaque par déni de service distribué (DDoS) qui perturbe le trafic normal du services sur le serveur cible en le submergeant de trafic de déchets, ou en exécutant un empoisonnement du cache du système de noms de domaine (DNS) ou un empoisonnement du protocole de résolution d'adresse (ARP) pour introduire des acteurs malveillants sur le réseau et aider à l'exfiltration des données. Une attaque de cybersécurité typique consiste en un attaquant essayant d'entrer dans un réseau aussi rapidement que possible, de voler les informations et de sortir. Cependant, l'objectif d'APT est de parvenir à une pénétration et à un accès continu aux systèmes qu'il a violés.

Quant aux solutions appliquées, elles reposent sur des logiciels et du matériel. Les pares-feux étaient dans le passé la principale défense contre les attaques de réseau, mais leurs fonctionnalités n'empêchent pas l'attaquant d'exfiltrer des données une fois qu'un serveur est compromis car il empêche le trafic entrant mais ne peut pas bloquer le trafic

sortant. Les pares-feux de nouvelle génération sont désormais utilisés comme dispositif de sécurité réseau car ils offrent des fonctionnalités supplémentaires par rapport au pare-feu traditionnel car ils sont plus sensibles aux applications et contiennent des modules de prévention des intrusions et de renseignement sur les menaces. Néanmoins, les solutions les plus performantes sont les Systèmes de Détection d'Intrusion (IDS). IDS est un système de sécurité de détection d'intrusion pour les actions malveillantes que les intrus utilisent pour compromettre le système. En appliquant des règles, il est possible d'arrêter et même de bloquer l'intrusion d'une ressource particulière. L'activité de base de l'IDS consiste à surveiller les paquets sur le réseau et le comportement du système, puis à afficher une alerte lorsqu'une activité anormale se produit sur le réseau ou l'hôte. Ils peuvent être classés en deux classes : basés sur l'hôte (HIDS) et basés sur le réseau (NIDS).

Deux entités majeures affectent la sécurité du réseau : le logiciel en cours d'exécution sur les périphériques réseaux connectés et les périphériques eux-mêmes. Plusieurs modèles ont été créés pour définir les différentes étapes d'une cyberattaque ; Le modèle de chaîne de destruction cybernétique de Lockheed-Martin est l'un des plus utilisés par les chercheurs en cybersécurité. La Cyber Kill Chain crée un format et un langage communs permettant au personnel de cybersécurité d'évaluer les événements de sécurité par association, motivation et intégration, où ils peuvent être agrégés et corrélés en fonction de l'objectif et du vecteur d'attaque. Ainsi, les solutions discutées ne peuvent pas totalement protéger l'organisation si elles ne parviennent pas à détecter la menace à un stade précoce. Ainsi, la possibilité qu'un malware persiste est toujours possible, et sa communication avec l'attaquant permettra l'extraction des données.

Telles que présentées, les solutions peuvent apporter de la sécurité mais dans une certaine mesure. Cela n'est pas acceptable dans certains environnements où la confidentialité, l'intégrité et la disponibilité des actifs et des données sont essentielles. Les entreprises ayant de telles préoccupations s'appuient sur l'espace d'air (Air Gapped) de leur réseau pour l'isoler des connexions externes. Le concept de réseau (Air Gapped) consiste à créer un réseau sans communication externe physique. Cependant, les exigences modernes d'aujourd'hui rendent ces réseaux isolés non pratiques étant donné le besoin de flux de données provenant de l'extérieur des réseaux insulaires. Ainsi, ces îlots de réseau peuvent être sécurisés, mais ils reposent sur des données extérieures, ce qui signifie que les données sont souvent transférées entre les réseaux à l'aide de supports difficiles à contrôler tels que des clés USB ou des DVD. Ces médias ne sont pas sans risque car ils peuvent être utilisés pour divulguer des données ou même pour infiltrer des logiciels malveillants tels que l'incident Stuxnet.

Pour mieux comprendre les menaces et les solutions, la méthodologie de modélisation de processus graphique Integrated DEFinition (IDEF0) a été utilisée pour modéliser ce niveau. Les variables de contrôle pointeront du côté supérieur tandis que les

ressources pointeront du côté inférieur. Les menaces pointeront en diagonale depuis le coin supérieur gauche tandis que les solutions de ces menaces pointeront depuis le coin inférieur gauche. Les variables d'entrée seront du côté gauche tandis que les variables de sortie pointeront vers l'extérieur du côté droit. Le modèle est facile à comprendre. De même, les besoins et les opportunités d'amélioration sont révélés clairement.

Après avoir présenté les différentes solutions disponibles concernant la sécurité de la couche réseau, notre solution aborde la résolution des principaux problèmes rencontrés dans ces solutions. La solution proposée est divisée en deux parties : matérielle et logicielle. La première repose sur l'utilisation d'une data diode tandis que le second est un logiciel personnalisé qui gère le trafic d'un côté à l'autre de la diode de données.

La data diode offre les avantages d'un réseau isolé tout en assurant la connectivité à partir du réseau externe. L'introduction d'une pile de sécurité pour les périmètres du réseau augmente la complexité de la gestion, de la maintenance et, parfois, cela peut être une source de vulnérabilité et de faiblesse. Ainsi, la data diode est la meilleure solution possible pour augmenter la sécurité du réseau en créant un flux réseau unidirectionnel et en interdisant ainsi à un logiciel malveillant de communiquer avec l'attaquant et en protégeant ainsi le réseau contre une exploitation ultérieure, le vol de données, l'altération et l'exposition.

La data diode est la meilleure solution pour protéger les secrets et les actifs. Elle protège les données classifiées du gouvernement et la propriété intellectuelle du secteur privé contre les adversaires. De plus, elle protège les actifs électroniques des cyberattaques par le biais du réseau. Une data diode peut assurer la résilience et éliminer les risques mentionnés précédemment.

La data diode fournit un mécanisme physique pour imposer une communication unidirectionnelle stricte entre deux réseaux. Elle ne peut envoyer des informations que d'un réseau dit réseau « bas » vers un autre réseau dit réseau « haut ». Le réseau haut contient souvent des données avec un niveau de classification plus élevé que le réseau bas. Ils sont souvent mis en œuvre en supprimant le composant de transmission d'un côté et le composant de réception de l'autre côté d'un système de communication bidirectionnel. Cependant, cette méthode nécessite qu'une troisième entité fournisse simplement un signal porteur à l'émetteur qui ne fonctionnera pas s'il ne reçoit pas le signal porteur approprié.

Cependant, la nouveauté de ce travail consiste à créer la diode de données avec seulement deux entités : un émetteur et un récepteur. Le signal porteur nécessaire à l'émetteur est fourni par l'émetteur lui-même, éliminant ainsi le besoin d'une troisième entité.

Le concept principal derrière la data diode est la séparation physique de la connectivité entre le réseau non sécurisé et le réseau sécurisé via une connexion réseau unidirectionnelle.

Dans les réseaux, les communications de réseau numérique assurent la livraison des données de bout en bout. Cela se fait par l'utilisation de cartes d'interface réseau (NIC). Elles relient l'ordinateur au réseau informatique permettant ainsi l'envoi et la réception de données. Elles sont considérées comme faisant partie de la couche liaison de données du modèle OSI. Le type de carte réseau est déterminé par l'interface du réseau utilisé : RJ45, SFP, BNC, AUI.

Toutes ces cartes réseau ont une interface d'envoi et une interface de réception. Étant donné que nous recherchons une solution rentable pouvant être mise en œuvre, nous avons utilisé la carte réseau RJ45 pour créer la diode de données.

Le câble de données RJ-45 contient 4 paires de fils composées d'un fil de couleur unie et d'une bande de la même couleur. Bien qu'il y ait 4 paires de fils, l'Ethernet 10BaseT/100BaseT n'utilise que 2 paires : Orange et Vert. Les deux autres couleurs (bleu et marron) peuvent être utilisées pour une deuxième ligne Ethernet ou pour les connexions téléphoniques. Le câblage du câble RJ45 suit deux normes : T-568A et T-568B. Les deux normes de câblage sont utilisées pour créer un câble croisé (T-568A à une extrémité et T-568B à l'autre extrémité) ou un câble droit (T-568B ou T-568A aux deux extrémités).

Les câbles croisés sont utilisés lors de la connexion d'un équipement de terminaison de données (DTE) à un DTE ou d'un équipement de communication de données (DCE) à un équipement DCE, tel qu'un ordinateur à un ordinateur, un ordinateur à un routeur ou une passerelle à des connexions de concentrateur. Les câbles directs sont utilisés lors de la connexion DTE à DCE, tels que des ordinateurs et des routeurs à des modems (passerelles) ou des concentrateurs (commutateurs Ethernet). L'équipement DTE termine le signal, contrairement à l'équipement DCE. Deux lignes TX doivent être connectées à deux RX afin de coupler et de connecter n'importe quel périphérique réseau. Donc, en théorie, pour mettre en place un trafic unidirectionnel, il suffit de déconnecter le RX de l'expéditeur. Cependant, cela transformera l'état de l'expéditeur en déconnecté pour le système d'exploitation. La technologie Ethernet utilise Carrier Signal (CS) pour garantir la présence d'une connexion de bout en bout. Une fois la connexion RX déconnectée, le CS sera perdu et la connexion Ethernet sera considérée comme déconnectée. Afin de surmonter ce problème, d'autres ont dû utiliser un troisième média pour fournir uniquement RX à l'expéditeur déconnecté. Comme déjà mentionné, la nouveauté de ce travail est de supprimer ce tiers ; ainsi, l'expéditeur travaille sans utiliser un troisième média en fournissant le CS de lui-même.

Pour mettre en œuvre le nouveau design, nous avons utilisé deux connecteurs femelles RJ45 CAT6. Cette conception est économique, simple à mettre en œuvre et fonctionne avec n'importe quel câble CAT6. La principale innovation de cette conception est que nous avons réussi à connecter les broches 1 et 2 de l'IN RJ45 aux broches 3 et 6 de la OUT RJ45 et, dans le même temps, aux broches 3 et 6 de l'IN RJ45. Ainsi, le résultat est une solution simple, efficace et facile à déployer.

De plus, étant donné que notre data diode est basée sur des cartes réseau Ethernet, l'Auto MDI-X sur les cartes réseau du Node IN et du Node OUT doit être désactivé afin de préserver et de protéger la data diode des manipulations physiques et de permettre l'inversion du flux de données. Pour connecter deux ports de la même configuration (MDI à MDI ou MDI-X à MDI-X), un câble croisé Ethernet est nécessaire pour transmettre les signaux sur le câble, de sorte que l'émetteur et le récepteur correspondent au niveau du connecteur. La fonction Auto MDI-X détecte automatiquement le type de connexion de câble requis et configure la connexion de manière appropriée, éliminant ainsi le besoin de câbles croisés pour interconnecter les commutateurs ou connecter des PC peer-to-peer.

Enfin, une limitation majeure du trafic unidirectionnel est que toutes les applications qui reposent sur TCP/IP sont rendues inutilisables en raison du fait qu'aucune réponse ne sera reçue et que la prise de contact échouera. La seule option disponible est l'utilisation de protocoles prenant en charge le protocole unidirectionnel tel que UDP ou le mode de transfert asynchrone (ATM) pour diffuser les données d'un bout à l'autre.

Avoir cette limitation dans la data diode rend nécessaire le développement d'un logiciel personnalisé ayant une double fonctionnalité : gérer les fichiers entrants, et gérer les fichiers reçus de l'autre côté.

Après la connections de la partie physique, un logiciel est nécessaire pour valider le bon fonctionnement du matériel. Pour envoyer des paquets UDP, le logiciel « UDPcast » a été choisi pour échanger. Il s'agit d'un outil de transfert de fichiers open source qui peut envoyer des données simultanément en utilisant le protocole UDP. Il possède de nombreuses fonctionnalités intégrées, mais la plus importante est de spécifier la bande passante à utiliser. Cette fonctionnalité nous a permis de faire des tests en utilisant plusieurs vitesses de connexion tout en faisant varier les tailles de fichiers afin de comparer le temps de réponse et les retards générés par la diode de données.

En raison de l'absence d'accusé de réception côté destinataire, nous avons utilisé le hachage pour vérifier la bonne réception du fichier. Afin de générer la valeur de hachage du fichier, le logiciel MD5sum a été utilisé. MD5sum calcule le hachage MD5 du fichier et agit comme une empreinte numérique compacte. Le fichier et son hachage sont archivés dans un fichier qui est envoyé sur la diode. Ainsi, puisqu'il n'y a pas de canal inverse, le logiciel doit : s'appuyer entièrement sur la correction d'erreur directe, effectuer des sommes de contrôle pour rejeter les mauvaises données, intégrer la redondance pour assurer la réception des données et générer un état opérationnel déterminé en aval. De nombreuses data diodes commerciales existent sur le marché. Elles sont coûteuses en raison de la complexité du réseau à portée de main. Ainsi, pour valider l'architecture proposée, nous avons créé un prototype de data diode à faible coût. La data diode NODE IN / OUT est un PC doté d'un processeur Intel (R) Core (TM) i7-7700HQ à 2,80 GHz et de 16 Go de RAM avec une carte Ethernet 100 Mbps. Trois fichiers DICOM différents, qui sont généralement transmis sur

l'architecture PACS, ont été utilisés pour étudier et analyser les performances du réseau, avec et sans l'utilisation de la diode de données. On peut remarquer que le pourcentage de latence augmente lorsque la vitesse de connexion augmente (2,73 % pour une connexion de 10 Mb/s contre 17,99 % pour une connexion de 100 Mb/s pour une taille de 223,8 Mo) et que la taille du fichier augmente (6,4 % pour le fichier de 4,9 Mo contre à 12,99 % pour le fichier de 223,8 Mb lors de l'utilisation d'une vitesse de connexion de 50 Mb/s).

Pour conclure, nous avons pu intégrer une data diode dans un système médical pleinement fonctionnel, exposé à plusieurs attaques. La solution proposée a souligné l'importance et le résultat de la data diode sans modifier le temps nécessaire à la transmission/réception du fichier.

Après avoir présenté la solution de la couche réseau, nous présentons au chapitre quatre la solution au niveau de l'appareil (device). La couche appareil représente la couche qui gère la partie logicielle de toute solution. La grande dépendance à l'égard des appareils connectés en a fait un atout essentiel. Ainsi, dans ce chapitre, les menaces essentielles, leur origine et la manière de sécuriser les données contre le vol, l'altération ou l'injection seront définies. En fonction de ces menaces, des dispositifs matériels et des configurations seront proposés afin d'assurer la sécurité maximale du réseau vis-à-vis des interfaces externes.

Les experts en sécurité tentent de dissuader les menaces liées à cette couche en utilisant des logiciels existants comme les antivirus et les pares-feux, et en proposant de nouvelles méthodes qui seront détaillées plus loin. Beaucoup de travail a été fait dans ce domaine, pourtant le nombre d'incidents de sécurité augmente jour après jour. En outre, l'adoption massive de l'Internet des objets (IoT) s'est accompagnée d'une augmentation des vulnérabilités et de l'exploitation de la sécurité. L'IoT est un système d'appareils informatiques interdépendants communiquant via Internet leur permettant d'envoyer et de recevoir des données. Il permet l'interaction entre le monde physique et le monde numérique. Les capteurs et les actionneurs interagissent avec le monde physique et transforment les données dans le monde numérique. Pour mieux comprendre les menaces et les solutions, IDEF0 a été utilisé pour modéliser ce niveau.

Il devient de plus en plus difficile pour un administrateur réseau de faire face au nombre énorme de cybermenaces. Le besoin d'une méthodologie efficace et simple est nécessaire pour faire face à un risque de sécurité actif auquel est confrontée l'entreprise. Par conséquent, une évaluation des risques de vulnérabilité est effectuée pour sélectionner celui qui présente le risque correspondant le plus élevé en tant que priorité pour le renforcement de la sécurité du réseau. Traditionnellement, les systèmes CVSS (Common Vulnerability Scoring Systems) de FIRST constituent principalement la base du risque de vulnérabilité. Le CVSS est un cadre ouvert pour communiquer les principales caractéristiques et la gravité des vulnérabilités logicielles. En fait, une petite enquête sur les scores CVSSv2 montrera que jusqu'à fin 2020, il y a 8330 CVE avec un score de 10 (HIGH) et 193 CVE ont un score

de 10 (CRITICAL) dans les scores CVSSv3. Ainsi, les scores ne peuvent pas être utilisés seuls comme échelle de priorité des risques. De plus, pour comprendre réellement le risque qu'une vulnérabilité impose, une analyse doit être effectuée autour de celle-ci en fonction de la probabilité qu'elle soit utilisée. La raison d'une telle conclusion est le fait que de nombreuses vulnérabilités n'ont jamais été exploitées même si elles ont un score de gravité élevé. FIRST fournit un calculateur CVSS pour saisir des informations contextuelles afin d'améliorer la notation d'une vulnérabilité donnée, comme les facteurs temporels (c'est-à-dire comment une vulnérabilité change dans le temps) et les facteurs environnementaux, mais la notation de ces facteurs est encore générale et repose en grande partie sur l'expérience de l'administrateur d'une organisation. Aucune directive systématique n'est fournie sur la manière de définir ces facteurs de manière appropriée. Pour résoudre ce problème plusieurs travaux ont proposé des solutions pour améliorer la méthodologie de classement et donner une vision plus claire sur les risques pouvant conduire à une meilleure approche de hiérarchisation des dangers.

Pour améliorer la sécurité de l'environnement des technologies de l'information, la nouveauté de ce travail est de proposer une nouvelle procédure pour calculer le score de risque (RS) des cybermenaces détectées et de produire un score de priorité (PS) pour classer les menaces. Les scores CVSS sont utilisés conjointement avec d'autres mesures pour produire un nouveau score qui peut être plus significatif pour améliorer la classification des cybermenaces.

L'évaluation de la cybersécurité d'une entreprise est une étape importante vers la sécurisation de son système et de ses ressources. L'identification des risques est mieux appelée découverte et clarification des risques. L'objectif est de classer les risques existants de manière à remédier à ceux qui peuvent être résolus et à englober ceux dont les conséquences sont acceptables. Notre solution proposée vise à fournir une procédure simple et fiable qui peut aider les administrateurs à estimer une probabilité de notation des vulnérabilités. Cette notation est calculée en se référant à plusieurs bases de données et ensembles de données mis à jour en permanence par leurs propriétaires. Les ressources utilisées pour calculer le score de vulnérabilité sont: NVD CVSS, The Exploit Database (EDB) et Computer Incident Response Center Luxembourg (CIRCL). A partir du NVD, nous avons utilisé le score de base de v2 et v3 pour chaque vulnérabilité, le nombre de citations, la balise Patch et la balise Vendor Advisory qui indiquent s'il existe ou non un Patch ou un Vendor Advisory concernant la vulnérabilité. À partir de l'EDB, nous collectons le nombre d'exploits et la date du premier exploit. Et à partir de CIRCL, nous recueillons le nombre de Common Attack Pattern Enumeration and Classification (CAPEC). En utilisant tout cela, une équation mathématique est utilisée pour créer un nouveau SCORE de vulnérabilité (VS) pour chaque vulnérabilité. Ensuite, un score de vulnérabilité standardisé (SVS) est calculé. S'il existe un Patch and Vendor Advisory, 3 est ajouté au SVS. S'il n'existe

qu'un Patch, 2 est ajouté. S'il n'existe qu'un avis fournisseur, 1 est seulement ajouté. Ainsi, un nouveau score de priorité (PS) a été créé.

Se référant au principe de Pareto, ou à la règle des 80/20, il considère qu'environ 80 % des effets sont déclenchés par 20 % des causes. Pour les administrateurs, le principal enseignement de cette règle est que tous les risques de cybersécurité ne sont pas égaux. Par conséquent, les ressources de sécurité doivent être consacrées aux risques susceptibles de causer le plus de dommages à l'organisation à protéger. Ainsi, sur la base du score de priorité obtenu, si nous avons, par exemple, 20 vulnérabilités, la remédiation du top 4 conduira à l'élimination de 80% du risque système total.

Pour valider la théorie et la formule mathématique, l'ensemble des données a été téléchargé à partir du site Web de NVD. À l'origine, l'ensemble de données contenait 148789 entrées, mais avant d'appliquer la formule, 208 entrées ont été supprimées car elles n'avaient aucun score, mais 23 CVE avec un score CVSSv2 égal à zéro ont été conservés pour maintenir l'intégrité de l'ensemble de données. Il est à noter que parmi les CVE restants, 73688 n'ont pas de score CVSSv3, ce qui est dû au fait qu'il a été publié en 2015. Après avoir appliqué la formule aux CVE, le VS a été obtenu. Après normalisation des valeurs, les valeurs SVS ont été obtenues. Ensuite, sur la base de la présence du correctif de balises et/ou de l'avis du fournisseur, un score de priorité a été acquis. Pour vérifier, la nouvelle liste, nous y avons pris le TOP 10. Et vérifié qu'il s'agissait de l'une des vulnérabilités les plus dangereuses sur la base des classifications de sites Web renommés qui suivent ces problèmes.

Outre le fait que la classification des CVE devient plus précise, cette nouvelle technique de notation élimine les scores redondants qui étaient largement présents dans les CVSSv2 et CVSSv3. Ainsi, on peut remarquer que les dix premières vulnérabilités ont des scores uniques. Ce qui n'est pas le cas des deux scores ci-dessus ; ces nouvelles valeurs peuvent faciliter l'exécution autonome des procédures de remédiation pour réduire les risques de cybersécurité.

De plus, pour mieux protéger le système au niveau de l'appareil, nous avons travaillé sur la protection des flux de données. Bien sûr, les algorithmes d'intelligence artificielle (IA) et le jeu d'apprentissage automatique sont devenus un pilier important de la « révolution de la cybersécurité ». Ces technologies font déjà partie de la vie quotidienne et sont utilisées par des organisations à diverses fins telles que la maintenance prédictive, la détection des fraudes, etc. Notre travail s'est concentré sur l'utilisation de l'IA pour protéger un système médical et s'assurer que les données reçues sont authentiques et dans le même temps pour détecter les comportements médicaux anormaux.

L'application proposée est mise en œuvre pour les personnes âgées et surveillée à distance par les prestataires de soins de santé ; ainsi, le système proposé reposait sur une

approche de télémédecine. Ainsi, l'objectif de ce travail est de s'assurer que les mêmes données, envoyées du côté du patient, sont reçues en toute sécurité pour analyse par le prestataire de soins afin d'assurer le meilleur suivi possible à cette catégorie de personnes. L'architecture du système avant la mise en œuvre de la solution est constituée des trois entités : les capteurs des patients, le serveur passerelle et le serveur du prestataire de soins. Le logiciel AI est installé sur le serveur de passerelle. Plus en détail, le patient est équipé d'un capteur de fréquence cardiaque et d'un capteur SpO2 pour vérifier son état de santé en conduisant. Le serveur passerelle reçoit les données de santé de plusieurs patients utilisant cette application. Le logiciel basé sur l'IA surveillera les données entrantes pour vérifier les irrégularités du point de vue du réseau et de l'appareil, c'est-à-dire l'adresse IP, le délai entre l'envoi de données consécutives, le formatage, les mesures, et avertit le prestataire de soins de réévaluer l'état de santé du patient et le contacter en cas de problème majeur. Et enfin, le serveur du fournisseur de soins de santé est utilisé pour surveiller la santé des patients. Ainsi, nous avons collecté un ensemble de données de 54 000 paquets à partir d'un logiciel qui envoie des données métriques médicales à un serveur et les avons utilisées dans Matlab R2018a. L'ensemble de données contenait 35 % de trafic de cyberattaques, 38 % de trafic lié à des problèmes médicaux et 27 % de trafic normal. Après avoir utilisé 22 algorithmes d'apprentissage automatique, l'algorithme K-Nearest Neighbor (KNN) a fourni le plus de précision. En utilisant ce résultat important, un script python a créé un modèle utilisant l'algorithme KNN et ce modèle a été utilisé dans le logiciel médical pour étiqueter à la volée les données reçues.

Au chapitre cinq, nous abordons la couche du facteur humain. La couche du facteur humain traite des actifs humains présents dans l'entreprise. Une chaîne est aussi solide que son maillon le plus faible. Cependant, dans la cybersécurité, les humains sont le maillon le plus faible. Les violations de la cybersécurité causées par les employés peuvent être causées par une intention malveillante de l'employé à la recherche de profit ou par ignorance. La seule façon de lutter contre l'ignorance consiste à suivre une formation efficace de sensibilisation à la cybersécurité. Cependant, maintenir les employés à jour et informés peut-être coûteux pour l'entreprise s'il est abordé par le biais de la formation classique. De plus, il sera difficile de regrouper les salariés sur un horaire commun précis. Ainsi, introduire des chats bots résoudrait la situation et ajouter quelques fonctionnalités intelligentes lui permettra d'être autonome.

Quant à la malignité, elle consiste à enfreindre intentionnellement les règles de l'entreprise en envoyant des données protégées à un tiers, causant des dommages aux actifs numériques et physiques de l'entreprise, etc. La nouveauté de ce travail résidera dans le développement d'un chat bot basé sur l'IA qui, non seulement initiera les procédures de sécurité de base aux employés, mais les tiendra également informés de la nouvelle technique émergente de défense contre les cyber-attaques et leur fournir un test d'auto-évaluation pour vérifier en permanence leur état de préparation.

Le chat bot proposé, assurera la sécurité au niveau du facteur humain dans un réseau en permettant aux utilisateurs de rechercher des sujets liés aux politiques et procédures mises en œuvre dans l'entreprise. Il permet également aux utilisateurs de rechercher des mots clés liés à la cybersécurité. En outre, il permet aux utilisateurs de passer un examen dirigé sur la cybersécurité en utilisant une liste dynamique de questions créée au hasard pour auto-évaluer un employé et générer un rapport montrant les faiblesses. Ces résultats seront sauvegardés et comparés ultérieurement pour suivre l'évolution de l'employé. Ce bot a également une fonctionnalité nouvelle et innovante qui est l'utilisation de WhatsApp pour faciliter la livraison d'informations en fournissant un moyen de communication familier sans compter sur l'API fournie par WhatsApp.

La première étape de la mise en œuvre de la solution est la collecte des données. Les organisations s'appuient sur les politiques en tant que déclaration de gestion de haut niveau dans le but d'influencer les décisions et de guider l'organisation pour atteindre les résultats souhaités. Les politiques sont appliquées par des normes et sont ensuite mises en œuvre par le biais de procédures visant à établir des exigences applicables et responsables. Les politiques ISO/IEC 27001, NIST et PCI DSS ont été utilisées pour fournir les données nécessaires pour enrichir les jeux de données des cyber bots. En outre, le questionnaire sur les aspects humains de la sécurité de l'information (HAIS-Q) a été utilisé. HAIS-Q était un questionnaire réalisé par *Prasons et al.* sur 1112 étudiants universitaires concernant les comportements naïfs et accidentels de l'utilisateur de l'ordinateur qui pourraient être à l'origine des failles de sécurité de l'information. De plus, trois moteurs de recherche (duckduckgo.com, google.com, bing.com) ont été utilisés pour trouver des tests d'évaluation de la cybersécurité à l'aide de mots-clés tels que « awareness test », « cyber security test » et « cyber awareness test ». Après filtrage des résultats, seuls les tests libres ont été pris en compte.

L'intelligence artificielle joue un rôle important dans la solution. La Machine Vision est utilisée pour identifier des objets à l'écran afin d'identifier les utilisateurs sur le Web WhatsAPP et de collecter les questions. Alors que le Natural Language Processing (NLP) a été utilisé pour créer un bot conversationnel basé sur BertForQuestionAnswering formé à l'aide de l'ensemble de données SQuad2 pour gérer des questions/réponses simples avec l'utilisateur.

Le questionnaire d'évaluation est composé de douze questions parmi une liste de 150 questions générales choisies au hasard, mais également, parmi trois catégories. Suivi de dix questions parmi une liste de 50 questions spécialisées choisies au hasard dans la catégorie à laquelle appartient l'utilisateur.

Ce test d'évaluation peut être passé par l'utilisateur à tout moment. Cependant, tous les trois mois, au moins un test doit être effectué ou à des périodes aléatoires de la semaine avec ces 12 questions pour assurer un flux continu de connaissances.

Ce bot sera capable de tenir à jour les dossiers de chaque employé, d'évaluer ses progrès et de proposer des formations pour réduire les faiblesses qu'il rencontre. La mise en œuvre de ce bot basé sur l'IA a montré un grand impact sur l'employé ainsi qu'un moyen plus simple d'enseigner aux employés et de les tenir au courant des menaces de sécurité.

Et enfin, le chat bot fournit également des alertes de cybersécurité de la Cybersecurity and Infrastructure Security Agency (CISA).

Après avoir présenté les différentes solutions pour les trois niveaux définis de l'architecture du système dans les chapitres trois, quatre et cinq, un dernier chapitre présentera une mise en œuvre de bout en bout des différentes solutions dans un système complet. Une application médicale sera utilisée pour cette implémentation car elle fait partie des systèmes les plus critiques.

Le système est une application de suivi médical. Il se compose de nombreux sous-systèmes qui assureront le transfert des données du patient vers le superviseur médical. Des mesures de cybersécurité seront appliquées pour assurer la confidentialité, l'intégrité et la disponibilité du système.

Le système est conçu pour surveiller la santé du patient tout en faisant son activité quotidienne normale. Les signes vitaux sont continuellement mesurés et recueillis pour informer les prestataires de soins de santé en cas de problème médical. Ceci est principalement appliqué pour les clients ayant des antécédents médicaux tels que l'insuffisance cardiaque, le diabète ou même l'épilepsie. Ainsi, l'objectif principal est de fournir un flux de données sécurisé du patient au prestataire de soins de santé pour surveiller la santé du patient et prendre la bonne décision chaque fois que cela est nécessaire.

Le système sera testé à l'aide de cybermenaces contrôlées pour garantir sa fiabilité et son endurance face aux cyberattaques réelles. La sécurité du système sera maintenue sur la base des trois niveaux décrits précédemment dans les chapitres précédents, à savoir le niveau réseau (NL), le niveau dispositif (DL) et le niveau facteur humain (HL).

Une analyse du système doit préciser l'état actuel des actifs et déterminer les points clés pouvant avoir un impact sur sa disponibilité, sa confidentialité et son intégrité. Ensuite, une estimation et une classification des risques sont réalisées afin de déterminer les « quick wins » pouvant avoir un fort impact sur la sécurité et définir le traitement adéquat à appliquer afin de minimiser l'ensemble des risques déjà identifiés.

Pour appliquer une sécurité à un système, les actifs doivent être clairement identifiés. Les principaux actifs de ce système sont les données et les fichiers de configuration. Quant aux actifs secondaires, ils représentent les composants du réseau, les logiciels, les appareils et les utilisateurs.

Après avoir noté les actifs, un modèle du système a été créé avant d'appliquer les mesures de sécurité. Ensuite, le modèle a été modifié pour montrer le système après l'application des mesures et des contrôles de sécurité.

Le système médical sera divisé en trois environnements principaux :

- Le premier regroupe les patients (ou clients) connectés au serveur passerelle, dit R2. Les données des capteurs sont regroupées et envoyées, via un appel API, avec quelques délimiteurs à ce serveur ;

- La deuxième partie représente le serveur de passerelle qui gère les demandes des clients, vérifie les anomalies de données et transmet les données au serveur principal, appelé R3. C'est le dernier composant connecté physiquement à Internet ;

- La troisième partie représente le serveur backend. Ce serveur est simplement connecté au serveur de passerelle sans aucune connexion Internet. Son rôle principal est de répondre aux demandes des prestataires de soins et de sauvegarder les données de l'ensemble du système en toute sécurité.

De plus, nous devons noter que les microcontrôleurs Raspberry Pi sont divisés comme suit :

- Raspberry Pi 4 : un pour le serveur passerelle, un pour le serveur backend, un pour le serveur chat bot WhatsApp (sera défini ultérieurement) et deux pour les prestataires de santé ;

- Raspberry Pi zéro : tous les trois sont connectés aux capteurs médicaux de trois patients différents utilisant ce système.

Ainsi, pour maintenir la sécurité de ce système, nous appliquerons ce qui suit :

- un logiciel basé sur l'IA pour filtrer le trafic et maintenir la sécurité des données transférées ;

- une data diode pour empêcher toute exfiltration de données sensibles hors de la zone restreinte ;

- une priorisation des vulnérabilités au niveau des deux serveurs ;

- un chat bot pour intégrer la cyber-sensibilisation des utilisateurs de ce système (soignants et patients) afin de limiter les carences en facteur humain au niveau de la sécurité.

Après application des mesures de sécurité, des tests de validation ont été effectués pour s'assurer du bon fonctionnement du système. Pour tester le transfert de données, plusieurs fichiers de différentes tailles ont été transférés avec et sans la diode de données. D'après les résultats, la data diode n'a pas affecté le temps nécessaire pour transférer les données entre les deux serveurs. Notez ici que la taille des paquets n'a pas seulement été choisie en fonction des données médicales à envoyer (car cela représente une très petite taille de fichier d'environ quelques Ko) mais l'objectif principal était de s'assurer que, même pour les fichiers de données volumineuses, la data diode n'affectera pas la vitesse de transfert entre les serveurs. De plus, les différents tests ont été effectués au niveau du réseau pour valider les solutions implantées et les résultats globaux ont montré que ces solutions proposées ne créent pas de délais et assurent un niveau d'intégrité supérieur à 99%.

Quant à la hiérarchisation des vulnérabilités, un logiciel open-source, WAZUH, a été utilisé sur les deux serveurs pour identifier toutes les vulnérabilités reçues afin de les comparer ultérieurement avec une liste actualisée de CVSS. L'intégration de la sécurité au niveau de l'appareil a été réalisée grâce à la mise en œuvre d'algorithmes avancés de chaque côté du serveur. Le logiciel de priorisation présenté au chapitre 4 a été utilisé et tournait périodiquement sur la passerelle et les serveurs backend. Les résultats ont montré que le top 10 des CVE ont un score CVSSv3 de 10, et notre système de notation des priorités a fourni une liste ordonnée que le Responsable Sécurité des Systèmes d'Information peut utiliser pour résoudre les vulnérabilités du système d'information.

Et pour la sécurité au niveau humain, le chat bot a été utilisé. Basé sur un petit questionnaire rempli par 12 utilisateurs (patients et prestataires de soins de santé), il a montré qu'ils avaient obtenu de nombreuses informations concernant la sécurité du système dont ils n'avaient jamais entendu parler auparavant. De plus, 11 des utilisateurs ont trouvé le quiz du chat bot facile à utiliser et 9 utilisateurs ont trouvé l'expérience d'apprentissage satisfaisante.

Pour conclure, on peut confirmer que les outils de sécurité ajoutés à ce système ont offert un haut niveau d'intégrité avec un coût très réduit. Ces solutions sont d'autant plus intéressantes qu'elles sont autonomes et ne nécessitent pas une intervention continue de l'administrateur système. De plus, la présence de logiciels basés sur l'IA aidera à mettre à jour en permanence les protocoles de sécurité de manière à maintenir le système toujours à jour face aux nouvelles menaces et vulnérabilités. Enfin, il convient de noter que, même avec la mise en œuvre des différents logiciels et algorithmes, le temps de réponse du système n'a pas été affecté ; cette fonctionnalité est si importante car, généralement, l'installation d'un logiciel aussi exigeant sur un processeur entraînera une réduction de ses performances.

Table of contents

List of figures

List of Tables

List of acronyms

AI	(Artificial Intelligence)
APT	(Advanced Persistent Threat)
APT	(Advanced Persistent Threats)
AUC	(Area Under Curve)
ARP	(Address Resolution Protocol)
ATM	(Asynchronous Transfer Mode)
C&C	(Command and Control)
CAPEC	(Common Attack Pattern Enumeration and Classification)
CIRCL	(Computer Incident Response Center Luxembourg)
CISA	(Cybersecurity and Infrastructure Security Agency)
CISO	(Chief Information Security Officer)
CNN	(Convolution Neural Networks)
CS	(Carrier Signal)
CVE	(Common Vulnerabilities and Exposures)
CVSS	(Common Vulnerability Scoring Systems)
DCE	(Data Communications Equipment)
DDOS	(Distributed Denial of Service)
DDoS	(Distributed Denial-of-Service)
DFD	(Data-Flow Diagrams)
DNS	(Domain Name System)
DTE	(Data Terminating Equipment)
EDB	(The Exploit Database)
GHSOM	(Growing Hierarchical Self-Organizing Map)
HAIS-Q	(Human Aspects of Information Security Questionnaire)
HIDS	(Host-based Intrusion Detection System)
HMM	(Hidden Markov Model)
ICS	(Industrial Control Systems)
IDEF	(Integrated DEFinition)
IDS	(Intrusion Detection System)
IDS	(Intrusion Detection Systems)
IoT	(Internet of Things)
IPS	(Intrusion Prevention System)
ITS	(Information Technology System)

KNN	(K-Nearest Neighbor)
MADE	(Malicious Activity Detection in Enterprises)
MAIL	(Malware Analysis Intermediate Language)
MIR	(Music Information Retrieval)
ML	(Machine Learning)
MitM	(Man-in-the-Middle)
NBM	(Naive Bayes Multinomial)
NIC	(Network Interface Cards)
NIDS	(Network-based Intrusion Detection Systems)
NLP	(Natural Language Processing)
PAN	(Primary Account Numbers)
PCi DSS	(Payment Card industry Data Security Standard)
PoC	(Proof of Concept)
POS	(Point of Sale)
SDLC	(Software Development Life Cycle)
SMB	(Medium-Sized Businesses)
Structured Language Query	(SQL)
SVM	(Support Vector Machine)
SWOD-CFWeight	(Sliding Window Of Difference and Control Flow Weight)

Chapter 1 Security Issues that encounter the Information Technology Systems

Outline

1.1 Introduction

Cyber security concerns have been the major driver of Information Technology System (ITS) spending. The fast pace of emerging threats and risk to ITS have pushed organizations to spend enormous amount of time and resources deploying and managing cyber security solutions to mitigate these threats and keep critical business services operating. In spite of that, most organizations remain inadequately protected and the number of security incidents is on the rise day after day (Jang-Jaccard & Nepal, 2014).

Data and its availability have become the world most valuable resources. Threat actors are diversifying their methods in order to acquire control of assets inside of organizations. Some of them do it for the glory and fame while others do it to inflict destruction and harm and at last there are the ones how do it for lucrative purposes.

The number of data breaches is rising every year. A report by ***Risk Based Security*** showed that 7.9 billion records have been exposed in the first nine months of 2019 which is more than 112% the number of records in the same period in 2018 (Goddijn, 2020).

In this chapter, an overview over the information technology system and its security parameters will be presented. Added to that, a brief introduction concerning the realized work will be also proposed. This chapter introduces the novelty of the work, shows the methodology applied in this work and ends up by presenting the report structure.

1.2 Security overview

In this section, the main security entities will be presented along with their main actors. Thus, the main security threats and vulnerabilities will be shown along with their actors. Some countermeasures, found in the bibliography, will be also mentioned.

1.2.1 Threats

A threat is an event or condition that causes asset loss and undesirable consequences or impact from such loss. The basis of asset loss constitutes all forms of intentional, unintentional, accidental, incidental, misuse, abuse, error, weakness, defect, fault, and/or failure events and associated conditions (Ross, McEvilley, & Oren, 2016). Added to that, a threat can refer to entities who attempt unauthorized access to organizations assets using a data communications pathway (CISA, Cyber Threat Source Descriptions, 2020). This access can be initialized from within an organization by trusted users or from remote locations by unknown entities using the Internet. Kaspersky classifies cyber threats into three folds (Kaspersky, 2020):

- Cyber-crime: an act that targets ITS for financial gain or causes disruption;
- Cyber-attack: an act that involves information gathering;
- Cyber-terrorism: an act that intends to undermine systems to inflict panic or fear.

The most common cyber security threats are:

- Malware: A malware is a malicious software that a cybercriminal creates to compromise a user's computer. It is often spread as an email attachment or as legitimate-looking software. There are different types of malwares:
    - Virus: it is a self-replicating program that injects itself in clean files;
    - Trojans: it is a program that disguises itself as legitimate software;
    - Spyware: it is a program that secretly records what a user does;
    - Ransomware: it is a program which locks down a user's files and data, with the threat of erasing it unless a ransom is paid;
    - Adware: it is advertising software which can be used to spread malware;
    - Rootkits: once a malware is installed on a system, it is useful if it stays concealed to avoid detection. Rootkits hide themselves inside the host's operating system, thus, making them hard to detect;
    - Worms: unlike a virus, worms self-propagate without attaching itself to legitimate software. It uses vulnerabilities in network access to send themselves to other computers on the network;
    - Botnets: it is a group of computers connected to the Internet that have been compromised by a hacker using a computer virus or Trojan horse. An individual computer in the group is known as a "zombie" computer;
- SQL Injection: SQL (Structured Language Query) injection is a type of cyber-attack that aims to control and steal data from a database. Vulnerabilities in data-driven applications are used to insert malicious code into a database via a malicious SQL statement. This attack mainly gives access to the sensitive information contained in the database by bypassing implemented security measures;
- Phishing: phishing is a type of online identity theft. It involves tricking people into handing over sensitive personal data, such as credit card numbers, passwords, bank account information or other information;
- Man-in-the-middle attack: they allow the attacker to eavesdrop on communication between two targets. They can listen to a communication which should, in normal settings, be private. Examples of these attacks are:
    - DNS spoofing
    - HTTPS spoofing
    - IP spoofing
    - ARP spoofing
    - SSL hijacking

    - Wi-Fi hacking
- Denial-of-service attack: it prevents a computer system from fulfilling legitimate requests by overwhelming the networks and servers with traffic. This renders the system unusable and prevents an organization from carrying out vital functions;
- Crypto-jacking: it is the malicious installation of cryptocurrency mining software. It harnesses the victim's processing power to mine for crypto-currency.

1.2.2 Vulnerabilities

Vulnerabilities are weaknesses in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source (NIST, Risk Management Framework for Information Systems and Organizations, 2018). Flaws, features or user error are often the source of vulnerabilities. Below is a description of each entity apart (NCSC, 2020):

- Flaws: it is an unintended functionality. It may be the result of poor design or mistakes made during implementation;
- Zero-day vulnerabilities: they are used in specially tailored attacks by capable and resourced attackers. Once it becomes publicly known, commodity attacks are developed for reusability. The ability of an attacker to find such vulnerabilities depends on their technical capabilities;
- Features: They are functionalities which can be misused by an attacker to compromise a system. Features may improve management, the user's experience or help diagnose problems. Most common example is JavaScript which is widely used in dynamic websites and continues to be used by attackers;
- User error: a system that is secured by design can minimize the exposure to threats. However, the human factor can undermine any security measure applied. A vulnerable feature that is not fixed or weak passwords are some of the many examples that weaken any secure system.

1.2.3 Threat Actors

A threat actor is a person or group that participates in an action or process that is hostile using computers, devices, systems, or networks. Threat actors are classified into five groups based on their motivations and affiliations (CIS, 2020):

- Cybercriminals: they are largely profit-driven and represent a long-term, global, and common threat. They target data to sell, hold for ransom, or otherwise exploit for monetary gain. Cybercriminals may work individually or in groups

to achieve their purposes. They include hackers, phishers, spammers and botnet operators;

- Insiders: they are current or former employees, contractors, or other partners who have access to an organization's networks, systems, or data. Malicious insiders intentionally exceed or misuse their access in a manner that negatively affects the confidentiality, integrity, or availability of the organization's information or information systems. This differs from unwitting insiders who unintentionally cause damage to their organization's information systems through their actions, such as clicking on malicious links in a phishing email;
- Nation-State actors: they aggressively target and gain persistent access to public and private sector networks to compromise, steal, change, or destroy information. They may be part of a state apparatus or receive direction, funding, or technical assistance from a nation-state. Nation-state has been used interchangeably with Advanced Persistent Threat (APT); however, APT refers to a type of activity conducted by a range of actor types;
- Hacktivists: Hacktivists (*a.k.a.* Ideologically-Motivated Criminal Hackers) are politically, socially, or ideologically motivated and target victims for publicity or to effect change, which can result in high profile operations;
- Terrorist Organizations: Their limited offensive cyber activity is typically disruptive or harassing in nature. Terrorist organization’s primarily use the internet for communications and recruitment.

1.2.4 Countermeasures

Security countermeasures are the controls used to protect the confidentiality, integrity, and availability of data and information systems. There is a wide array of security controls available at every layer of the stack. Overall security can be greatly enhanced by adding additional security measures, removing unneeded services, hardening systems, and limiting access. Some of the most used protection methods are listed below (without taking into consideration their order of importance):

- Intrusion Detection System (IDS): it is a passive system that monitors traffic to/from all devices on the network, performs an analysis of passing traffic and matches it to the database of known attacks;
- Intrusion Prevention System (IPS): it is an inline (in the direct communication path between source and destination), actively analyzing and taking automated actions on all traffic flows that enter the network to detect and prevent vulnerability exploits;
- Firewall: it is a system that monitors and controls incoming and outgoing network traffic based on predetermined security rules;

- Network segmentation / Data Diode: it is the act of splitting a computer network into sub-networks, each being a network segment. It allows isolating and filtering access between network segments;
- Network Encryption: it secures the data transmitted between different network entities by making it unreadable except to the entities communicating;
- Antivirus: it is a software used to prevent, detect, and remove malware;
- Host-based Intrusion Detection System (HIDS): it is a passive system that monitors and analyses files and processes of the internals of a computing system and monitors network traffic as it enters the host;
- Software Update: it involves the deployment of updates and security patches when necessary to keep the device safe from exploitation;
- Device Encryption: encryption of the data on the device helps from being compromised in case of theft and the encryption of certain parts of the device helps in protecting itself from malwares in case of infection;
- Physical security: it involves the protection from physical actions and events that could cause serious loss or damage. Measures applied include locks, access control and fire suppression systems;
- Limit the use of removable media: removable media is useful for quickly transferring files from one device to another but are a common source of malware and can be used to remove large amounts of data from corporate systems very quickly;
- Secure Configuration: it involves the utilization of the best configuration for the software installed by deactivating unneeded features;
- Scheduled Backups: scheduled backups allow the host to recover from attacks and to maintain availability;
- Documented policies and procedures: they help employees to use the ITS in accordance with the directives of the Chief Information Security Officer (CISO) by setting the rules and expectations for behavior. They also provide the guidelines for CISO to monitor and investigate when needed, and define the consequences of violations;
- Frequent training: It helps employees become accustomed to cyber threats and about corporate policies and procedures for working with Information Technology System (ITS). Added to that, it is necessary to the advancement of the Cyber Security team;
- Identity and access management: it refers to a framework of policies and technologies for ensuring that the right individuals in an enterprise have the appropriate access to technology resources at the right times for the right reasons.

To sum up, after presenting the main threats, vulnerabilities and their actors along with the countermeasures that are integrated in today's systems in order to increase security, confidentiality, authenticity and availability of data, one can find clearly that these solutions are so sophisticated and require lots of resources, human interaction and cost to be implemented. Added to that, the absence of a full and complete strategy that can help the network administrator is remarkable.

1.3 Evolution of threats and solutions: History and first state of the art

In this section, a brief history of the threats will be shown since the first attack in 1971 till the beginning of the third decade. Added to that, the solutions that were undertaken to limit these threats will be also emphasized. This section will introduce for the first time the distinction between the three different layers of the system: the network, the device and the human.

1.3.1 History of threats

Although cyber-attacks have started widely at the end of the 1980's, the beginning of the current millennium has yielded more frequent attacks that created extreme damage to data credibility and hardware performance. These attacks focused basically on network and/or host security breaches. Here below, the most notorious cyber-attacks that made history are presented:

- In 1971, Bob Thomas created a program capable of moving across a network. He named it the *Creeper* (Chen & Robert, 2004). It was the first computer virus to infect and spread to PDP-10 mainframe computers made by Digital Equipment Corporation;

- In 1986, The German computer hacker Marcus Hess hacked an internet gateway in Berkeley, and used that connection to infiltrate Arpanet. He hacked 400 military computers, including mainframes at the Pentagon, with the intent of selling their secrets to the KGB. Marcus Hess operation was discovered by Cliff Stoll who had a Ph.D. in astronomy at Lawrence Berkeley Lab's Keck Observatory but was transferred to the computer center in the basement of the same building and started work as a computer systems manager after that his grant money ran out. His work was manual because no automated system was present at the time;

- In 1987, The Vienna virus is a virus that infects *.com* files on DOS-based systems (Burger, 1988);

- In 1989, *NASA Ames Research* Center in California was hit by a virus attack. Peter Yee, an employee at the NASA, sent a memo by email to his colleagues that read, "We are currently under attack from an Internet VIRUS! It has hit Berkeley, UC San Diego, Lawrence Livermore, Stanford, and NASA Ames.";

- In 1995, Kevin Poulsen hacked the telephone lines of Los Angeles radio station to guaranty his winning of a Porsche 944 S2;

- In 2006, Greg Chung was arrested for cyber espionage in favor of the Republic of China after working more than 30 years at Boeing. Over $2 billion USD worth of aerospace docs were delivered to China (Justice, 2009);

- In December 2006, *NASA* was forced to block all emails containing attachments before launching the shuttle out of fear that they would be hacked. Rumors say that foreign intruders have gained access to the plans for the latest US space launch vehicle (Boyle, 2006);

- In June 2007, the People's Liberation Army was accused of hacking into the US Secretary of Defense's unclassified email account as part of larger series of attacks aiming to access the Pentagon's network (NEWS, 2007);

- In January 2009, at least 5,000,000 computers attacked Israel's internet infrastructure focusing on government websites (Pfeffer, 2009);

- In October 2010, Iran was hit by a cyber-attack by the malware Stuxnet designed to interfere with Siemens industrial controls used in the Iranian nuclear program (Shakarian, 2011);

- In April 2011, *Sony's PlayStation Network* was attacked and the personal data of 77 million users were leaked as a result (Quinn & Arthur, 2011);

- In July 2011, a U.S. Department of Defense contractor was hacked. As a result, 24,000 files belonging to the department were stolen (SHANKER & BUMILLER, 2011);

- In 2013, *Yahoo* was the victim of a cyber-attack that led to the compromise of one billion accounts. However, it wasn't until 2016 that Yahoo announced the incident and revised its estimation of the incident to declare that over 3 billion accounts were affected (PERLROTH, 2017);

- In 2014, *JPMORGAN chase* was hit by a cyber-attack that compromised the accounts of 76 million households and 7 million small businesses (SILVER-GREENBERG, GOLDSTEIN, & PERLROTH, 2014);

- In October 2016, 412.2 million accounts from *Adult Friend Finder* were leaked (WEISE, 2016);
- In March 2017, the gaming site *Rune.live* had 9,618 user accounts exposed;
- In May 2017, educational website *Edmodo.com* suffered a 77 million user accounts breach (DELLINGER, 2017);
- Also in May 2017, the *WannaCry* ransomware affected more than 200,000 computers across 150 countries. On May 12, the National Health Service hospitals in England had to turn away non-critical emergencies because over 70,000 devices were affected by this attack (HERN & GIBBS, 2017);
- In September 2017, *Equifax* announced that it was a victim of a major breach that extended from May till July of this same year. 143 million user accounts were exposed and the hackers gained access to around 209,000 consumer credit cards (O'Brien, 2017);
- In June 2018, 340 million records were leaked by the marketing firm *Exactis*. Poor cyber security measures made the database wide open for hackers. Sensitive information was in the leak, including smoking habits, adhering to a particular religion, interests and other habits (GREENBERG, 2018);
- In September 2018, the *Marriot* declared that a leak of information has been spotted going back to 2014 affected its Starwood hotel group owned by Marriott. Approximately 327 million customers suffered a breach of personal information (Perlroth, Tsang, & Satariano, 2018);

After presenting a history of the most notorious cyber-attacks, Table 1.1 presents a timeline of the cyber threats and the respective solutions created to solve them.

Table 1.1 Timeline of the cyber threats and the respective solutions

Year	Cyber Threat	Cyber Resolution
1971	The Creeper (Chen & Robert, 2004)	Reaper was a program created by Ray Tomlinson to combat and disinfect the computers affected by this virus. It was a first attempt to create ANTIVIRUS software.
1986	The German computer hacker Marcus Hess hacked an internet	Dorothy Denning and Peter Neumann researched and developed the first model of real-time IDS that they started developing since 1984 (Denning,

	gateway in Berkeley, and used that connection to infiltrate Arpanet	1987). This prototype was named the Intrusion Detection Expert System (IDES). Their work was supported by the U.S. Space and Naval Warfare Command (SPAWAR) and by the U.S. National Science Foundation.
1987	The Vienna virus (Burger, 1988)	Bernd Robert Fix created a program to get rid of this virus. This event was the first documented removal of a computer virus by an actual ANTIVIRUS program. McAfee Antivirus and NOD32 Antivirus also appeared in this same year.
1989	NASA Ames Research Center in California was hit by a virus attack	A NASA researcher created the very first firewall program design through a virtual "firewall" which was modeled on the physical structures that prevent the spread of actual fires within buildings or structures. However, the first-generation FIREWALL, or packet filter, was presented by Jeffrey C. Mogul of Digital Equipment Corp. (DEC) (Mogul, Rashid, & Accetta, 1987). The second-generation FIREWALL, or Circuit Level Gateway, was created by AT&T Bell Labs by Dave Presotto. It was the first Stateful Firewall. It keeps information about the active sessions and connection states (Kenneth & Forrest, 2002). Today's firewalls are a mix of these two concepts.
1990 - …	All attacks.	Updates to policies and amelioration of the performance of the different solutions.

1.3.2 History of solutions

In this section, the main contributions proposed by researchers to ensure and increase the security of each level, *e.g.*, Network Level, Device Level and Human factor Level, along with the threats they wanted to handle, will be presented. As already mentioned, three different levels will be treated in order to assure the maximum security of the system. Hereafter is a list of publications treating respectively the different security levels.

1.3.2.1 Network threats and actions

In 2019, Alsughayyir *et al.* from Qassim University have used Deep Learning Auto-encoders model to create an effective Intrusion Detection System (IDS) based on classifying normal behavior from abnormal behavior found on the network. They used the NSL-KDD dataset which is an improved version of KDD'99 Cup Dataset. Their approach achieved an accuracy of 99% for training and 91.28% for the testing phase compared to Random Forest classifiers that achieved 96.77% for training and 75.81% for the testing phase (Alsughayyir, Qamar, & Khan, 2019).

In 2018, Ajagekar and Jadhav from Kharghar Mumbai University worked on a solution to detect Distributed Denial of Service (DDOS) attacks. Their solution relies on removing irrelevant text information such as delimiters and stop-words from the collected traffic before using it as training data for the Naive Bayes Multinomial (NBM) machine learning algorithm. The experimental analysis showed that their proposed solution achieved 93% F-measure, 91% recall rate, 82% precision rate, and 97% accuracy (Ajagekar & Jadhav, 2018).

In 2016, Eigner *et al.*, from the University of Applied Sciences St. Pölten, proposed a method to detect Man-in-the-Middle (MitM) attacks on Industrial Control Systems (ICS) using machine learning. They created a Proof of Concept (PoC) of an ICS, and used the data collected from the sensor and actuator data of the conveyor belt to extract 32 features. After experimenting with several machine learning algorithms, they realized that the best one fit for their case was the *k-Nearest Neighbors algorithm with Bregman divergence algorithim*. At first, they collected these features during normal behavior of their system, then they collected them after launching MitM attack on the system using ETHERCAP. The higher transmission time of the data, which was caused by the interception of the attacker, resulted in slight deviations of the values. Their process managed to identify anomaly attack behaviors with no false positives. It should be noted that the experiment was made in a very controlled environment where no unnecessary protocols or other data or noise were transmitted over the network. Moreover, the realistic settings in ICS regarding the number of sensors and actuators involved would make the process of feature selection during the machine learning a more challenging task because a larger number of elements are present in real industrial environments (Eigner, Kreimel, & Tavolato, 2016).

In 2015, Celik *et al.*, in a joint work between the Pennsylvania State University and USA Army Research Laboratory, proposed a framework to give insight of malicious traffic and complement diagnostic measures of existing detection systems. Their work focused on detecting malwares that are in continuous contact with their C&C server connected over TCP. In their study, they focused on 10 features extracted from the network traffic flow. They used the legitimate traffic collected at the University of Twente from May to June 2007, and they compared it to the network traces of 21 malware families collected from 2007 to 2014.

To detect anomalies, four machine learning algorithms were used: OCSVM, k-NN, LSAD and k-means. By using AUC (Area Under Curve) as a metric evaluation, k-NN outperformed the other algorithms. For example, for the DonBot malware, K-NN scored 0.99 while LSAD scored 0.73. However, they deduced that the AUC decreased with the evolution of malwares especially if the malware disguises its traffic as HTTP (Celik, Walls, McDaniel , & Swami, 2015).

In 2014, Roy *et al.* from VIT University and the University of Southern Mississippi have proposed an intrusion detection system based on the Bagging Classifier which is an ensemble meta-estimator machine learning algorithm. They used in their work the KDD'99 Cup Dataset and they achieved an accuracy rate of 82.72% with 10-fold cross validation (Roy, Krishna, & Yenduri, 2014).

In 2013, Huang and Huang from Academia Sinica used Growing Hierarchical Self-Organizing Map (GHSOM) with Support Vector Machine (SVM) machine learning classifiers to better understand the network traffic, to classify its flow and to detect network anomalies. GHSOM was used to cluster the collected traffic. The output was a labeled cluster used as an input for SVM classifiers to start training the model. Finally, the applied performance tests led them to conclude that the combination used is a promising approach to detect patterns of anomalous network traffic (Huang & Huang, 2013).

Based on the Stuxnet case that hit industrial control systems in 2010, Jeon and Na proposed a solution to protect highly classified computer networks by separating them from the less classified networks using a data diode. They presented different approaches to realize a data diode and listed the different commercial data diodes available. However, no measurements were shown in the paper (Jeon & Na, 2016).

1.3.2.2 Device treats and actions

In 2019, Zhang *et al.*, in a joint work between the University of Hong Kong and the University of Technology and Design Singapore, proposed a new approach to handle malicious files. Their format-based file cleansing process named *File Guard* aims at preventing software vulnerabilities from being triggered by malicious files yet keeping it functional as much as possible. Their solution relies on repairing the file according to his original format standard, thus cutting off any attempt to manipulate it in order to trigger vulnerabilities in the software responsible for opening it. They focused their work on GIF, PNG and MPEG4 file formats because these types of files were the most abused to infiltrate Android phones according to their research. The evaluation of the system showed that the malicious files were sanitized and became non harmful, the sanitized files remained functional. Moreover, *File Guard* is efficient enough to be implemented on smart phones (Zhang, Lee, Gao, & Zhou, 2019).

In 2018, Kumar *et al.,* in a joint work between the University of Electronic Science and Technology of China and Quaid-e-Azam University Islamabad of Pakistan, used CNN (Convolution Neural Networks) image similarity technique to detect unknown or new types of malwares. The files were decompiled and converted into gray scale images. A dataset from Vision Research Lab of 9,458 of 25 different malware families and 3,000 different kinds of benign software was used in the work and achieved 98% of accuracy (Kumar, Xiaosong, Khan, Ahad, & Kumar, 2018).

In 2016, Farrokhmanesh and Hamzeh from Shiraz University proposed in their paper a byte-level method for detecting malware by using audio signal processing techniques. Their method relied on converting program's bytes to a meaningful audio signal, then Music Information Retrieval (MIR) techniques were employed to detect new and unseen instances by constructing a machine learning music classification model from the resulting audio signals. They applied kNN, AdaBoost, and Random Forest classifiers. The AdaBoost classifier gave the best results with a 92%TPR, 0.7% FPR, 92.2% precision, 92.2% accuracy and 92.2% F-Mesure (Farrokhmanesh & Hamzeh, 2018).

In 2015, in a joint work between Nokia Bell Labs and Siemens AG Corporate Technology, Khatri and Abendroth presented a solution for mobile phone threats by developing the *Mobile Guard* malware detection system. This application was able to detect malicious activities in networks, especially in the mobile operator's network. The solution consisted of creating a top-level layer through which all traffic is being directed in order to detect any malicious activity by analyzing the pattern of the flow. The authors didn't mention the accuracy of the system (Khatri & Abendroth, 2015).

In the same year, Saxe and Berlin introduced, while working at Invincea Labs LLC, an approach to detect malwares using deep learning applied on a dataset of over 400,000 software binaries. Their work consisted of analyzing benign malwares by extracting the features needed to train the deep learning model using static analysis without de-obfuscation of the binary. The machine learning Python libraries SciPy and NumPy were used to extract the features, while the machine learning Python library Keras was used to create the neural network. They relied on the Bayesian model calibration to identify whether a file is malicious or not. The results of the tests showed a 95% detection rate and a false positive of 0.1% (Saxe & Berlin, 2015).

Also in 2015, Gharacheh *et al.,* from Yazd University and Shiraz University, proposed to tackle the malware detection process through the classification of the Operation Code (*opcode*) extracted from the malwares. At first, they trained a Hidden Markov Model (HMM). Then they determined the important sequences of *opcodes* by eliminating the less important ones and using them to train a new HMM. They achieved a detection rate of 92.5%, a false positive rate of 0.77 % and an overall accuracy of 92.4% (Gharacheh, Derhami, Hashemi, & Fard, 2015).

In 2013, Alam *et al.*, from the University of Victoria and Gebze Institute of Technology, presented a technique named SWOD-CFWeight (Sliding Window Of Difference and Control Flow Weight). It relied on the extraction of *opcodes* to detect malwares. They demonstrated their approach using an intermediate language named MAIL (Malware Analysis Intermediate Language). They used 5305 sample programs: 1020 metamorphic malware, 4285 benign Windows and Cygwin programs. Their technique showed 94% detection rate, 3.1% false positive rate and a mean maximum accuracy of 96% (Alam, Horspool, & Issa, MAIL: Malware Analysis Intermediate Language - A Step Towards Automating and Optimizing Malware Detection, 2013) (Alam, Sogukpinar, Issa, & Horspool, 2015).

1.3.2.3 Human Factor treats and actions

In 2018, Geng *et al.*, from China Internet Network Information Center and Henan University of Technology, proposed their plug-in *RRPhish* that analyses the website's resources to determine if the site is legitimate or a phishing one. The authors used the WebExtensions to develop a FireFox extension and intercepted the traffic using the webRequest API. Then their extension searched the pages for input box, sensitive words, copyright notice and other parameters to determine the authenticity of the website. The evaluation of the system revealed that out of 200 phishing websites, their plug-in managed to detect 176 of them, surpassing Firefox blacklist and Trend Micro Smart Protection Network (Geng, Yan, Zeng , & Jin, 2018).

In the same year, in a joint work between Northeastern University, University of California Irvine, EMC/Dell CIRC and RSA, Oprea *et al.* designed a system called MADE (Malicious Activity Detection in Enterprises) that proactively detects HTTP network connections resulting from malware communication using supervised learning techniques applied to 40 highest ranked features extracted from web proxy logs. Using Random Forest Model, MADE achieved 97% precision in the set of 100 detected domains of highest risk, at a very small false positive rate (Oprea, Li, Norris, & Bowers, 2018).

In 2015, Li and Wang from the National Taiwan University approached the phishing problem by providing a tool named PhishBox that can be used to validate and detect malicious websites. At first, the tool collects features from the website like the Host information, the URL, the count of elements in the content and a screenshot. Three indexes were used to flag a website as invalid: being offline, redirected to another website or having invalid content. To determine if the website is offline, the authors used selenium to check the availability of the website. As for the redirection and content validity detection, they created a classifier for the metadata, a classifier for text content, and a classifier for image comparison. These classifiers used SVM classifiers with a Natural Language Processing

(NLP) to regroup the site as legitimate or not. As for the result of the classifiers, they were compared with the data found on the website PhishTank that is specialized in detection of phishing websites. To validate their work, they used 28607 phishing websites and 18671 legitimate websites. Accuracy and FPR were used as evaluation metrics. They managed to achieve an accuracy of 95% and FPR of 3.9% (Li & Wang, 2017).

In 2015, Sahu and Shrivastava from SATI College Vidisha proposed Kernel k-means clustering to categorize malware and phishing website. Term Frequency (TF) and Inverse Document Frequency (IDF) was used for the classification of Websites, while Instruction Frequency (IF) converted to TF and IDF was used in the classification of malwares. Their method achieved 43.58% accuracy on 2000 sample websites and 55.97% error rate on 1400 sample websites. On the other hand, they achieved 48.31% accuracy on a sample of 40 malwares and 45.00 % error rate on 45 malwares (Sahu & Shrivastava, 2015).

Based on the different findings, the most proposed and implemented solutions relay on neural-networks software based. Thus, one can identify the need of a near follow up and coordination between the different software that assure system security by the network administrator. Added to that, it is always safer to include a hardware part in the solution and to make sure that employees know how to deal with frequent threats and vulnerabilities they encounter during their daily work.

1.4 Analysis and Problem Statement

In this section, a more advanced classification of the above state of art will be presented. Added to that, in the last part, the problem statement will be introduced.

1.4.1 Analysis

Table 1.2 regroups the different threats for the three already listed levels and shows how each research contributed to limit these problems.

Table 1.2 Threats and solutions provided by the different research papers

Authors	NL	DL	HL	Methodology
Zhang *et al.* (2019)		X		software
Alsughayyir *et al.* (2019)	X			software machine learning
Geng *et al.* (2018)			X	software
Ajagekar and Jadhav (2018)	X			software machine learning
Kumar *et al.* (2018)		X		software machine learning
Oprea *et al.* (2018)			X	software machine learning
Eigner *et al.* (2016)	X			software machine learning
Farrokhmanesh and Hamzeh (2016)		X		software machine learning
Celik *et al.* (2015)	X			software machine learning
Li and Wang (2015)			X	software machine learning
Khatri and Abendroth (2015)		X		software
Saxe and Berlin (2015)		X		software machine learning
Gharacheh *et al.* (2015)		X		software machine learning
Sahu and Shrivastava(2015)			X	software machine learning
Roy *et al.* (2014)	X			software machine learning
Huang and Huang (2013)	X			software machine learning
Alam *et al.* (2013)		X		software
Jeon and Na (2010)	X			hardware

To the best of our knowledge, it is remarkable that not even a single paper treated the three threat levels all together. Hence, the remaining of this section will regroup the most frequent threats highlighted in the above references, and the applied actions to in order to provide safety, security and integrity of the system, hence of the data.

1.4.2 Problem Statement

The most recurring threats can be summarized by the following:

TH1: Denial of Service (DoS) attack;

TH2: Malware infections;

TH3: Phishing;

As for the proposed solutions, they are summarized by the below five techniques:

TE1: Machine Learning – it was the dominant choice by most of the papers mentioned previously. They used models like Naive Bayes Multinomial, K-Nearest Neighbor (k-NN), Hidden Markov Model, AdaBoost, Convolution Neural Network (CNN), Bayesian Model, Support Vector Machine (SVM), Bagging Classifier, Deep Learning Auto-encoders and k-MEANS;

TE2: Data Diode – It is hardware device that only allow unidirectional network traffic;

TE3: Sliding Window of Difference and Control Flow Weight (SWOD-CFWeight) algorithm – SWOD is a window that represents differences in MAIL (Malware Analysis Intermediate Language) patterns distributions, while CFWeight captures the control flow semantics of a program;

TE4: Repairing of the files – It is the reconstruction of files according to their original ISO file format;

TE5: Network based malware detection system – It consisted of creating a top-level layer through which all traffic is being directed in order to detect any malicious activity by analyzing the pattern of the flow;

Table 1.3. summarizes the correlation between these three parameters.

Table 1.3 Correlation between the threats and the applied techniques

	Threats	TH1	TH2	TH3
Techniques used	TE1	**X**	**X**	**X**
	TE2		**X**	
	TE3		**X**	
	TE4		**X**	
	TE5		**X**	

1.5 Problems encountered in the methodology phase

A lot of methodologies and software exist for modeling threats that can affect the organizations, but the one that are open source rely more on the individual who is in charge of the security of the organization. Even commercial ones can model the system but eventually human intervention is needed to tweak the final model to reflect truly the overall view of the threats that can impact the security of the organization.

Added to that, a limited work, when it exists, tackles the vulnerabilities and the threats at the different levels, *e.g.* the network, the device and the human factor levels. Such systems need a high technically skilled network and safety administrator(s) to be able to implement, update and maintain the system. In addition, the integrity degree of the three layers is not equally consistent regarding these three layers.

As for the simplicity of updating / maintaining such software, it was noted that most of the threats-based software require interference from the users to assess the real vulnerabilities on their systems for a classification purpose.

At the end, the absence of self and efficient learning modules for the end-users in order to increase their safety-based knowledge and awareness is remarkable.

1.6 Novelty of this work

From the analysis of the state of art presented earlier, it is noticed the absence of published works that tackled the security of the whole system. So, the novelty of this work is the decomposition of the solution to the three levels and presenting them in a single

coherent solution. The work will present solutions per level in one separate chapter, and then demonstrate the way it can be used together to provide organizations an overall protection against most threats that can are descried in OWASP top 10 (OWASP, OWASP API Security Project, 2020).

In more details, the proposed solution will address three main issues:

- Physical solution by integrating data diode to protect the network layer;
- Software solution by automatically updating the list of top vulnerabilities due to an active calculation process; this new calculation process aims to prioritize the vulnerabilities that need to be tackled in a first instant in order to increase the safety and security of the system;
- Software solution to harden the human factor in an organization by providing a simple and efficient self-learning and self-testing module for updating the employee's cyber-security awareness.

1.7 Methodology

The methodology emphasized in this report consists of defining the main threats at the three levels (*i.e.*, network, device and human), to model their sources and to propose some technical solutions. Added to that, the main focus relies on developing a complete secured proposition that does not need huge resources neither for its implementation nor its the application.

The first step of this work focuses on showing the proper solution for each level. This solution will be tested alone to test its efficiency and measure some performance indicators in order to validate it or to make some changes to ameliorate it.

Once the security at each level is confirmed, a full solution will be proposed and tested over a network where several types of threats and vulnerabilities will be injected. To validate this system, a modelling will be first shown than some measurements, in particular the response time and the security level, will be performed.

At the end, some technical comparisons between the proposed method and the existing solutions will be achieved. Such a comparison will help in showing the power of the proposed system and the possibility of reducing human intervention whenever this secured application is running.

1.8 Report structure

After presenting a general introduction as well as the main sources of threats and vulnerabilities in systems and the major contributions that were proposed so far to increase system security, this chapter has ended by proposing the methodology and the novelty of work.

This manuscript will be divided into six chapters as follow: in chapter 2, the modeling and security measurements of security and threats will be presented. In chapter 3 the network layer will be detailed with its security issues and the solution will be presented. In chapter 4, the device level security issues will be shown and a novel solution will be presented. In chapter 5, the security gap that the human factor will be presented and a solution will be delivered to solve some of the aspects around the security issues. Finally, in chapter 6, an implementation containing the three solutions presented in chapters 3,4 and 5 will be detailed.

Chapter 2 Modeling and Measurements of Security and Threats

Outline

2.1 Introduction

Threat modeling is a way to plan and optimize security operations. Security teams lay out their goals, identify vulnerabilities and outline defense plans to prevent and remediate cyber-security threats. It is a structured approach for identifying, quantifying, and addressing threats. It allows system security staff to communicate the potential damage of security flaws and prioritize remediation efforts. Thinking about security requirements with threat modeling can lead to proactive architectural decisions that allow for threats to be reduced from the start. Threat modeling could be defined as a set of expectations about malicious actions against the organization. Hereafter is a list of the most used threat modeling classification methodologies.

2.2 Threat Modelling Classification Methodologies

In this section, the most used classification methods, applied in the software engineering domain, will be presented briefly. The main parameters of each method will be shown. Due to space restrictions, we will not present applications examples; however, in the upcoming chapters, some of these techniques will be applied along with their applications.

2.2.1 Attack/threat trees

To model threats against computer systems, Bruce Schneier (Schneier, 2015) defined Attacks trees. By understanding all the different ways in which a system can be attacked, one can likely design countermeasures to thwart those attacks. Further, by understanding who the attackers are -- not to mention their abilities, motivations, and goals - maybe the system administrator can install the proper countermeasures to deal with the real threats.

Attack Trees provide a formal, methodical way of describing the security of systems, based on varying attacks. A tree structure is used to represent attacks against a system where the goal is the root node and the leaf nodes are the different ways of achieving that goal. Each node becomes a sub goal, and children of that node are ways to achieve that sub goal. “OR” nodes are used to represent alternatives and “AND” nodes are used to represent different steps toward achieving the same goal.

Once the tree is built, one can assign values to the various leaf nodes, then make calculations about the nodes. Once the values are assigned, one can calculate the security of the goal.

The attack attributes assist in associating risk with an attack. An Attack Tree can include special knowledge or equipment that is needed, the time required to complete a step, and the physical and legal risks assumed by the attacker. The values in the Attack Tree could also be operational or development expenses. An Attack Tree supports design and requirement decisions. If an attack costs the perpetrator more than the benefit, that attack will most likely not occur. However, if there are easy attacks that may result in benefit, then those need a defence.

2.2.2 STRIDE

STRIDE is an acronym that stands for Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. It was developed by Praerit Garg and Loren Kohnfelder at Microsoft for identifying computer security threats in 1999. It evaluates the system detail design by building Data-Flow Diagrams (DFDs). It provides a mnemonic for security threats in six categories. These categories are (Microsoft Threat Modeling Tool threats, 2022):

- Spoofing
- Tampering
- Repudiation
- Information disclosure
- Denial of service
- Elevation of privilege

Each threat is a violation of a desirable property for a system as shown in table 2.1. As for figure 2.1, it represents the six categories for security threats in a graphical representation.

Table 2.1 STRIDE threats and property definitions.

	Threat	Property Violated	Threat Definition
S	Spoofing identity	Authenticity	Pretending to be someone else
T	Tampering data	Integrity	Modifying data that belongs to others
R	Repudiation	Non-reputability	Claiming non-responsibility for an act
I	Information disclosure	Confidentiality	Providing none authorized information

D	Denial of Service	Availability	Exhausting needed service resources
E	Elevation of Privilege	Authorization	Allowing someone to do something not authorized to do.

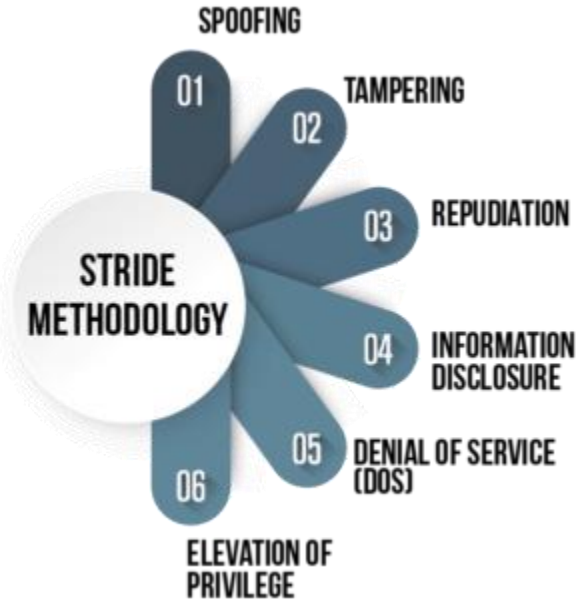


Figure 2.1 STRIDE Methodology set of threats

2.2.3 DREAD

DREAD model is proposed by Microsoft, and it is used to assess and rank threats based on their risk. It outlines five risk attributes to estimate the probability of an exploitation of a vulnerability from distinct aspects. These attributes are Damage (D), Reproducibility (R), Exploitability (E), Affected users (A), and Discoverability (Di).

- Damage (D): How much are the assets affected?
- Reproducibility (R): How easily the attack can be reproduced?
- Exploitability (E): How easily the attack can be launched?
- Affected users (A): What's the number of affected users?
- Discoverability (Di): How easily the vulnerability can be found?

Each risk attribute is scaled into three qualitative levels as high, medium, and low. Due to the property of a concrete threat, one of the three qualitative levels can be assigned for each risk attribute. All the five aspects need to be considered to assess the risk of a threat. The threat risk ranges from 0 to 10, and the DREAD model uses three integers 0, 5, and 10, to represent the three corresponding levels numerically (Microsoft, Microsoft Improving Web Application Security Threats and Countermeasures, 2003).

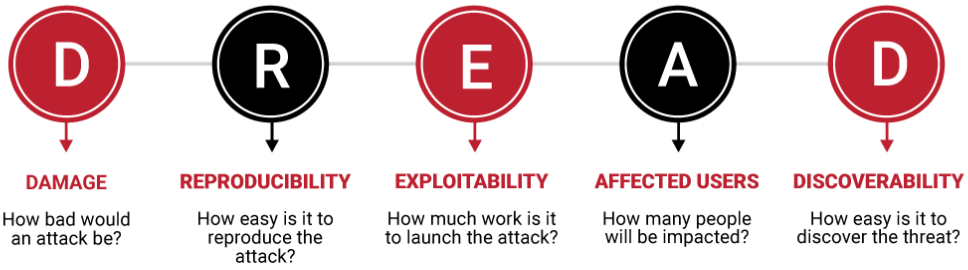


Figure 2.2 The DREAD model

2.2.4 Data Flow Diagrams (DFD)

Data-Flow Diagrams (DFDs) model a perspective of the system that is most readily understood by users – the flow of information through the system and the activities that process this information. DFDs provide a graphical representation of the system that aims to be accessible to computer specialist and non-specialist users alike. The models enable software engineers, customers and users to work together effectively during the analysis and specification of requirements. Although this means that the customers are required to understand the modelling techniques and constructs, in data-flow modelling only a limited set of constructs are used, and the rules applied are designed to be simple and easy to follow. These same rules and constructs apply to all Data-Flow Diagrams (*i.e.*, for each of the different software process activities in which DFDs can be used) (Kaufmann, 2003).

2.2.5 Adversarial Tactics, Techniques, and Common Knowledge or ATT&CK

The MITRE ATT&CK framework is a curated knowledge-base and model for cyber adversary behavior, reflecting the various phases of an adversary's attack lifecycle and the platforms they are known to target. The tactics and techniques abstraction in the model provide a common taxonomy of individual adversary actions understood by both offensive and defensive sides of cybersecurity. It also provides an appropriate level of categorization for adversary action and specific ways of defending against it.

The behavioral model presented by ATT&CK contains the following core components:

- **Tactics** denoting short-term, tactical adversary goals during an attack (the columns);
- **Techniques** describing the means by which adversaries achieve tactical goals (the individual cells);
- **Documented adversary** usage of techniques and other metadata (linked to techniques).

MITRE ATT&CK was created in 2013 as a result of MITRE's Fort Meade eXperiment (FMX) where researchers emulated both adversary and defender behavior in an effort to improve post-compromise detection of threats through telemetry sensing and behavioral analysis. The key question for the researchers was "How well are we doing at detecting documented adversary behavior?" To answer that question, the researchers developed ATT&CK, which was used as a tool to categorize adversary behavior.

The MITRE ATT&CK matrix contains a set of techniques used by adversaries to accomplish a specific objective. Those objectives are categorized as tactics in the ATT&CK Matrix. The objectives are presented linearly from the point of reconnaissance to the final goal of exfiltration or "impact". Looking at the broadest version of ATT&CK for Enterprise, which includes Windows, MacOS, Linux, AWS, GCP, Azure, Azure AD, Office 365, SaaS, and Network environments, the following adversary tactics are categorized:

- *Reconnaissance:* gathering information to plan future adversary operations, *i.e.*, information about the target organization;
- *Resource Development:* establishing resources to support operations, *i.e.*, setting up command and control infrastructure;
- *Initial Access:* trying to get into your network, *i.e.*, spear phishing;
- *Execution:* trying the run malicious code, *i.e.*, running a remote access tool;
- *Persistence:* trying to maintain their foothold, *i.e.*, changing configurations;
- *Privilege Escalation:* trying to gain higher-level permissions, *i.e.*, leveraging a vulnerability to elevate access;
- *Defense Evasion:* trying to avoid being detected, *i.e.* using trusted processes to hide malware;
- *Credential Access:* stealing accounts names and passwords, *i.e.*, keylogging;
- *Discovery:* trying to figure out your environment, *i.e.*, exploring what they can control;
- *Lateral Movement:* moving through your environment, *i.e.*, using legitimate credentials to pivot through multiple systems;
- *Collection:* gathering data of interest to the adversary goal, *i.e.*, accessing data in cloud storage;

- *Command and Control:* communicating with compromised systems to control them, *i.e.*, mimicking normal web traffic to communicate with a victim network;
- *Exfiltration:* stealing data, *i.e.*, transfer data to cloud account;
- *Impact:* manipulate, interrupt, or destroy systems and data, *i.e.*, encrypting data with ransomware.

Lots of techniques have been developed to detect threats over a network and to classify them based on their priorities and impact. However, a main limitation has been identified for all proposed and applied methods: the network administrator should always handle these threats and follow up on their implementation.

Thus, one of the features of the system that we will be proposing in the upcoming chapter is the autonomy feature which will be insured by applying some artificial intelligence method that will be used for updating the threats database and prioritizing the threats.

2.3 Threat Modelling Tools

In this section, the most used modelling techniques, applied in the software engineering domain, will be presented briefly. The main techniques of each method will be shown. As for the previous paragraph, we will not present applications examples; however, in the upcoming chapters, some of these tools will be applied along with their applications.

- ***IriusRisk:*** it offers both a community and a commercial version of the tool. This tool focuses on the creation and maintenance of a live Threat Model throughout the entire Software Development Life Cycle (SDLC). It drives the process by using fully customizable questionnaires and Risk Pattern Libraries, with flow diagramming and integration with DevSecOps (OWASP ZAP, BDD-Security, Threadfix...) to empower automation (Iriusrisk, 2022);

- ***Microsoft's free threat modeling tool:*** it uses the Microsoft threat modeling methodology, is DFD-based, and identifies threats based on the STRIDE threat classification scheme. It is intended primarily for general use (Microsoft, Microsoft, 2016);

- ***ThreatModeler:*** it is an automated threat modeling solution that fortifies an enterprise's SDLC by identifying, predicting and defining threats, empowering security and DevOps teams to make proactive security decisions. It provides a holistic view of the entire attack surface, enabling enterprises to minimize their

overall risk. It utilizes the VAST methodology to identify threats based on a customizable comprehensive threat library. It is intended for collaborative use across all organizational stakeholders (Threatmodeler, 2022);

- ***PyTM:*** it is an open-source Pythonic framework for threat modeling. It encodes threat information in python code, and processes that code into a variety of forms (Tarandach, 2022);

- ***SecuriCAD:*** it is a threat modeling and risk management tool by the Scandinavian company foreseeti. It is intended for company cyber security management, from CISO, to security engineer, to technician. SecuriCAD conducts automated attack simulations to current and future IT architectures, identifies and quantifies risks holistically including structural vulnerabilities, and provides decision support based on the findings. SecuriCAD is offered in both commercial and community editions (Foreseeti, 2022);

- ***Tutamantic "Automated Design Analysis":*** it is an interesting tool which provides microservices for threat modeling. In contrast to integrated tools, users upload a Visio file, and receive a spreadsheet of threats (Tutamantic, 2022);

- ***OWASP Threat Dragon Project:*** it is a free, open source, online threat modeling web application including system diagramming and a rule engine to auto-generate threats/mitigations (OWASP, OWASP Threat Dragon, 2022);

- ***Mozilla SeaSponge:*** it is a free, open source, accessible web-based threat modeling tool developed for Mozilla Winter of Security 2014 (Mozilla, 2014);

- ***OVVL the "Open Weakness and Vulnerability Modeller":*** it is free, open source threat modeling tool based on STRIDE with a particular focus on providing support for later stages in the secure development lifecycle (Schaad & Reski, 2019);

- ***SD Elements:*** it is a software security requirements management platform that includes automated threat modeling capabilities provided by Security Compass. A set of threats is generated by completing a short questionnaire about the technical details and compliance drivers of the application. Countermeasures are included in the form of actionable tasks for developers that can be tracked and managed throughout the entire Software Development Life Cycle (SDLC) (Securitycompass, 2022);

- ***Draw.io For threat modeling:*** it is a library for Draw.io (Diagrams.net, 2022) software that leverages the software's abilities for threat modeling by allowing the creation of Data Flow Diagrams (DFD) and Attack Trees (Henriksen, 2018).

To sum up, there is a large list of tools that are provided, even freely, to model threats. Most of these tools are open source and they are provided by well-known companies. Some autonomy is provided through these tools and the interaction with users is not a must.

This is the main reason why threats modelling was not considered largely in this work as the developed and provided techniques are sufficient to represents these viruses.

2.4 Cyber Security KPIs

Measuring the performance of a cyber-security system is essential for improving its efficiency resilience. Well-thought KPIs will look different in every organisation. They make problems and anomalies stand out in an actionable and solutions-oriented manner and can therefore help transform meaningful but copious amount of security data into quicker-to-digest information for senior management. Tracking too many KPIs can become a burden to both the analyst and the decision makers having to deal with an overload of data and information. When determining which KPIs should stay on the top list, one must check whether they help in decision making and still tick the boxes of the SMART criteria:

• Are they **Simple** to measure and have a clear purpose on how it impacts the security program?

• Are they **Measurable** in some way, quantitatively or qualitatively, with a method for measurement clearly defined and kept consistent?

• Are they **Actionable** and used as a driver for decisions to be made?

• Are they still **Relevant** to the security program?

• Are they **Time based**, so that variations and patterns are revealed over time?

KPIs should reflect each organisation's priorities, goals and objectives, but some examples of cyber security KPIs that are applicable to most organisations can inspire ideas when the IT administrator draft the institution own KPIs, such as:

1. Number of devices being monitored – Is this number increasing or decreasing? Why? Assess the security operation's workload and adjust if necessary.

2. Total number of events – Is this increasing or decreasing? Why? Assess the cost to value of incidents detection, response and recovery processes, and look for patterns to identify key risks.

3. Number of events per device or host – Are there any devices or hosts which are more prone to security issues than others, causing increased risk? Why? Assess detection success rates and key risks per device or host.

4. Mean Time to Detect – How long is it taking your organisation to detect a security event? Are there ways to reduce this time? How? Assess the detection success rates and your processes.

5. Mean Time to Resolve – How long is your organisation taking to resolve an actual security event? Are there processes or technologies that can help you reduce this time? What are they? Assess your mitigation success and processes.

2.5 Norms and standards

2.5.1 ISO/IEC 27001

ISO / IEC 27001 is an official standard for the information security of organizations. The ISO 27001 standard is focused on the higher-level goal of making sure that organizations have a structure (called a management system in ISO-speak) that ensures that the organization improves on information security. This ISMS is not an IT system, but a description of processes in your organization. It consists of goals, resources, policies and process descriptions. Only these higher-level elements are required by ISO 27001. There are two ideas that are not explicitly mentioned in ISO 27001 but that are important for understanding ISO 27001:

- The first idea is related to risk management: before taking any action, teams should understand which assets are worth protecting, what the risks are and how these risks are controlled;
- The second idea is the plan-do-check-act cycle: before acting, one needs to have a clear goal (plan) and think how he will check if the action works and what to do after the check. Below is a short checklist of all items that are described (ISO/IEC, 2022):
    - Organisation context description (4.1)
    - Stakeholders / interested parties in information security (4.2)
    - The ISMS scope (4.3)
    - Commitment from top management (5.1)

- Availability of an information security policy document (5.2)
- Roles and responsibilities for information security (5.3)
- Determining risks and opportunities (6.1.1)
- Defining and executing a process for risk assessment (6.1.2) and risk treatment (6.1.3). Part of this is to create a statement of applicability that indicates which best practice controls are or are not implemented
- Creating measurable security objectives (6.2)
- Resources for the ISMS (7.1)
- Appropriate training / competencies for the staff responsible for the ISMS (7.2) - see also the Information Security NL Special Interest Group as one way to fulfil this requirement
- Awareness for all staff in scope (7.3)
- Communication plan for internal and external communication about information security (7.4)
- Sufficient documentation about your ISMS including size of your organisation, complexity and competence of people (7.5.1). It must be updated appropriately (7.5.1) and controlled (7.5.3)
- Planning and control of operational aspects. Basically, this is about doing plan-do-check-act and prove this using documentation. (8.1)
- Planning a security risk assessment at regular intervals (8.2)
- Implementing the treatment plan (8.2, for treatment plan see 6.1.3)
- Monitoring the effectiveness of the ISMS, by seeing if the goals are reached (9.1)
- Planning and execution of regular internal audits (9.2)
- Planning and execution of regular management reviews (9.3)
- Taking management action if things do not go as planned (10.1). Again, this is part of doing plan-do-check-act correctly
- Making sure there is continuous improvement (10.2). This is not just about plan-do-check-act but also about collecting feedback on each meeting from participants and similar improvement steps.

Figure 2.3 ISO/IEC 27001

2.5.2 NIST Framework

The NIST Framework is voluntary guidance, based on existing standards, guidelines, and practices for organizations to better manage and reduce cybersecurity risk. This Cybersecurity Framework consists of three main components: The Core, Implementation Tiers, and Profiles.

Figure 2.4 The Cybersecurity Framework

As figure 2.4 shows, the framework categorizes all cybersecurity capabilities, projects, processes, daily activities into these 5 core functions (NIST, CYBERSECURITY FRAMEWORK, 2022):

IDENTIFY

The *Identify* function is focused on laying the groundwork for an effective cybersecurity program. This function assists in developing an organizational understanding to manage cybersecurity risk to systems, people, assets, data, and capabilities. To enable an organization to focus and prioritize its efforts, consistent with its risk management strategy and business needs, this function stressed the importance of understanding the business context, the resources that support critical functions, and the related cybersecurity risks. Essential activities in this function include:

- Identifying physical and software assets to establish the basis of an asset management program;
- Identifying the organization's business environment including its role in the supply chain;
- Identifying established cybersecurity policies to define the governance program, and the legal and regulatory requirements of the organization;
- Identifying asset vulnerabilities, threats to internal and external organizational resources, and risk response activities to assess risk;
- Establishing a risk management strategy including identifying risk tolerance;

- Identifying a supply chain risk management strategy including priorities, constraints, risk tolerances, and assumptions used to support risk decisions associated with managing supply chain risks.

PROTECT

The *Protect* function outlines appropriate safeguards to ensure delivery of critical infrastructure services and support the ability to limit or contain the impact of a potential cybersecurity event. Critical activities in this function include:

- Implementing protections for Identity Management and Access Control within the organization including physical and remote access;
- Empowering staff through security awareness training including role based and privileged user training;
- Establishing data security protection consistent with the organization's risk strategy to protect the confidentiality, integrity, and availability of information;
- Implementing processes and procedures to maintain and manage the protections of information systems and assets;
- Protecting organizational resources through maintenance, including remote maintenance activities;
- Managing technology to ensure the security and resilience of systems, consistent with organizational policies, procedures, and agreements.

DETECT

Detecting potential cybersecurity incidents is critical and this function defines the appropriate activities to identify the occurrence of a cybersecurity event in a timely manner. Activities in this function include:

- Ensuring the detection of anomalies and events, and their potential impact is understood;
- Implementing continuous monitoring capabilities to monitor cybersecurity events and to verify the effectiveness of protective measures including network and physical activities.

RESPOND

The *Respond* function focuses on appropriate activities to take action in case of a detected cybersecurity incident and supports the ability to contain the impact of a potential cybersecurity incident. The essential activities for this function include:

- Ensuring response planning process that are executed during and after an incident;
- Managing communications with internal and external stakeholders during and after an event;
- Analysing the incident to ensure effective response and supporting recovery activities including forensic analysis and determining the impact of incidents;
- Performing mitigation activities to prevent expansion of an event and to resolve the incident;
- Implementing improvements by incorporating lessons learned from current and previous detection / response activities.

RECOVER

The *Recover* function identifies appropriate activities to renew and maintain plans for resilience and to restore any capabilities or services that were impaired due to a cybersecurity incident. Timely recovery to normal operations is impressed upon, to reduce the impact from a cybersecurity incident. Essential activities for this function somewhat overlap with those of Respond and include:

- Ensuring the organization implements recovery planning processes and procedures to restore systems and/or assets affected by cybersecurity incidents;
- Implementing improvements based on lessons learned and reviews of existing strategies;
- Coordinating internal and external communications during and following the recovery from a cybersecurity incident.

2.5.3 The Payment Card industry Data Security Standard (PCi DSS)

It is a set of requirements intended to ensure that all companies that process, store, or transmit credit card information maintain a secure environment. It was launched on September 7, 2006, to manage PCI security standards and improve account security throughout the transaction process. An independent body created by Visa, MasterCard,

American Express, Discover, and JCB, the PCI Security Standards Council (PCi SSC) administers and manages the PCi DSS. Interestingly, the payment brands and acquirers are responsible for enforcing compliance, rather than the PCi SSC.

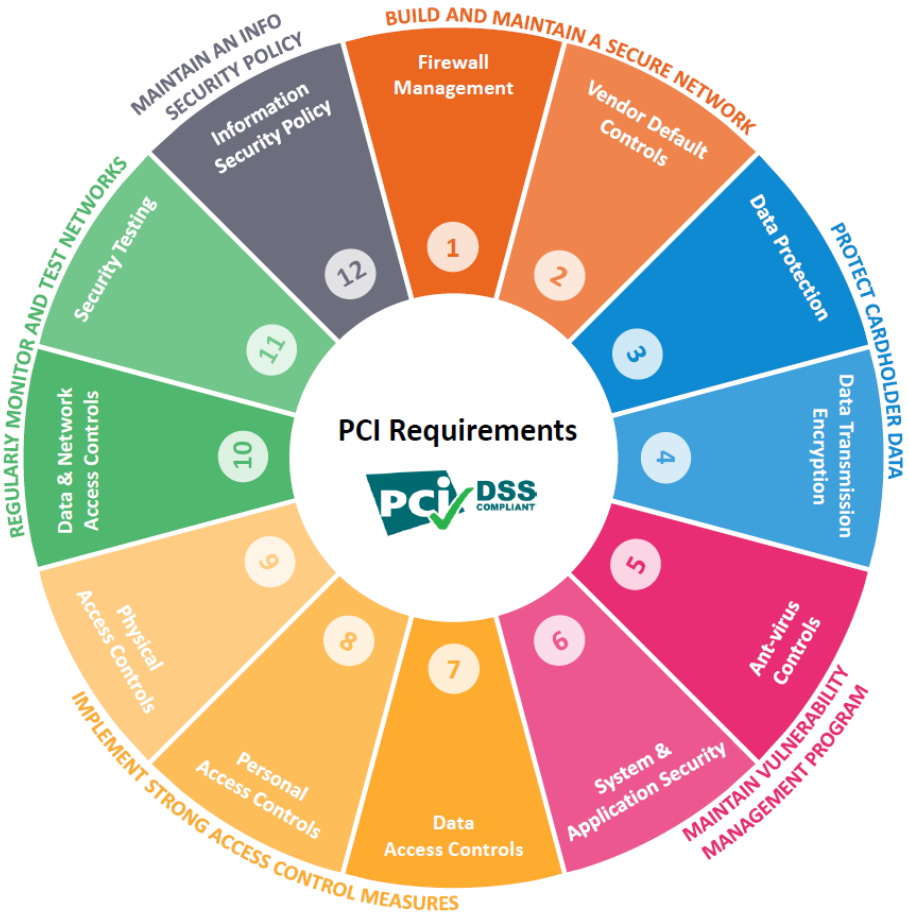


Figure 2.5 PCi DSS 12 requirements for compliance

Referring to figure 2.5, the 12 requirements for PCi DSS compliance are provided as follow:

2.5.3.1. Use and maintain firewalls

Firewalls essentially block access of foreign or unknown entities attempting to access private data. These prevention systems are often the first line of defense against hackers (malicious or otherwise). Firewalls are required for PCi DSS compliance because of their effectiveness in preventing unauthorized access.

2.5.3.2. Proper password protections

Routers, modems, Point of Sale (POS) systems, and other third-party products often come with generic passwords and security measures easily accessed by the public. Too

often, businesses fail to secure these vulnerabilities. Ensuring compliance in this area includes keeping a list of all devices and software which require a password (or other security to access). In addition to a device/password inventory, basic precautions and configurations should also be enacted (e.g., changing the password).

2.5.3.3. Protect cardholder data

The third requirement of PCi DSS compliance is a two-fold protection of cardholder data. Card data must be encrypted with certain algorithms. These encryptions are put into place with encryption keys — which are also required to be encrypted for compliance. Regular maintenance and scanning of Primary Account Numbers (PAN) are needed to ensure no unencrypted data exists.

2.5.3.4. Encrypt transmitted data

Cardholder data is sent across multiple ordinary channels (i.e., payment processors, home office from local stores, etc.). This data must be encrypted whenever it is sent to these known locations. Account numbers should also never be sent to locations that are unknown.

2.5.3.5. Use and maintain anti-virus

Installing anti-virus software is a good practice outside of PCi DSS compliance. However, anti-virus software is required for all devices that interact with and/or store PAN. This software should be regularly patched and updated. The POS provider should also employ anti-virus measures where it cannot be directly installed.

2.5.3.6. Properly updated software

Firewalls and anti-virus software will often require updates. It is also a good idea to update every piece of software in a business. Most software products will include security measures, such as patches to address recently discovered vulnerabilities, in their updates, which add another level of protection. These updates are especially required for all software on devices that interact with or store cardholder data.

2.5.3.7. Restrict data access

Cardholder data is required to be strictly "need to know." All staff, executives, and third parties who do not need access to this data should not have it. The roles that do need sensitive data should be well-documented and regularly updated — as required by PCi DSS.

2.5.3.8. Unique IDS for access

Individuals who do have access to cardholder data should have individual credentials and identification for access. For instance, there should not be a single login to the encrypted data with multiple employees knowing the username and password. Unique IDs creates less vulnerability and a quicker response time in the event data is compromised.

2.5.3.9. Restrict physical access

Any cardholder data must be physically kept in a secure location. Both data that is physically written or typed and data that is digitally-kept (*e.g.*, on a hard drive) should be locked in a secure room, drawer, or cabinet; not only should access be limited, but anytime the sensitive data is accessed, it should be kept in a log to remain compliant.

2.5.3.10. Create and maintain access logs

All activity dealing with cardholder data and PAN require a log entry. Perhaps the most common non-compliance issue is a lack of proper record keeping and documentation when it comes to accessing sensitive data. Compliance requires documenting on how data flows into the organization and the number of times access is needed. Software products to log access are also needed to ensure accuracy.

2.5.3.11. Scan and test for vulnerabilities

All ten of the previous compliance standards involve several software products, physical locations, and likely a few employees. There are many things that can malfunction, go out of date, or suffer from human error. These threats can be limited by fulfilling the PCi DSS requirement for regular scans and vulnerability testing.

2.5.3.12. Document policies

Inventory of equipment, software, and employees that have access will need to be documented for compliance. The logs of accessing cardholder data will also require

documentation. How information flows into your company, where it is stored, and how it is used after the point of sale will also all need to be documented.

In this section, the three well-known norms and standards to define system safety and security were proposed. Added to that, the main frameworks of each method are presented. These standards will serve to validate the method assuring the system security that we will develop all along this report.

2.6 Security threat detection

Threat detection is the practice of analyzing the total of a security ecosystem to identify any malicious activity that could compromise the network. If a threat is detected, then mitigation efforts must be enacted to properly neutralize the threat before it can exploit any present vulnerabilities.

A robust threat detection program should employ:

- **Security event threat detection technology** to aggregate data from events across the network, including authentication, network access, and logs from critical systems;
- **Network threat detection technology** to understand traffic patterns on the network and monitor traffic within and between trusted networks, as well as with the internet;
- **Endpoint threat detection technology** to provide detailed information about possibly malicious events on user machines, as well as any behavioural or forensic information to aid in investigating threats.

By employing a combination of these defensive methods, the IT administrator will be increasing the chances of detecting and mitigating a threat quickly and efficiently. Security is a continuous process, and nothing is guaranteed. It will be up to the system administrator and the deployed resources and processes put in place to keep the business as secured as possible.

2.7 Conclusion

In this chapter, all models, techniques, tools and norms, related to the system security and vulnerability, have been presented. However, the applications were not provided at this level due to space constraints.

Thus, in the next three chapters, the threats and the security solutions regarding the network, the device and the human factor layers will be presented. In these chapters, the modelling, the proposed solutions, the validation and the implementation of these solution will be projected.

So, in chapter 3, we will start by tackling all security threats related to the system level by deploying some of the tools that were introduced in this chapter. This chapter will start by showing the methodology of work of the applied work.

Chapter 3 Proposed Solutions and Procedures for Network Layer Security

Outline

3.1 Definition

The network layer represents the part of the ITS that handles the network traffic. It incorporates the devices along with their peripherals that purely handles the network flows. This is the first attack vector where threats can be injected in the system. Thus, the main contribution of this chapter is to define the essential threats, their origin and the way to secure data from being theft, altered or injected. Based on these threats, some hardware devices and configurations will be proposed in order to ensure the maximum security of the network with respect to external interfaces.

3.2 Methodology of work

In this chapter, the security issues related to the network layer will be presented, and a solution will be offered to remediate these issues. The IDEF0 will be used to model the different threats, control variables, solutions and resources to have better visibility of the work needed to be done to secure this level. An application of the proposed solution will be applied to a medical PACS (Picture Archiving and Communication System) system and measurements are used to emphasize on the usability of the solution.

3.3 Overview of Network Layer

Although the threats and their solutions have been presented in details in chapter 1, this section aims to focus on the most active and efficient threats and viruses that address the network layer. This overview will present what our proposed solution aims to solve. It is a major constraint to consider the efficiency and the simplicity of the derived solution(s).

Network security is a major concern for stakeholders that have sensitive data stored or transmitted between devices. The failure to design secure networks results in creating a gap that can let hackers infiltrate the network and jeopardize the integrity of the data stored or exchanged (Papadimitratos & Haas, 2006). The nature of the harm can be the destruction, the modification or the exposure of data (Rupprecht, Dabrowski, Holz , Weippl, & Pöpper, 2018). This forces all companies to invest in securing their networks, mainly when they have some end-points exposed to internet, in order to minimize the risk of losing data.

The main threats that have been encountered during the past decade on the network layer are caused by Advanced Persistent Threats (APT) that apply a wide variety of attacks such as Distributed Denial-of-Service (DDoS) attack that disrupts the normal traffic of the services on the target server through overwhelming it with garbage traffic, or by executing Domain Name System (DNS) cache poisoning or Address Resolution Protocol (ARP) poisoning to introduce malicious actors to the network and help in the exfiltration of the data [63]. A typical cyber security attack consists of an attacker trying to enter a network as

quickly as possible, steal the information, and exit. However, APTs goal is to achieve penetration and a continuous access to the systems it breached.

As for the applied solutions, they rely on software and hardware. Firewalls were in the past the primary defense from network attacks, but its functionality doesn't prevent the attacker from exfiltrating data once a server is compromised because it prevents incoming traffic but can't block outgoing traffic. Next-generation firewalls are now used as a network security device because it provides additional functionalities than the traditional firewall as they are more application aware since they contain intrusion prevention and threat intelligence modules. Nevertheless, the most performing solutions are the Intrusion Detection Systems (IDS). IDS is an intrusion detection security system for malicious actions that intruders use to compromise the system. By applying rules, it is possible to halt and even block the intrusion from a particular resource. The IDS basic activity is monitoring the packets on the network and the system behavior, then display an alert when an abnormal activity happens in the network or host (Ali & Yong, 2011). They can be classified into two classes: Host-based (HIDS) and Network-based (NIDS).

Basically, intrusion detection systems monitor, evaluate, detect, and respond to the security threats. The presence of two classes is correlated to the position of the entity inside the architecture of the network. The HIDS detects intrusions by collecting and analyzing information from the computer that hosts the service; the installed agent is responsible for monitoring and reporting. The NIDS, on the other hand, is more and more deployed because it monitors the network traffic for particular network segments and detects attacks by analyzing captured network packets thus giving large visibility on what is going across the network. Several studies have been conducted on network security and the ways to increase it.

Two major entities affect the network security: the running software on the connected network devices and the devices themselves. Several models were created to define the various stages of a cyber-attack; Lockheed-martin cyber kill chain model is one of the most used by cybersecurity researchers. The Cyber Kill Chain creates a common format and language for cybersecurity personnel to evaluate security events by association, motivation, and integration, where they could be aggregated and correlated according to objective and attack vector. So, the discussed solutions cannot totally protect the organization if they failed to detect the threat at an early stage. Thus, the possibility of a malware to persist is always possible, and its communication with the attacker will allow the extraction of the data.

As presented, the solutions can provide security but to a certain extent. This is not acceptable in some environments where the confidentiality, integrity and availability of the assets and the data is critical. Enterprises with such concerns rely on air gapping their network to isolate it from external connections. The concept of Air Gapped network consists

of creating a network with no physical external communication. However, today's modern demands make these isolated networks non-practical given the need for flows of data from outside the island networks. So, these network islands might be secure, but they rely on outside data which means that data is often transferred between networks using hard-to-control media such as USB sticks or DVDs. Such media is not without risks because they can be used to leak data out or even can be used to infiltrate malwares such as the Stuxnet incident.

To further understand the threats and solutions, the graphical *Process Modeling Methodology* Integrated DEFinition (IDEF0) will be used to model this level. Thus, the diagram presented in Figure 3.1 shows the general form of the IDEF0 diagram along with all its parameters (Menzel & Mayer, 1998). The control variables will be pointing from the upper side while the resources will be pointing from the lower side. The threats will be pointing diagonally from the upper left corner while the solutions of these threats will be pointing from the lower left corner. The input variables will be from the left side while the output variables will be pointing outwards from the right side.

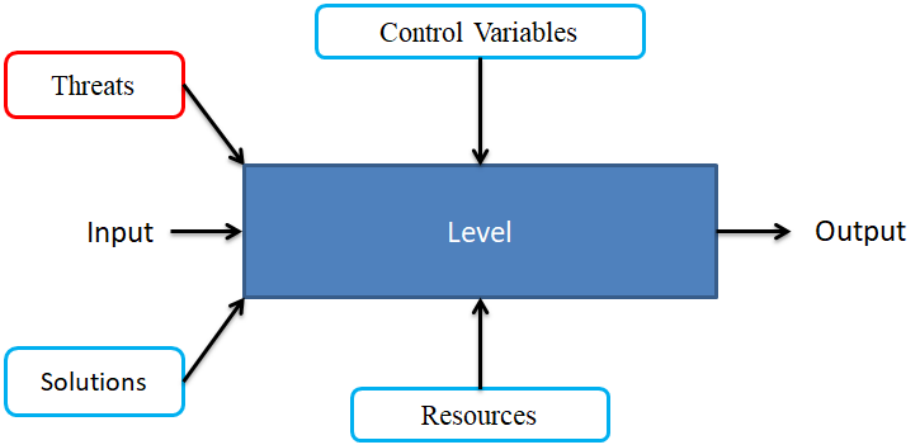


Figure 3.1 General form of the block diagram

So, concerning the network level, an extended IDEF0 representation is emphasized in Figure 3.2. From this diagram, one can identify the following:

4 ***Control variables***, which represent the main protocols to assure security at the level of networks, are highlighted by the use of ISO/IEC standards, NIST, SOC3,… norms;
5 ***Inputs***, which are the main sources of traffic and threats/vulnerabilities in networks, depend on the internet and intranet connections as well as the use of external devices/memory sticks;
6 ***Threats,*** which represent the types of malicious actors that affect the proper functionality of the network, can be represented mainly by the DOS Attack;

7 ***Solutions,*** which represent the list of actions (software and hardware) that will limit the threats effects on the network, are implemented through the use of one or more protection solutions such as NIDS, IPS, firewalls, segmentation of networks and encryption;

8 ***Output,*** which represent the traffic flow over the network after being filtered by the solutions and governed by the control variables, designates the secured flow of data.

As an example, let's consider the following scenario: a DOS cyber-attack is occurring on the network from an unauthorized entity. IDS will detect it and stop it; furthermore, applying the standards defined in ISO/IEC 27002 can be efficient in handling this cyber-attack mitigating its consequences.

Figure 3.2 shows the block diagram for the network level along with all actors following the IDEF0 model.

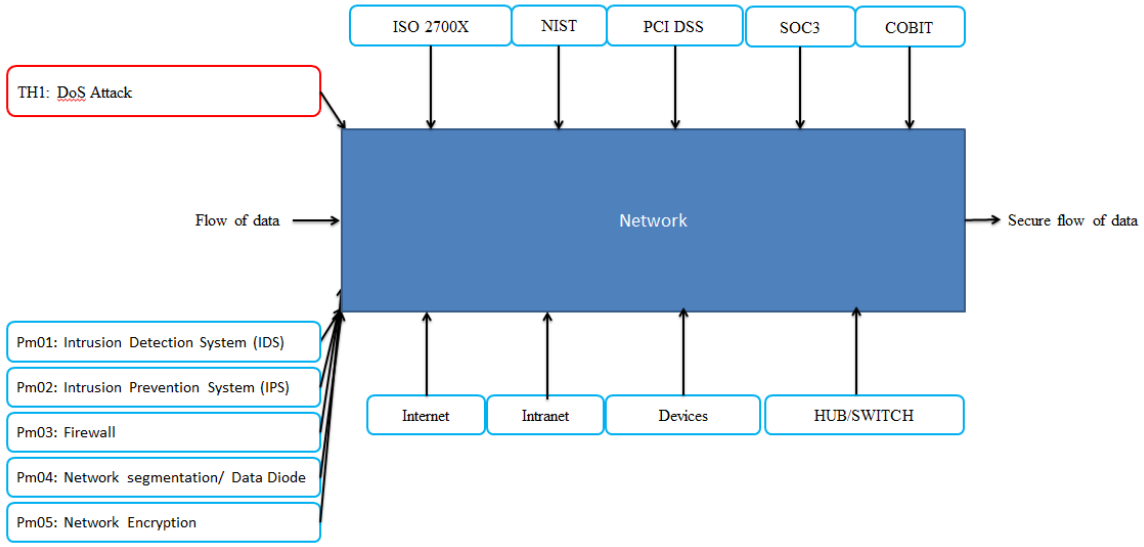


Figure 3.2 Block diagram for the Network level

To sum up, this section has presented a modelling of the threats and the vulnerabilities over the network layer using IDEF0 scheme. Added to that, the most relevant solution to limit attacks over network have been projected along with the limitation of each method. This leads to propose, in the next section, our solution that will ensure a better security for the network layer with lower cost and resources.

3.4 Proposed solution: The data diode

After presenting the different available solutions regarding the network layer security, our solution deals with solving the main issues encountered in these solutions. The

proposed solution is divided into two parts: hardware and software. The first one relies on the use of a data diode whereas the second is a customized software that handles the traffic from one side to the other side of the data diode.

Data diodes provide the benefits of an air gapped network yet still provide connectivity from the external network. Introducing security stack for network perimeters increases the complexity of handling, maintenance and, sometimes, it can be a source of vulnerability and weakness. Thus, the data diode is the best solution possible to increase network security by creating a unidirectional network flow and thus disallowing a malware to communicate back to the attacker and thus protecting the network from further exploitation, data theft, alteration and exposure.

Data diodes are the best solutions in protecting secrets and assets. They protect government classified data, and private sector intellectual property from adversaries. Added to that, they protect electronic assets from cyber-attacks by means of the network. One data diode can provide resiliency and eliminate the previously stated risks. Figure 3.3 represents the way data diode controls the flow of data.

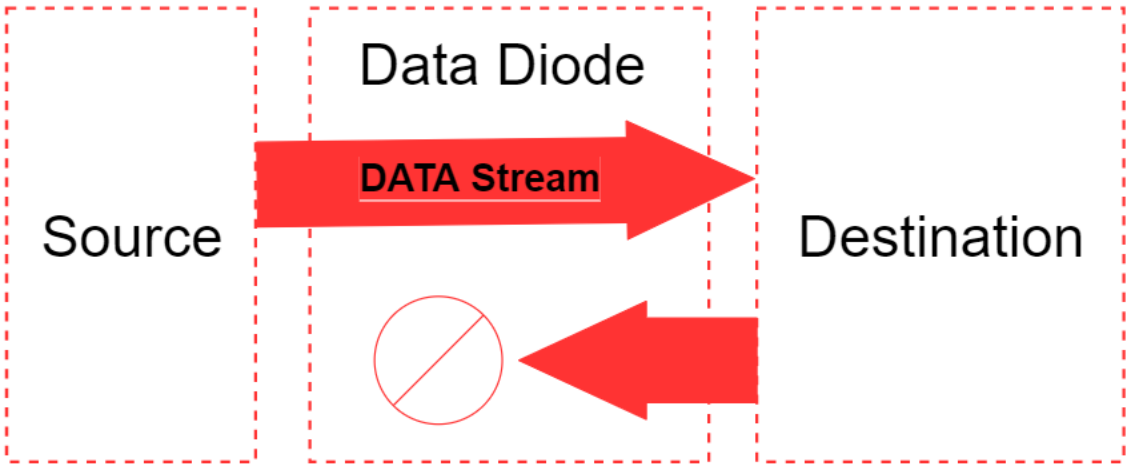


Figure 3.3 Data diode unidirectional communication

Data diodes provide a physical mechanism for enforcing strict unidirectional communication between two networks. They can only send information from one network (*a.k.a.* the "low" network) to another network (*a.k.a.* the "high" network). The high network often contains data with higher classification level than the low network. They are often implemented by removing the transmitting component from one side and the receiving component from the other side of a bidirectional communication system. However, this method requires a third entity simply to supply a carrier signal to the transmitter which will not work if it does not receive the appropriate carrier signal.

However, the novelty of this work consists of creating the data diode with only two entities: a transmitter and a receiver. The carrier signal needed by the transmitter is provided by the transmitter itself, thus eliminating the need for the third entity.

Figure 3.4 shows how data diode prevents the hacker from extracting the data by limiting its communication back to him. Based on the UML sequence diagram, the attack begins when the hacker sends a malicious document to the system. Since the data diode is implemented, it will handle the document, and scan it with the antivirus. If the antivirus didn't find anything wrong with the document – it may contain a Zero-Day exploit –, the document is forwarded to the designated destination. The receiver of the document opens it and the exploit is executed. The trojan will try to connect back to the hacker to start receiving the commands. However, since the data diode is a unidirectional flow of data, the trojan will fail to contact its Command and Control (C&C). In this case, either it will self-destruct to erase its presence, or with time, it will become obsolete and eventually the antivirus will become aware of its presence and removes it.

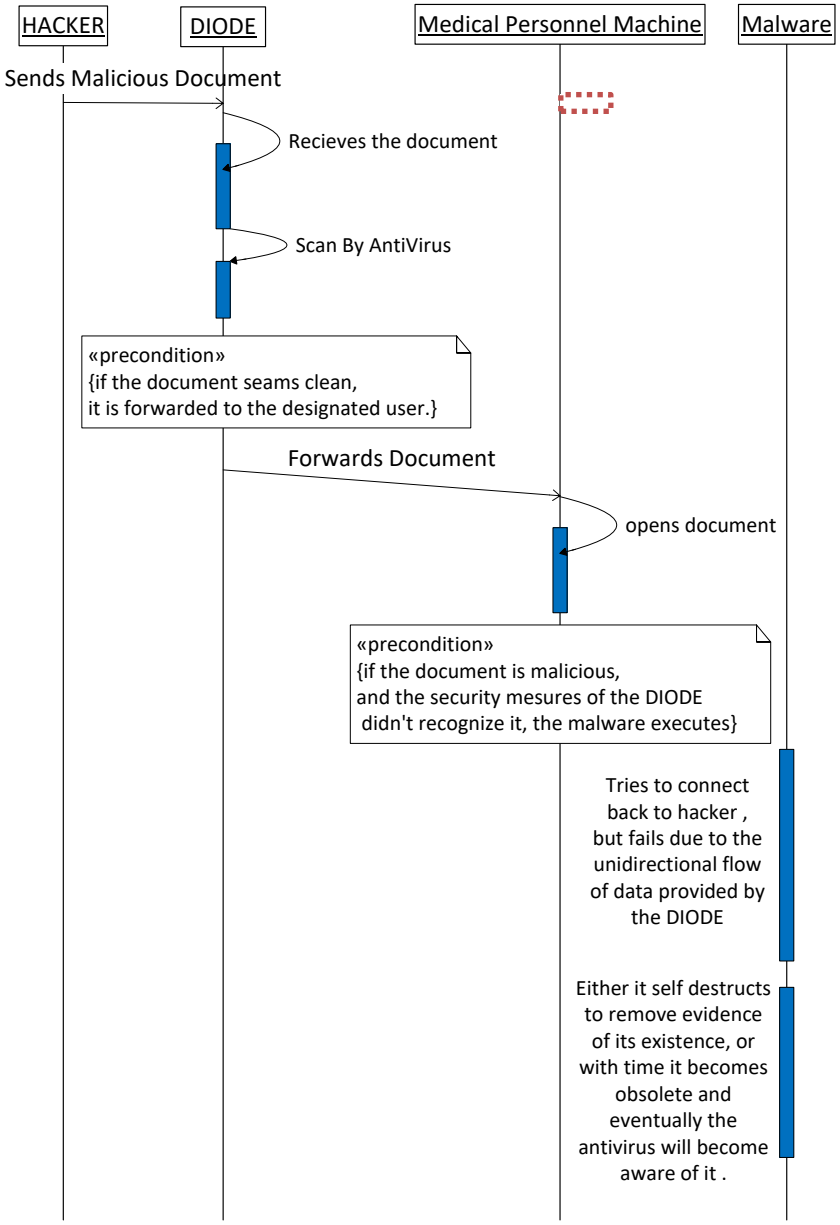


Figure 3.4 UML sequence diagram representing the proposed secured solution

3.5 Implementation of the solution

After presenting the usability and importance of a data diode in assuring maximum security for the network layer, this section is dedicated to show how we implemented it. As already mentioned, a data diode consists of two major components: the hardware component, and the software component.

3.5.1 Hardware component

The main concept behind the data diode is the physical separation of connectivity between the non-secure and the secure network through a unidirectional network connection.

In networks, digital network communications ensure end-to-end data delivery. This is done by the use of Network Interface Cards (NIC). It connects the computer to the computer network thus allowing sending and receiving of data. It is considered part of the Data Link Layer of the OSI model. The network card type is determined by the interface of the network in use: RJ45, SFP, BNC, AUI.

All of these NICs have a sending interface and a receiving one. Since we are looking for a cost-effective solution that can be implemented, we used the RJ45 NIC to create the data diode.

RJ-45 data cable contains 4 pairs of wires consisting of a solid-colored wire and a strip of the same color. Although there are 4 pairs of wires, 10BaseT/100BaseT Ethernet uses only 2 pairs: Orange and Green. The other two colors (blue and brown) may be used for a second Ethernet line or for phone connections. Wiring of the RJ45 cable follows two standards: T-568A and T-568B. The two wiring standards are used to create a cross-over cable (T-568A on one end, and T-568B on the other end), or a straight-through cable (T-568B or T-568A on both ends). Figure 3.5 represents the difference in the wiring of cables between the T-568A and the T-568B in terms of cable colors.

The cross-over cables are used when connecting Data Terminating Equipment (DTE) to DTE, or Data Communications Equipment (DCE) to DCE equipment, such as computer to computer, computer to router or gateway to hub connections. The straight-through cables are used when connecting DTE to DCE, such as computers and routers to modems (gateways) or hubs (Ethernet Switches). The DTE equipment terminates the signal, while DCE equipment do not. Figure 3.6 shows how the wiring is done in the straight-through cabling versus the cross-over.

As figure 3.6 shows, two TX lines must be connected to two RX in order to pair and connect any network device. So, in theory, to implement a unidirectional traffic, one has just to disconnect the RX of the sender. Yet, doing this will turn the sender state to disconnected for the operating system. Ethernet technology uses Carrier Signal (CS) to ensure an end-to-end connection is present. Once the RX connection is disconnected, the CS will be lost, and the ethernet connection will be considered disconnected. In order to overcome this issue, Scott (Austin, 2015) used a third media to provide only RX to the disconnected sender. Figure 3.7 shows how it was implemented.

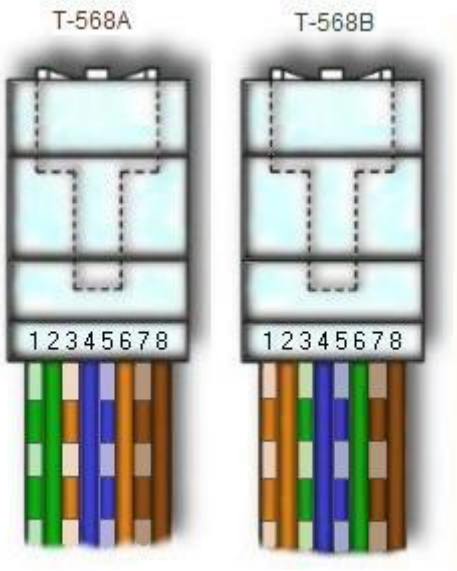


Figure 3.5 T-568A vs T-568B cable wiring

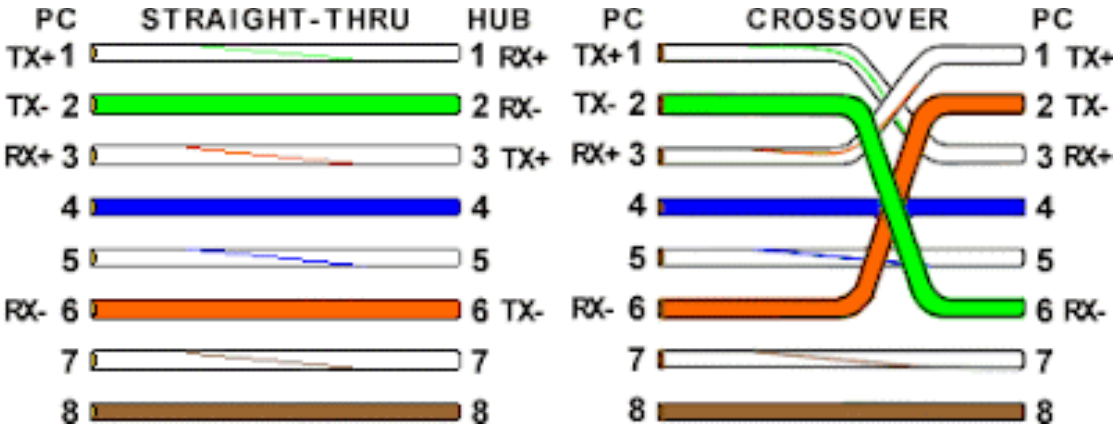


Figure 3.6 The straight-through vs cross-over wiring

As already mentioned, the novelty of this work is to remove this third party; thus, the sender works without using a third media by providing the CS from itself. Figure 3.8 shows the new design of the data diode.

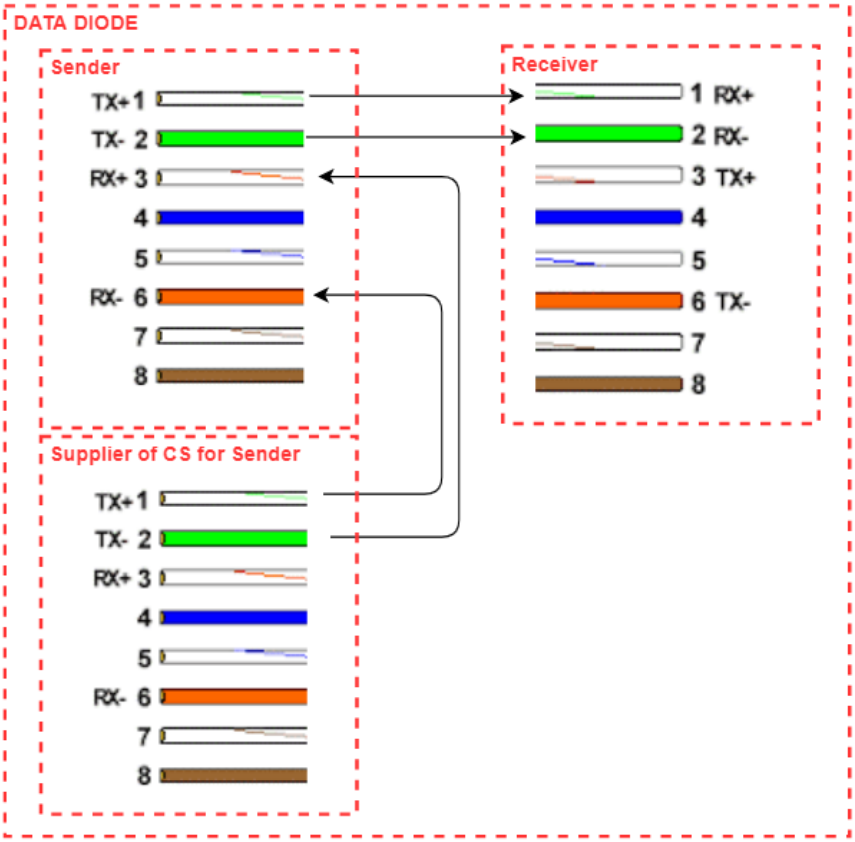


Figure 3.7 Data Diode using a third media to supply CS for the sender

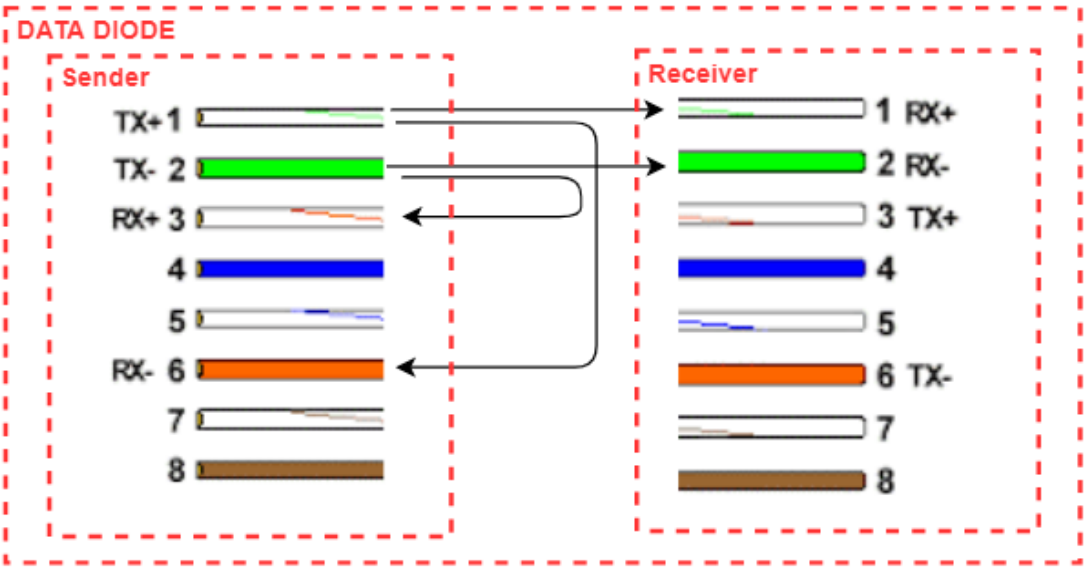


Figure 3.8 Data Diode without the need of a third media to provide CS

To implement the new design, we used two RJ45 CAT6 female connectors as shown in Figure 3.9. This design is cost effective, simple to implement and works with any CAT6 cable. The main innovation in this design is that we managed to connect pins 1 and 2 on the IN RJ45 to pins 3 and 6 of the OUT RJ45 and, in the same time, to pins 3 and 6 of the IN RJ45. Thus, the result is a solution that is simple, effective and easy to deploy.

Figure 3.9 Data Diode unidirectional physical link

Furthermore, since our data diode is based on ethernet network cards, the Auto MDI-X on the network cards of the Node IN and Node OUT need to be disabled in order to preserve and protect the data diode from physical manipulation and allow the reversing of the data flow. To connect two ports of the same configuration (MDI to MDI or MDI-X to MDI-X), an ethernet crossover cable is needed to transmit the signals over the cable, so the transmitter and receiver match at the connector level. The Auto MDI-X feature automatically detects the required cable connection type and configures the connection appropriately, removing the need for crossover cables to interconnect switches or connecting PCs peer-to-peer.

Finally, a major limitation of having unidirectional traffic makes all applications that rely on TCP/IP non-usable due to the fact that no reply will be received and thus the handshake will fail. The only option available is the use of protocols that support unidirectional protocol such as UDP or Asynchronous Transfer Mode (ATM) to broadcast the data from one end to the other.

Having that limitation in the data diode makes it necessary to develop custom software having a double functionality:

- handling the files coming in;
- handling the files received on the other side.

3.5.2 Software component

After connecting the physical part, a software is needed to validate the proper functioning of the hardware. To send UDP packets, the "UDPcast" software was chosen to exchange files (Lee, et al., 2014). It is an open-source file transfer tool that can send data simultaneously using UDP protocol. It has many built-in features, but the most important is specifying the bandwidth to use. This feature allowed us to make tests using several connection speeds while varying the file sizes in order to compare the response time and the delays generated by the data diode.

Due the absence of receipt acknowledgment from the receiver side, we used hashing to verify the proper reception of the file. In order to generate the Hash value of the file, MD5sum software was used (Den Boer & Bosselaers, 1993). MD5sum calculates the MD5 hash of the file and acts as a compact digital fingerprint. The file and its hash are archived in one file that is sent over the diode. So, since there is no reverse channel, the software must:

- rely entirely on forward error correction;
- perform checksums (or better) to reject bad data;
- integrate redundancy to assure data reception;
- generate operational status determined from downstream.

As an example of the full procedure, figure 3.10 shows the developed data diode while being integrated in the network. In more details, it serves as a junction between both networks whereas the software is operating at the NodeIn and the NodeOut extremums. In addition, figure 3.11 shows a full UML sequence diagram presenting the complete scenario since receiving the data at the input of the network until sending another data to another network.

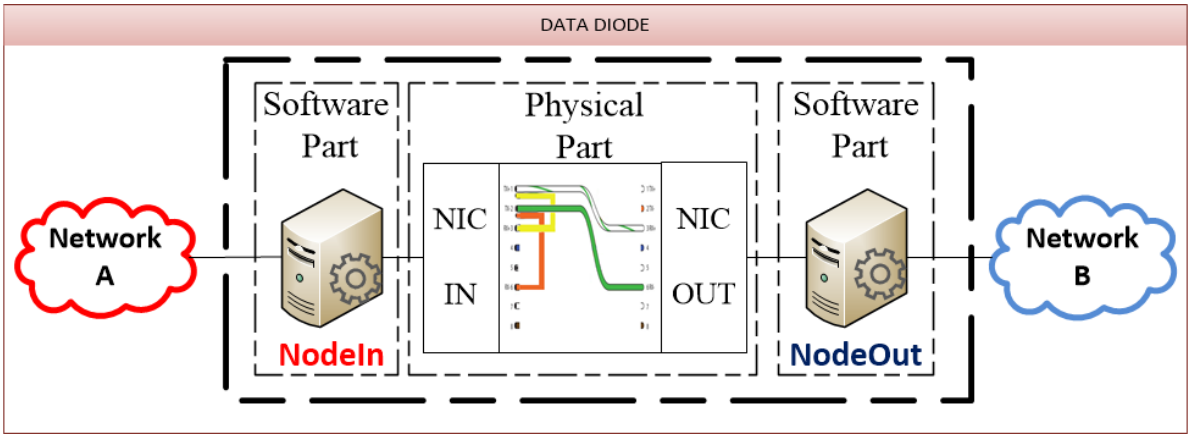


Figure 3.10 Implementation of the developed diode within the network

Figure 3.11 shows the flow of a file from source to destination. First the user sends the file to the **PC A** which is on the receiving edge of the Data Diode. The file is scanned with the antivirus then a **SHA256** hash is created for it. Both the file and its hash will be put in a zipped file and sent over UDP to **PC B** which is on the receiving edge of the data diode. The zipped file will be unzipped and a SHA256 hash will be made for the file. If both the received hash and the calculated one match, the file is forwarded to the destination. In case of match failure, and since there is no way to notify the sender to send the file again, the procedure of sending the file is made three times for redundancy and when the first one matches in hashes, the others are discarded.

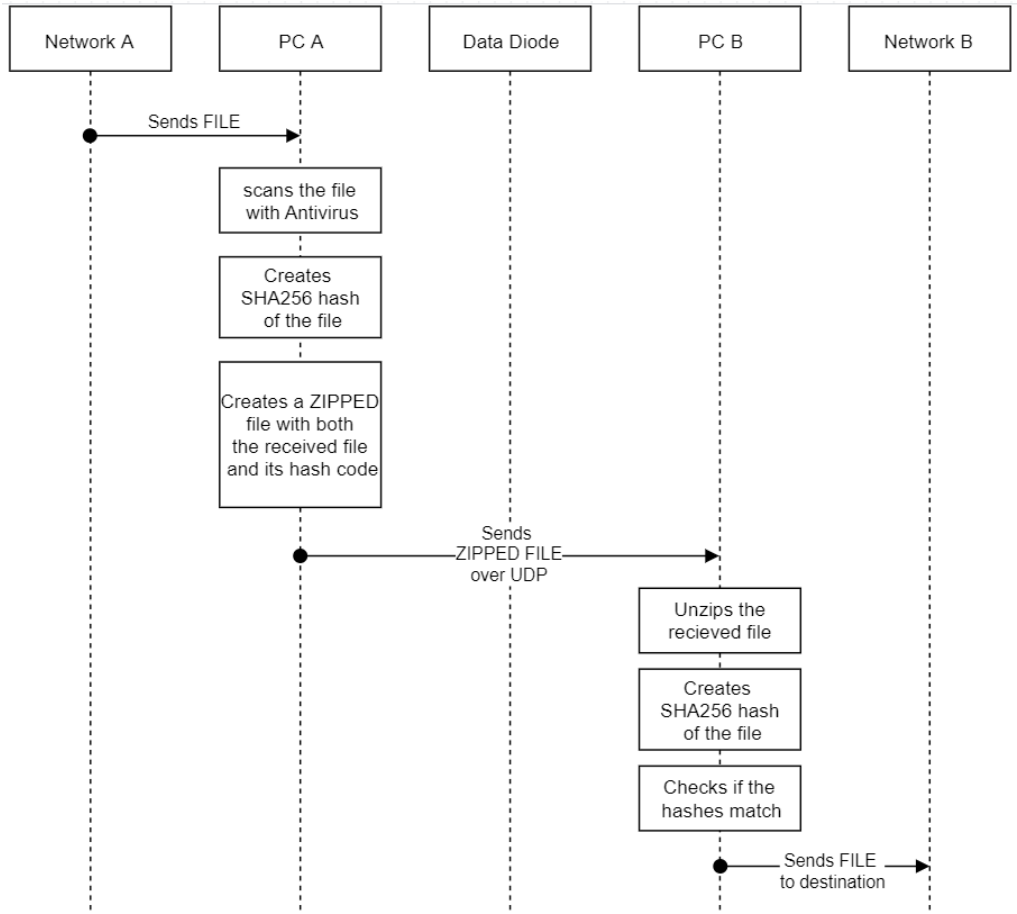


Figure 3.11 UML sequence diagram representing the full communication between the secured network and other networks

3.6 Case study: PACS system

Medical Imaging Devices (MIDs) play an important role in medicine today. MIDs are being connected to hospital networks for archiving Magnetic Resonance Imaging (MRI), Computed Tomography (CT), Radiography (Rx) and Echography using the Picture Archiving and Communication System (PACS) and to integrate them with the Hospital Information System (HIS) (Yu, Li, Zhang, & Yu, 2012) (Liu, Li, Liu, Yuan, & Yin, 2008).

Medical imaging techniques are becoming more digital and more connected, making them vulnerable to network related cyber-threats (Oh, Lee, Jung, & Yeom, 2008). These medical tests are used for various reasons such as supporting life-saving treatments which makes them a critical asset for hospitals. Therefore, the failure of one device will disrupt the entire hospital's operation, influencing the patients' wellbeing and confidentiality.

On the other hand, medical imaging storage technologies, such as PACS, are becoming more and more essential as the volume of digital medical images grows throughout the healthcare institutions and the analysis of this data becomes more crucial for clinical diagnosis. The use of PACS eliminates the need to store, retrieve and send sensitive information, films and reports manually (Zhang, Yu, Sun, Yang, & Liang, 2007) (Rostrom & Teng, 2011).

Concerning the medical imaging study stored within the PACS system, it consists of the MID image(s), the patient identification data and proprietary information (Fridell, Aspelin, Edgren, Lindsköld, & Lundberg, 2009) (Mansoori, Erhard, & Sunshine, 2012). Moreover, the transmission of this sensitive data should be controlled using security measures to ensure that nothing is leaked or even manipulated during its transmission. Robust privacy and security practices must be applied on PACS to preserve the confidentiality, integrity and availability of the information (Mahlaola & van Dyk, 2016).

Attackers' skills are improving and the number of unpatched devices with known vulnerabilities that can be easily exploited is growing, which is a major challenge to device manufacturers and healthcare providers. This is making MIDs vulnerable to sophisticated cyber-attacks targeting the devices infrastructure and components, thus disrupting digital patient records, and even posing a threat to patients' health.

3.6.1 Definition of the medical environment risks and threats

Medical institutes must be aware of the risks and new approaches for treat detection and prevention that should be deployed and implemented within the PACS system. Understanding the mechanisms behind these potential attacks can help in the prevention and the recovery in case of an incident because attackers may cause:

1. Manipulation of the MIDs predefined values;

2. Mechanical disruption;

3. Denial-of-Service attacks;

4. Exposure or destruction of medical documents or images stored by PACS.

Other treats, which have to be considered, deal with human factor and security changes over time. On one hand, humans are prone to errors and to negligence, which make them one of the weakest rings in the security of any system. Lack of security awareness and clear security policies lead to weak passwords and successful phishing attacks that allow an attacker to easily bypass any security measures found, especially if this person has a high level of privileges in the system. This issue will be handled in details in chapter 5 when introducing the security at the human factor layer. On the other hand, software and network systems are in a race with time; failure to update virus definition makes the security measures useless. Development of a well-structured PACS system takes time; sometimes, due to cost or negligence, the design is not updated to reflect up-to-date security issues. This act will lead to having a non-secured PACS system.

Several cyber-attacks have been identified in the medical field. One of the most notorious attacks was the WannaCry ransomware that had paralyzed Britain's National Health Service in May 2017 (Corporation), 2017) (Adams, 2018). The ransomware worm spread rapidly across a number of computer networks. After infecting Windows computers, it encrypts the files on the PC's hard drive, making them impossible for users to access, and then demands a ransom payment in Bitcoin in order to decrypt them. It exploited a Windows vulnerability discovered by the United States National Security Agency (USNSA) and leaked to the public by the hacking group *Shadow Brokers* in April 2017. The attack vector for WannaCry is more interesting than the ransomware itself. WannaCry exploited a vulnerability in the Windows implementation of the Server Message Block (SMB) protocol that is responsible for helping various nodes on a network to communicate (Kubovič, 2018).

3.6.2 Case study Implementation

A PACS infrastructure contains multiple assets, including a workstation, some imaging devices and acquisition gateways, a PACS controller, and a database for archiving. Most PACS are compliant to DICOM (Digital Imaging and Communications in Medicine) standard. This latter is a protocol that defines the file format for exchanging medical images and associated information (Zhou, Liu, & Le, 2007). Compliance with this standard enables the integration of various medical devices from multiple manufacturers.

Several network architectures exist for PACS. Usually, it is connected to the HIS or RIS network in the hospital (Peck, 2018). In order to be generic, we will consider that it is connected to the main network of the hospital as shown in figure 3.12.

Figure 3.12 shows the general architecture of a PACS system integrated within the network of the hospital. The proposed architecture is designed in a way to include the biggest number of physical threats that may affect the network. Thus, the PACS server is

connected to the hospital servers through a switch to allow connectivity with the HIS or RIS servers and other appliances within the hospital network.

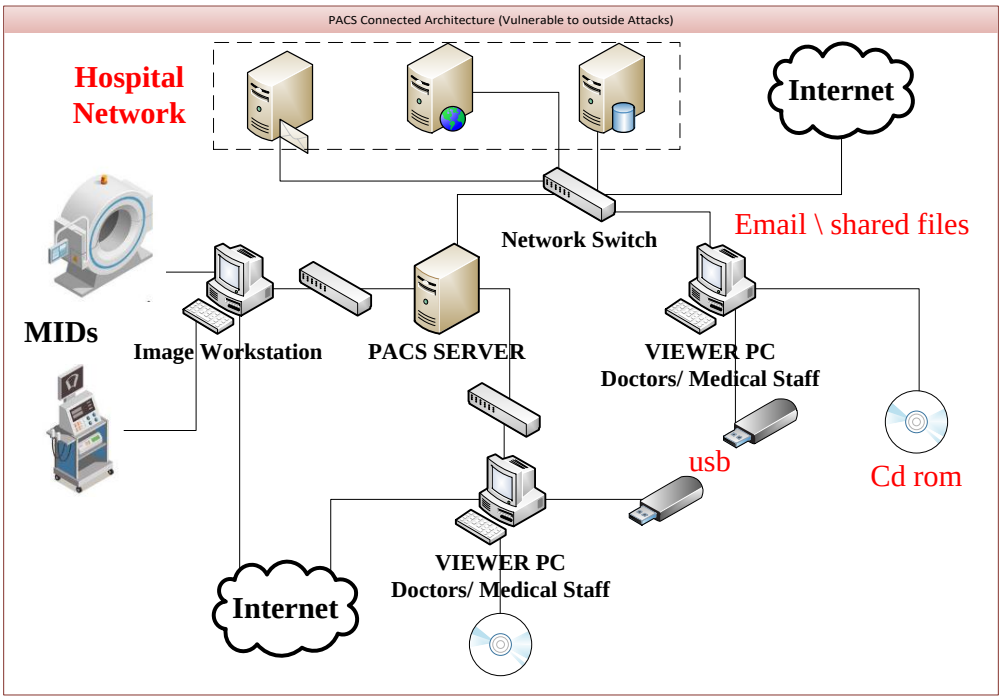


Figure 3.12 PACS architecture within Hospital network

Three different types of computers are connected to the PACS system. The first is the workstation that is connected to the MIDs, the second represents the computers of the medical staff directly connected to the PACS system, and the third presents the computers of the medical staff connected to the PACS system through the hospital network.

The PACS architecture reveals that it is made up of two major components: the software and the hardware. Thus, the PACS system is vulnerable in both domains and needs to be hardened in order to be resistant to any cyber-attack. The source of the threat can be a hacker, criminals, competitors, an unsatisfied employee within the organization, even partners etc … (Mansfield-Devine, 2018). Some of them just do it for the glory and fame while others want to punish the organization by inflicting destructive actions, such as deleting clients' data or sabotaging the hardware, which results normal work halting thus leading to loss of money and loss of confidence in the organization in question. Finally, in the past two years, the main focus was lucrative either by stealing valuable data and selling it on the dark web, or by encrypting the data and demanding ransom money.

Threats on the PACS system create risks on the data stored or exchanged, and on the normal functionality of the medical devices connected. The deletion of data is a major risk;

however, the modification of this data has a higher and more dangerous effect. Such modification, if not identified, could lead to wrong perception of the patients' medical conditions and lead medical professionals to an error in diagnosing the patients' health thus affecting negatively their lives.

The main focus will be on covering malwares that could infect the PACS system through one or many of the connected computers through internet. Social Engineering is becoming the most notorious way to infect networks with malware, thus phishing has become the ultimate method to penetrate into well secured networks. An email containing an infected document or file, if opened, can infect the machine, and thus opens the door wide for the infection of other computers in the network. Most malwares need to connect back to the C&C (Command and Control) server to receive instructions and to send it the stolen data. A more practical example of attack is shown in Figure 3.13 representing a UML sequence diagram of the attack described previously.

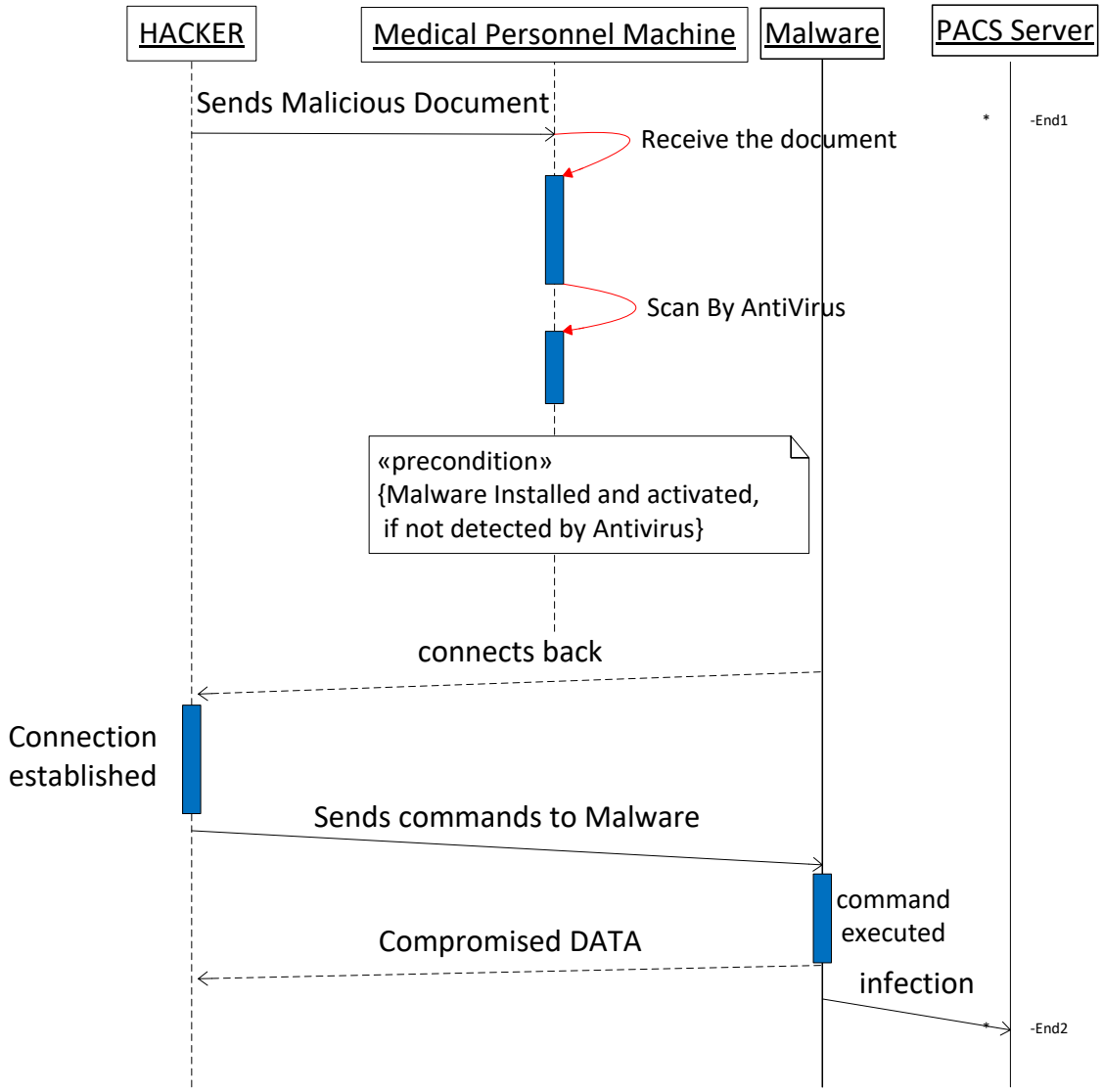


Figure 3.13 UML sequence diagram representing the attack steps

From Figure 3.13, one of the most common scenarios to inject a threat is shown: the hacker creates a malicious file and sends it to the medical personnel either by Flash Memory or email. When the victim receives the file, the antivirus should check it. Having an antivirus can help, but the negligence to update virus signatures or the absence of a signature in case of a Zero Day vulnerability will make the antivirus fail to detect the malicious file. So, the victim will open it and allow the execution of the malware. Upon execution, the malware will collect data related to the infected machine, and will try to communicate it back to the hacker. If the hacker sees that the victim is worth it, he will command the malware to send more data, thus compromising the privacy and security of the victim. Of course, the malware will try to infect other machines connected to the victim PC using either Zero Day attacks or even known vulnerabilities relying on the laziness of the information security officer to update the machines using the latest patches.

Concerning the already proposed scenario of Figure 3.13, upon infection, the malware needs to call the C&C server to get instructions on how to proceed. If this phase fails, the malware self-destructs thinking that it may be on a testing machine for analysis, or with time the malware becomes obsolete and the antivirus becomes aware of it and removes it. To achieve this, the easiest approach consists of applying physical separation (Air Gap) between the PACS network and any external network; however, this will affect the productivity of the medical staff, especially that most of them have little to none experience in computers.

However, a more efficient approach consists in integrating a data diode in the PACS architecture. The implementation of this approach leads to the secured solution proposed in Figure 3.14. The solution creates a one-way point of entry for the data into the network, thus minimizing the risk of getting infected. Figure 3.14 shows the new PACS network architecture after adding the data diodes to it.

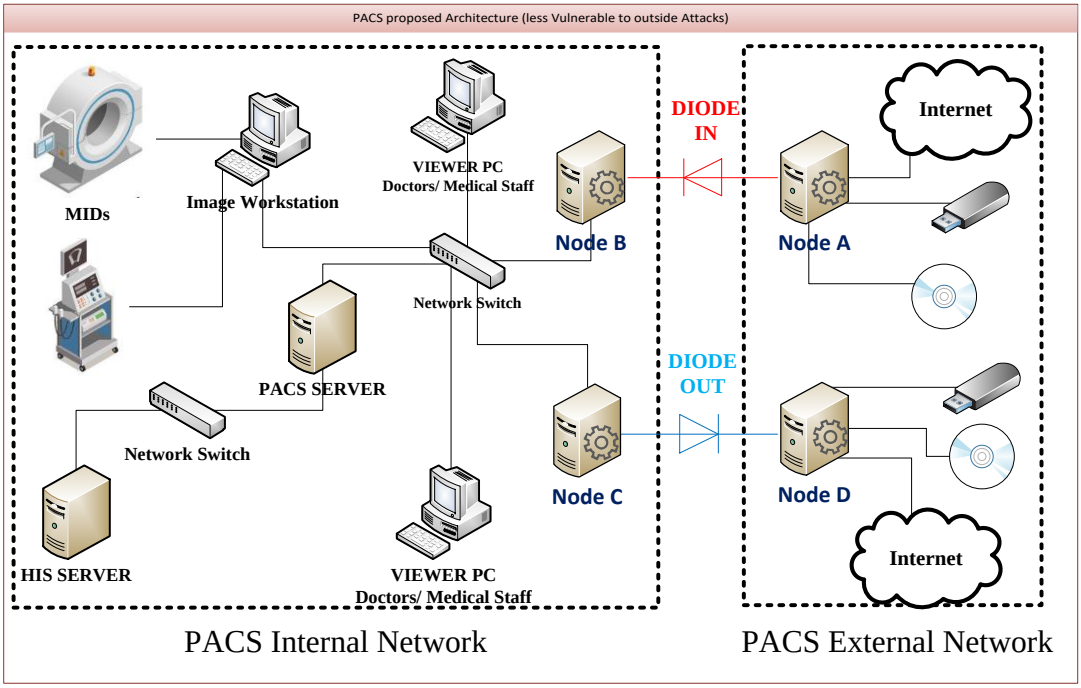


Figure 3.14 : PACS Network containing the data diode

From Figure 3.14, Node A is a server connected to the PACS external network, while Node B is connected to the PACS internal network. Node A receives the files from external sources like the internet or CD/DVD or USB memory sticks. It filters and scans the files, and then sends them through the DIODE IN to Node B. Node B receives the files and transfers them to the destination user/machine. When the user/machine wants to send an email or a file outside the PACS internal network, it forwards the request to Node C that, in turn, forwards the request to Node D through DIODE OUT. When the Node D receives the request, it processes it accordingly. If it is an email, it will send it, and if it is a file, it would put it on a CD/DVD or a USB memory stick according to the user's choice.

3.6.3 Testing and Validation

A lot of commercial data diodes exist on the market. They are expensive due to the complexity of the network at hand. Thus, to validate the proposed architecture, we created a low-cost prototype data diode. The data diode NODE IN / OUT is a PC having Intel(R) Core (TM) i7-7700HQ CPU @ 2.80GHz and 16 GB ram with a 100Mbps ethernet card.

Three different DICOM files, which are usually transmitted over PACS architecture, were used to study and analyze the performance of the network, with and without the use of the data diode as shown in Table 3.1.

Table 3.1 Time needed for files with different sizes to be transmitted using different bandwidths

	Time with DIODE(sec)			Time without DIODE(sec)		
specified BW	223.8 MB	14.5 MB	4.9 MB	223.8 MB	14.5 MB	4.9 MB
10 Mb\s	227.979	16.976	7.47	221.748	16.434	7.228
50 Mb\s	65.131	6.207	3.515	56.672	5.598	3.29
80 Mb\s	50.468	5.189	3.163	41.889	4.64	2.924
100 Mb\s	47.857	5.058	3.086	39.247	4.457	2.85

Table 3.1 shows the difference in time of file transmissions between the two architectures. The “time with DIODE” includes the addition of the time for hashing and archiving the file with its hash before transmission, and the time for extraction and hashing of the file after receipt (when using the diode only). This procedure is needed to ensure the eligibility of the received file.

Added to that, Figure 3.15 shows a graphical representation of the results obtained in Table 1. The *x*-axis shows the connection speed whereas the *y*-axis represents the latency ratio measured after adding the data diode to the network. These measurements are tested over the three DICOM files. So, one can notice the latency percentage increasing:

- while the connection speed increases (2.73% for 10Mb/s connection compared to 17.99% for a 100Mb/s connection for 223.8Mb size);
- while the file size increases (6.4% for the 4.9Mb file compared to 12.99% for the 223.8Mb file when using a connection speed of 50Mb/s).

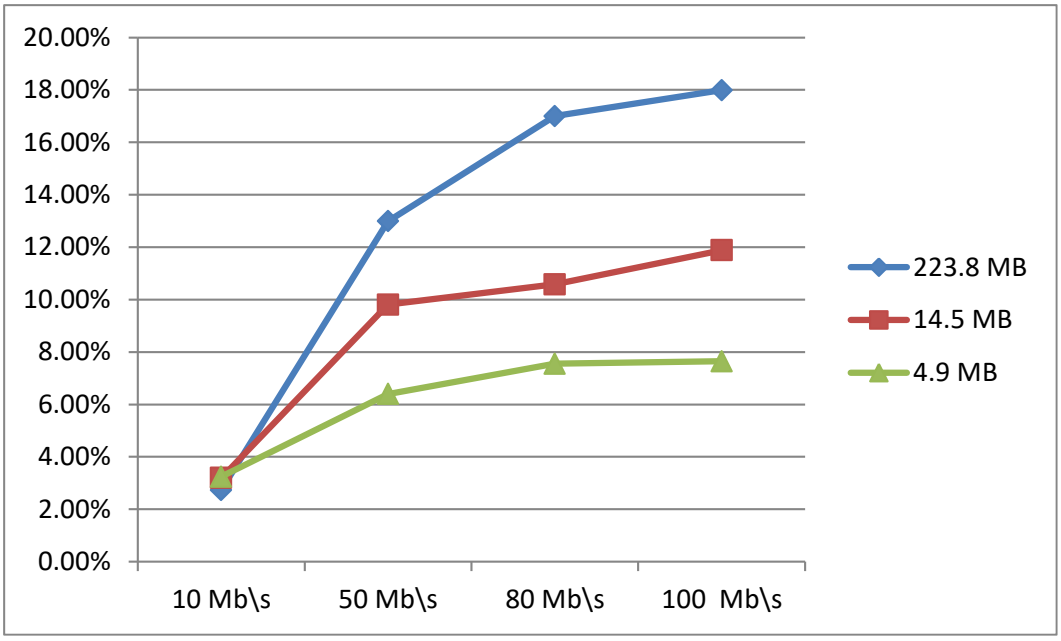


Figure 3.15 Latency percentage (with/without Data DIODE)

Moreover, a main drawback of the use of the data diode resides in the absence of receipt acknowledgment from the receiver side. Thus, to overcome this problem, sending the file more than once would decrease the probability of bad reception of the sent file. Sending the file two or even three times would be a good choice and the software on the receiving end should compare between the different received files and to choose the most adequate.

To conclude, we have been able to integrate a data diode in a fully functional medical system, subject normally to several attacks. The proposed solution has underlined the importance and the outcome of data diode without modifying the required time to transmitting/receiving the file.

3.7 Conclusion

In this chapter, we have presented a solution, consisting of a hardware and a software tool, to protect the system on the network level. As the first part of the security puzzle is being achieved, the next chapter will propose, implement and validate a solution to protect the system from a device level perspective.

Chapter 4 Proposed Solution and Procedures for Device Layer

Outline

4.1 Introduction

The device layer represents the entities in the information system that are, or that support the business logic of the organization. Nowadays, the vast reliance on connected devices made them a critical asset of the organization that needs to be secured. Data and its availability have become the world most valuable resource (Economist, 2017). Based on this fact, threat actors are creating more and more malware to infiltrate and steal data from devices. Thus, the main contribution of this chapter is to define the essential threats, their origin and the way to secure data from being theft, altered or injected. Based on these threats, solutions will be proposed in order to ensure the maximum security of this layer.

4.2 Overview of Device Layer

Although the threats and their solutions have been presented in details in chapter 1, this section aims to focus on the most active and efficient threats and viruses that address the device layer. This overview will present what our proposed solution aims to solve. It is a major constraint to consider the efficiency and the simplicity of the derived solution(s).

Security experts try to deter threats related to this layer by using existing software like antivirus and firewalls, and by proposing new methods that will be detailed later on. A lot of work has been made in this field, yet the number of security incidents is on the rise day after day (Galov, 2019).

Furthermore, the large adoption of the Internet of Things (IoT) has been coupled with the increase of security vulnerabilities and exploitation. IoT is a system of interrelated computing devices communicating through the internet allowing them to send and receive data. It allows the interaction between the physical world and the digital world. Sensors and actuators interact with the physical world and transform the data to the digital world. IoT technologies are contributing in the significant growth of generated data from appliances. Moreover, data security, confidentiality, integrity and availability are the challenges facing IoT applications and platforms due to the fast involvement of distributed diverse devices (Granjal, Monteiro, & Sá Silva, 2015). Most IoT systems utilize an architecture focusing on the connectivity with the cloud servers via gateways. Unfortunately, privacy and security risks are severe consequences of this architecture (Garg & Dave, 2019).

Along the years, malware (*Mal*icious Soft*ware*) acquired different names based on their purpose and behavior such as adware, spyware, virus, backdoor, Command and Control (C&C) bot, worm, Trojan, rootkit and ransomware (Aycock, 2006) (Mathur & Hiranwal, 2013). Cyber-attacks do not discriminate among governments and companies. Data breaches happen at any time and anywhere, and data can be sold or bought in the DarkNet (Biryukov, Pustogarov, Thill, & Weinmann, 2014).

To further understand the threats and solutions, IDEF0 will also be used to model this level. Thus, the diagram presented in Figure 4.1 shows the general form of the IDEF0 diagram along with all its parameters (Menzel & Mayer, 1998). The control variables will be pointing from the upper side while the resources will be pointing from the lower side. The threats will be pointing diagonally from the upper left corner while the solutions of these threats will be pointing from the lower left corner. The input variables will be from the left side while the output variables will be pointing outwards from the right side.

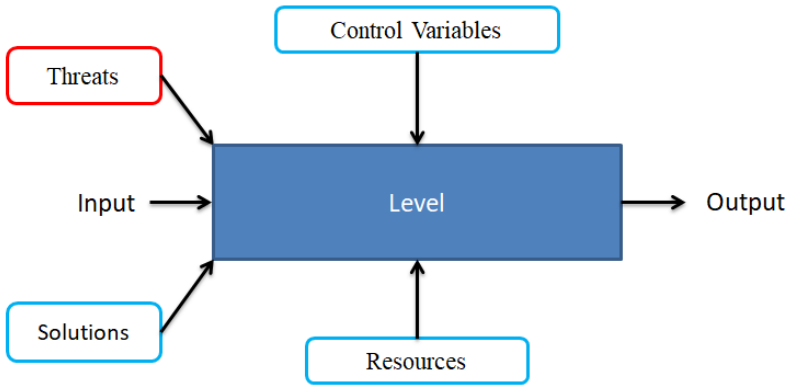


Figure 4.1 General form of the block diagram

So, concerning the device level, an extended IDEF0 representation is emphasized in Figure 4.1. From this diagram, one can identify the following:

1 ***Control variables***, which represent the main protocols to assure security at the level of devices, are highlighted by the use of ISO/IEC standards, NIST, SOC3,… norms;
2 ***Inputs***, which are the main sources of threats/vulnerabilities affecting a device, depend on the use of external devices/memory sticks, network shares, emails as well as malicious websites;
3 ***Threats,*** which represent the types of malicious actors that affect the proper functionality of the network, can be represented mainly by malware;
4 ***Solutions,*** which represent the list of actions (software and hardware) that will limit the threats effects on the device, are implemented through the use of one or more protection solutions such as antivirus, HIDS, software update, device encryption, physical security, secure configuration and scheduled backups;
5 ***Output,*** which represents the final state of the device after implantation of the solutions and after being governed by the control variables.

As an example, let's consider the following scenario: an email with malicious content is received by a user. The antivirus will detect it and stop it; furthermore, applying the standards defined in ISO/IEC 27002 can be efficient in handling the proper reception of emails from unsolicited sources.

Figure 4.2 shows the block diagram for the device level along with all actors following the IDEF0 model.

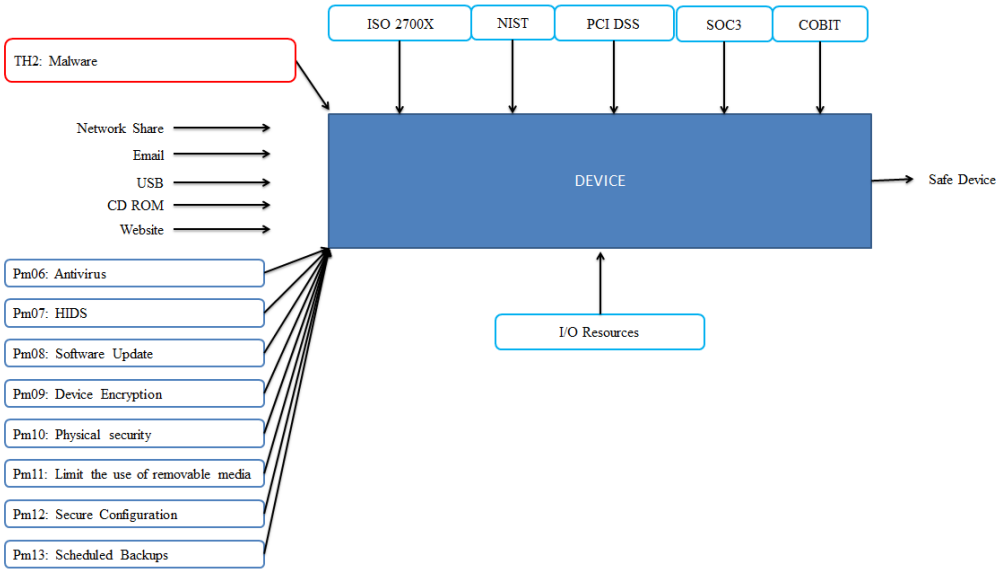


Figure 4.2 Block diagram for the Device level

According to a study realized by Ponemon Institute LLC published in 2017, 54% of Small and Medium-sized Businesses (SMB) experienced a cyber-attack in 2016, and 61% in 2017. The study showed that 48% of the companies had a Phishing attack, 36% had Malware attacks, and 26% experienced a DoS attack in 2017 (Ponemon Institute LLC, 2017).

Based on these studies, it is becoming more and more difficult for a network administrator to cope with the enormous number of cyber threats. The need for an effective and simple methodology is necessary to face an active security risk facing the enterprise. Therefore, vulnerability risk assessment is performed to select the one with the highest corresponding risk as a priority for network security reinforcement (López, Pastor, & Villalba, 2013). Traditionally, the FIRST's Common Vulnerability Scoring Systems (CVSS) (FIRST, 2022) (NIST, NATIONAL VULNERABILITY DATABASE, 2022) is primarily the basis for vulnerability risk assessments (Houmb, Franqueira, & Engum, 2010) (Chatzipoulidis, Michalopoulos, & Mavridis, 2015). The CVSS is an open framework for communicating the principle characteristics and the severity of software vulnerabilities. The CVSS metric ignores the impact of the vulnerability in a specific network, which leads to identical vulnerability values for different network environments (Wang, Shi, Zhang, Xu, & Zheng, 2020) (Holm & Khan, 2015). In fact, a little survey of the CVSSv2 scores will show that until the end of 2020, there are 8330 CVEs with a score of 10 (HIGH) and 193 CVEs have a score of 10 (CRITICAL) in CVSSv3 scores. So, the scores cannot be used alone as a scale for risk priority. Furthermore, to actually understand the risk that a

vulnerability imposes, an analysis should be performed around it based on the likelihood of it being used. The reason for such a conclusion is the fact that a lot of vulnerabilities were never exploited even though they have a high severity score (Nayak, Marino, Efstathopoulos, & Dumitraş, 2014). FIRST provides a CVSS calculator to input context information in order to improve the scoring of a given vulnerability such as temporal factors (*i.e*., how a vulnerability changes over time) and environmental factors, but the scoring of such factors is still general, and largely relies on the experience of the administrator of an organization. No systematic guideline is provided on how to appropriately set these factors (Jiang, Ding, Zhai, & Yu, 2012). To solve this problem several works have proposed solutions to improve the ranking methodology and give a clearer vision on the risks that can lead to a better approach of prioritizing the dangers.

For instance, Othmane et *al*. proposed a risk estimation method that incorporates attacker capabilities in estimating the likelihood of threats to improve the accuracy (Othmane, Ranchal, Fernando, Bhargava, & Bodden, 2015). In a similar approach, Dobrovoljc et *al*. used the CVSS values and attacker characteristics to develop a model which they say achieves the highest effectiveness among existing methods (Dobrovoljc, Trček, & Likar, 2017). Also, Maghrabi et *al*. enhanced the CVSS scoring through the use of game theory by modeling an attacker-defender scenario for achieving the prioritization in vulnerability patching (Maghrabi, et al., 2017). Keramati and Keramati ranked and prioritized vulnerabilities by proposing attack graphs based on CVSS (Keramati & Keramati, 2014). Allodi and Massacci proposed the black market as an index of risk of vulnerability exploitation. Their approach assesses the risk of vulnerability exploitation based on the volumes of the attacks due to the vulnerability exploits sold in the black market (Allodi & Massacci, 2012).

To sum up, this section has presented a modelling of the threats and the vulnerabilities over the device layer using IDEF0 model. Added to that, it presented the most relevant solution for vulnerability risk assessment along with the limitation of each method. This leads to propose, in the next section, our solution that will ensure a better security for the device layer.

4.3 Proposed solutions

After presenting the problems affecting the device level, the proposed solution treats two main issues: cyber-risk prioritization and response to anomalies. Concerning the first issue, a novel approach is proposed for the calculation of cyber security risk by proposing a new methodology to define risks level whereas, for the second, Artificial Intelligence (AI) will be used to handle anomalies without human intervention.

To improve the security of the Information Technology environment, the novelty of this work is to propose a novel procedure to calculate the Risk Score (RS) of detected cyber threats and to produce a Priority Score (PS) to rank the threats. CVSS scores are used in conjunction with other metrics to produce a new score that can be more significant to enhance cyber threat classification.

Evaluating the cyber security of an enterprise is an important step towards securing its system and resources. Risk identification is better called risk discovery and clarification. The objective is to classify existing risks in such a way to remediate the ones that can be solved and to embrace those whose consequences are acceptable. Our proposed solution aims at providing a simple and reliable procedure that can help administrators estimate a scoring probability of vulnerabilities. This scoring is being calculated referring to several databases and datasets that are being updated continuously by their owners. Here below are the different resources that are used to calculate the vulnerability score:

1) Common Vulnerability Scoring Systems (CVSS)

CVSS Scores have been in wide use in vulnerability management programs for more than a decade. CVSS attempts to assign severity scores to vulnerabilities, allowing responders to prioritize responses and resources according to threat. The MITRE corporation (MITRE, MITRE, 2022) gives every publicly known information security vulnerabilities a Common Vulnerabilities and Exposures (CVE) identifier which makes it easier to share data across separate network security databases and tools. First released in 2005, the scoring mechanism has gone through three major revisions. The most recent revision was the move from CVSSv2 to CVSSv3 (in 2015), with CVSSv3.1 (in 2019) being the current revision. CVSSv3, designed to correct shortcomings in v2, has been judged by the security community as a whole to have closed some, but not all, of the shortcomings of v2. Table 4.1 shows the difference in the metrics used between CVSSv2 and CVSSv3.

Table 4.1 The difference between CVSSv2 and CVSSv3

Base Score Metrics V2	***Base Score Metrics V3***
Exploitability Metrics	Exploitability Metrics
Access Vector (Local, Adjacent Network, Network)	Access Vector (Local, Adjacent Network, Network, Physical)
Attack Complexity (High, Medium, Low)	Attack Complexity (Low, High)

Base Score Metrics V2	*Base Score Metrics V3*
Authentication (Multiple, Single, None)	Privileges Required (None, Low, High)
	User Interaction (None, Required)
	Scope (Unchanged, Changed)
Impact Metrics	Impact Metrics
Confidentiality Impact (None, Partial, Complete)	Confidentiality Impact (None, Low, High)
Integrity Impact (None, Partial, Complete)	Integrity Impact (None, Low, High)
Availability Impact (None, Partial, Complete)	Availability Impact (None, Low, High)

While CVSS scores can and should be an important part of the vulnerability management program, it is important to keep in mind that widely published CVSS scores for a vulnerability can be misleading, as these typically represents the base score only. Thus, the base score (V2/V3) will constitute a part of the total score of the vulnerability classification introduced in the solution. Data is available through an API for online access or by downloading the entire National Vulnerability Database (NVD) for offline use which is provided by the National Institute of Standards and Technology (NIST) data feeds (NIST, data-feeds, 2022). NVD contains all the CVEs and their corresponding metrics and offers the vulnerability data feed using the JSON format. Files are stored offline in order to have more flexibility in extracting the values and producing the new score. The base scores of both CVSSs, CVE published date and counted the number of citations associated to the CVE were extracted. Two important parameters can be also extracted: The Patch tag that indicates if the vulnerability has a patch or not, and the Vendor Advisory tag which indicates if the vendor has declared an advisory for this vulnerability or not. Table 4.2 represents a sample of the final extracted values for CVE-2018-19458.

Table 4.2 Data extracted from NVD for CVE-2018-19458

CVE	Published	BaseScore V2	BaseScore V3	Number of Citations	Patch	Vendor Advisory
CVE-2018-19458	2018-11-22T20:29Z	5	7.5	2	FALSE	FALSE

2) The Exploit Database (EDB)

EDB is an archive of public exploits and corresponding vulnerable software (Offensive_Security, Exploit Database, 2022). It is maintained by Offensive Security, an information security training company that provides various Information Security Certifications as well as high end penetration testing services. The Exploit Database is a non-profit project that is provided as a public service by Offensive Security. It is used by penetration testers, vulnerability researchers, and security fanatics. It reports vulnerability for which there is a proof-of-concept exploit. EDB is considered as the white market for exploits. According to surveys, it is also preferred by many contemporary exploit developers (Fang & Hafiz, 2014). It provides an updated CSV file in its GitHub repository of all its exploits to be used offline (Offensive_Security, The Exploit Database Git Repository, 2022); however, this file contains the ExploitID and the exploit date but not the CVE related to the exploit. So, to get the CVE connected to the corresponding exploit, the website had to be scraped in order to correlate the ExploitID with the corresponding CVE then calculate the number of exploits for each CVE and the date of the first published exploit in case of multiple exploits. Table 4.3 represents a sample of the final extracted values.

Table 4.3 : Measures extracted from EDB for CVE-2018-19458

CVE	Number of Exploits	Minimum Date of all exploits
CVE-2018-19458	1	05/11/2018

3) Computer Incident Response Center Luxembourg (CIRCL)

CIRCL is a government-driven initiative designed to provide a systematic response facility to computer security threats and incidents (CIRCL, Computer Incident Response Center Luxembourg, 2022). CIRCL provides a contextual feed containing all software vulnerabilities including visibility ranking in Luxembourg. The data feed originates from the aggregated data-sources. It has an API to provide online queries and it provides a daily CVE JSON file to be used offline (CIRCL, Open Data at CIRCL, 2022). The number of Common Attack Pattern Enumeration and Classification (CAPEC) (MITRE, Common Attack Pattern Enumeration and Classification, 2022) associated to each CVE was extracted using the API. Table 4.4 represents a sample of the final extracted values.

Table 4.4 Measure extracted from CIRCL for CVE-2020-17518

CVE	Count of CAPEC
CVE-2020-17518	10

After describing all the resources, the five parameters, used to calculate the new vulnerability score and the way they are created from the aggregation of different values extracted from the three already listed sources, will be detailed. Thus, Table 4.5 contains the detailed calculations, definitions and weights of these parameters. The **first parameter** is the average of CVSS base scores for each CVE. It contributes 60% of the new score since it is calculated using values from a trusted source. The **second parameter** is the number of citations because the more citations exists the more serious is the vulnerability and assigned it a weight of 2.5%. As for the **third parameter**, it is calculated from the difference in days between the published date of the CVE and the date of the first appearance of an exploit. A value of 4 is attributed for this parameter if the difference is less than 7 days, a value of 3 if the difference is within 30 days, a value of 2 if the difference is within one year and a value of 1 if the difference is more than one year or doesn't exist. The values reflect the severity of the CVE, because if the CVE has an exploit produced within less than a week, it means it is easily exploitable. However, if within a year or more, an exploit was produced, it means it is hard to exploit. A weight of 5% is attributed to this parameter because the ease of exploit plays an important role in determining the seriousness of a CVE. As for the **fourth parameter**, a 30% weight is attributed to it based on the existence or not of an exploit for the vulnerability. The presence of a publicly available exploit elevates the risk of exploitation even if the vulnerability itself is not that critical because it can be easily used and chained to other exploits in order to achieve a more severe level of exploitation. For **the**

fifth parameter the count of the CAPECs is used and a weight of 2.5% has been attributed to it. CAPECs are common attack patterns that describe how adversaries exploit weaknesses in applications and other cyber-enabled capabilities.

Thus, Table 4.5 summarizes the above information and shows the formula to calculate the Vulnerability Score (VS) as well as its Standardized Vulnerability Score (SVS).

Table 4.5 Classification of the CVE based on the proposed technique

<table>
<tr><th>Parameters</th><th colspan="2">Description</th><th>Weight</th></tr>
<tr><td>Average Base Score (BS)</td><td colspan="2">The average of the base scores from CVSSv2 and CVSSv3; if this latter doesn’t exist, BS will consist of CVSSv2 only.</td><td>60%</td></tr>
<tr><td>nbrCitations (NC)</td><td colspan="2">Internet links that provides additional information about the vulnerability.</td><td>2.5 %</td></tr>
<tr><td rowspan="5">Pub – dateExploit (PDE)</td><td colspan="2">Difference between the Publication Date of the CVE and the date of the first available Exploit. According to the obtained difference, the following values will be used:</td><td rowspan="5">5 %</td></tr>
<tr><td>Less than 7 days</td><td>4</td></tr>
<tr><td>Between 8 and 30 days</td><td>3</td></tr>
<tr><td>Between 31 and 365 days</td><td>2</td></tr>
<tr><td>More than 365 days</td><td>1</td></tr>
<tr><td>ExploitExists (EE)</td><td colspan="2">(TRUE=1, FALSE=0) if the CVE has at least one exploit in EDB</td><td>30 %</td></tr>
<tr><td>CountCAPEC (CC)</td><td colspan="2">The number of Common Attack Pattern Enumeration and Classification (CAPEC) ids associated with every CVE.</td><td>2.5 %</td></tr>
<tr><td>Vulnerability SCORE (VS)</td><td colspan="3">= 0.6*BS + 0.025*NC + 0.05*PDE + 0.3*EE + 0.025*CC</td></tr>
</table>

Standardized Vulnerability Score (SVS)	= VS /Max(VS)

Calculating the vulnerability score is just the first step to prioritize their risk. To improve the visibility, the presence of a patch or a vendor advisory for the CVE (as was discussed previously in the CVSS paragraph) was also taken into consideration. Using the SVS score from Table 4.5, the priority score is obtained after adding a certain constant as shown in Table 4.6. The highest value is assigned when both tags (*i.e.*, Patch and Vendor Advisory) are available.

Table 4.6 Priority Score constants

Priority Score (PS)	Value added to SVS
Patch and Vendor Advisory exists	3
Patch only exists	2
Vendor Advisory only exists	1

Referring to Pareto Principle (Gittens, Kim, & Godwin, 2005), or the 80/20 rule, it considers that about 80 percent of effects are triggered by 20 percent of causes. For administrators, the main takeaway from this rule is that not all cyber security risks are equal. Therefore, security resources should be devoted to the risks that are likely to cause the most damage to the enterprise. So, based on the priority score obtained, if we have, for example, 20 vulnerabilities the remediation of top 4 will lead to the elimination of 80% of the total system risk.

4.4 Implementation of the proposed solution

In this paragraph, the procedure used for implementation will be described in details. Figure 4.3 represents the methodology used in order to implement the proposed procedure. As it shows, the first step is to acquire the files from the three sources cited previously in paragraph 4.3. Python scripts were used to automate the tasks. The first script downloads the JSON files from the NVD website. Every file is parsed to extract the name of the CVE, its publish date, the CVSSv2 base score, the CVSSv3 base score, the number of citations, the presence of the tags Patch and the Vendor Advisory. The second downloads the CSV files from the GITHUB of EDB. The files are parsed for the ExploitDB, date of exploit. Then, for every ExploitDB, the EDB page related to this ExploitID is scraped for the CVE

linked to it and then calculate its number of exploits and the date of the first published exploit in case of multiple exploits. Added to that, the third script downloads the files from CIRCL and extractds the count of CAPEC per each CVE. At this point, the three datasets are merged to create one MS Excel file.

In Microsoft Excel, the difference in days was calculated, between the date the CVE was published and the date the first exploit of this CVE was announced. A constant value is assigned as shown in Table 4.5. Then a parameter was created indicating if the CVE has an exploit and gave it a value of **0** for no exploit found, and **1** if it has at least one exploit. Originally the dataset contained 148789 entries, but before applying the formula, 208 entries were removed because they had no scores at all however 23 CVEs with CVSSv2 score equal to zero were kept to maintain the integrity of the dataset. It is to note that out of the remaining CVEs, 73688 don't have a CVSSv3 score which is due to the fact that it was released in 2015. After applying the formula to the CVEs, the VS was obtained. After standardizing the values the SVS values were obtained. Afterwards, based on the presence of the tags patch and/or Vendor Advisory, a new score will be acquired. Sorting, in descending order, the PS will give a priority list that can be used by the administrator. Of course, we applied the formula on all the CVEs as a proof-of-concept. However, for a real system, the procedure will be applied on the CVEs that the administrator detects in his environment to obtain a list with priority scores which can help in deterring the cyber risks in an organized manner.

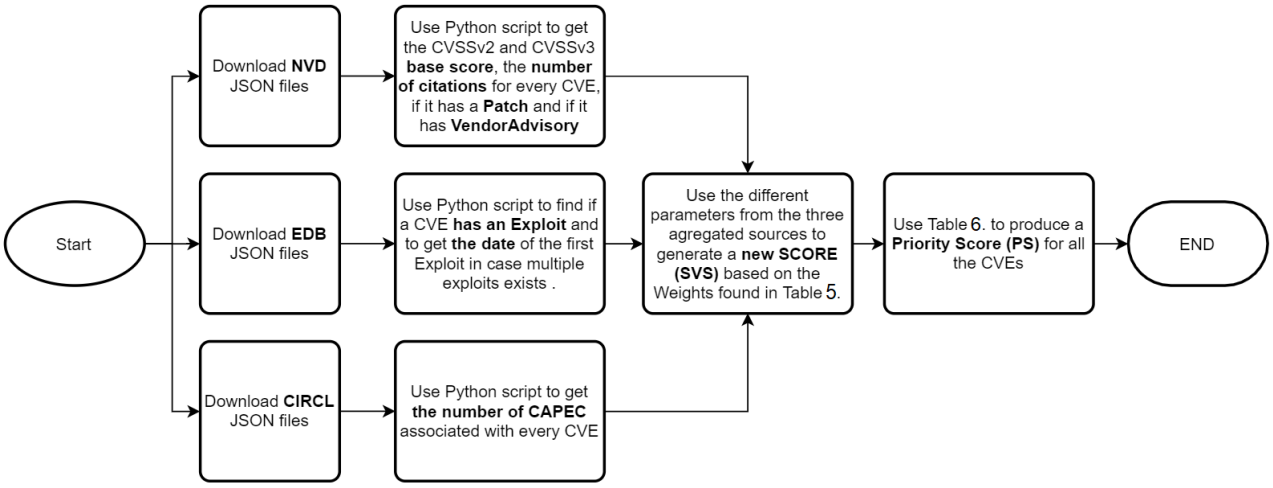


Figure 4.3 Procedure used to calculate the proposed Priority Score

4.4.1 Validation

Figure 4.4 represents the ten highest CVEs (we have shown only the first 10 because of space limitation) based on the proposed *Priority Score* while showing the CVSSv2, CVSSv3 and the SVS. The higher the SVS, the more serious the vulnerability is, and the higher the PS, the easier the vulnerability can be addressed. One can notice that the CVE-

2014-0224, with a CVSSv2 score of 5.8 and a CVSSv3 score of 7.4, has the highest classification in both scores. This can be explained by the fact that this is a serious vulnerability in OpenSSL. It allows man-in-the-middle attackers to hijack sessions and obtain sensitive information, via a crafted TLS handshake. Yet, based on its Priority Score, one can conclude that it is easily remediable since both tags are available.

As for CVE-2008-1447, known as the Kaminsky Bug, it is a DNS vulnerability that allowed attackers to send users to malicious sites and impersonate any legitimate website and steal data. It is classified as number 8 on the list of the Top Ten Worst Vulnerabilities by infosecurity-magazine (Raywood, 2020).

Concerning CVE-2014-3566, it is a vulnerability in the SSL protocol 3.0 which makes it easier for man-in-the-middle attackers to obtain clear-text data via a padding-oracle attack, aka the "POODLE" issue.

	CVE	BaseScoreV2	BaseScoreV3	SVS	PS
1	CVE-2014-0224	5.8	7.4	1	4
2	CVE-2014-6271	10	9.8	0.925225612	3.925225612
3	CVE-2006-3738	10		0.814353245	3.814353245
4	CVE-2008-1447	5	6.8	0.770519983	3.770519983
5	CVE-2007-2446	10		0.745595187	3.745595187
6	CVE-2009-0846	10		0.73700043	3.73700043
7	CVE-2017-5638	10	10	0.73485174	3.73485174
8	CVE-2014-3566	4.3	3.4	0.720670391	3.720670391
9	CVE-2006-0884	9.3		0.720240653	3.720240653
10	CVE-2011-1018	10		0.717662226	3.717662226

Figure 4.4 CVE sorted based on the PS value

In addition to the fact that the classification of the CVEs is becoming more accurate, this new scoring technique eliminates the redundant scores which were widely present in the CVSSv2 and CVSSv3. Thus, one can notice that the first ten vulnerabilities have unique scores. which is not the case of the two above scores; these new values can facilitate the autonomous execution of remediation procedures to reduce cyber security risks.

As for the execution time to run all this proposed procedure it needs few minutes to classify and sort all vulnerabilities; thus, calculating the threats in a certain system will require no time.

4.5 Case Study

The communication between sensors and the IoT gateways is very susceptible to cyber-attacks. Distributed Denial-of-Service (DDoS) attacks can result in significant infrastructure collapse when targeting cloud servers. Moreover, a centralized server presents a risk to the entire system in case of a malware infection by generating a single point of failure. Man-In-The-Middle (MitM) attacks can alter the data in transit and causes severe disruption for data security, confidentiality, integrity and availability.

In addition, IoT systems are not homogeneous in terms of security requirements, resource availability and networked devices. The devices operate in an open environment, which yields to increasing cyber physical risks and accessibility by adversaries.

So, in this case study, we will leverage the use of AI to protect a medical system and make sure the data received is genuine and at the same time, to detect abnormal medical behaviors.

The proposed application is implemented for elderly people and being monitored remotely by healthcare providers; so, the proposed system relied on a telemedicine approach. Thus, the objective of this work is to make sure that the same data, sent from the patient side, is securely received for analysis by the healthcare provider to ensure the best follow up possible to this category of people.

Safety and security are highly recommended since this application deals with medical data. Thus, this work will present a novel approach to ensure that the data, sent by the patient's device and processed by the healthcare provider tools, is authentic, confidential and available. The architecture of this systems consists of three main entities: the patient part where the sensors are placed, a gateway server that receives the data collected from the different patients to forward them to the required healthcare provider and the healthcare backend server that receives and analyses the data.

The AI-based software is installed on the gateway server. This software will identify, authenticate the receiver and check the data provided by an AI-based software.

4.5.1 Architecture

The architecture of the system before implementing the solution is shown in Figure 4.5. In more details, the figure represents the three entities (*i.e.*, the patients' sensors, the gateway server, and the healthcare provider server). The AI software is installed on the *Receiving Server*.

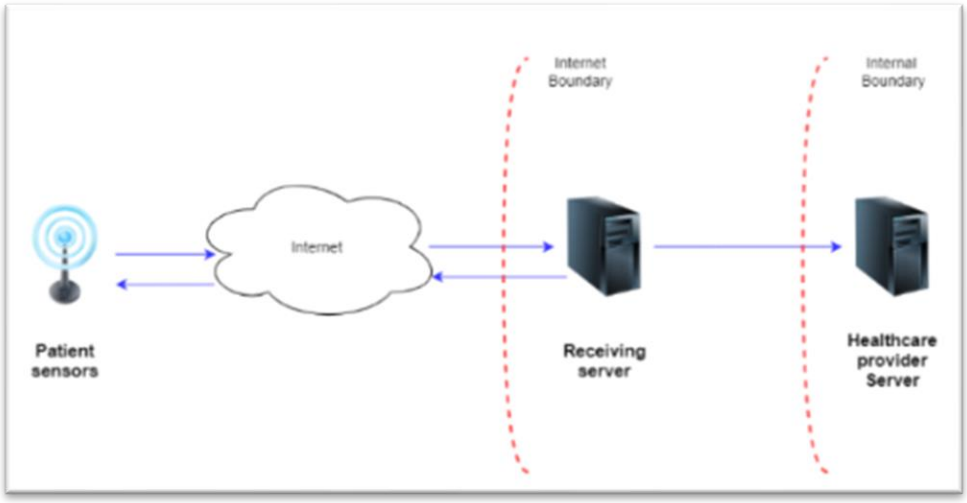


Figure 4.5 System architecture without the AI software

In more details, the system consists of three entities:

- A patient equipped by a heart rate sensor and a SpO2 sensor to check his health while driving;
- A gateway server (IoT gateway) to receive the health data of multiple patients using this application. An AI-based software will monitor the incoming data to check for irregularities from the network and device perspectives, *i.e.* the IP, the time frame between sending of consecutive data, the formatting, etc… Also, it checks the health of the patient for abnormal measurements, and notifies the healthcare provider to reassess the patient health to contact him in case of a major problem;
- The healthcare provider's server that is used to monitor the patients' health.

Although not all vulnerabilities have been resolved, this work deals with most frequent and common ones according to OWASP API top ten security list (OWASP, OWASP API Security Project, 2020).

AI-based software is installed on the input side of the exchange server to assess data transferred to the inside and to block any malicious action. Using Artificial Intelligence applications provide great security features, high transparency and enhance efficiency (Dhingra, Jain, & Jadon, 2016). The development of edge computing permitted data generated by the IoT devices to be transferred to the edge gateways for further process and analysis before being forwarded to the servers that are used by the monitoring personnel.

Figure 4.6 represents a DFD to show the way data flows from the patient's side to the healthcare provider end considering all security measurements proposed in this solution.

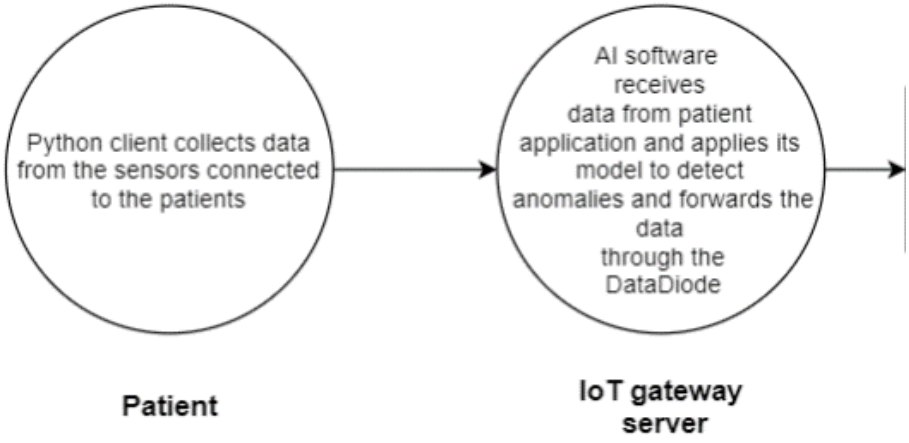


Figure 4.6 Scenario for the flow of data from the patient side to the healthcare provider backend

4.5.2 Implementation

The system is made of two applications: The first application is the one on the patient (P) and the second is on the exchange server (the IoT gateway in this application) (IoTg).

On the patient's side, a *raspberry pi zero* is equipped by an SPo2 sensor, heart rate sensor, and a GPS hat with 3G/4G & LTE Base HAT. The data of the patient is sent using a python script that collects all data from the sensors and sends them to a Python Web application on the IoTg server via an Internet connection. The IoTg is connected to the input part of the Data Diode and the output part of the Data Diode is connected the server of the medical healthcare personnel. The AI application controls the sending of data from the input part and a python application handles data reception on the output part of the Data Diode to insert it in the database of the healthcare provider analytics application.

The preparation of the data for machine learning is essential in order to generate a model capable of detecting the anomalies later on. So, at first, the data was collected and analysed. The anomalies detected and considered a priority to be handled by the system were:

1- If the SPo2 value is different with more than 20% value deviation relative to the patient's average values;
2- If the heart rate value is different with more than 20% value deviation relative to the patient's average values;
3- Attack on the API itself by a malicious actor;
4- Traffic is coming from the same user agent but from the different IP;
5- Traffic is coming from the different user agent but from the same IP;
6- SQL injection on the Web Application;
7- All data is correct but different IP of original patient;

8- The data sent by the user is not within the specified interval.

To train the AI/ML algorithms, the following features were selected: user, spo2 value, heart rate value, GPS longitude value, GPS latitude value, date and time, IP, user-agent as shown in Figure 4.7 below.

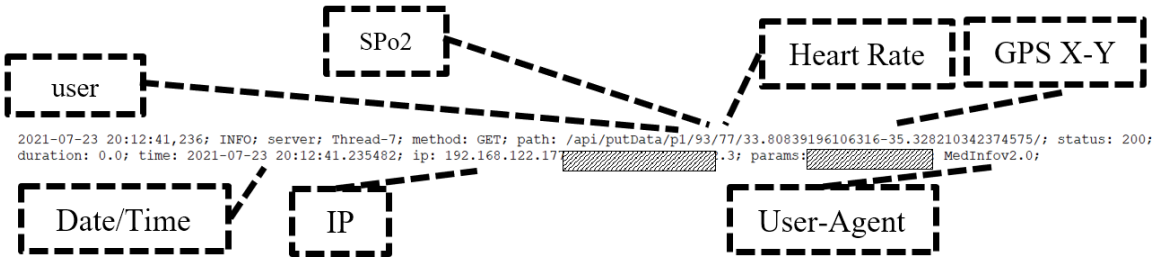


Figure 4.7 Features used for training AI/ML algorithms

The second step was to collect and label the data. The data was collected for the values of normal traffic, then data values with medical issues, and at last, data values after simulating cyber-attacks using hping3 (hping3, 2021) to perform a DOS attack and Ettercap (ETTERCAP, 2021) for Man In the Middle attack from a KALI Linux machine. The final dataset contained 27% normal traffic, 38% medical issue traffic and 35% cyber-attack related traffic.

Matlab R2018a was used to help us determine the best machine learning classifier that can achieve the highest accuracy. After applying 23 machine learning algorithms, as shown in Figure 4.8, some of them gave us a 100% accuracy while others gave us 37.9% accuracy. Added to that, we noticed that the K-Nearest Neighbor (KNN) algorithm was the best for our data since it gave us 100% accuracy in all its variances. Based on these results, *KNeighborsClassifier* was used from "scikit-learn" (Pedregosa, et al., 2011) to be implemented in the python AI application.

KNN is a supervised classification algorithm. It generates new data points based on the closest data points or the k number. The latter should be large enough to limit noise in data, but also small enough in order not to interfere with the other classes (Taunk, De, Verma, & Swetapadma, 2019). So, different values of *k* were calculated in order to preserve the accuracy. For each value, the mean error was calculated. Figure 4.9 shows the accuracy value with respect to the value of *k*. Thus, one can find that any value below 19 is a good value for *k*.

1.1	Tree	Accuracy: **100.0%**	8/8 features
	Last change: Fine Tree		
1.2	Tree	Accuracy: **100.0%**	8/8 features
	Last change: Medium Tree		
1.3	Tree	Accuracy: 96.4%	8/8 features
	Last change: Coarse Tree		
1.4	Linear Discriminant	Accuracy: 83.2%	8/8 features
	Last change: Linear Discriminant		
1.5	Quadratic Discriminant	Failed	8/8 features
	Last change: Quadratic Discriminant		
1.6	SVM	Accuracy: 89.7%	8/8 features
	Last change: Linear SVM		
1.7	SVM	Accuracy: **100.0%**	8/8 features
	Last change: Quadratic SVM		
1.8	SVM	Accuracy: **100.0%**	8/8 features
	Last change: Cubic SVM		
1.9	SVM	Accuracy: **100.0%**	8/8 features
	Last change: Fine Gaussian SVM		
1.10	SVM	Accuracy: 99.7%	8/8 features
	Last change: Medium Gaussian SVM		
1.11	SVM	Accuracy: 96.5%	8/8 features
	Last change: Coarse Gaussian SVM		
1.12	KNN	Accuracy: **100.0%**	8/8 features
	Last change: Fine KNN		
1.13	KNN	Accuracy: **100.0%**	8/8 features
	Last change: Medium KNN		
1.14	KNN	Accuracy: 100.0%	8/8 features
	Last change: Coarse KNN		
1.15	KNN	Accuracy: **100.0%**	8/8 features
	Last change: Cosine KNN		
1.16	KNN	Accuracy: **100.0%**	8/8 features
	Last change: Cubic KNN		
1.17	KNN	Accuracy: **100.0%**	8/8 features
	Last change: Weighted KNN		
1.18	Ensemble	Accuracy: 37.9%	8/8 features
	Last change: Boosted Trees		
1.19	Ensemble	Accuracy: **100.0%**	8/8 features
	Last change: Bagged Trees		
1.20	Ensemble	Accuracy: 79.7%	8/8 features
	Last change: Subspace Discriminant		
1.21	Ensemble	Accuracy: 99.4%	8/8 features
	Last change: Subspace KNN		
1.22	Ensemble	Accuracy: 37.9%	8/8 features
	Last change: RUSBoosted Trees		
2	Quadratic Discriminant	Accuracy: 42.3%	8/8 features
	Last change: 'Covariance structure' = 'Diagonal'		

Figure 4.8 Applied Machine learning algorithms and results

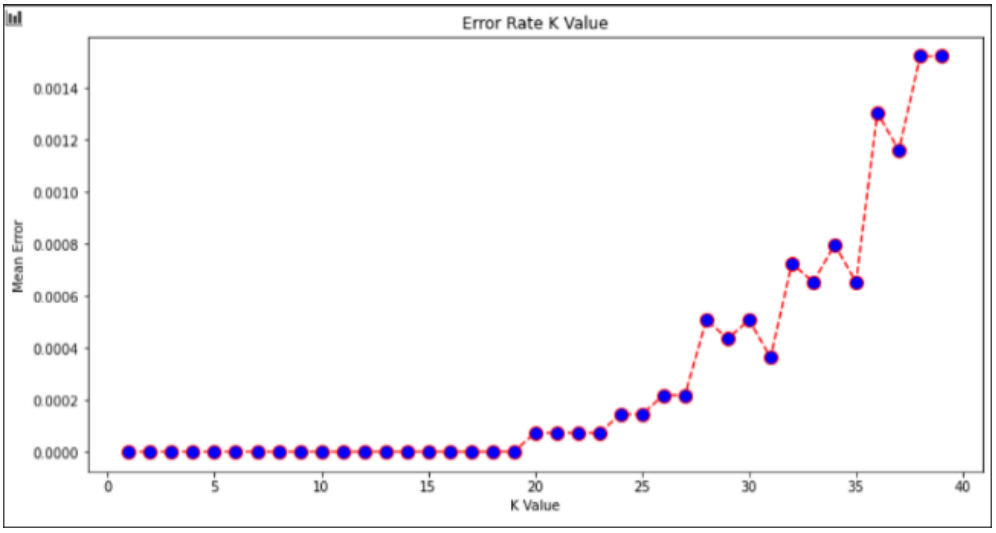


Figure 4.9 K-value vs Mean Error Value

Finally, the provided classifier was used in the AI application. The application continuously reads the received data and uses the classifier to predict its nature and acts accordingly upon it:

- If it is normal traffic, the data is forwarded to the healthcare provider through the data diode;

- If it is a medical issue, the data will be labelled with a warning to notify the healthcare provider upon receipt;
- If it is classified as a cyber-attack, the application will block the IP of the sender and notifies the IT admin of the system of the issue.

Figure 4.10 shows how the AI application is working flawlessly. In the first, second and last line, it shows a detection of medical anomalies due to abnormal values of the spo2 or heart rate. While in the third line, it shows the detection of a cyber issue.

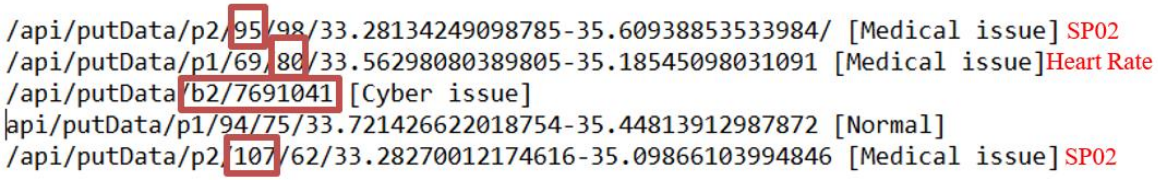


Figure 4.10 AI application detecting normal or medical issues or Cyber issues

From the Healthcare provider end, a python application will receive the data and forward them to the application used by the healthcare personnel to monitor the system.

After applying the AI algorithm, the hitting percentage was 100 for the KNN algorithm as 75% of the 54.000 packets were used for the training and 25% were used for testing. Added to that, the AI algorithm haven't caused any delay on data transmission between the patient device and the healthcare provider as time recordings were achieved first without implementing the security tools and it was then achieved after implementing the different peripherals and the time differences were very small, almost negligible.

4.6 Conclusion

This chapter presented the solution of the second part of the system, *i.e.* the device level. Sorting the cyber risk prioritization procedure while using an AI-based software was the novel approach proposed in this work. The results, after being applied to a mobile system transferring data about the health of a patient, showed very promising output with a high accuracy and a good management to eliminate threats while considering timing issues.

The next chapter will present the third and final piece of the puzzle by proposing, implementing, and validating a system that assures the readiness of the employees to tackle all threats and to act in a way to deny external entities to steal or to alter the data within the system. To do so, a chatbot will be developed with a double functionality: continuous evaluation of the knowledge and the know-how of the employee as well as an up-to-date cyber awareness of upcoming threats and viruses tackling the different networks.

Chapter 5 Proposed solution and procedures for the Human Factor layer

Outline

5.1 Introduction

The human factor layer deals with the human assets found in the enterprise. A chain is as strong as its weakest link; however, in cyber security, humans are the weakest link. Cyber security breaches caused by employees can be caused by malicious intent of the employee seeking profit, or through ignorance. The only way to tackle ignorance is through an effective cyber security awareness training. However, maintaining employees updated and informed can be costly on the enterprise if approached through classical training. Moreover, it will be hard to group employees on a specific common timetable. Thus, introducing chatbots would resolve the situation and adding some intelligent features will allow it to be autonomous.

5.2 Overview of Human Factor Layer

Cyber incidents are increasing every day. Organizations are investing more and more in security and specially in the technologies that can provide autonomous detection of malicious activities. Added to that, little is being invested in securing the human element (Disparte & Furlow, 2016).

Every company or organization faces the insider threat. Intentional or unintentional abuse of the access granted to the digital assets of the organization may be done by any current or former employee, partner or contractor. Insider threats can be divided into two most common types: first is the malicious insider who acts intentionally, and second is the negligent insider who does not comply with the policies and security instructions and becomes compromised thus serving as an attack vector for the true attacker. The acknowledgement of the existence of the insider threat is crucial for the enterprise in order to define a clear strategy for security and protection of its data.

So, the errors generated by humans can be mainly partitioned in two levels: ignorance and malignancy. The first one is due the lack of training that the employee must undergo when starting his job and during his work (*e.g.*, the IT department does not inform the employee on the way to act when receiving untrusted mails, noticing unusual activity on his PC, finding sent mails that were not generated by him and so on). As for malignancy, it consists of breaking intentionally the rules of the company by sending censored data to a third party, causing damage to the company's digital and physical assets and so on.

According to a survey by Fortinet, the following percentages were revealed related to the human factor (Schulze, 2019):

- 68% of organizations feel moderately to extremely vulnerable to insider attacks;
- 68% of organizations confirm insider attacks are becoming more frequent;

- 56% believe detecting insider attacks has become significantly to somewhat harder since migrating to the cloud;
- 62% think that privileged IT users pose the biggest insider security risk to organizations.

Malicious insiders are motivated by monetary gain and the theft of intellectual property. Also, they can be linked to sabotage, espionage, and reputation damage of the enterprise.

Whereas the negligent insider can be the victim of a spear phishing attack, poor passwords implementation, and browsing of suspicious sites.

The insider threat compromises mainly confidential business information such as data related to the financials of the enterprise, to the customers and employees. Also, privileged account information such as credentials and passwords are an objective (Sfakianakis, Douligeris, Marinos, Lourenço, & Raghimi, 2019).

Examples of damages created by employees, intentionally or by ignorance, are many. Below is a brief representation of the most known and published ones:

- In 2017, Equifax was breached, via one-member staff. The attackers used his credentials to acquire the information of 143 million people (Bernard & Cowley, 2017);

- In 2018, Veeam customer records were compromised by database left unprotected due to the lack of a password. 200 gigabytes worth of data were exposed (Whittaker, 2018);

- In 2018, the Defense Travel System (DTS) of the United States Department of Defense (DOD) sent out an unencrypted email with an attachment to the wrong distribution list. The email exposed the personal information of approximately 21,500 Marines, sailors and civilians (Egnash, 2018).

In 2017, a survey made by Ipsos MORI about cyber security and breaches showed that 72% of the breaches were by staff receiving fraudulent emails, followed by 33% related to malware, 27% related to impersonation of legitimate entities, and 17% related to ransomware. All these types of breaches can be linked to human factor either through clicking a malicious link or by not realizing impersonation (Klahr, et al., 2017). These revelations clearly demonstrate that employee awareness is important to a business's cyber security.

Specific actions for the insider threat contain the following elements:

- Implementing Data Loss Prevention (DLP) software based on human behavior-driven data to increase the effectiveness of a traditional DLP by applying user activity monitoring, behavior analytics and forensics;
- Using Single-Sign-On (SSO) access for enterprise applications;
- Use a multifactor authentication;
- Implement a security policy targeting insider threats, particularly based on user awareness;
- Use Identity and Access Management (IAM) solutions by implementing segregation of duties;
- Integrating Artificial intelligence and machine learning solutions to leverage the behavior analytical tools;
- Regular audit and user monitoring;
- Organizing training and awareness activities.

Cyber security awareness training is a solution used by organizations to educate its employees. It helps them getting acquainted with the most tactics, techniques, and procedures used as human-based attacks. Furthermore, regulatory frameworks like ISO 27001 (ISO/IEC, 2022) and General Data Protection Regulation (GDPR) (GDPR, 2022) require organizations to carry such trainings.

The main limitation of the trainings is the follow up and in keeping the employees up-to-day with the new technologies adopted to restrict cyber-attacks. Hence, having an external resource or an educational team in the IT department is of major importance to keep delivering new hints and tips for employees to reduce human errors. As this solution requires lots of efforts and, eventually an important budget of the IT department, new and innovative solutions would be in more favor. Thus, introducing chatbots would resolve the situation and adding some *intelligent features* will allow it to be autonomous. Chatbots are easier to interact with and humans find it natural (Heller, Proctor, Dean Mah, Jewell, & Cheung, 2005). This would be the objective of this chapter. This work is not unique. Several previous attempts were made by other researchers. For instance, we can list the work of Molnár and Szüts for creating a chatbot for education (Molnár & Szüts, 2018), Riikkinen *et al.* for using chatbots in insurance (Riikkinen, Saarijärvi, Sarlin, & Lähteenmäki, 2018), Santoso *et al.* for university Admission Services (Agus Santoso, et al., 2018), and Rosruen and Samanchuen as a medical Consultant (Rosruen & Samanchuen, 2018).

5.3 Proposed solution

Chatbots nowadays are becoming an essential element of the Human Interface Mediums (HIMs) like the internet and mobile phones (Nuruzzaman & Hussain, 2018). A chatbot is a service governed by rules and recently by artificial intelligence. It becomes restricted if it functions based on rules. It will only reply to predefined commands, thus if the query of the user is not part of its predefined keywords it will fail to comprehend and will not reply. However, the chatbot that leverages the power of machine learning can understand commands and the human language. Chatbots simulates a human conversation using machine learning techniques such as Natural Language Processing (NLP), computer vision and audio analysis. Chatbots evolve over time through learning from previous interactions making it more and more intelligent. This makes chatbots gain the name AI bot or Smart Bot.

Many current generations of chatbots are gaining popularity due to their ability to recognize complex questions via voice and respond to them orally. AI assistances from Amazon and Google are a clear example of this. Amazon's ALEXA and Google assistant help in the knowledge enrichment of their owners and help in controlling Smart Home appliances *i.e.*, turn off lights, control the home thermostat…

The novelty of this work will reside in developing an AI-based chatbot that will, not only initiate the basic security procedures to the employees, but also keep them updated about the new and emerging technique for defense towards cyber-attacks and it will provide a self-assessment test to check continuously about their readiness.

However, the proposed chatbot, that will assure safety at the level of human factor in a network, has the following features:

- allowing the users to search for topics related to the policies and procedures implemented in the company;

- allowing the users to search for keywords related to cyber security;

- allowing the users to take a cyber oriented exam using a randomly created dynamic list of questions to self-evaluate an employee and generate a report showing the weaknesses. These results will be saved and compared later on to monitor the evolution of the employee;

- sending alarms to all employees in case of new security measures needed to be implemented.

This chatbot has also a novel and innovative feature which is the use of WhatsApp (WhatsApp, Whatsapp LCC, 2022)to facilitate the delivery of information by providing a familiar medium of communication without relying on the API provided by WhatsApp.

Figure 5.1 shows the main functionalities that the chatbot will be able to do.

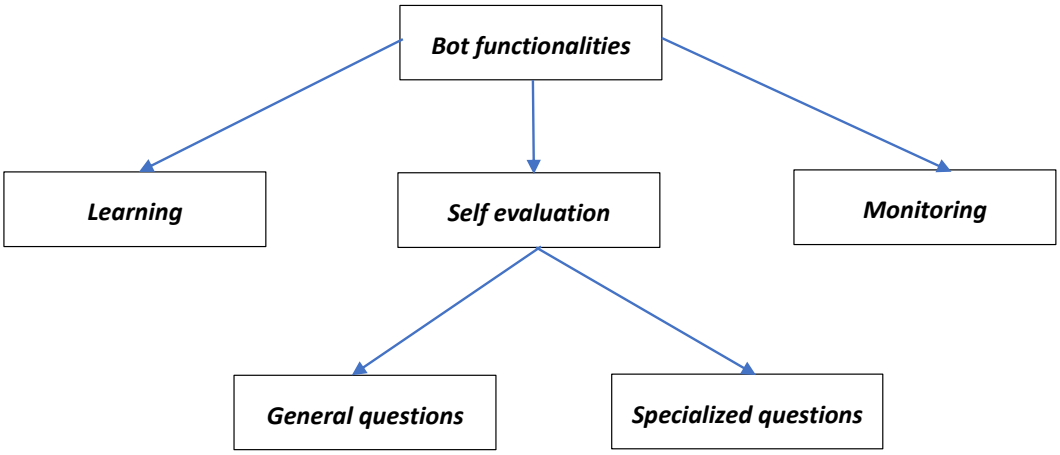


Figure 5.1 Flowchart of the chatbot main functionalities

5.4 Implementation of the solution

The solution is made of two parts: the collection of the data and the delivery of the data.

The data that needs to be collected are:

- Procedures and policies that are specific to security in the enterprise;
- Renowned best practices that are used in securing the human factor;
- A pool of security questions to be used in the self-assessment made by the employees;
- Cyber security news to be shared upon major breaches worldwide relative to the industry of the enterprise.

Whereas the delivery of these data is handled by Python through the use of WhatsApp as a medium of communication.

5.4.1 Data Collection

Organizations rely on policies as a high-level statement of management with the intent to influence decisions and guide the organization to achieve the desired outcomes. Policies are enforced by standards and are further implemented through procedures to establish actionable and accountable requirements.

The ISO/IEC 27001 (ISO/IEC, 2022), the NIST (Ross, McEvilley, & Oren, 2016) and the PCI DSS (PCI_DSS, 2022) policies were used to provide the data needed to enrich the cyber bot datasets. Furthermore, the Human Aspects of Information Security Questionnaire (HAIS-Q) was used. HAIS-Q was a questionnaire conducted by Prasons *et al*. (Parsons, et al., 2017) on 1, 112 university students relating to the computer user naïve and accidental behaviors that could lead the cause of the information security breaches. Figure 5.2 shows the items of the HAIS-Q. They center around seven areas of information security that are Password management, Email use, Internet use, social media use, Mobile devices, Information handling, and Incident reporting. After that, each focus area is further divided into three specific sub-areas.

	Knowledge	Attitude	Behaviour
Focus area: Password management			
Using the same password	It's acceptable to use my social media passwords on my work accounts. ^	It's safe to use the same password for social media and work accounts. ^	I use a different password for my social media and work accounts.
Sharing passwords	I am allowed to share my work passwords with colleagues. ^	It's a bad idea to share my work passwords, even if a colleague asks for it.	I share my work passwords with colleagues. ^
Using a strong password	A mixture of letters, numbers and symbols is necessary for work passwords.	It's safe to have a work password with just letters. ^	I use a combination of letters, numbers and symbols in my work passwords.
Focus area: Email use			
Clicking on links in emails from known senders	I am allowed to click on any links in emails from people I know. ^	It's always safe to click on links in emails from people I know. ^	I don't always click on links in emails just because they come from someone I know.
Clicking on links in emails from unknown senders	I am not permitted to click on a link in an email from an unknown sender.	Nothing bad can happen if I click on a link in an email from an unknown sender. ^	If an email from an unknown sender looks interesting, I click on a link within it. ^
Opening attachments in emails from unknown senders	I am allowed to open email attachments from unknown senders. ^	It's risky to open an email attachment from an unknown sender.	I don't open email attachments if the sender is unknown to me.
Focus area: Internet use			
Downloading files	I am allowed to download any files onto my work computer if they help me to do my job. ^	It can be risky to download files on my work computer.	I download any files onto my work computer that will help me get the job done. ^
Accessing dubious websites	While I am at work, I shouldn't access certain websites.	Just because I can access a website at work, doesn't mean that it's safe.	When accessing the Internet at work, I visit any website that I want to. ^
Entering information online	I am allowed to enter any information on any website if it helps me do my job. ^	If it helps me to do my job, it doesn't matter what information I put on a website. ^	I assess the safety of websites before entering information.
Focus area: Social media use			
SM privacy settings	I must periodically review the privacy settings on my social media accounts.	It's a good idea to regularly review my social media privacy settings.	I don't regularly review my social media privacy settings. ^
Considering consequences	I can't be fired for something I post on social media. ^	It doesn't matter if I post things on social media that I wouldn't normally say in public. ^	I don't post anything on social media before considering any negative consequences.
Posting about work	I can post what I want about work on social media. ^	It's risky to post certain information about my work on social media.	I post whatever I want about my work on social media. ^
Focus area: Mobile devices			
Physically securing mobile devices	When working in a public place, I have to keep my laptop with me at all times.	When working in a café, it's safe to leave my laptop unattended for a minute. ^	When working in a public place, I leave my laptop unattended. ^
Sending sensitive information via Wi-Fi	I am allowed to send sensitive work files via a public Wi-Fi network. ^	It's risky to send sensitive work files using a public Wi-Fi network.	I send sensitive work files using a public Wi-Fi network. ^
Shoulder surfing	When working on a sensitive document, I must ensure that strangers can't see my laptop screen.	It's risky to access sensitive work files on a laptop if strangers can see my screen.	I check that strangers can't see my laptop screen if I'm working on a sensitive document.
Focus area: Information handling			
Disposing of sensitive print-outs	Sensitive print-outs can be disposed of in the same way as non-sensitive ones.^	Disposing of sensitive print-outs by putting them in the rubbish bin is safe. ^	When sensitive print-outs need to be disposed of, I ensure that they are shredded or destroyed.
Inserting removable media	If I find a USB stick in a public place, I shouldn't plug it into my work computer.	If I find a USB stick in a public place, nothing bad can happen if I plug it into my work computer. ^	I wouldn't plug a USB stick found in a public place into my work computer.
Leaving sensitive material	I am allowed to leave print-outs containing sensitive information on my desk overnight. ^	It's risky to leave print-outs that contain sensitive information on my desk overnight.	I leave print-outs that contain sensitive information on my desk when I'm not there. ^
Focus area: Incident reporting			
Reporting suspicious behaviour	If I see someone acting suspiciously in my workplace, I should report it.	If I ignore someone acting suspiciously in my workplace, nothing bad can happen. ^	If I saw someone acting suspiciously in my workplace, I would do something about it.
Ignoring poor security behaviour by colleagues	I must not ignore poor security behaviour by my colleagues.	Nothing bad can happen if I ignore poor security behaviour by a colleague. ^	If I noticed my colleague ignoring security rules, I wouldn't take any action. ^
Reporting all incidents	It's optional to report security incidents. ^	It's risky to ignore security incidents, even if I think they're not significant.	If I noticed a security incident, I would report it.

Figure 5.2 HAIS-Q items

Added to that, three search engines duckduckgo.com, google.com, bing.com) were used to find cyber security assessment tests using keywords such as “awareness test”, “cyber

security test" and "cyber awareness test". After filtering the results, only the free tests were considered. Table 5.1 lists all the online resources used.

Table 5.1 Online resources used for gathering sample questions

	Website	***Description***
1	http://netsecurity.about.com/cs/compsecurity101/	Computer Security 101 course with short quizzes in the end of each chapter
2	http://www.proprofs.com/quizschool/story.php?title=it-securityquiz	36 IT-related questions testing the knowledge.
3	http://www.proprofs.com/quizschool/story.php?title=end-usersecurity-awareness-quiz	20 questions testing end user security knowledge.
4	http://www.softwareunlimited.com/securityquiz.htm	Simple test with 10 yes/no questions.
5	http://computer.howstuffworks.com/computer-security-quiz.htm	10 questions testing IT terminology.
6	http://www.bankersonline.com/technology/tech_infosecquiz.html	5 questions long information security quiz
7	http://www.cio.gov.bc.ca/local/cio/informationsecurity/March2015MatchingQuiz/March2015MatchingQuiz.htm	Three test pages each containing 8 terms in two columns that need to be paired.
8	http://www.cio.gov.bc.ca/local/cio/informationsecurity/Feb2015Quiz/Feb2015Quiz.htm	8 questions testing factual knowledge of IT security and related events
9	http://www.gocertify.com/quizzes/comptia/security-plussy0301.html	15 questions long practice test for Comptia Security+ certification exam.

5.4.2 Delivery of the Data

The success of any security awareness training programs significantly relies on the delivery method by which training material were delivered to trainees (Abawajy & Kim, 2010). Delivery method should make security as an essential attention within its targeted trainees. Based on that the social media chatting platform WhatsApp was chosen as a medium to transfer and communicate between the cyber bot and the employees due to the fact of its popularity.

The bot is hosted on a Raspberry Pi 4 Model Band 4GB Ram. The code is written using the Python language. The WhatsApp application provides an API to use in applications, but due to the complexity to acquire it and usage limitation quotas, the decision was to benefit from the web interface that it provides.

Computer vision is used to automate the manipulation of the web interface of the WhatsApp application and uses NLP to handle conversations with the end user.

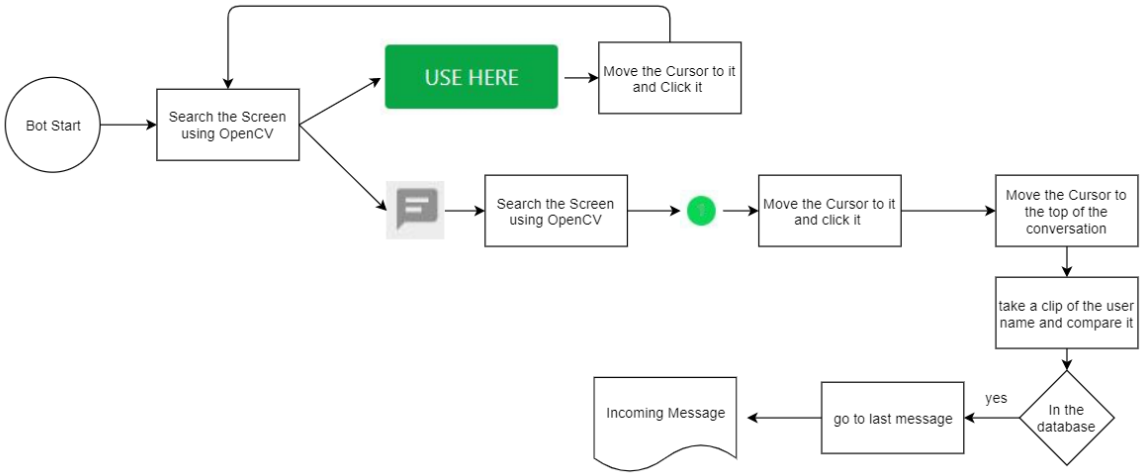


Figure 5.3 First part of the flow of the events used in the implementation

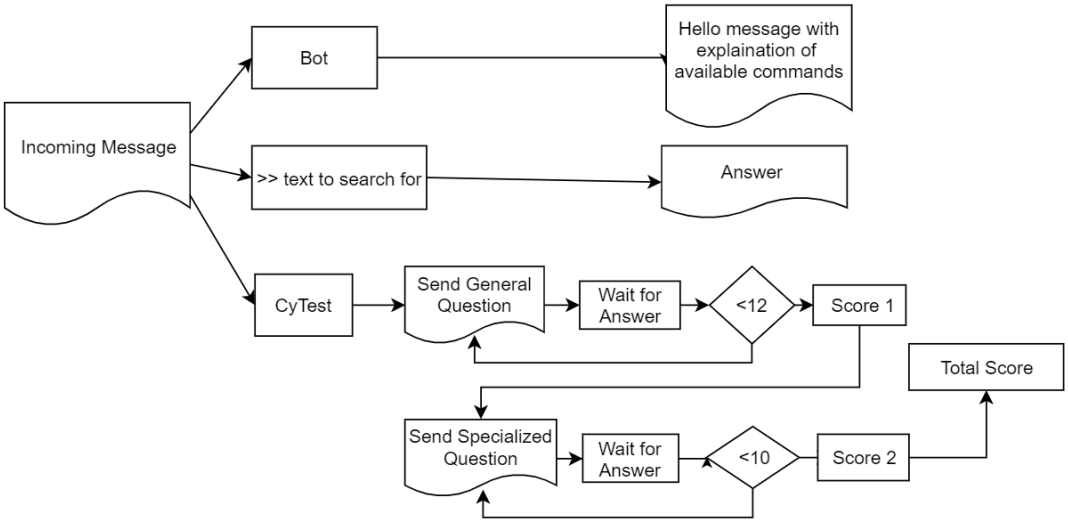


Figure 5.4 Second part of the flow of events used in the implementation

So, Python was used to automate the use of the web version of WhatsApp in a browser.

The python application opens “web.whatsapp.com” in the browser. Then, it uses OpenCV to capture events on the webpage. When the bot opens the browser, it searched the screen for USE HERE or the icon showed in the Figure 5.3 above, if it sees the USE HERE, it means the web.whasapp.com is opened in a different page, so the application moves the cursor to this button and simulates a click, and then searches the screen for this sign. If it recognizes it, it will start monitoring the screen for the green dot. The green dot appears when a user sends a message to the bot. When spotted, the application moves the cursor over the green dot and simulates a click, then it moves the cursor to the top of the conversation and takes a clip of the name present. It compares it to the ones I the database, if it is not one of the employees, it ignores the conversation, else moves the cursor to the last message and reads it.

After it reads the incoming message, if it is not **BOT** , or **two greater sign** or **pytest**, the **NLP** responds based on ***BertForQuestionAnswering*** trained using *the* ***SQuad2*** ***dataset*** to handle simple Q/A with the user.

However, if the application spots the word **BOT**, it will send this message to the user. If it spots **two greater signs**, it will take any set of words after them and provides the convenient answer. And if it spots **cytest**, the application will start to send at first 12 general information questions and then 10 specific questions. And it will display the score to the user. The general questions are common information technology questions, and the 10 specialized are according to the employee’s department. Questions for HR employees are different from those of the IT. All the scores will be linked to his account, to be assessed by the organization to get a clear view of the level of awareness its employees have, and to follow the progress they are making.

5.5 Case study: Cyber Chatbot

The solution was applied on five university students and five employees. The diversity of the age was necessary to deduce the simplicity of the solutions and the ease of use. The students come from different backgrounds related to the biomedical environment, while the employees are not technically savvy.

5.5.1 Definition of the environment

The environment depends on the WhatsApp accounts of the study group, so the people involved in the test were asked to share their WhatsApp number and to add the WhatsApp of the bot to their contacts. Then their accounts were introduced inside the chatbot so it will respond to their commands and to maintain a record of their grades for later on comparison. Figure 5.5 shows the high-level architecture of the solution that is used. To the right of the internet, it shows that the end-users only need to have a WhatsApp account. While, to the left of the internet, the chatbot is installed on the raspberry pi.

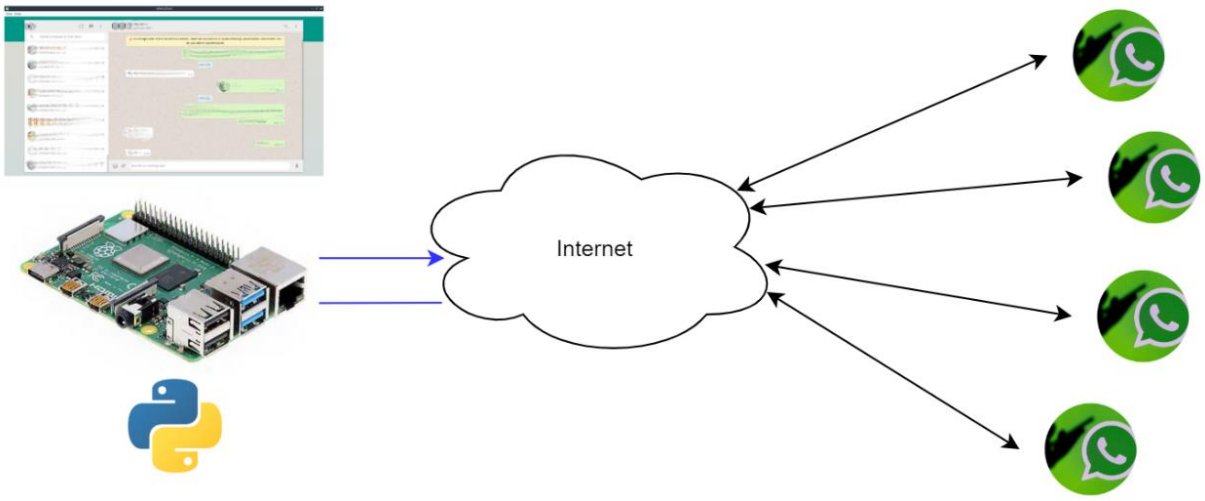


Figure 5.5 The environment of the Chatbot solution

5.5.2 Implementation

The chatbot is implemented on a Raspberry Pi and its code is written using Python language. It uses computer vision to automate the use of web interface of the WhatsApp application and uses NLP to handle conversations with the end user.

In order for the cyber bot to manipulate the web interface of WhatsApp, the browser must be linked to the WhatsApp account of the mobile. A simple scan of the QR code will

achieve that. Figure 5.6 shows the web interface when the browser is still not linked to the account.

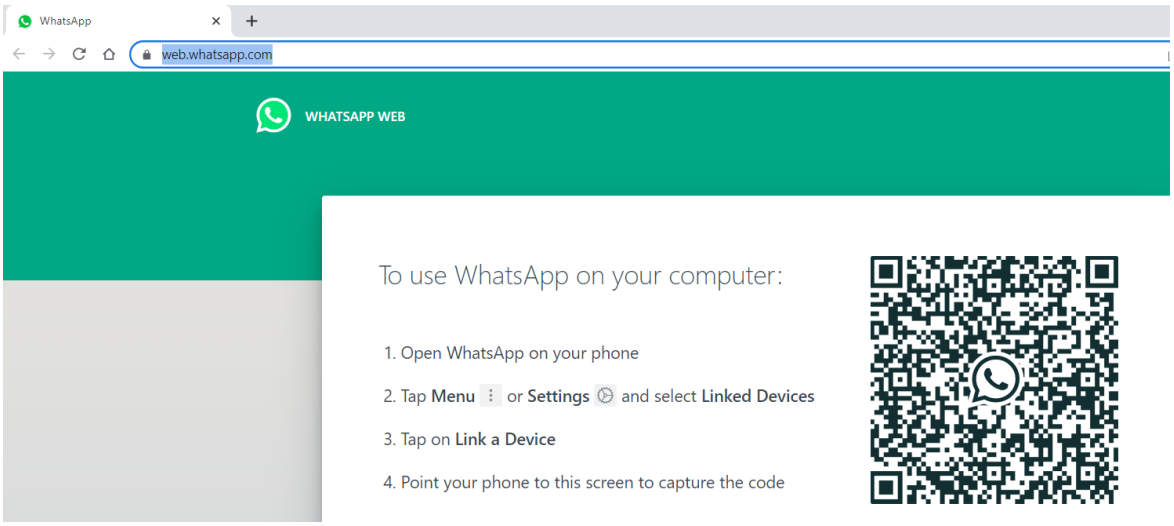


Figure 5.6 The web page of WhatsApp opens like this, when the device is not linked to the browser

While the browser that is opened by the chatbot is still not connected to the Whatsapp account the chatbot will stay in the waiting mode as shown in figure 5.7.

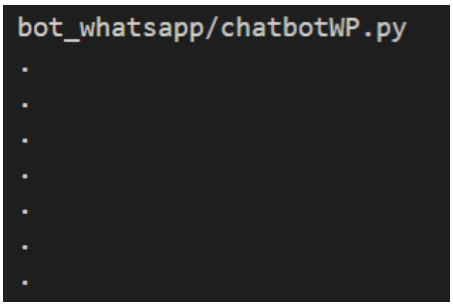


Figure 5.7 the chatbot waits for the browser to be linked to the device

When the browser gets connected to the Whatsapp account it will display the message shown in figure 5.8. After that, the cyber bot will be on alert to respond based on the input from the WhatsApp web interface.

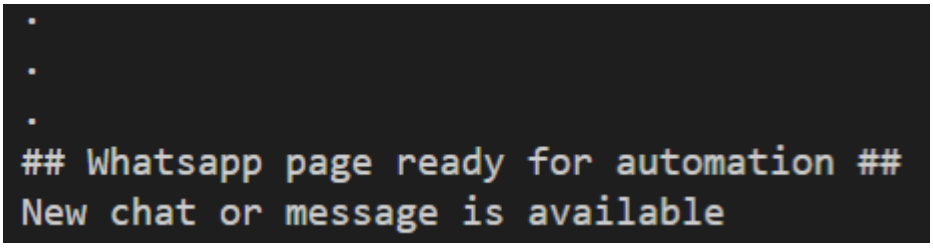


Figure 5.8 the browser is linked to the device

When the user sends the keyword “BOT”, the chatbot replies with the greeting message shown in the figure 5.9 and reminding the user of the different options available for him to use.

Figure 5.9 Greeting message of the chatbot

When the user sends the keyword “>>” followed by a word or phrase, the chatbot replies with a definition of related to this word or phrase as shown in figure 5.10 and 5.11.

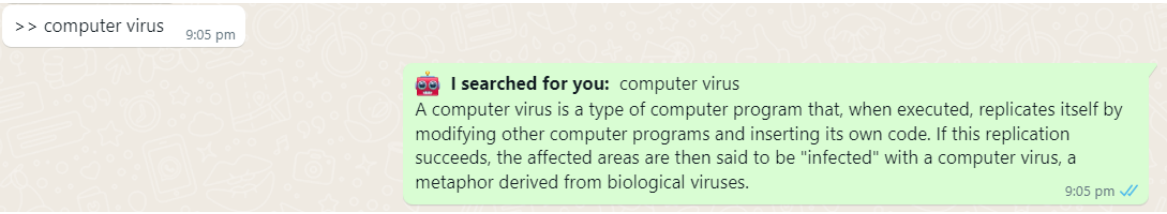


Figure 5.10 The user is searching for the phrase “computer virus”

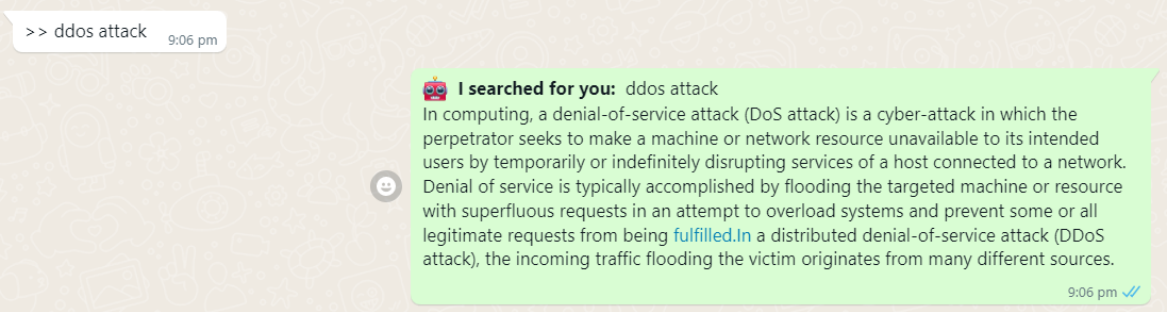


Figure 5.11 the user is searching for the phrase “ddos attack”

When the user sends the keyword “Cytest”, the chatbot starts the quiz mode and starts sending the user the different questions and waits for the answer as shown in figure 5.12.

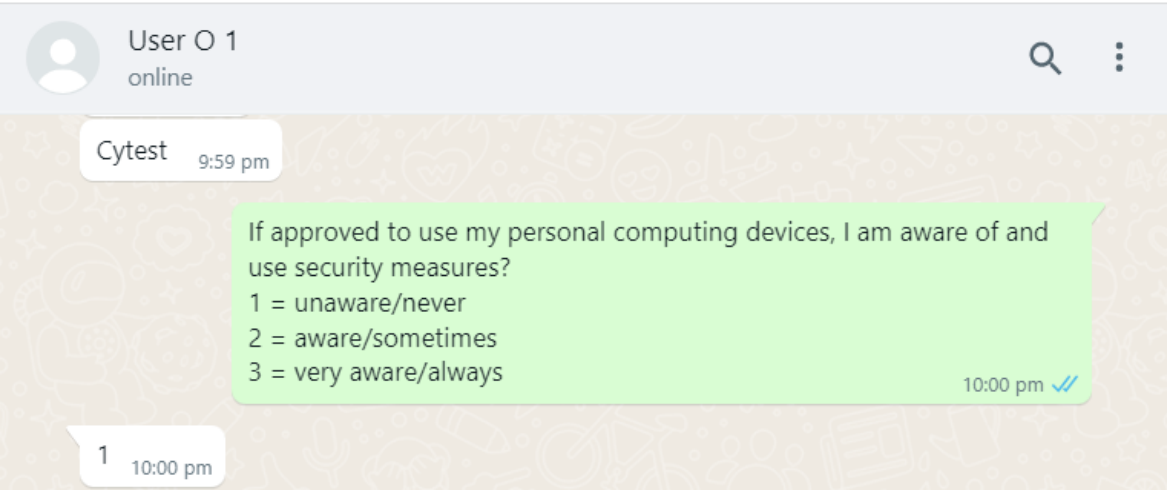


Figure 5.12 the user is presented with the questions after typing the keyword “Cytest”

If the user replied other than the numbers specified, the chatbot will reject his response and ask him to send a valid answer instead as shown in figure 5.13

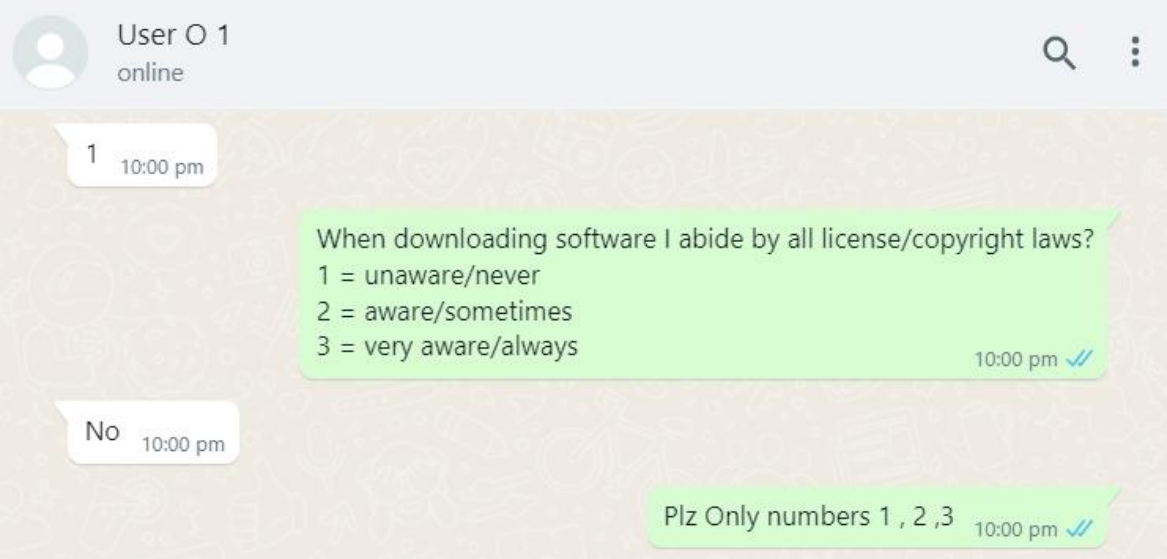


Figure 5.13 The user has to answer using the specified numbers only

After the user completes the first 12 questions, and score is displayed for him to assess his responses. And after he finishes the specialized 10 questions, a score for these 10 answers will be displayed followed by the total score achieved

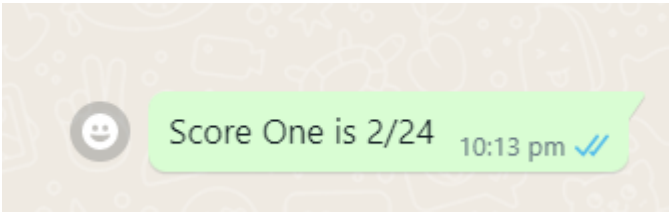


Figure 5.14 the user gets first score based on the answers he provided for the General Questions

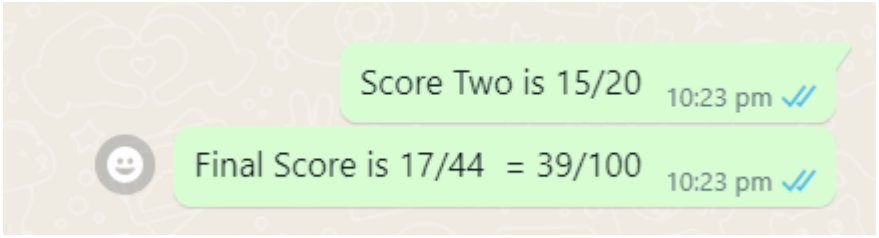


Figure 5.15 The user gets the second score based on his answers for the specialized questions and his final score

The scores are saved per user and are accessed by the Cyber Information Security Officer (CISO). The CISO can monitor the overall scores of the enterprise and can help the employees to focus more on the weak points that appeared. Added to that, he can assess the progress made after repeating the test by the same user.

The IT administrator could, at any time, update this list of questions by updating the config file that is used by the chatbot to update its bank of information/questions.

Thus, the evaluation questionnaire consists of the following (note that the bank of questions is updated continuously):

- Twelve questions from a list of 150 general questions chosen randomly, but equally, from three categories;
- Ten questions from a list of 50 specialized questions chosen randomly from the category where the user belongs;

This evaluation test can be taken by the user anytime. However, each three months, at least one test should be performed or at random periods of the week with 12 questions to ensure continuous flow of knowledge.

Added to that, the chatbot intelligent system can be used by the IT administrator to send alarms when a cyber incident happens or in case of cyber-attacks. These alerts are sent every day in the morning. Three top alerts are displayed and the source is CyberSecurity & Infrastructure Security Agency (CISA) (CISA, National Cyber Awareness System, 2022).

As for the NLP part, the *"BERT"* (Devlin, Chang, Lee, & Toutanova, 2018) model was used and more specifically *"BertForQuestionAnswering"*.

5.6 Testing and Validation

The system was tested by several users. Figure. 5.16 shows the conversations of two users doing the quick test in the same time. The chatbot was able to handle them simultaneously as shown by the time in the chat window.

(a)

(b)

Figure 5.16 Simultaneous test done by two users (a) and (b)

The users showed satisfaction from the bot's performance. Further testing will be performed to evaluate the effect of the bot on the overall company cyber awareness and the reduction of cyber incidents related to human factors.

Moreover, figure 5.17 shows the alerts sent to the users every day. Currently the top three alerts are being sent from the Cybersecurity and Infrastructure Security Agency (CISA, Cyber Threat Source Descriptions, 2020).

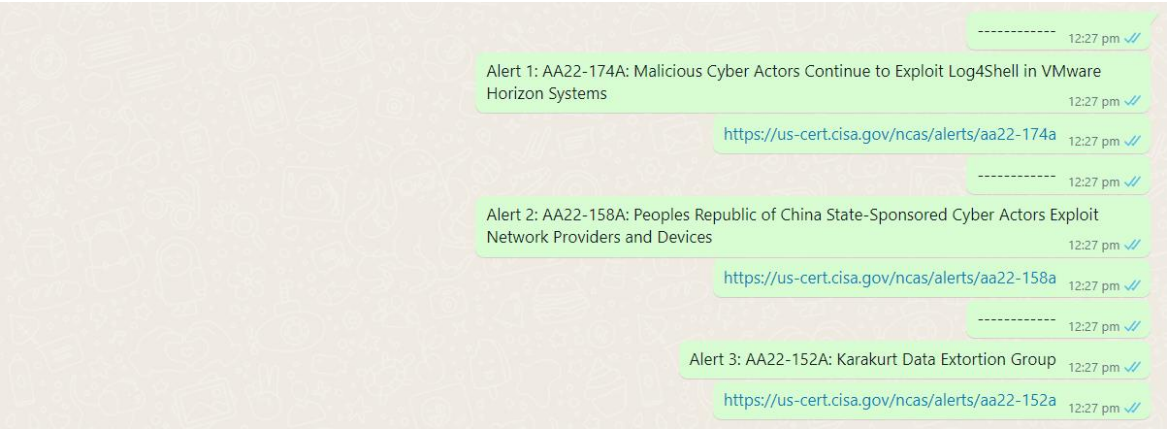


Figure 5.17 The cyber alerts *automatically* r*eceived every day*

5.7 Conclusion

To sum up, this chapter has presented a novel idea to design chatbots capable of delivering the newest information and trainings to the employees of a company. The IT department prepares a list of Q&A and fills the database of the chatbot. This latter will be able to train the employees, automatically update its database of any security breach and the way to behave in case of attack and the self-assessment of these employees. This chatbot will be able to maintain the records of each employee, to evaluate his progress and to propose some trainings to reduce the weaknesses he is encountering. The implementation of this AI-based bot has shown a great impact on the employee as well as an easier way to teach employees and to keep them up-to-date concerning security threat.

Chapter 6 Implementation of the solution

Outline

6.1 Introduction

The rapid pace of digital transformation showed the need for a methodology to help the responsible of cyber security in an enterprise to maintain security of the Information Technology System (ITS). Small to Medium Enterprises (SME) are always victims of cyber-attacks because they focus on their core business and fail to secure their information technology security. This is due to the complexity related to the implementation of available cyber security frameworks which needs a skilled professional to achieve it.

After presenting different solutions for the three defined levels of the system architecture in chapters three, four and five, this chapter will present an end-to-end implementation of the different solutions in one whole system. A medical application will be used for this implementation as it is among the most critical systems.

The system is a medical monitoring application. It consists of many sub systems that will ensure the transfer of data from the patient to the medical supervisor. Cyber security measures will be applied to ensure the confidentiality, integrity and availability of the system.

The system is designated to monitor patient's health while doing their normal daily activity. Vital signs are continuously measured and collected to inform healthcare providers whenever there is a medical issue. This is mainly applied for clients having a medical history as heart failure, diabetes or even epilepsy. Thus, the main objective is to provide a secure data flow from the patient to the healthcare provider to monitor the patient's health and take right decision whenever it is necessary.

The system will be tested using controlled cyberthreats to ensure its reliability and endurance facing real life cyber-attacks.

6.2 System description

The system is made of two main components: hardware and software. The security of the system will be maintained based on the three levels as described earlier in the previous chapters, *i.e.* Network Level (NL), Device Level (DL), and Human factor Level (HL).

An analysis of the system needs to specify the current state of the assets and to determine the key points that can have an impact on its availability, confidentiality and integrity. Then, an estimation and a classification of the risks is made in order to determine the quick wins that can have a high impact on the security and define the adequate treatment to be applied in order to minimize all the risks already identified.

6.2.1 Identification of Assets

Assets are the information technology resources that are valuable for the enterprise and need to be protected. Any damage or loss of one or more of these assets will have a bad consequence either financially or on the reputation of the enterprise. Assets will be divided into two categories: primary and secondary. The primary assets consist of the device data and the configuration data. Device data is the data generated by the system and sent/received by the servers. Configuration data is the data used in the implementation of the different security mechanisms and policies. As for the secondary assets, they represent the network components, the software, the devices, and the users.

6.2.2 Modeling

Modeling of the ITS is necessary. It gives the CISO an overview of the whole system (Wang, Chao, Lo, & Wang, 2017), and it helps determine the Points of Entry (PoE). PoEs are the entry points that can be used by an adversary in order to penetrate to the ITS and compromise the assets. Data Flow Diagrams are an ideal modeling methodology because they can accommodate the NL, DL and HL in the same diagram. Figure 6.1 shows the data flow diagram of the ITS before implementing any security. The clients (*i.e.*, patients) connect to a webserver that will handle the request and process the data received. Figure 6.2 shows the treat model of this system. The system is composed of three boundaries that needs to be handled in a different way security wise. The client requests should be handled and filtered in a DMZ zone, the true data needs to be handled in a protected zone and finally the data should be in a restricted zone.

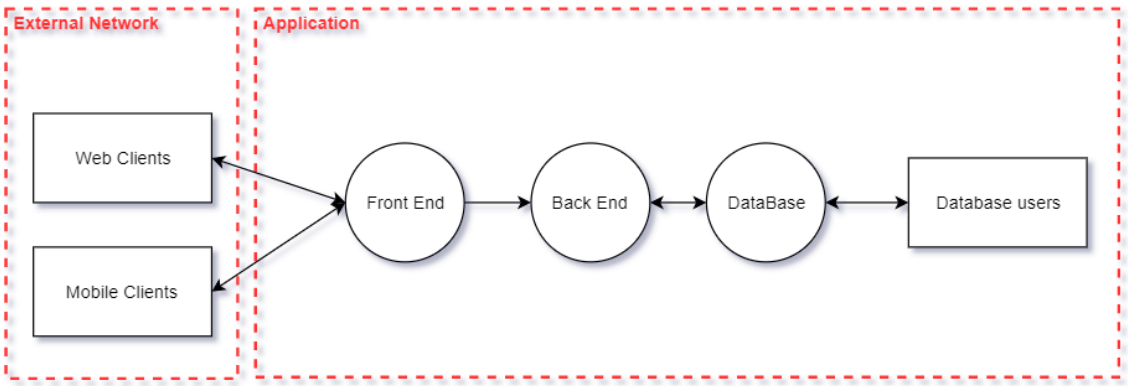


Figure 6.1 Data flow diagram of the ITS before applying the security measures

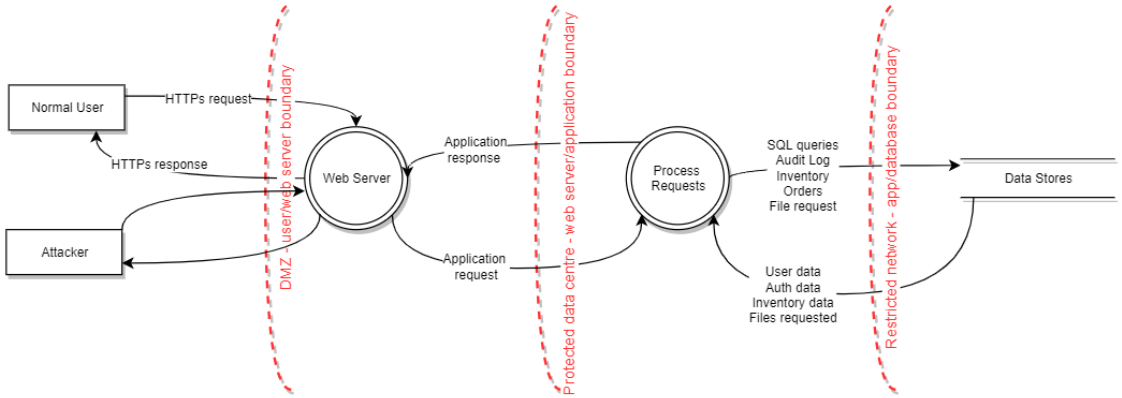


Figure 6.2 Threat modelling of the ITS

Back to Figure 6.2, the medical system will be divided into three main environments:

- The first one regroups the patients (or clients) who are connected to the gateway server, known as *R2*. The data of the sensors are regrouped and sent, via API call, along with some delimiters to this server;
- The second part represents the gateway server that handles the requests of the clients, checks for data anomaly and forwards the data to the backend server, known as *R3*. This is the last components that is connected physically to internet;
- The third part represents the backend server. This server is just connected to the gateway server without any internet connections. Its main role is to respond to the healthcare provider requests and to save the data of the whole system securely.

Based on the above description, Table 6.1 lists in details all types of assets found in this medical system.

Table 6.1 Primary and secondary assets present in the system

Primary assets	
Device data	Patients' medical data consisting of the different metrics collected from the connected IoT device
Configuration data	• Configuration files for security, for devices, and for applications; • Backup policies; • Data retention policies.
Secondary assets	
Network components	• 3G/4G network • Switch

Devices *	• Five Raspberry pi 4 • Three Raspberry pi 0
Humans	• Patients • Healthcare providers • IT supervisor

Kindly note that the Raspberry Pi microcontrollers are divided as follow:

- Raspberry Pi 4: one for the gateway server, one for the backend server, one for the WhatsApp chatbot server (will be defined later on) and two for the healthcare providers;
- Raspberry Pi zero: all three are connected to the medical sensors of three different patients using this system.

6.3 Solution implementation

Security is the utmost objective of any enterprise, and especially if the data is related to medical vital signs. Based on Figure 6.1, one can identify three main entities in the system: the patient sending the data, the server receiving the data, and the healthcare personnel monitoring the patient through the data analysis.

So, to maintain the security of this system, a solution will be defined based on the procedures that were already applied in the previous three chapters. Thus, we propose the following:

- an AI-based software to filter traffic and maintain security of the transferred data;
- a unidirectional data diode to prevent any exfiltration of sensitive data from the restricted zone;
- a chatbot to integrate cyber awareness for the users of this system (healthcare providers and patients) in order to limit the human factor deficiencies at the security level.

Figure 6.3 shows the ITS model after applying the security measures to mitigate any cyber-attack that can hinder the proper functionality of the system.

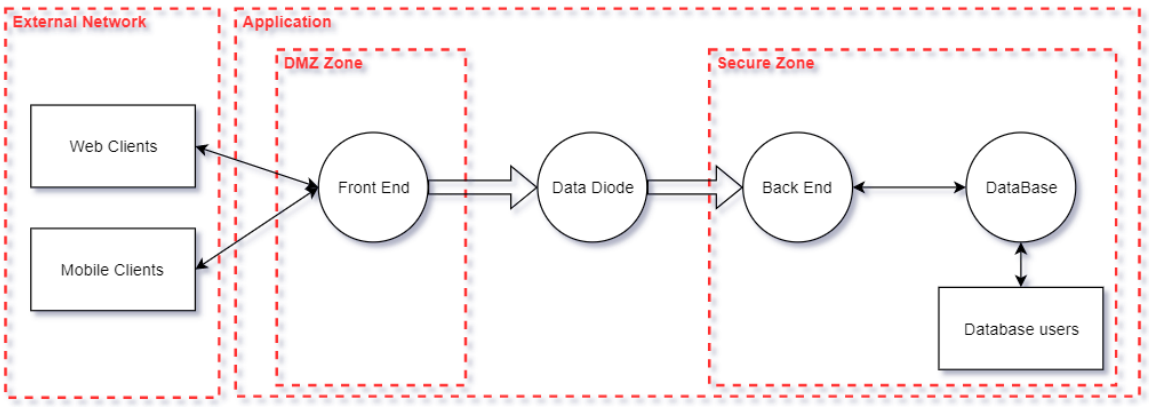


Figure 6.3 Data flow diagram of the ITS after applying the security measures

In more details, the system is consisting of four entities:

- A patient equipped by a heart rate and a SpO2 sensors to check his health while doing his/her normal daily activity;
- A gateway server (IoT gateway) placed in a DMZ to receive the health data of multiple patients. The AI-based software presented in chapter four will be responsible for:
    - monitoring the health of the patients for abnormality;
    - notifying the healthcare provider to handle the issue and reassess the data from the patient in case of abnormality in the medical measurements;
    - monitoring incoming data (type, length, and conformity) for irregularities from the network and device perspectives.
- The unidirectional network device (Data Diode) presented in chapter three is installed between the IoT gateway and the healthcare provider's server in order to deny data getting out of the backend server;
- The healthcare provider's server that is used to monitor the patients' health. The healthcare provider must be connected directly to the secured network without going through internet. Once an abnormality is detected, the system flags it, the healthcare provider must confirm it and he contacts the patient through WhatsApp application for further follow up.

Although not all vulnerabilities are resolved, this work deals with the most frequent and common ones according to OWASP API top ten security list (OWASP, OWASP API Security Project, 2020).

Concerning the hardware layer, the use of air gapped (isolated) network has been the most efficient way to ensure that a network can't be compromised remotely. Air gapped systems ensure that, in case of a cyber security attack, no data can be communicated to the outside. The two most famous frameworks for cyber-attacks, the Lockheed Martin Cyber Kill Chain (Martin, 2021) and the MITRE ATT&CK (Corporation, 2021), have described in their approaches the reconnaissance and the communication with the Command and Control (C&C) as vital for the success of any Cyber-attack. Data diodes provide security from these two phases because of the physical nature of unidirectional flow of data that doesn't allow the devices to interchange information.

As for the software layer, an AI-based software is installed on the input side of the gateway server to assess data transferred to the inside and to block any malicious action. Using Artificial Intelligence applications provide great security features, high transparency and enhance efficiency (Dhingra, Jain, & Jadon, 2016). The development of edge computing permits data, generated by the IoT devices, to be transferred to the edge gateways for further process and analysis before being forwarded to the backend server that is used by the monitoring healthcare provider.

Before starting the testing and the validation of the system, figure 6.4 presents the system architecture. It consists of three patients (connected to three Raspberry Pi 0), a gateway server (*R2* – Raspberry Pi 4), a backend server (*R3* – Raspberry Pi 4), a chatbot server (*R1* – Raspberry Pi 4) and two healthcare providers (two Raspberry Pi 4).

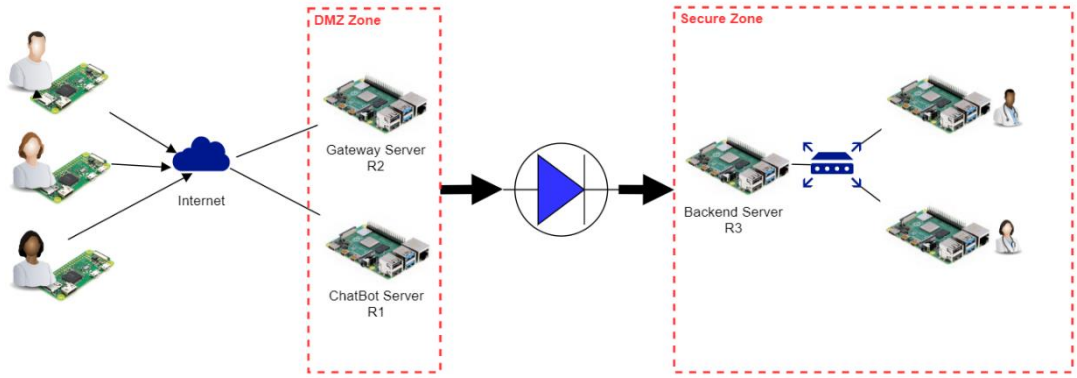
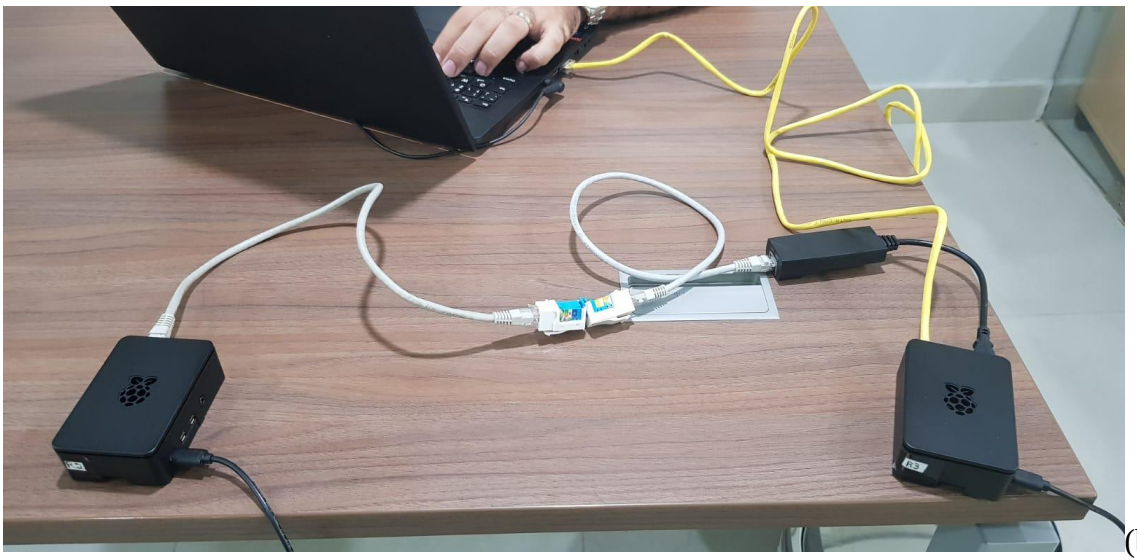


Figure 6.4 Medical system architecture

Figure 6.5 shows the real implementation of this system. However, due to size restrictions, only one patient and one healthcare provider were integrated in these photos.

(a)


(b)

(c)

Figure 6.5 Photos of the implemented system along with one patient and one healthcare provider

6.4 System testing

This section will be divided into three parts, relative to the three levels already defined:

- Measurement and Validation of data transfer between the patient, the gateway server and the backend server and of the delay, generated due to the data diode, of the data sent from the gateway server to the backend server;
- Prioritization of the vulnerabilities at the level of both servers;

- Education and evaluation of cyber awareness using chatbot for all human assets using this system.

6.4.1 Data transfer

Having a secure transfer between the DMZ and the secure zone is solved by using the Data Diode. NextCloud is installed on the gateway and backend servers in order to transfer data securely through data diode with the use of any hardware while using a user graphical interface. For every user, an account is created on both servers. Figure 6.6 shows the process of data transfer:

1- the user first uploads a file to his folder in the DMZ side; when the software notices a new file, it moves it to a new location for treatment – the file is scanned by ClamAV for viruses and malwares:
    a. if it is a malware, it is deleted and the user is notified as shown in figure 6.7;
    b. if it is clean, an MD5 hash is created, and both the file and the hash are bundled in one file.
2- Using the UDP, over the data diode, the file is transferred to the server in the internal network. There, it is unbundled and an MD5 hash is calculated for the file and compared to the value the came in the same bundle.
    a. If they match, it is presented to the user in the folder.
    b. Otherwise, it is deleted because it could be damaged during the transfer.

Figure 6.8 represents the flow of action from the beginning till the end as already presented in the above paragraph.

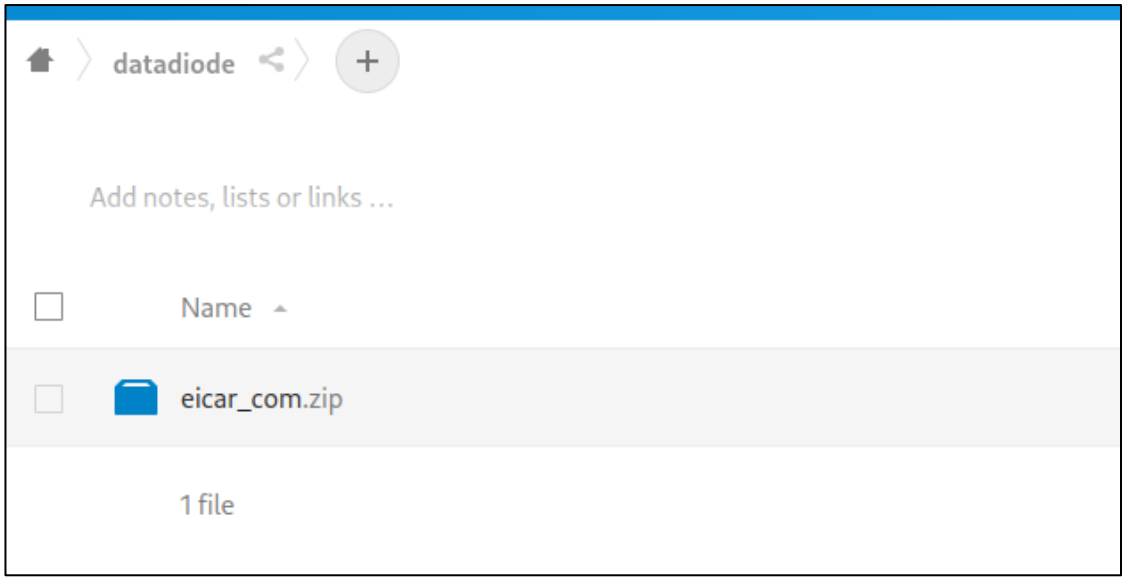


Figure 6.6 Zip file containing a virus

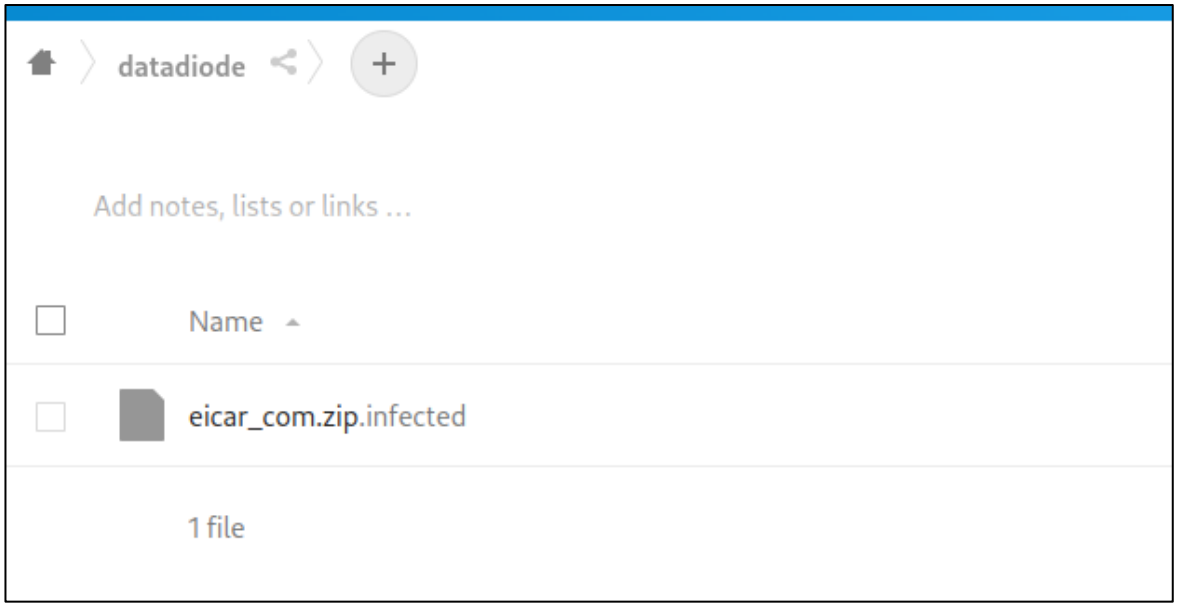


Figure 6.7 File rejected with message to the user

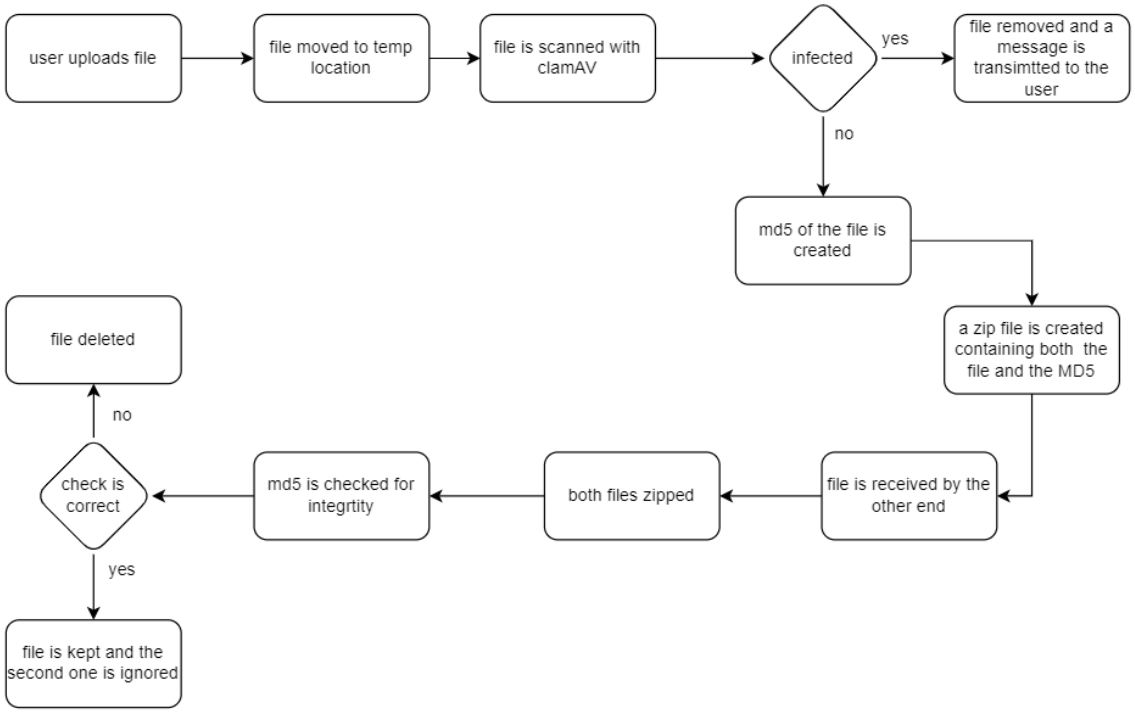


Figure 6.8 Flow of action from the beginning till the end

After presenting a high-level description of the implemented solution, here after the details will be presented. Bash scripting is used to create the software responsible for maintaining the connection and for preparing the files to be transferred. CalmAv was installed to provide protection against malwares and viruses. The data diode presented in chapter three is used to secure the communication between DMZ and internal zone.

As for the measurements shown in table 6.2, it contains the time needed to transfer a file, with a predefined size, from the gateway to the backend, with and without the use of the data diode. As for figure 6.9, it represents a chart comparing the timing for the transfer of data between both servers. As it is clear from both outputs, the data diode did not affect the time needed to transfer data between both servers. Note here that the size of the packets was not only chosen according to the medical data to be sent (as it represents a very small file size of about some KB), but the main purpose was to make sure that, even for big data files, the data diode will not affect the speed of transfer between the servers.

Table 6.2 Primary and secondary assets present in the system

Size MB	***Time in sec w/diode***	***Time in sec w/o diode***
0.98	75	75
1.5	72	72
3.2	71	71

Size MB	*Time in sec w/diode*	*Time in sec w/o diode*
5.24	78	78
34.8	78	78
38.6	78	79
53.4	83	84
143	110	108
381	170	171

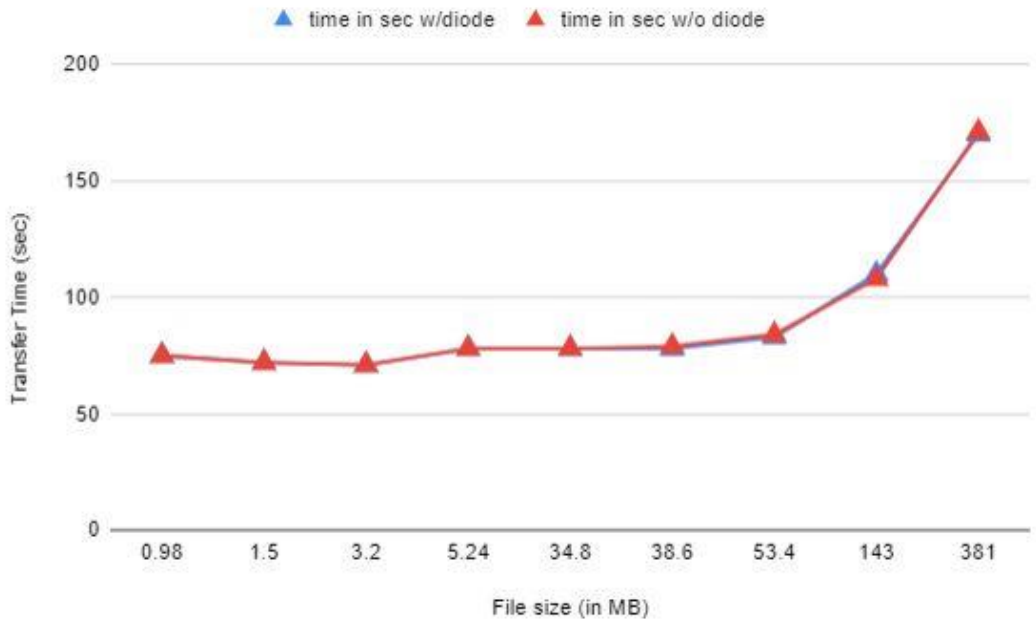


Figure 6.9 Chart comparing the time required to send data between two servers with and without the use of data diode taking into consideration the file size

After presenting the first part of the measurements, the second part of this section will show the data integrity detected when sending the patient data at two levels:

- Between the patient terminal and the gateway server;
- Between the gateway server and the backend server.

To do so, the first test consisted of using real data from the three patients connected to the system. However, due to the need of applying a stress test, the original data was replayed by a virtual machine (that was acting as exciter instead of the patient) and data integrity has been measured. Table 6.3 shows the obtained results. In this table, the number of records has been defined as well as the hitting and missing percentages and the number

of connected patients. For better visualization, a graph is represented in Figure 6.10 to compare between the hitting percentages per test level.

Table 6.3 Testing data integrity in the system

Test Level	Num. of records	Num. of patients	Hitting percentage	Failure percentage
Patient → Gateway	10 000	1	100.0%	0.00%
Gateway → Backend	10 000	1	100.0%	0.00%
Patient → Gateway	10 000	5	100.0%	0.00%
Gateway → Backend	10 000	5	99.87%	0.13%
Patient → Gateway	10 000	10	99.91%	0.09%
Gateway → Backend	10 000	10	97.31%	2.69%

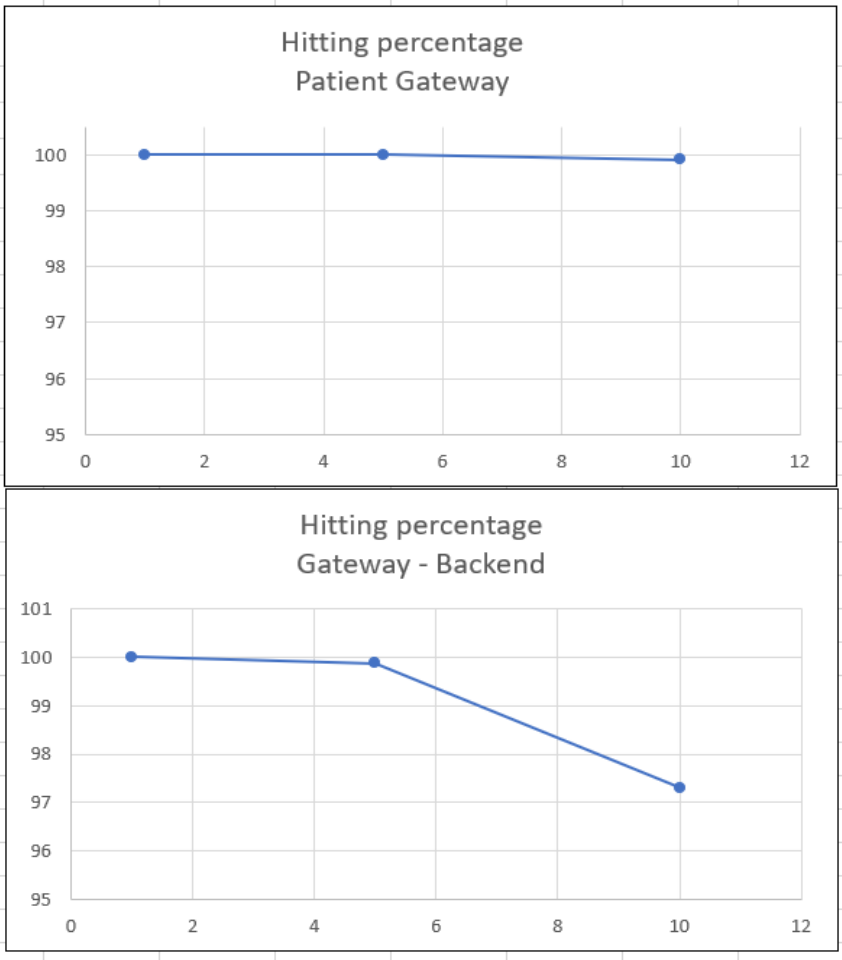


Figure 6.10 Chart comparing hitting percentage from patient to gateway and gateway to backend

Based on the obtained results, one can identify that the failure rate when transferring the data between both servers is much lower than the rate obtained when the patient sends his data to the gateway. In fact, the only vulnerability that exists between the servers is the congestion that may lead to data loss. This can be identified due to the comparison between logs at the level of both servers. However, the threats between the patient and the gateway have multiple causes as the presence of a third-party entity that can inject or alter the original data or the loss of the data between both ends due to connectivity issue for example.

To sum up, different tests have been made at the network level to validate the implanted solutions and the overall results showed that these proposed solutions do not create delays and assure an integrity level above 99%.

6.4.2 Prioritization of vulnerabilities

The integration of the security at the device level was achieved through the implementation of advances algorithms at each server side. In fact, two steps are required to update the prioritization list of vulnerabilities:

1. Data is collected from different sources (NIST, ExploitDB, CIRCL) to generate a new priority list for vulnerabilities. The complete procedure was presented in chapter 4 and it was implemented in this system as is (as shown in figure 6.11). Note that this procedure is just applied on the gateway server and the list is transferred regularly to the backend to process in the second step;
2. WAZUH software is installed on both servers (gateway and backend) to identify all vulnerabilities. A report will be generated and the servers have to cross reference between the identified vulnerabilities and the ones collected from step 1. An updated list will be generated at the server side containing the vulnerabilities to be solved by priority. Figure 6.12 shows the complete procedure of this phase.

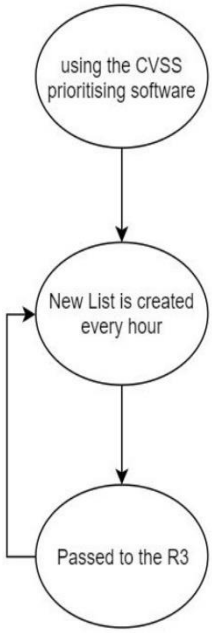


Figure 6.11 Procedure to generate an update list of the CVSS for the gateway server

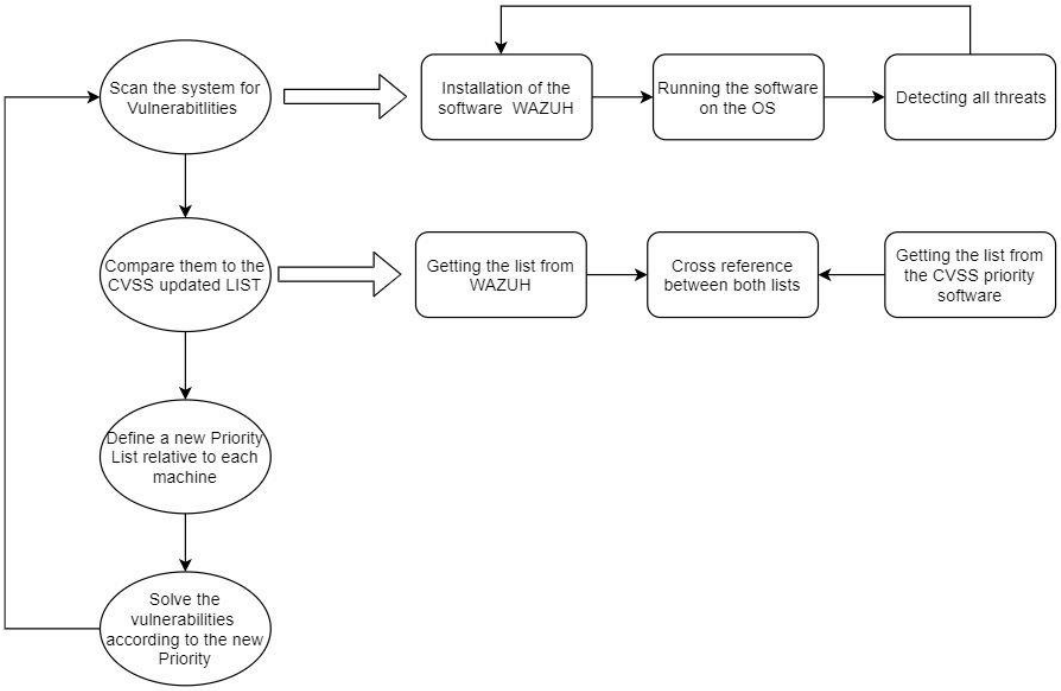


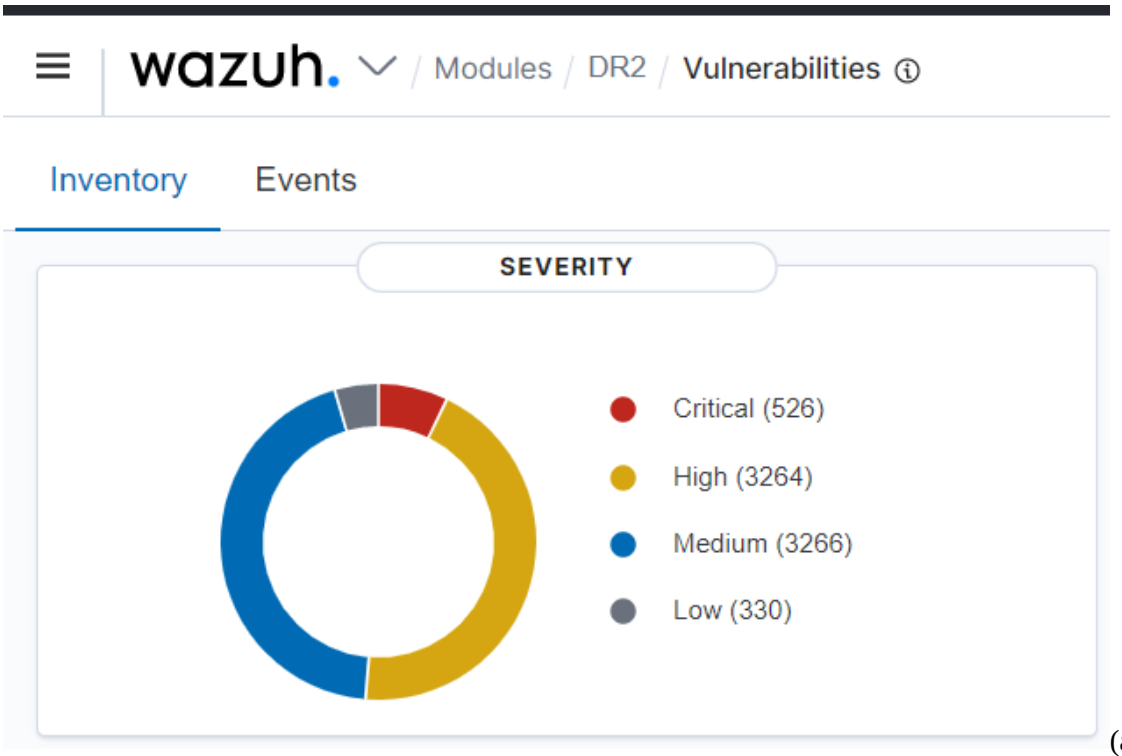
Figure 6.12 Procedure for updating the prioritization list for the backend and the gateway servers

After presenting the procedure, figures 6.13 and 6.14 show respectively the CVSS updated list displayed on the gateway server and the WAZUH report obtained from the

gateway server (same output could be obtained for the backend server). As for figure 6.15, it shows the new classification of vulnerabilities on the gateway server based on the cross referencing between the values obtained in figures 6.13 and 6.14.

CVE	BaseScoreV2	BaseScoreV3	PS3
CVE-2014-0224	5.8	7.4	4
CVE-2014-6271	10	9.8	3.75
CVE-2008-1447	5	6.8	3.731461864
CVE-2014-3566	4.3	3.4	3.729872881
CVE-2015-4000	4.3	3.7	3.670815678
CVE-2017-5638	10	10	3.63559322
CVE-2014-0160	5	7.5	3.623675847
CVE-2017-5754	4.7	5.6	3.623411017
CVE-2010-2943	6.4	8.1	3.611493644
CVE-2018-7600	7.5	9.8	3.601694915
CVE-2016-3714	10	8.4	3.594809322
CVE-2020-1945	3.3	6.3	3.584480932
CVE-2016-3092	7.8	7.5	3.583951271
CVE-2016-10175	5	9.8	3.58315678
CVE-2017-9788	6.4	9.1	3.582362288

Figure 6.13 CVSS updated list



(a)

Vulnerabilities (7386)

Filter or search

Name	Version	Architecture	Severity	CVE	CVSS2 Score	CVSS3 Score ↓
gir1.2-javascriptcoregtk-4.0	2.20.1-1	amd64	Critical	CVE-2020-13753	7.5	10
gir1.2-webkit2-4.0	2.20.1-1	amd64	Critical	CVE-2020-13753	7.5	10
libjavascriptcoregtk-4.0-18	2.20.1-1	amd64	Critical	CVE-2020-13753	7.5	10
libwebkit2gtk-4.0-37	2.20.1-1	amd64	Critical	CVE-2020-13753	7.5	10
firefox	59.0.2+build1-0ubuntu1	amd64	Critical	CVE-2019-11708	10	10
firefox	59.0.2+build1-0ubuntu1	amd64	Critical	CVE-2018-18505	7.5	10
libsmbclient	2:4.7.6+dfsg~ubuntu-0u...	amd64	Critical	CVE-2020-1472	9.3	10
libwbclient0	2:4.7.6+dfsg~ubuntu-0u...	amd64	Critical	CVE-2020-1472	9.3	10
samba-libs	2:4.7.6+dfsg~ubuntu-0u...	amd64	Critical	CVE-2020-1472	9.3	10
ghostscript	9.22~dfsg+1-0ubuntu1	amd64	Critical	CVE-2021-3781	9.3	9.9
ghostscript-x	9.22~dfsg+1-0ubuntu1	amd64	Critical	CVE-2021-3781	9.3	9.9
libgs9	9.22~dfsg+1-0ubuntu1	amd64	Critical	CVE-2021-3781	9.3	9.9
libgs9-common	9.22~dfsg+1-0ubuntu1	all	Critical	CVE-2021-3781	9.3	9.9

(b)

Figure 6.14 WAZUH report

1	CVE	BaseScoreV2	BaseScoreV3	PS
75	CVE-2020-13753	7.5	10	3.534957627
2513	CVE-2018-18505	7.5	10	3.346927966
3047	CVE-2020-1472	9.3	10	3.304555085
23711	CVE-2019-11708	10	10	1.566737288

Figure 6.15 Prioritization list of vulnerabilities

The results show that even though all the CVEs in figure 6.14(b) have a CVSSv3 score of 10, our priority scoring system provided an ordered list that the CISO can use to resolve the vulnerabilities of the information system.

6.4.3 Chatbot testing and evaluation

A separate server has been implemented to service the chatbot feature for the users. However, all logs and data will be sent to the backend continuously in order to allow the IT to monitor the users' activities. Added to that, a data diode will be integrated between both servers to ensure the privacy of data from the backend side.

A database is upload on the gateway server and periodic updates are being applied. As for the log and results transferred from the gateway to the backend, it is done periodically. Whenever sent to the backend, the logs are completely removed from the gateway; however, the results of the self-tests are kept on the gateway for further comparison with future tests, but they are encrypted to protect them in case of server being compromised.

To launch these features, the user (patient or healthcare provider) must initiate the connection with the chatbot via WhatsApp. The same procedure, as presented in chapter 5, will apply. Thus, the user can ask for information related to security or can undergo a cyber awareness test. All logs (user ID and login time) is saved in a database on the gateway server before being transmitted to the backend server. Figures 6.16 shows the database schema. The users table contains all the information about the users specially the department he belongs to because accordingly the questions will be presented in the quiz. Also, it contains the last time the user used the chatbot and how many times he took the quiz. These last two values are important for the CISO of the enterprise to follow up on the users that are not using the Chatbot cyber awareness features and quizzes. The scores table contains the user score per quiz. The value in this table are used by the CISO to track the weak users in the enterprise and act accordingly to reduce their possible impact on the security of the information technology system. The department table contains the name of departments of the enterprise. The questions table contain all the questions used in the quizzes. The questions are organized per type *e.g.* general, specialized. Each record contains the question and the id of the correct answer. Finally, the answers table contains the possible answers per question that is displayed for the user.

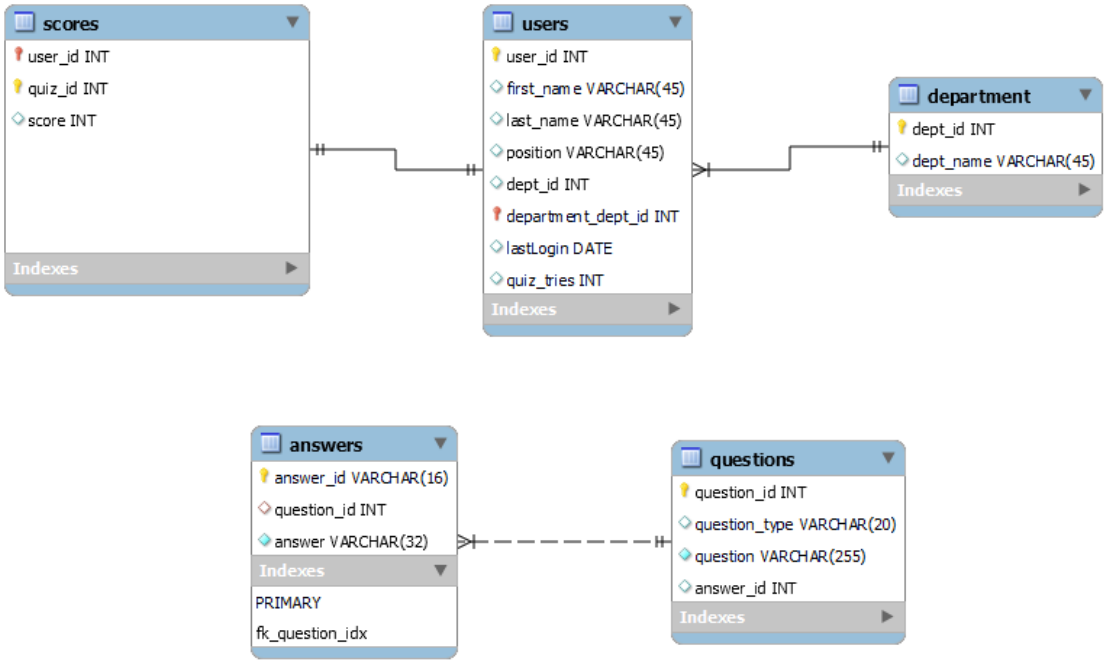
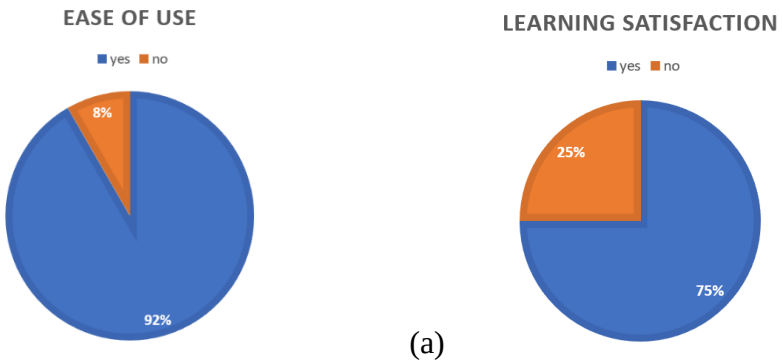


Figure 6.16 Database of the chatbot

As this section does not contain any measurements for vulnerability and threats attacking the system, the only purpose was to educate and evaluate the users' security knowledge to help limiting threats coming from the human factor.

However, one should mention that the users that tested this feature were satisfied, and a small questionnaire showed that they got lots of information concerning system security that they haven't heard of before. In more details, figure 6.17 shows a chart presenting the output of the questionnaire filled by 12 users (patients and healthcare providers). 11 out of the users found the chatbot quiz easy to use, and 9 users found the learning experience satisfactory.



(a) (b)

Figure 6.17 Pie charts representing the satisfaction and the usability of the chatbot

6.5 Validation and Performance

After presenting the different security features of the medical system that was proposed in this chapter, this section will conclude of the validation of the system and will determine its performance with and without the use of these additional tools.

To start with the validation of the secured system, the measurements related to the network and the device levels showed that the data had arrived at the backend server without any delay. Added to that, the software installed on the backend and the gateway servers were able to identify the loss or the falsification of the medical data sent by the patient. Added to that, in order to increase security, the healthcare provider was obliged to contact the patient by phone or via WhatsApp whenever an abnormality is encountered. Combined together, the data losses and the vulnerabilities detected were very tiny (below 1%) compared to the size of the sent records.

As for the prioritization of the vulnerabilities, an open-source software, WAZUH, was used on both servers to identify all received vulnerabilities in order to compare them later on with an updated list of CVSS. The up-to-date prioritization list will identify which vulnerability to solve first. This procedure is running periodically on the gateway and the backend servers. The proposed solution helps in defining a road map to solve vulnerabilities based on their simplicity of resolution and impact on the system.

Finally, a chatbot has been realized and implemented to increase the users' cyber awareness as a big portion of the cyber-attacks are related to users' ignorance. After using this feature, that is provided easily through WhatsApp, the satisfaction and the usability percentages were very high.

To conclude, one can confirm that the security tools added to this system have offered a high integrity level with a very reduced cost. Such solution is so interesting as they are autonomous and do not require a continuous input from the system administrator. Added to that, the presence of AI-based software will help in continuously updating the security protocols in a way to keep the system always up-to-date towards new threats and vulnerabilities. Finally, one should note that, even with the implementation of the different software and algorithms, the response time of the system was not affected; this feature is so important because, usually, installing such demanding software on a processor will lead to reduce its performance.

6.6 Conclusion

This chapter was dedicated for the validation of the proposed solution at the level of all the components of the system. After validating the solutions that were presented at the network level, device level and human factor level, this chapter combined all solutions together while offering a technical way to implement them without having any conflict. The obtained results were very promising, especially that the proposed work was one of the few, if not the only one, that tackles simultaneously all security threats present in the system.

Added to that, the other contribution that was achieved in this work is the modelling that is absent in almost all similar works. The IDEF0 models that were presented per layer in the previous chapters were reinforced by several flowcharts, data flow diagrams and sequence diagram UMLs to identify and solve threats attacking the system.

To conclude, the system was completely implemented and the measured performance showed that the security has been widely settled without creating any delays on the system response, reducing the users' productivity or requiring a big budget for features purchasing.

Conclusion and future works

The research work presented in this thesis brings a lot of contributions related to cyber security. It is a solution that can help small to middle companies protect their information system using opensource based software and DIY (do it yourself) hardware. The cost of implementing the solution presented in the thesis is nearly negligible and the simplification of the solution makes it easily implementable with no extensive knowledge in security or machine learning.

Our solution is based on segregating the system into three levels of security: Network Level (NL), Device Level (DL), and Human factor Level (HL). The network level consists of all security related to the data flow in the network and the devices handling the network flow. The device level consists of the software part on the devices where the information system is running and the security related to it. Whereas the human factor level consists of the humans that are part of the information system and the threats they may induce either on purpose of due to ignorance.

To develop our solution, we relied on bibliographic work and on the deep knowledge we have in modeling, cyber security, artificial intelligence and machine learning.

Based on the bibliographic work performed, we deduced that there was a gap in the solutions. No work dealt with these three levels as a complete solution. Some researchers tackled the network part as a security measure, while others tackled the software part as a security measure and finally some addressed the human factor. We noticed that most of them are using machine learning to resolve and detect malicious behavior that may occur on the network level and the device level. Few provided a pure hardware or software solution. This thesis provided an end-to-end security solution that can be implemented and can also complement existing solutions in enterprises that can't afford high-end cyber security solutions.

For the network level, we presented a hardware solution that can effectively protect the data in restricted zones from being exfiltrated. We presented a DIY unidirectional network device *A.K.A. data diode* that allows the network traffic to go only in one direction. The physical approach is so effective and costs negligible money and can be easily made with off-the-shelf network appliances. The data diode can allow transfer of data from the DMZ zone to the restricted zone. In this thesis, we demonstrated how this network device has no effect on the speed of data transferred. Data of different sizes were moved from one zone to another using a data diode and without one, and the time latency was nearly zero. Of course, when talking about unidirectional network devices, we have to lighten up the

notion that only network protocols that work in unidirectional way can be used. TCP/IP can't be used because it needs to establish a handshake before beginning the transmission of data across. So, the use of UDP was the alternative in designing and implanting the solution. Because the UDP protocol allows the transmission of data without any handshake.

For the device level, we presented two solutions. First, a prioritization solution and second, an AI/ML solution. Regarding the first solution, the majority of CISOs have a problem in prioritization of vulnerabilities they have on their system. This problem comes from the fact that the CVSS version 2 or version 3 tend to give a score that can be sometimes misleading and other times equal in value to a lot of other vulnerabilities. Therefore, the CISO will have a very hard time patching and resolving the issues since they have similar priorities. To solve this, we provided a novelty approach to prioritize the vulnerabilities based on the aggregation of multiple sources and applying a weighted formula to generate a list that can be used to cross reference the vulnerabilities he has with it, and create a prioritized list that he can use as reference to patch and resolve them. As for the second solution, we used AI/ML to create a model that helped in detecting the anomalies in the data coming to the server. These anomalies can be cyber related or medical related to the health of the patients.

For the human factor level, we presented an AI driven chatbot as a solution. We utilized WhatsApp as a means of communication between the users and the chatbot server. The chatbot allows the users to search for policies and keywords related to their work environment and to cyber security. It allows them to take a quiz that helps them assess their general knowledge in security and in their specific domain of work. These results are stored to let them assess their progress and motivate them to further deepen their knowledge.

To implement and test these solutions, we chose to secure a medical system. The system is made of patients sending medical metrics using an attached raspberry pi zero with sensors connected to it. The data is sent to a server where it is inspected by a healthcare provider for medical anomalies. To secure this information system, we separated the system into two zones, DMZ zone and restricted zone. In the DMZ zone, the server will receive the data from the patient. The AI/ML solution will inspect the data and label it as normal, medical issue or cyber issue. Medical issue is when the vital signs of the patient are not normal while the cyber issue is when the data receives is not consistent with what the server should receive. After it get inspected, it is forwarded through the data diode to the internal server where the healthcare provider can see the labeled data and act upon it if necessary in case of a medical issue. And finally, the AI chatbot was used to help the patients and the healthcare providers gain more insight to cybersecurity and thus help in maintaining the whole information system as secure as possible.

This work is just a milestone in a future plan of ameliorations and modifications. The data diode can be ameliorated by replacing the copper cables by optic fiber. This can help in attaining higher bandwidth and thus allow more sophisticated data with big sizes to pass smoothly. And the fiber optic nature can help protect the data diode from any electromagnetic interferences that can occur if placed in a highly condensed server room. As for the software part of the data diode, it has a large potential of updates to accommodate new type of applications like video surveillance, emails, and other types of data replications.

As for the prioritization software, the plan is to integrate it with the open source WAZUH security platform to make it easily accessible for the CISO. Having an easy interface can help in minifying the tasks and help in delegating them to other personnel in the security team. Currently the software is using three sources to calculate the list of priority, so the future work will be to add more sources and ameliorate the formula accordingly. Regarding the AI/ML software, adding new medical vital metrics will imply recreating a new model, so the future work will focus on automating this whole phase and making it transparent to the integator of the solution.

The AI chatbot has a lot of potential regarding its improvement and utilization. For example, if an employee opens an attachment sent by mail from an unauthorized party, the IT team is informed immediately to be able to handle any fatal attack. Added to that, adding a voice enabled chatbot will also be interesting in order to help the employee stay focused on his current screen and asking the chatbot to send voice message whenever a security issue is encountered. And at last, making the bot scrape data from the most up-to-date security webpages and databases will make it inform autonomously the employees and the IT department about every new breach.

Bibliography

Parsons, K., Calic, D., Pattinson, M., Butavicius, M., McCormac, A., & Zwaans, T. (2017). The Human Aspects of Information Security Questionnaire (HAIS-Q): Two further validation studies. *Computers & Security*, *66*, 40-51.

Abawajy, J., & Kim, T. (2010). Performance Analysis of Cyber Security Awareness Delivery Methods. In *Security Technology, Disaster Recovery and Business Continuity* (pp. 142--148). Berlin: Springer.

Adams, C. (2018). Learning the lessons of WannaCry. *Computer Fraud & Security*(9), 6-9.

Agus Santoso, H., Anisa Sri Winarsih, N., Mulyanto, E., Wilujeng saraswati, G., Enggar Sukmana, S., Rustad, S., . . . Firdausillah, F. (2018). Dinus Intelligent Assistance (DINA) Chatbot for University Admission Services. *International Seminar on Application for Technology of Information and Communication.* Semarang.

Ajagekar, S., & Jadhav, V. (2018). Automated Approach for DDOS Attacks Detection Based on Naive Bayes Multinomial Classifier. *International Conference on Trends in Electronics and Informatics (ICOEI).* Tirunelveli.

Alam, S., Horspool, R., & Issa, T. (2013). MAIL: Malware Analysis Intermediate Language - A Step Towards Automating and Optimizing Malware Detection. *ACM 6th International Conference on Security of Information and Networks.* Aksaray.

Alam, S., Sogukpinar, I., Issa, T., & Horspool, R. (2015, 05). Sliding window and control flow weight for metamorphic malware detection. *Journal of Computer Virology and Hacking Techniques, 11*(2), 75-88.

Ali, F. A., & Yong, L. Y. (2011). Development of host based intrusion detection system for log files. *IEEE Symposium on Business, Engineering and Industrial Applications (ISBEIA),* pp. 281-285.

Allodi, L., & Massacci, F. (2012). A preliminary analysis of vulnerability scores for attacks in wild. *ACM Proc. of CCS BADGERS*, (pp. 17-24).

Alsughayyir, B., Qamar, A. M., & Khan, R. (2019). Developing a Network Attack Detection System Using Deep Learning. *International Conference on Computer and Information Sciences (ICCIS),* (pp. 1-5). Sakaka.

Austin, S. (2015, 06 30). *Tactical Data Diodes in Industrial Automation and Control Systems*. Retrieved from SANS Institute: https://www.sans.org/white-papers/36057/

Aycock, J. (2006). *Computer Viruses and Malware.* Boston: Springer.

Bernard, T., & Cowley, S. (2017, 10). *Equifax Breach Caused by Lone Employee's Error*. (The NewYork Times) Retrieved from https://www.nytimes.com/2017/10/03/business/equifax-congress-data-breach.html

Biryukov, A., Pustogarov, I., Thill, F., & Weinmann, R. (2014). Content and Popularity Analysis of Tor Hidden Services. *34th International Conference on Distributed Computing Systems Workshops (ICDCSW),* (pp. 188-193). Madrid.

Boyle, A. (2006, December). Retrieved from NBC NEWS: http://www.nbcnews.com/id/16095705/ns/technology_and_science-security/t/security-conscious-nasa-tightens-e-mail-policy/#.XXwBHCgzYgw

Burger, R. (1988). *Computer viruses : a high-tech disease.* United Kingdom: Abacus.

Celik, Z., Walls, R., McDaniel , P., & Swami, A. (2015). Malware traffic detection using tamper resistant features. *MILCOM 2015 - 2015 IEEE Military Communications Conference.* Tampa.

Chatzipoulidis, A., Michalopoulos, D., & Mavridis, I. (2015, Aug.). Information infrastructure risk prediction through platform vulnerability analysis. *Journal of Systems and Software, 106*, 28-41.

Chen, T., & Robert, J. (2004). The Evolution of Viruses and Worms.

CIRCL. (2022). *Computer Incident Response Center Luxembourg*. Retrieved from Computer Incident Response Center Luxembourg: https://www.circl.lu/

CIRCL. (2022). *Open Data at CIRCL*. Retrieved from Computer Incident Response Center Luxembourg: https://www.circl.lu/opendata/

CIS. (2020, 2). *Election Security Spotlight – Cyber Threat Actors*. Retrieved from Center for Internet Security: https://www.cisecurity.org/spotlight/cybersecurity-spotlight-cyber-threat-actors/

CISA. (2020, 2). *Cyber Threat Source Descriptions*. Retrieved from Cybersecurity and Infrastructure Security Agency: https://www.us-cert.gov/ics/content/cyber-threat-source-descriptions

CISA. (2022). *National Cyber Awareness System*. Retrieved from National Cyber Awareness System: https://www.cisa.gov/uscert

Corporation), B. B. (2017). Retrieved from Cyber-attack: Europol says it was unprecedented in scale: http://www.bbc.com/news/world-europe-39907965.

Corporation, T. M. (2021, 07). *MITRE ATT&CK*. Retrieved from https://attack.mitre.org/

DELLINGER, A. (2017, May). Retrieved from International Business Times: https://www.ibtimes.com/edmodo-hacked-77-million-accounts-students-teachers-parents-stolen-education-social-2540073

Den Boer, B., & Bosselaers, A. (1993). Collisions for the compression function of MD5, Advances in Cryptology. *Eurocrypt.* Norway.

Denning, D. (1987, February). An Intrusion-Detection Model. *IEEE Transactions on Software Engineering, 13*(2), 222-232.

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language. *CoRR, abs/1810.04805*.

Dhingra, M., Jain, M., & Jadon, R. (2016). Role of artificial intelligence in enterprise information security: A review. *Fourth International Conference on Parallel, Distributed and Grid Computing (PDGC).*

Diagrams.net. (2022). *Security-first diagramming for teams*. Retrieved from Draw.io: https://www.draw.io/

Disparte, D., & Furlow, C. (2016, 16). *The Best Cybersecurity Investment You Can Make Is Better Training*. (Harvard Business Review) Retrieved 08 01, 2021, from https://hbr.org/2017/05/the-best-cybersecurity-investment-you-can-make-is-better-training

Dobrovoljc, A., Trček, D., & Likar, B. (2017). Predicting Exploitations of Information Systems Vulnerabilities Through Attackers' Characteristics. *IEEE Access, 5*, 26063-26075.

Economist, T. (2017, 05). Retrieved from The Economist: https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data

Egnash, M. (2018, 03). *Defense Travel System Data Breach Leaves Thousands Open to ID Theft*. Retrieved from https://www.military.com/daily-news/2018/03/01/defense-travel-system-data-breach-leaves-thousands-open-id-theft.html

Eigner, O., Kreimel, P., & Tavolato, P. (2016). Detection of Man-in-the-Middle Attacks on Industrial Control Networks. *International Conference on Software Security and Assurance (ICSSA)*, (pp. 64-69). St. Polten.

EL HAJAL, G., ABI ZEID DAOU, R., & DUCQ, Y. (2021). A novel approach to classify vulnerabilities based on authenticated measurements. *iCatse International Conference on IT Convergence and Security.*

El Hajal, G., Abi Zeid Daou, R., Ducq, Y., & Börcsök, J. (2019). Designing and validating a cost effective safe network: application to a PACS system. *Fifth International Conference on Advances in Biomedical Engineering (ICABME).* Tripoli, Lebanon.

EL HAJAL, G., ABI ZEID DAOU, R., DUCQ, Y., & Börcsök, J. (2019). Designing and validating a cost effective safe network: application to a PACS system. *5th International Conference on Advances in Biomedical Engineering (ICABME).* Tripoli, Lebanon.

ETTERCAP. (2021). Retrieved from https://www.ettercap-project.org/

Facebook. (2022). *Capacity, Quality Rating, and Messaging Limits*. Retrieved 08 2021, from https://developers.facebook.com/docs/whatsapp/api/rate-limits/

FactMonster. (2001). *Families of Musical Instruments*. (KidsSAFE Seal Program) Retrieved 10 28, 2018, from www.factmonster.com

Fang, M., & Hafiz, M. (2014). Discovering Buffer Overflow Vulnerabilities in the Wild: An Empirical Study. *Empirical Software Engineering and Measurement (ESEM 2014)*, (pp. 1-10). Torino.

Farrokhmanesh, M., & Hamzeh, A. (2018, 06). Music classification as a new approach for malware detection. *Journal of Computer Virology and Hacking Techniques, 15*(2), 77–96.

FIRST. (2022). *Common Vulnerability Scoring System SIG*. Retrieved from Forum of Incident Response and Security Teams: https://www.first.org/cvss/

Foreseeti. (2022). *SecuriCAD*. Retrieved from Foreseeti: https://www.foreseeti.com/

Foundation, P. S. (n.d.). *Python*. Retrieved 08 2021, from https://www.python.org/

Fridell, K., Aspelin, P., Edgren, L., Lindsköld, L., & Lundberg, N. (2009). PACS influence the radiographer's work. *Radiography, 15*(2), 121-133.

Galov, N. (2019, 3). Retrieved from Cyber Defense Magazine: https://www.cyberdefensemagazine.com/cyber-security-statistics-for-2019/

Garg, H., & Dave, M. (2019). Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware. *4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU).*

GDPR. (2022). *GENERAL DATA PROTECTION REGULATION (GDPR)*. Retrieved from GENERAL DATA PROTECTION REGULATION (GDPR): https://gdpr.eu/

Geng, G., Yan, Z., Zeng , Y., & Jin, X. (2018). RRPhish: Anti-phishing via mining brand resources request. *IEEE International Conference on Consumer Electronics (ICCE).* Las Vegas.

Gharacheh, M., Derhami, V., Hashemi, S., & Fard, S. M. (2015). Proposing an HMM-based approach to detect metamorphic malware. *4th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS).* Zahedan.

Gittens, M., Kim, Y., & Godwin, D. (2005). The vital few versus the trivial many: examining the Pareto principle for software. *29th Annual International Computer Software and Applications Conference (COMPSAC'05)*, (pp. 179-185). Edinburgh.

Goddijn, I. (2020, 2). *Data Breach QuickView Report 2019 Q3*. Richmond: Risk Based Security. Retrieved from Risk Based Security: https://www.riskbasedsecurity.com/2019/11/12/number-of-records-exposed-up-112/

Granjal, J., Monteiro, E., & Sá Silva, J. (2015). Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues. *IEEE Communications Surveys Tutorials, 17*(3), 1294-1312.

GREENBERG, A. (2018, June). Retrieved from WIRED: https://www.wired.com/story/exactis-database-leak-340-million-records/

Heller, B., Proctor, M., Dean Mah, D., Jewell, L., & Cheung, B. (2005). Freudbot: An Investigation of Chatbot Technology in Distance Education. *EdMedia: World Conference on Educational Media & Technology.* Montreal.

Henriksen, M. (2018). *Draw.io for threat modeling*. Retrieved from michenriksen.com: https://michenriksen.com/blog/drawio-for-threat-modeling/

HERN, A., & GIBBS, S. (2017, May). Retrieved from The Guardian: https://www.theguardian.com/technology/2017/may/12/nhs-ransomware-cyber-attack-what-is-wanacrypt0r-20

Holm, H., & Khan, K. (2015, Sep.). An expert-based investigation of the common vulnerability scoring system. *Computers & Security, 53*, 18-30.

Houmb, S. H., Franqueira, V. N., & Engum, E. A. (2010). Quantifying security risk level from CVSS estimates of frequency and impact. *Journal of Systems and Software, 83*(9), 1622-1634.

hping3. (2021). Retrieved from https://tools.kali.org/information-gathering/hping3

Huang, S., & Huang, Y. (2013). Network traffic anomaly detection based on growing hierarchical SOM. *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN).* Budapest.

Iriusrisk. (2022). *threat-modeling-platform*. Retrieved from Iriusrisk: https://iriusrisk.com/threat-modeling-tool

ISO/IEC. (2022). *ISO/IEC 27001 — Information security management*. (International Organization for Standardization) Retrieved 2022, from International Organization for Standardization: https://www.iso.org/isoiec-27001-information-security.html

Jang-Jaccard, j., & Nepal, S. (2014). A survey of emerging threats in cybersecurity. *Journal of Computer and System Sciences, 80*(5), 973-993.

Jeon, B., & Na, J. (2016). A study of cyber security policy in industrial control system using data diodes. *International Conference on Advanced Communication Technology (ICACT).* Pyeongchang.

Jiang, J., Ding, L., Zhai, E., & Yu, T. (2012). VRank: A Context-Aware Approach to Vulnerability Scoring and Ranking in SOA. *IEEE Sixth International Conference on Software Security and Reliability*, (pp. 61-70). Gaithersburg.

Justice, U. D. (2009, July). Retrieved from U.S Department of Justice: https://www.justice.gov/opa/pr/former-boeing-engineer-convicted-economic-espionage-theft-space-shuttle-secrets-china

Kaspersky. (2020, 2). Retrieved from Kaspersky : https://www.kaspersky.com/resource-center/definitions/what-is-cyber-security

Kaufmann, M. (2003). Chapter 15 - Data Flow Diagrams. In *Design Methods for Reactive Systems* (pp. 185-200). R.J. Wieringa.

Kenneth, I., & Forrest, S. (2002, January). A history and survey of network firewalls. *ACM Journal Name*, 1-42.

Keramati, M., & Keramati, M. (2014). Novel security metrics for ranking vulnerabilities in computer networks. *7th International Symposium on Telecommunications (IST),* (pp. 883-888). Tehran.

Khanna, A., Pandey, B., Vashishta, K., Kalia, K., Bhale, P., & Das, T. (2015). A study of today's A.I. through chatbots and rediscovery of machine intelligence. *International Journal of U- and e-Service, Science and Technology, 8*, 277-284.

Khatri, V., & Abendroth, J. (2015). Mobile Guard Demo: Network Based Malware Detection. *IEEE Trustcom/BigDataSE/ISPA.* Helsinki.

Klahr, R., Shah, J., Sheriffs, P., Rossington, T., Pestell, G., Button, M., & Victoria Wang, V. (2017). *Cyber security breaches survey 2017.* Ipsos MORI.

Kubovič, O. (2018). *One year later: EternalBlue exploit more popular now than during WannaCryptor outbreak.* Eset.

Kumar, R., Xiaosong, Z., Khan, R., Ahad, I., & Kumar, J. (2018). Malicious Code Detection based on Image Processing Using Deep Learning. *International Conference on Computing and Artificial Intelligence.* Chengdu.

Lee, K.-M., Teng, W.-G., Wu, J.-N., Huang, K.-M., Ko, Y.-H., & Hou, T.-W. (2014). Multicast and customized deployment of large-scale operating systems. *Autom. Softw. Eng., 21,* 443–460.

Li, J., & Wang, S. (2017). PhishBox: An Approach for Phishing Validation and Detection. *IEEE 15th Intl Conf on Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence and Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech).* Orlando.

Liu, B., Li, X., Liu, Z., Yuan, Q., & Yin, X. (2008). *Design and implementation of information exchange between HIS and PACS based on HL7 standard.* Shenzhen, China: International Conference on Information Technology and Applications in Biomedicine.

López, D., Pastor, O., & Villalba, L. G. (2013). Dynamic risk assessment in information systems: State-of-the-art. *Proc. 6th Int. Conf. Inf. Technol.*, (pp. 8-10).

M., S. M. (1995). Data Diodes.

Maghrabi, L., Pfluegel, E., Al-Fagih, L., Graf, R., Settanni, G., & Skopik, F. (2017). Improved software vulnerability patching techniques using CVSS and game theory. *International Conference on Cyber Security And Protection Of Digital Services*, (pp. 1-6). London.

Mahlaola, T., & van Dyk, B. (2016). Reasons for Picture Archiving and Communication System (PACS) data security breaches: Intentional versus non-intentional breaches. *Health SA Gesondheid, 21*, 271-279.

Mansfield-Devine, S. (2018). Extreme prejudice: securing networks by treating all data as a threat. *Computer Fraud & Security, 2018*(6), 16-20.

Mansoori, B., Erhard, K., & Sunshine, J. (2012). Picture Archiving and Communication System (PACS) Implementation, Integration & Benefits in an Integrated Health System. *Academic Radiology, 19*(2), 229-235.

Martin, L. (2021, 07). *the Cyber Kill Chain*. Retrieved from https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

Mathur, K., & Hiranwal, S. (2013). A Survey on Techniques in Detection and Analyzing Malware Executables. *International Journal of Advanced Research in Computer Science and Software Engineering*, 422-428.

Menzel, C., & Mayer, R. (1998). The IDEF Family of Languages. In P. Bernus, K. Mertins, & G. Schmidt, *Handbook on Architectures of Information Systems. International Handbooks on Information Systems.* (pp. 209-241). Berlin: Springer.

Microsoft. (2003). Microsoft Improving Web Application Security Threats and Countermeasures. *Microsoft patterns & practices*.

Microsoft. (2016). Retrieved from Microsoft: https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling

Microsoft Threat Modeling Tool threats. (2022, 03 01). (Microsoft) Retrieved 06 16, 2022, from https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats

MITRE. (2022). *Common Attack Pattern Enumeration and Classification*. Retrieved from Common Attack Pattern Enumeration and Classification (CAPEC): https://capec.mitre.org/

MITRE. (2022). *MITRE*. Retrieved from MITRE: https://www.mitre.org/

Mogul, J., Rashid, R., & Accetta, M. (1987). The Packet Filter An Efficient Mechanism for User-Level Network Code. *11th Symposium on Operating Systems Principles, ACM SIGOPS.* Austin.

Molnár, G., & Szüts, Z. (2018). The Role of Chatbots in Formal Education. *IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY).* Subotica.

Mozilla. (2014). Retrieved from https://github.com/mozilla/seasponge

Nayak, K., Marino, D., Efstathopoulos, P., & Dumitraş, T. (2014). Some Vulnerabilities Are Different Than Others. In *Research in Attacks, Intrusions and Defenses* (pp. 426--446). Cham, Switzerland: Springer International Publishing.

NCSC. (2020, 2). *Understanding vulnerabilities*. Retrieved from National Cyber Security Center: https://www.ncsc.gov.uk/information/understanding-vulnerabilities

NEWS, F. (2007, September). Retrieved from FOX NEWS: https://www.foxnews.com/story/pentagon-source-says-china-hacked-defense-department-computers

NIST. (2018). *Risk Management Framework for Information Systems and Organizations.* Gaithersburg: National Institute of Standards and Technology. Retrieved from National Institute of Standards and Technology: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf

NIST. (2022). *CYBERSECURITY FRAMEWORK*. Retrieved 2022, from National Institute of Standards and Technology: https://www.nist.gov/cyberframework

NIST. (2022). *data-feeds*. Retrieved from NATIONAL VULNERABILITY DATABASE: https://nvd.nist.gov/vuln/data-feeds

NIST. (2022). *NATIONAL VULNERABILITY DATABASE*. Retrieved from National Institute of Standards and Technology: https://nvd.nist.gov/vuln-metrics/cvss

Nuruzzaman, M., & Hussain, O. K. (2018). A Survey on Chatbot Implementation in Customer Service Industry through Deep Neural Networks. *IEEE 15th International Conference on e-Business Engineering (ICEBE).* Xi'an.

O'Brien, S. (2017, September). Retrieved from CNN: https://money.cnn.com/2017/09/07/technology/business/equifax-data-breach/index.html

Offensive_Security. (2022). Retrieved from The Exploit Database Git Repository: https://github.com/offensive-security/exploitdb

Offensive_Security. (2022). *Exploit Database*. Retrieved from Exploit Database: https://www.exploit-db.com/

Oh, G., Lee, Y., Jung, M., & Yeom, S. (2008). Design of a Robust Watermarking Algorithm against the Geometric Distortion for Medical Image Security. *Second International Conference on Future Generation Communication and Networking Symposia.* Sanya, China.

Oprea, A., Li, Z., Norris, R., & Bowers, K. (2018). MADE: Security Analytics for Enterprise Threat Detection. *the 34th Annual Computer Security Applications Conference.* San Juan.

Othmane, L. B., Ranchal, R., Fernando, R., Bhargava, B., & Bodden, E. (2015). Incorporating attacker capabilities in risk estimation and mitigation. *Computers & Security, 51,* 41 - 61.

OWASP. (2020). *OWASP API Security Project*. Retrieved 07 2021, from Open Web Application Security Project: https://owasp.org/www-project-api-security/

OWASP. (2022). *OWASP Threat Dragon*. Retrieved from Open Web Application Security Project: https://owasp.org/www-project-threat-dragon/

Papadimitratos , P., & Haas, Z. (2006). Secure data communication in mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications , 24*(2), 343 - 356.

Pawar, M., & Anuradha, J. (2015). Network Security and Types of Attacks in Network. *Procedia Computer Science, 48,* 503-506.

PCI_DSS. (2022). *Payment Card Industry Data Security Standard*. Retrieved from Payment Card Industry Security Standards: https://www.pcisecuritystandards.org/

Peck, A. (2018). *Essential PACS, RIS and Imaging Informatics.* New York: CRCC Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Duchesnay, E. (2011). Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research, 12*, 2825--2830. Retrieved from https://scikit-learn.org/stable/

PERLROTH, N. (2017, March). Retrieved from The New York Times: https://www.nytimes.com/2017/10/03/technology/yahoo-hack-3-billion-users.html

Perlroth, N., Tsang, A., & Satariano, A. (2018, November). Retrieved from The New York Times: https://www.nytimes.com/2018/11/30/business/marriott-data-breach.html

Pfeffer, A. (2009, June). Retrieved from HAARETZ: https://www.haaretz.com/1.5065382

Ponemon Institute LLC. (2017, September). Retrieved from CSR Privacy Solution's Inc (CSR): https://csrps.com/wp-content/uploads/2019/03/2017-Ponemon-State-of-Cybersecurity-in-Small-and-Medium-Sized-Businesses-SMB.pdf

Quinn, B., & Arthur, C. (2011, April). Retrieved from The Guardian: https://www.theguardian.com/technology/2011/apr/26/playstation-network-hackers-data

Raywood, D. (2020, FEB). Retrieved from InfoSecurity: https://www.infosecurity-magazine.com/magazine-features/top-worst-vulnerabilities/

Riikkinen, M., Saarijärvi, H., Sarlin, P., & Lähteenmäki, I. (2018). Using artificial intelligence to create value in insurance. *International Journal of Bank Marketing, 36*(6), 1145-1168.

Rosruen, N., & Samanchuen, T. (2018). Chatbot Utilization for Medical Consultant System. *3rd Technology Innovation Management and Engineering Science International Conference (TIMES-iCON).* Bangkok.

ROSS, R., McEVILLEY, M., & OREN, J. (2016). *Systems Security Engineering.* Washington: National Institute of Standards and Technology. Retrieved from National Institute of Standards and Technology: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160.pdf

Rostrom, T., & Teng, C. (2011). Secure communications for PACS in a cloud environment. *International Conference of the IEEE Engineering in Medicine and Biology Society.* Boston, MA, USA.

Roy, S. S., Krishna, P. V., & Yenduri, S. (2014). Analyzing Intrusion Detection System: An ensemble based stacking approach. *IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, (pp. 307-309). Noida.

Rupprecht, D., Dabrowski, A., Holz , T., Weippl, E., & Pöpper, C. (2018). On Security Research Towards Future Mobile Network Generations. *IEEE Communications Surveys & Tutorials, 20*(3), 2518 - 2542.

Sahu, K., & Shrivastava, S. (2015, February). Kernel k-Means Clustering for Phishing Website and Malware Categorization. *International Journal of Computer Applications, 111*(9).

Saxe, J., & Berlin, K. (2015). Deep neural network based malware detection using two dimensional binary program features. *International Conference on Malicious and Unwanted Software (MALWARE).* Fajardo.

Schaad, A., & Reski, T. (2019). Open Weakness and Vulnerability Modeler" (OVVL): An Updated Approach to Threat Modeling. *16th International Joint Conference on e-Business and Telecommunications - SECRYPT.* Prague.

Schneier, B. (2015). *Secrets and Lies: Digital Security in a Networked World.* Wiley.

Schulze, H. (2019). *Insider Threat Report: Trends and Analysis.* Retrieved from https://www.fortinet.com/content/dam/fortinet/assets/threat-reports/insider-threat-report.pdf

Securitycompass. (2022). *SD Elements*. Retrieved from Securitycompass: https://www.securitycompass.com/

Sfakianakis, A., Douligeris, C., Marinos, L., Lourenço, M., & Raghimi, O. (2019, january 28). *ENISA Threat Landscape Report 2018.* Retrieved from https://www.enisa.europa.eu/publications/enisa-threat-landscape-report-2018

Shakarian, P. (2011, April). Stuxnet: Cyberwar Revolution in Military Affairs. 23. *Small Wars Journal., 23*.

SHANKER, T., & BUMILLER, E. (2011, July). Retrieved from The New York Times: https://www.nytimes.com/2011/07/15/world/15cyber.html

SILVER-GREENBERG, J., GOLDSTEIN, M., & PERLROTH, N. (2014, October). Retrieved from The New York Times: https://dealbook.nytimes.com/2014/10/02/jpmorgan-discovers-further-cyber-security-issues/

Sobhi Abou Chahine, A. A. (2008). *Quality Assurance for Higher Education in Lebanon, Guide II: Self Evaluation In Higher Education Institutions.* Beirut: Tempus Project.

Srinivas, J., Das, A., & Kumar, N. (2019). Government regulations in cyber security: Framework, standards and recommendations. *Future Generation Computer Systems, 92*, 178-188.

Tarandach, I. (2022). *pytm: A Pythonic framework for threat modeling*. Retrieved from https://github.com/izar/pytm

Taunk, K., De, S., Verma, S., & Swetapadma, A. (2019). A Brief Review of Nearest Neighbor Algorithm for Learning and Classification. *International Conference on Intelligent Computing and Control Systems (ICCS).* Madurai, India.

Threatmodeler. (2022). *Threatmodeler*. Retrieved from Threatmodeler: https://threatmodeler.com/

Tutamantic. (2022). *Tutamen Threat Model Automator*. Retrieved from Tutamantic: http://www.tutamantic.com/

Wang, P., Chao, K., Lo, C., & Wang, Y. (2017, March). Using ontologies to perform threat analysis and develop defensive strategies for mobile security. *Information Technology and Management*, 1–25.

Wang, W., Shi, F., Zhang, M., Xu, C., & Zheng, J. (2020). A Vulnerability Risk Assessment Method Based on Heterogeneous Information Network. *IEEE Access, 8*, 148315-148330.

WEISE, E. (2016, November). Retrieved from USA TODAY: https://www.usatoday.com/story/tech/news/2016/11/14/412-million-adult-meeting-accounts-possibly-hacked/93818404/

WhatsApp. (2022). *Whatsapp LCC*. Retrieved 08 2021, from https://www.whatsapp.com/?lang=en

WhatsApp. (2022). *WhatsApp Web*. Retrieved 08 2021, from https://web.whatsapp.com/

Whittaker, Z. (2018, 09). *Veeam server lapse leaks over 440 million email addresses*. Retrieved from https://techcrunch.com/2018/09/11/veeam-security-lapse-leaked-over-440-million-email-addresses/

Wreski, D. (2012, 12). Retrieved from Linux Security: https://linuxsecurity.com/news/intrusion-detection/ddos-attacks-against-us-banks-peaked-at-60-gbps

Yu, H., Li, J., Zhang, L., & Yu, Z. (2012). *A three-dimensional visualization PACS display system integrated with HIS.* Chongqing, China: 5th International Conference on BioMedical Engineering and Informatics.

Zhang, J., Yu, F., Sun, J., Yang, Y., & Liang, C. (2007). DICOM Image Secure Communications With Internet Protocols IPv6 and IPv4. *IEEE Transactions on Information Technology in Biomedicine , 11*(1), 70-80.

Zhang, T., Lee, W., Gao, M., & Zhou, J. (2019, June). File Guard: automatic format-based media file sanitization. *International Journal of Information Security*, 1-13.

Zhou, Z., Liu, B., & Le, A. (2007). CAD–PACS integration tool kit based on DICOM secondary capture, structured report and IHE workflow profiles. *Computerized Medical Imaging and Graphics, 31*(4-5), 346-352.

Zou, Q., Singhal, A., Sun, X., & Liu, P. (2021). Deep Learning for Detecting Network Attacks: An End-to-End Approach. In *Data and Applications Security and Privacy XXXV* (pp. 221--234). Springer International Publishing.