Classification des systèmes IoT vulnérables basée sur les CVE

Grzegorz J. Blinowski [1][0000-0002-0869-2828] et Paweł Piotrowski[2]

[1,2] Institut d'informatique, Université de technologie de Varsovie, Nowowiejska 15/19, 00-665 Warszawa, Pologne
g.blinowski@ii.pw.edu.pl

Résumé. La base de données Common Vulnerabilities and Exposures (CVE) est l'une des plus grandes sources publiques de données et de rapports sur les vulnérabilités logicielles et matérielles. Dans ce travail, nous analysons la base de données CVE dans le contexte des vulnérabilités des dispositifs et systèmes IoT. Nous introduisons une classification des systèmes IoT basée sur le monde réel. Ensuite, nous employons un algorithme SVM sur un sous-ensemble sélectionné de la base de données CVE pour classer les "nouveaux" enregistrements de vulnérabilité dans ce cadre. Le sous-ensemble qui nous intéresse est constitué d'enregistrements qui décrivent les vulnérabilités de dispositifs IdO potentiels de différentes applications, telles que : la maison, l'industrie, les contrôleurs mobiles, les réseaux, etc. L'objectif de la classification est de développer et de tester un système automatique de reconnaissance des dispositifs IoT vulnérables et de tester l'exhaustivité, la suffisance et la fiabilité des données CVE à cet égard.

Mots-clés : Internet des objets, sécurité de l'IdO, classification de la vulnérabilité des systèmes, CVE, NVD, SVM.

1 Introduction et contexte

1.1 Architecture IoT - Grandes lignes

L'IoT peut être le plus largement défini comme une interconnexion de divers objets adressables de manière unique par le biais de protocoles de communication. En réduisant ce qui précède, nous pouvons le décrire comme un paradigme de système de communication dans lequel les objets de la vie quotidienne, équipés de microcontrôleurs, d'émetteurs réseau et de piles de protocoles adaptées qui leur permettent de communiquer entre eux et, via une infrastructure cloud omniprésente et également avec les utilisateurs, deviennent une partie intégrante de l'environnement Internet [1].

Ici, nous allons considérer un modèle IoT composé de trois niveaux majeurs :

- **La couche de perception et d'exécution**, qui englobe un large éventail de dispositifs "intelligents" allant des étiquettes RFID et NFC, des capteurs et actionneurs environnementaux, de divers appareils domestiques, de terminaux mobiles, de téléphones intelligents, etc. Cela inclut également un large éventail de dispositifs SCADA industriels et de composants de véhicules intelligents, omniprésents dans les véhicules d'aujourd'hui (voitures, camions, etc.). Un rôle distinct doit être attribué aux contrôleurs ou aux stations de gestion utilisés à la fois dans les applications domestiques et industrielles. Un contrôleur est simplement un PC, une tablette ou un téléphone mobile utilisé pour gérer l'infrastructure IoT locale et pour visualiser les données.

- **Couche réseau** qui fournit une infrastructure de communication hétérogène basée sur de multiples normes de réseau telles que : s Wi-Fi, 3G/LTE, Z-wave, ZigBee, 6LoWPAN, VLC, mIP [2] et Ethernet, ainsi que la suite de protocoles Internet standard (IPv4/IPv6 et pile de la couche transport UDP/TCP).
- **Couche cloud ou applicative** qui intègre, gère et analyse les données provenant de Dispositifs IoT. Le nuage ne se contente pas de rassembler les données et de gérer les couches "objets" et "cœur", mais agit comme un fournisseur de services omniprésent pour les utilisateurs finaux, selon le paradigme de l'approche orientée services (SOA).

- Le modèle IoT ci-dessus est compatible avec le modèle d'architecture de référence proposé par le projet IoT-A du 7e PC de l'UE [3] et avec l'arborescence IoT-A [4].

1.2 Applications IoT

L'IoT est largement, bien que principalement anecdotique, connu comme un réseau d'appli- cations domestiques - des équipements et périphériques de PC aux réfrigérateurs, machines à café, etc. Cependant, le champ d'application des déploiements de l'IdO est beaucoup plus large et couvre les domaines suivants [1,5,6,7] :

- Villes intelligentes - surveillance de la santé structurelle des bâtiments, cartographie du bruit, surveillance de la gestion du trafic et "routes intelligentes" ; éclairage intelligent ; gestion des déchets.
- Environnement intelligent - surveillance météorologique ; systèmes d'alerte précoce en cas de catastrophe, par exemple.
  détection des inondations et surveillance des volcans) ; surveillance de la qualité de l'eau ; détection des fuites chimiques et du niveau de pollution.
- Agriculture et élevage intelligents - surveillance des engrais, des pesticides et de l'irrigation, surveillance du niveau des cultures ; surveillance et contrôle des plantes hydroponiques ; suivi des animaux.
- Smart Grid - surveillance et gestion de la consommation d'énergie électrique.
- Fabrication - cela couvre un large éventail de systèmes de contrôle des processus industriels - mécaniques, chimiques, etc. Cette gamme d'applications est souvent désignée par le terme IToT (Industrial IoT) ; les systèmes eux-mêmes sont désignés par le terme SCAD (Supervisory Control and Data Acquisition).
- Sécurité industrielle et détection - détection des niveaux de gaz et des fuites dans les environnements industriels, mesure du niveau de radiation.
- eHealth - surveillance et assistance des patients dans les établissements médicaux.
- Domotique ("maisons intelligentes") - surveillance de la consommation d'énergie et d'eau, appareils électroménagers contrôlés à distance, caméras de porte, serrures, alarmes, etc.

1.3 Problèmes de sécurité des systèmes IoT

Les problèmes de sécurité des environnements IoT ont été largement discutés et rendus publics. Dans certains cas, lorsque le système compromis était largement utilisé, par exemple comme un appareil ménager de type com- mune, ou lorsque les effets de l'exploitation de la sécurité étaient largement visibles (par exemple dans le cas du botnet Mirai [8]), la prise de conscience de l'insécurité de l'IdO a même atteint le grand public.

Nous pouvons distinguer deux types généraux de menaces liées à l'IdO : 1. les **menaces contre l'IdO** et 2. les **menaces provenant de l'IdO**. 1 Les menaces contre l'IdO se produisent lorsqu'une faille dans un appareil ou une application IdO, au niveau de la perception, du réseau ou du nuage, est exploitée par le pirate, et que l'appareil ou l'application est endommagé.

dispositif ou l'application est compromis - c'est-à-dire qu'un attaquant obtient un accès complet ou limité à ses fonctions et à ses données. 2 Dans le cas des menaces liées à l'IdO, l'infrastructure compromise est utilisée pour mener diverses attaques contre l'IdO ou les dispositifs connectés à Internet. Là encore, le botnet Mirai peut servir d'exemple - lorsqu'une multitude de webcams et d'autres appareils compromis ont été utilisés pour mener une attaque DDoS massive.

Dans [9], les auteurs ont proposé cinq " dimensions " relatives à la sécurité de l'IdO : le matériel, le système d'exploitation/firmware, le logiciel, le réseau et les données :

- La **sécurité matérielle** est critique lorsqu'un attaquant peut accéder physiquement à l'appareil. Grâce aux portes dérobées matérielles, le contrôle d'intégrité au niveau logiciel peut être contourné en désactivant la fonctionnalité de contrôle ou en démarrant via un firmware falsifié. Presque tous les dispositifs IoT présentent des vulnérabilités matérielles qui peuvent être exploitées (voir une base de données de référence - [10]). Les mécanismes de la liste des vulnérabilités comprennent, sans s'y limiter : les ports de débogage, les options de démarrage multiples et la mémoire flash non chiffrée [11]-[13]. Les microcontrôleurs (MCU) qui sont largement utilisés dans les applications industrielles (SCADA) ainsi que dans les automobiles et la domotique sont également sujets à des vulnérabilités au niveau matériel. Il s'agit par exemple d'attaques contre le contenu d'EEPROM via les ports JTAG/SPI, d'attaques par défaut d'horloge qui ont permis de compromettre le cryptage AES via des techniques d'injection de fautes [14].
- **Système d'exploitation, firmware et sécurité et confidentialité des logiciels** - concerne les trois couches de l'IdO : perception, réseau et cloud. Les problèmes de sécurité logicielle sont sim- laires à ceux des systèmes informatiques traditionnels. Des systèmes d'exploitation dignes de confiance devraient être utilisés au niveau de la couche perception pour réduire le risque de compromission à distance. Cependant, dans la pratique, c'est rarement le cas. L'application du contrôleur est souvent installée sur un PC ou un smartphone et des mesures de sécurité logicielles doivent être appliquées afin d'empêcher toute attaque à son encontre. La sécurité de la couche cloud ne peut pas non plus faire l'objet d'une confiance aveugle, par exemple : les serveurs installés sur Amazon EC2 sont sécurisés du point de vue du fournisseur de cloud, mais pas du point de vue de l'application installée et doivent être sécurisés par celui qui déploie les serveurs. D'autres risques de sécurité au niveau des logiciels, spécifiques aux environnements IoT, ont été décrits récemment : des paires de clés SSL publiques et privées découvertes par analyse statique sur un grand nombre de firmwares non emballés ; une analyse dynamique automatisée à grande échelle utilisant le cadre Metasploit de divers firmwares a été menée et un grand nombre d'exploits potentiels ont été découverts [15] ; un exploit de buffer overflow (qui peut être utilisé pour exécuter n'importe quel code sur le dispositif) a été découvert en analysant le protocole d'administration du réseau domestique (HNAP) [16] ; un buffer overflow basé sur la pile de la bibliothèque générale glibc [17] a été exploité pour attaquer plusieurs home hubs [18].
- **Sécurité et confidentialité du réseau** : en tant que système en réseau, l'ensemble de l'environnement IoT doit être sécurisé de bout en bout. Le cryptage et l'authentification devraient être utilisés de manière cohérente, mais ce n'est souvent pas le cas. Deux fonctions spécifiques aux dispositifs IoT domestiques sont le couplage et la liaison. Dans le processus de couplage, le contrôleur doit se connecter au dispositif IoT afin de configurer la " chose ". La plupart des dispositifs SOHO IoT permettent à tout contrôleur à proximité de procéder à l'appariement sans sécurité supplémentaire.

mesures. Cela peut être acceptable dans un environnement domestique, mais dans un dé- ploiement à grande échelle dans un environnement public, toute personne ayant accès aux appareils peut reconfigure et s'introduire dans le système. Le processus de liaison commence après la réussite de l'appairage et établit les informations d'identification d'accès nécessaires pour la chose afin de la contrôler. Les mots de passe hebdomadaires sont l'un des problèmes de sécurité typiques de cette phase. De nombreux fabricants ne fournissent pas la protection nécessaire à leurs appareils IoT en réseau, l'attaque Mirai citée plus haut n'étant qu'un des exemples d'exploitation de mots de passe faibles. De nombreuses attaques relatives à la liaison ont été analysées et décrites, par exemple pour : les dispositifs portables [19], les systèmes de caméras de surveillance [20], le système d'ampoules Phillips Hue [21], etc.

- **Le nuage et les données** - le nuage recueille les données de la couche de perception et est responsable du maintien d'une sécurité adéquate des données. Le nuage gère souvent l'authentification et les services associés et est un pair dans le chiffrement de bout en bout des données transmises. Les applications compromises au niveau du cloud exposent une quantité importante, voire la totalité, des données collectées. Cependant, le big data collecté par le cloud peut également contribuer à renforcer la sécurité. Par exemple, elles peuvent être utilisées pour distinguer les schémas d'utilisation légitimes des schémas illégaux et pour prévenir (du moins dans une certaine mesure) les attaques DDoS.

Pour résumer cette section : la majorité des problèmes de sécurité qui apparaissent dans les systèmes IoT actuels résultent directement d'implémentations logicielles et matérielles boguées, incomplètes ou obsolètes. Une erreur de conception majeure d'un protocole (comme Heartbleed [22] et DROWN [23]) est beaucoup plus rare. Comme on peut facilement le vérifier dans les bases de données de vulnérabilités du domaine public, le nombre de produits signalés avec de graves failles de sécurité augmente d'année en année.

1.4 Portée de ce travail et recherche connexe

Dans ce travail, nous proposons une classification des données de vulnérabilité liées aux appareils (c'est-à-dire qui ne sont pas des " logiciels purs ") pour les équipements IoT et IIoT. Nous avons divisé les enregistrements CVE d'une base de données publique en 7 catégories distinctes (par exemple : équipements domestiques, dispositifs SCADA, systèmes d'infrastructure réseau, etc.) Les échantillons de la base de données ont été classés à la main par nous sur la base des connaissances des experts. Nous avons ensuite utilisé un classificateur SVM (Support vector machine) sur les données relatives aux appareils et aux vulnérabilités pour prédire les catégories de "nouvelles" vulnérabilités - par exemple, les données de l'année 2017 ont été utilisées pour classer les données de 2018, etc. Le but était de prédire, et (si possible) de prévenir et d'atténuer les menaces résultant des nouvelles vulnérabilités. Il s'agit d'une tâche difficile compte tenu de la taille de la base de données et de son rythme de croissance - chaque jour, des dizaines de nouveaux enregistrements sont ajoutés à la seule base de données CVE. Par conséquent, lorsqu'une nouvelle vulnérabilité ou un nouvel exploit est découvert, il est souvent essentiel d'en connaître la portée par des moyens automatiques, aussi rapidement que possible.

Il y a eu quelques recherches antérieures sur l'analyse et la classification automatiques des bases de données de vulnérabilité : Dans [24,25], des modèles et des méthodologies de catégorisation des vulnérabilités de la base de données CVE en fonction de leurs types de sécurité basés sur des réseaux bayésiens. Dans [26], des modèles thématiques ont été utilisés pour analyser les tendances en matière de sécurité dans la base de données CVE, sans qu'il soit nécessaire de recourir à un système de classification.

connaissances préalables (expert). Huang et. al. [27] ont récemment proposé une classification automatique des enregistrements de la base de données NVD basée sur un réseau neuronal profond, les auteurs ont comparé leur modèle aux modèles Bayes et KNN et l'ont trouvé supérieur. Toutes les recherches citées ci-dessus se sont concentrées sur la catégorisation de l'aspect logiciel des vulnérabilités, avec des catégories telles que par exemple : Injection SQL, race condition, erreurs cryptographiques, injection de com- mande, etc. A notre connaissance, aucun travail préalable n'a été effectué concernant la catégorisation de l'équipement touché : système ou dispositif - notre travail tente de combler cette lacune.

Cet article est organisé comme suit : dans la section 2, nous décrivons le contenu et la structure de la base de données CVE ; nous décrivons également les enregistrements connexes : CPE (Common Platform Enumer- ation) et NVD (Network Vulnerability Data). Dans la section 3, nous présentons les classes de dispositifs IoT que nous proposons ; nous discutons brièvement des méthodes de classification SVM et des mesures que nous avons utilisées pour tester la qualité des classifications. Dans la section 4, nous présentons les résultats de la classification. Notre travail est résumé dans la section 5.

2 Structure et contenu de la base de données CVE

2.1 La base de données CVE (Common Vulnerability and Exposures)

La base de données Common Vulnerability and Exposures (CVE) hébergée par MITRE est l'une des plus grandes sources publiques d'informations sur les vulnérabilités. Comme l'indique la FAQ du CVE [28] : "CVE *est une liste de vulnérabilités et d'expositions en matière de sécurité de l'information qui vise à fournir des noms communs pour les problèmes connus du public. Le but de CVE est de faciliter le partage des données entre des capacités de vulnérabilité distinctes (outils, repos- itories et services) avec cette " énumération commune. " CVE attribue des identifiers (CVE- ID) aux vulnérabilités publiquement connues des produits informatiques.* Au sein des organisations, des fournisseurs de so- lutions de sécurité informatique et des experts en sécurité, CVE est devenu la norme de facto de partage des informations sur les vulnérabilités et les expositions connues.

Dans ce travail, nous utilisons une version annotée de la base de données CVE, connue sous le nom de National Vulnerability Database (NVD) qui est hébergée par le National Institute of Standards and Technology (NIST). La NVD est créée sur la base des informations fournies par MITRE (et par le site public CVE). Le NIST ajoute d'autres informations telles que les noms de produits structurés et les versions, et fait également correspondre les entrées aux noms CVE. Le flux NVD est fourni à la fois en format XML et JSON, structuré en fichiers année par année, en un seul fichier de base de données complète et en un flux incrémentiel reflétant les vulnérabilités de l'année en cours.

```
< ? xml version='1.0' encoding='UTF-8'?>
  < nvd xmlns:scap-core= "http://scap.nist.gov/schema/scap-
core/0.1" xmlns:cvss= "http://scap.nist.gov/schema/cvss-v2/0.2"
xmlns:vuln= "http://scap.nist.gov/schema/vulnerability/0.4"
xmlns:xsi= "http://www.w3.org/2001/XMLSchema-instance" ...>
   < entry id="CVE-2017-3741">
     < vuln:vulnerable-configuration id="http://nvd.nist.gov/">
       <cpe-lang:logical-test operator="OR" negate="false">
           <cpe-lang:fact-ref name=
             "cpe:/a:lenovo:power_management:1.67.12.19"/>
           <cpe-lang:fact-ref name=
             "cpe:/a:lenovo:power_management:1.67.12.23"/>
         </cpe-lang:logical-test>
     </vuln:vulnerable-configuration>
     < vuln:vulnerable-software-list>
       < vuln:product>
         cpe:/a:lenovo:power_management:1.67.12.19</vuln:product>
     </vuln:vulnerable-software-list>
     < vuln:cve-id>CVE-2017-3741</vuln:cve-id>
     < vuln:published-datetime> 2017-06-
       04T17:29:00.387- 04:00</vuln:published-
       datetime>
     < vuln:last-modified-datetime> 2017-06-13T13:13:17.
       827-04:00</vuln:last-modified-datetime>
     < vuln:cvss>
       < cvss:base_metrics>
         < cvss:score> 2.1</cvss:score>>.
         < cvss:authentication> NONE</cvss:authentication>
         < cvss:confidentiality-impact> AUCUNE
          </cvss:confidentiality-impact>
         < cvss:integrity-impact> PARTIEL</cvss:integrity-impact>.
         < cvss:availability-impact> AUCUNE</cvss:availability-impact>.
       </cvss:base_metrics>>.
     </vuln:cvss>
     < vuln:cwe id="CWE-254"/>
     < vuln:references xml:lang="en" reference_type="VENDOR_ADVISORY">
       <vuln:source> CONFIRM</vuln:source>
       < vuln:reference
         href="https://support.lenovo.com/us/en/product_security/
          LEN-14440" xml:lang="en"> https :...//.../LEN-14440
       </vuln:reference>
     </vuln:references>
     < vuln:summary> Dans le pilote Lenovo Power Management
       antérieur à la version 1.67.12.24, un utilisateur local
       peut modifier .... Ce site
       Le problème ne concerne que la génération ThinkPad X1 ...
       </vuln:summary>
```

```
</entry>
</nvd>
```

Fig. 1. Un seul enregistrement NVD simplifié provenant du flux CVE du NIST (certains champs moins pertinents ont été abrégés ou omis).

La figure 1 contient un exemple d'enregistrement (simplifié) de la base de données NVD. Les champs pertinents pour la suite de la discussion sont les suivants :

- L'**`entrée`** contient l'identifiant de l'enregistrement tel qu'émis par MITRE, l'identifiant est sous la forme : CVE- yyyy-nnnnn (par exemple CVE-2017-3741) et est couramment utilisé dans d'autres bases de données, documents, etc. pour faire référence à une vulnérabilité donnée.
- **`La liste vuln:vulnerable-configuration`** et **`vuln:vulnerable-software`** identifie les produits logiciels et matériels affected par une vulnérabilité. Cet enregistrement contient la description d'un produit et suit les spécifications de la norme Common Platform Enumeration (CPE). Comme la portée de la vulnérabilité peut être complexe - par exemple, elle peut se référer à une ver- sion particulière du logiciel sur une plate-forme matérielle donnée, la description du produit est for- matée comme une expression structurée, logique, ET-OU.
- **`cpe-lang`** - L'enregistrement de base de la structure de la vulnérabilité De plus amples informations sur le for- mat CPE seront fournies dans la section suivante.
- **`vuln:cvs`** et **`cvss:base_metrics`** décrivent la portée et l'impact de la vulnérabilité. Ces données permettent d'identifier les conséquences réelles de la vulnérabilité, c'est-à-dire ses impacts en termes d'accès, de disponibilité et de confidentialité. Par exemple, elles indiquent si le bogue permet une prise de contrôle à distance du système, s'il s'agit d'une violation de données, etc.
- **`vuln:cwe`** contient une référence à une base de données développée par la communauté sur les faiblesses communes de sécurité des logiciels (CWE) [29]. CWE est hébergée par MITRE et contient une liste formelle de types de faiblesses logicielles. En termes simples, l'ID CWE identifie le type de bogue qui a causé la vulnérabilité.
- **`vuln:references`** peut contenir une URL fournissant des informations supplémentaires sur la vulnérabilité.
- **`vuln:summary`** contient une brève description informelle de la vulnérabilité.

2.2 Énumération de la plate-forme commune (CPE)

Le CPE est un système de dénomination formel permettant d'identifier : les applications, les dispositifs matériels et les systèmes d'exploitation. CPE fait partie de la norme Security Content Automation Protocol (SCAP) 5 [30], proposée par le National Institute of Standards and Technol- ogy (NIST). Nous ferons référence ici à la version 2.3 la plus récente de CPE. Le système de dénomination CPE est basé sur un ensemble d'attributs appelé Well-Formed CPE Name (WFN) [31]. Les attributs suivants font partie de ce format : partie, vendeur, produit, version, mise à jour, édition, langue, édition du logiciel, logiciel cible, matériel cible et autre (tous les attributs ne sont pas toujours présents dans l'enregistrement CPE, très souvent "mise à jour" et les suivants sont omis de l'enregistrement). Actuellement, la CPE supporte deux formats : URI (defined

dans la version 2.2 de CPE) et chaîne formatée (définée dans la version 2.3 de CPE). La base de données CVE utilise le format URI et nous ne parlerons que de ce format par la suite.

Par exemple, dans l'enregistrement CPE suivant **`cpe:/h:moxa:edr-g903:-,`** les attributs sont les suivants : **`part:h`** (indiquant le dispositif matériel), **`vendor:moxa`**, **`prod- uct:edr-903`**, version, et les attributs suivants ne sont pas fournis. Comme deuxième exemple, considérons un enregistrement complexe `vuln:vulnerable-configuration` - **Fig. 2** ci-dessous.

```
< vuln:vulnerable-configuration
    id="http://nvd.nist.gov/">
  < cpe-lang:logical-test operator="AND" negate="false">
    < cpe-lang:logical-test operator="OR" negate="false">
      < cpe-lang:fact-ref name=
        "cpe:/o:d-link:dgs-1100_firmware:1.01.018"/>
      < /cpe-lang:logical-test>
      < cpe-lang:logical-test operator="OR" negate="false">
      < cpe-lang:fact-ref name="cpe:/h:d-link:dgs-1100-05:-"/>
      < cpe-lang:fact-ref name="cpe:/h:d-link:dgs-1100-05pd:-"/>
      < cpe-lang:fact-ref name="cpe:/h:d-link:dgs-1100-08:-"/>
      < cpe-lang:fact-ref name="cpe:/h:d-link:dgs-1100-08p:-"/>
      < /cpe-lang:logical-test>
    < /cpe-lang:logical-test>
  < /vuln:vulnerable-configuration>
```

Fig. 2. Un enregistrement de configuration vulnérable provenant de CVE - une expression logique construite à partir de CPE identifiants.

L'enregistrement présenté sur la **figure 3** fait référence à une version particulière du système d'exploitation (firmware), à savoir : `cpe:/o:d-link:dgs-1100_firmware:1.01.018` sur une liste de dispositifs matériels distincts : `cpe:/h:d-link:dgs-1100-05:-`, `cpe:/h:d-link:dgs- 1100-05pd:-`, etc. La CPE n'identifie pas des instanciations uniques de produits sur des systèmes, mais plutôt des classes abstraites de produits. Le premier composant du dé-scripteur CPE est "part" ; il peut prendre les valeurs suivantes : a - pour application, h - pour matériel, o - pour système d'exploitation.

Les enregistrements CPE sont maintenus dans une base de données séparée "CPE Dictionary" [31] qui est distribuée au format XML, elle contient des enregistrements de produits avec une référence URL à la description du produit - **Fig. 1Fig. 3**.

```
< cpe-item name="cpe:/h:d-link:dap-1320:-">
  < title xml:lang="en-US"> D-Link DAP-1320</title>
    < références>
      < reference href="http://us.dlink.com/products/access-
points- range-extenders-and-bridges/wireless-range-extender/">Site
web du fournisseur</reference>
    </references>
    < cpe-23:cpe23-item name="cpe:2.3:h:d-link:dap-1320:-
:*:*:*:*:*:*:*"/>
</cpe-item>
```

Fig. 3. Un enregistrement CPE définissant un appareil nommé.

2.3 Discussion

La base de données NVD est distribuée sous forme de flux XML et JSON, il est également possible de télécharger l'ensemble des données historiques (à partir de 1999, mais les enregistrements conformes à la spécification actuelle sont disponibles pour les données générées depuis 2002). En outre, il existe également une interface de recherche en ligne. La base de données, au début de 2020, contient plus de 120 000 enregistrements au total, et le nombre d'enregistrements augmente en moyenne d'année en année. Pour des raisons historiques - une longue période de collecte de données - la base n'est ni totalement cohérente, ni exempte d'erreurs. Les enregistrements les plus anciens manquent d'informations (par exemple le score de vulnérabilité) ; l'URL dans le champ de référence peut être obsolète ; il y a environ 900 enregistrements sans identifiant CPE ; il existe un grand nombre d'enregistrements avec des CPE incohérents ou non présents dans le dictionnaire CPE (environ 100 000 CPE). En général, le lien entre la description de la vulnérabilité et le produit concerné peut être problématique, car il n'est fourni que par le CPE - par exemple, il n'y a pas de description de produit "en langage clair", ou de classification présente dans la base de données. Les noms de produits contenant des caractères non ASCII ou non européens posent également un problème, car ils sont recodés en ASCII souvent de manière incohérente ou erronée.

L'absence de classification des enregistrements au niveau CVE ou CPE (à l'exception de l'attribut "application, OS ou matériel" dans le CPE) est particulièrement lourde, car il n'existe aucun moyen facile ou évident de différencier les produits. Essentiellement, il est impossible d'extraire les données relatives, par exemple : aux serveurs web, aux routeurs domestiques, aux appareils domestiques IoT, aux caméras de sécurité, aux voitures, aux systèmes SCADA, etc. sans une connaissance a priori des produits et des ven- drains.

3 CVE Classification et analyse des données

3.1 Sélection des données

Pour les besoins de la classification, nous n'avons sélectionné que les enregistrements dont l'attribut "part" du CPE est défini sur "h" (enregistrements matériels), c'est-à-dire que les critères de sélection étaient les suivants : si l'un des enregistrements de la section `vuln:vulnerable-configuration` contient un CPE avec la partie =
= "h", l'enregistrement a été sélectionné pour un examen plus approfondi. Les autres enregistrements ont été écartés. La raison est la suivante : tous ou la plupart des enregistrements de type " matériel " font référence à des appareils ou des systèmes qui peuvent potentiellement être un composant de la couche perception ou réseau de l'architecture IoT ou IIoT. Nous avons également réduit l'horizon temporel aux données des années 2010-2019 (les données du premier trimestre 2019 ont été prises en compte). La **figure 4** montre le nombre de tous les enregistrements de cette période.

L'analyse manuelle des données de vulnérabilité sélectionnées nous a conduits à regrouper les enregistrements en 7 classes distinctes comme suit :

- **H** - Appareils domestiques et SOHO ; routeurs, caméras et surveillance en ligne, autres appareils de qualité client.
- **S** - SCADA et systèmes industriels, automatisation, systèmes de capteurs, non domestique
  Appareils IoT, voitures et véhicules (sous-systèmes), dispositifs médicaux, enregistreurs vidéo industriels et systèmes de surveillance.

- **E** - Matériel d'entreprise, de fournisseur de services (SP) (routeurs, commutateurs, Wi-Fi d'entreprise et mise en réseau) - il s'agit principalement du niveau réseau de l'infrastructure de l'IdO.
- **M** - téléphones mobiles, tablettes, montres intelligentes et appareils portables - ce consti-
  tutes les "contrôleurs" des systèmes IoT.
- **P** - PC, ordinateurs portables, appareils informatiques de type PC et serveurs PC (entreprise) : ils constituent les "contrôleurs" des systèmes IdO.
- **A** - autres appareils non domestiques : imprimantes et systèmes d'impression d'entreprise, photocopieuses.
  les machines, les appareils de stockage et multimédia non destinés aux clients.

La raison de cette classification est d'ordre pratique : la principale distinction pour un composant IoT en ce qui concerne sa vulnérabilité en matière de sécurité est le marché et le champ d'application (utilisation domestique, utilisation industrielle, couche réseau, etc.) D'autre part, nous sommes limités par la description des données disponibles - il serait impossible d'utiliser une classification plus fine. De plus, il n'est pas pratique d'introduire de nombreuses classes avec un petit nombre de membres, car la qualité de la classification automatique en souffre (comme nous l'avons appris dans les mêmes cas). Des exemples de dispositifs de la base de données NVD sont présentés dans le tableau 1.

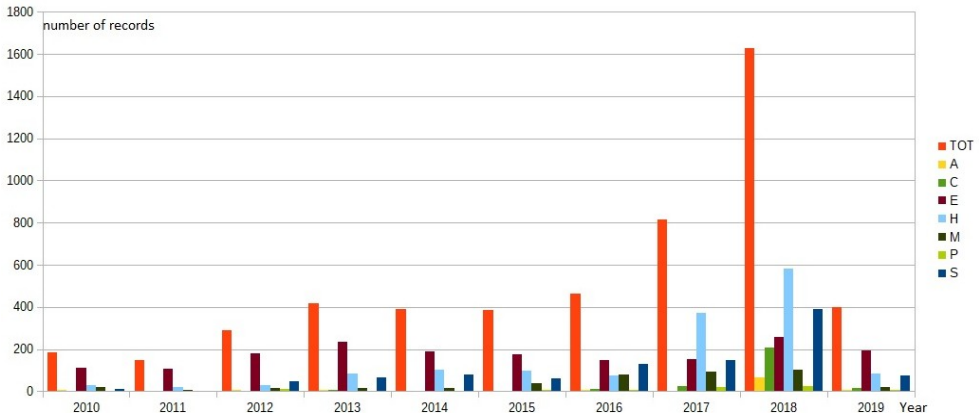


Fig. 4. Nombre d'enregistrements "h" classés dans la période 2010-2019. TOT est le nombre total d'enregistrements, les autres barres font référence aux classes attribuées. Seul le premier trimestre de 2019 a été pris en compte, d'où un nombre d'enregistrements plus faible.

Tableau 1. Appareils de la base de données NVD - échantillons de chacune des classes proposées.

Classe	Systèmes d'échantillonnage (fournisseur, nom)	Notes
H	• D-link Dir-815 • Opticam i5 • Meetcircle Circle With Disney • Amazon Amazon_key	• Réseau et routeur Wi-Fi • Caméra de surveillance domestique • Dispositif de contrôle parental • Dispositif d'accès à domicile
S	• Siemens Sinumerik 828d • Yokogawa FCI • Cockpit du Mbusa • Appareil photo Vivotek	• contrôleur de machine-outil industrielle • contrôleur autonome • automatisation du cockpit de la voiture • caméra de surveillance
E	• Juniper SRX100 SRX110 • CISCO staros asr_5000 • Passerelle Citrix Netscaler • Polycom QDX6000	• Famille de pare-feu de réseau • Routeur et dispositif d'accès • Système d'équilibrage de la charge du réseau • Système de vidéoconférence
M	• Samsung Galaxy S6 • Amazon Kindle Fire • Mi mi router 3 • Huawei Watch 2	• Téléphone intelligent • Tablette / lecteur de livres électroniques • Routeur Wi-Fi/LTE portable • Dispositif de montre intelligente
P	• Éclairage intégré HP • HP nonstop_server • Intel s7200ap	• Module de gestion du serveur • Plate-forme de serveur • Carte mère du serveur
A	• Drobo 5n2 • TBK Vision tbk-dvr4216 • Ricoh d2200	• Système de stockage de données d'entreprise • Système DVR d'entreprise • Système d'impression d'entreprise

3.2 Méthodologie d'analyse des données

Nous construisons des classificateurs en entraînant des machines à vecteurs de support linéaires (SVM) [32] sur les caractéristiques des enregistrements de vulnérabilité "matérielle" extraits de la base de données NVD. Le vecteur de caractéristiques contient :

- Nom du vendeur,
- le nom du produit et d'autres données sur le produit provenant du CPE (s'il est fourni),
- la description de la vulnérabilité,
- code d'erreur (CWE).

Les étapes du processus de construction d'un classificateur sont les suivantes :

- Prétraitement des données d'entrée (suppression des mots d'arrêt, lemmatisation, etc.),
- l'extraction de caractéristiques, c'est-à-dire la conversion des données textuelles en espace vectoriel,

- formation du SVM linéaire.

Nous utilisons un SVM linéaire standard, qui calcule l'hyperplan de marge maximale qui sépare les exemples positifs et négatifs dans l'espace des caractéristiques. La classification basée sur un SVM linéaire génère l'hyperplan qui maximise la distance des exemples de formation les plus limites par rapport au plan de décision linéaire (ou frontière). Parmi les autres méthodes possibles, citons : le k-plus proche voisin, les classificateurs bayésiens et les réseaux neuronaux. Nous avons mené quelques expériences avec les réseaux neuronaux, mais nous avons finalement décidé d'utiliser la méthode SVM, car elle s'est avérée rapide, efficace et bien adaptée à la classification des données textuelles. Avec la méthode SVM, la limite de décision est non seulement spécifiée de manière unique, mais la théorie de l'apprentissage statistique montre qu'elle produit des taux d'erreur attendus plus faibles lorsqu'elle est utilisée pour classer des exemples non vus auparavant [32,33] - c'est-à-dire qu'elle donne de bons résultats lors de la classification de nouvelles données.

Nous avons utilisé Python 3.7.1 avec les bibliothèques NLTK 3.4.1 [34] et scikit-learn 0.21.3 [35]. NLTK a été utilisé pour prétraiter les données textuelles, tandis que scikit contient des algorithmes SVM ainsi que des outils pour calculer les mesures de qualité de la classification.

3.3 Mesures de classification

Pour évaluer les résultats de la classification, nous utilisons deux mesures standard : la précision et le rappel. Nous définissons la *précision* (eq. (1)) comme une fraction d'instances pertinentes parmi les instances récupérées ; nous définissons le *rappel* (eq. (2)) comme la fraction de la quantité totale d'instances pertinentes qui ont été effectivement récupérées. En d'autres termes, la précision indique le rapport entre les vrais positifs et la somme des vrais positifs et des faux positifs, tandis que le rappel est calculé comme le rapport entre les vrais positifs et la somme des vrais positifs et des faux négatifs (éléments appartenant à la catégorie actuelle mais non classés comme tels). Enfin, comme mesure concise de la qualité, nous utiliserons le score F1, également connu sous le nom de score F équilibré. Le score F1 peut être interprété comme une moyenne pondérée de la précision et du rappel, où un score F1 atteint sa meilleure valeur à 1 et son pire score à 0. La contribution relative de la précision et du rappel au score F1 est égale. La formule pour le score *F1 est* donnée par l'équation (3)

$$precission = TP/(TP + FP) \quad (1)$$

$$recall = TP/(TP + FN) \quad (2)$$

$$F1 = 2 * \frac{precision*recalll}{(prceision+recall)} \quad (3)$$

4 Résultats de la classification

4.1 Sélection et classification des données

Nous avons testé le classificateur pour des données historiques par intervalles d'un an. Par exemple, pour classifier les données de 2018, nous avons utilisé les enregistrements des plages suivantes : 2014-2017, 2015-2017, 2016-2017 et 2017, etc. Sur la **figure 5**, nous montrons les matrices de confusion formées sur les données allant de 2014 à 2017 utilisées pour classer les données de 2018. D'un bon classificateur

on s'attendrait à une majorité d'enregistrements sur la diagonale. Ici, la classification n'est pas parfaite, par exemple - pour les données d'entraînement de la gamme d'années 2014-2017 : 109 enregistrements de type H ont été marqués comme étant de classe E et 108 de classe S ; seuls 62 % ont été correctement classés (rappel). Lorsque seules les données de 2017 ont été utilisées, de manière peut-être surprenante, la classification est plus précise : 489 / 85 % des enregistrements de type H ont été étiquetés correctement (rappel) ; cependant, pour les classes S et E, seuls 55 % ont été correctement identifiés. Pour les classes avec un faible nombre d'enregistrements (C, M, P), la classification tombe en dessous de 50%. Sur la **Fig. 6.** Précision et rappel pour les enregistrements de 2018 basés sur les données d'entraînement de : 2014-2017, 2015-2017, 2016-2017 et 2017 (la cat- égorie "A" a été supprimée).

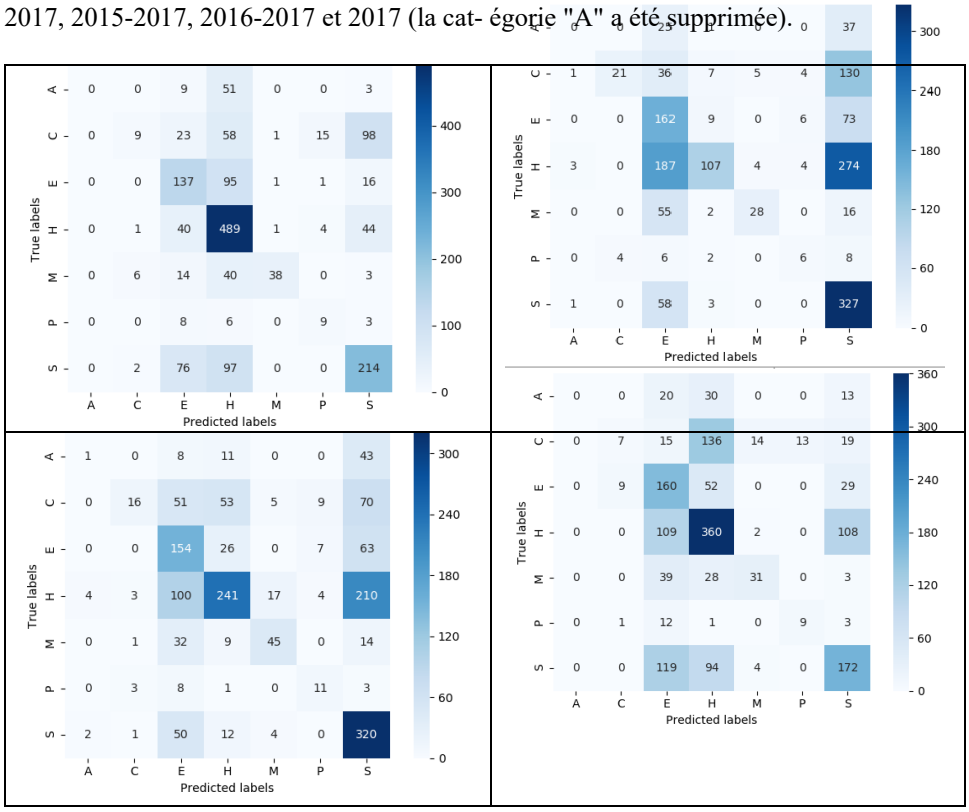


Fig. 5. Classification des enregistrements de 2018 basée sur les données d'entraînement de : 2017, 2016-2017, 2015-2017 et 2014-2017 (de gauche à droite, de haut en bas). Les chiffres indiquent le nombre d'enregistrements.

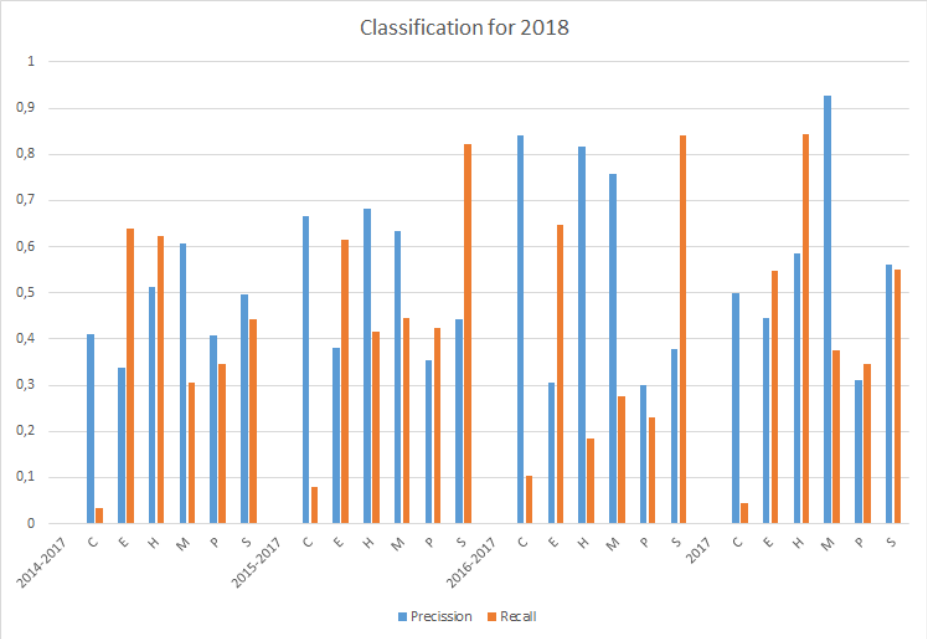
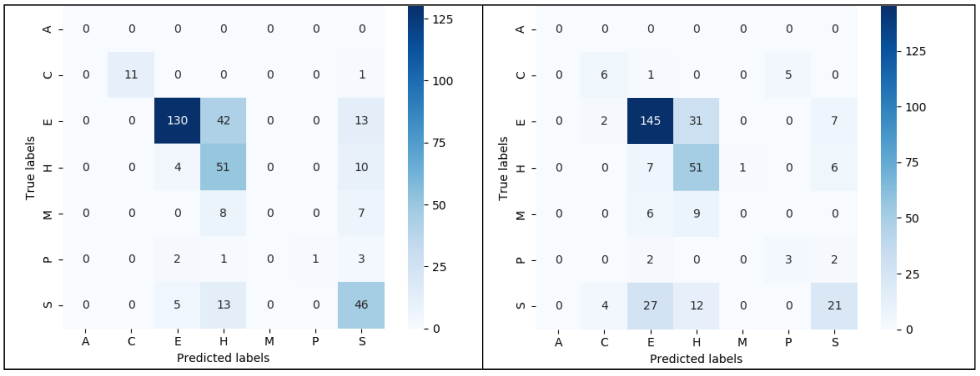


Fig. 6. Précision et rappel pour les enregistrements de 2018 basés sur les données d'entraînement de : 2014-2017, 2015- 2017, 2016-2017 et 2017 (la catégorie " A " a été supprimée).

Des tendances similaires sont visibles pour les données classées du premier trimestre 2019 basées sur le SVM entraîné sur des enregistrements de la gamme 2015-2018 - pour les classes E, H et S, la précision se situe dans la fourchette de 70 % à 90 % (à l'exception de la classe H où elle n'est que de 44 %), et le rappel tombe dans la fourchette similaire de 70 % à 90 %.

Sur la **figure 8**, nous avons montré les résultats pour les années 2015 et 2016 - pour les classes principales, la classification pour les années allant de 2011 à 214 et 2017 montrent également des tendances similaires.



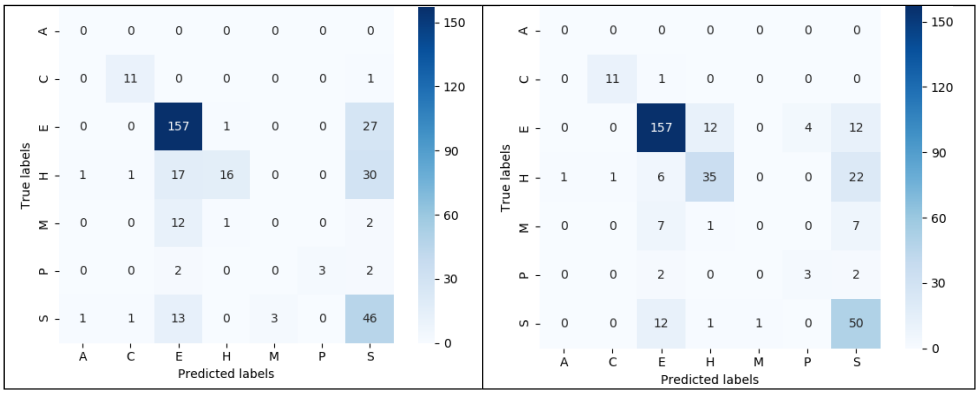


Fig. 7. Classification des enregistrements du 1st trimestre 2019 sur la base des données de : 2018, 2017-2018, 2016-2018 et 2015-2016 (de gauche à droite, de haut en bas). Les chiffres indiquent le nombre d'enregistrements.

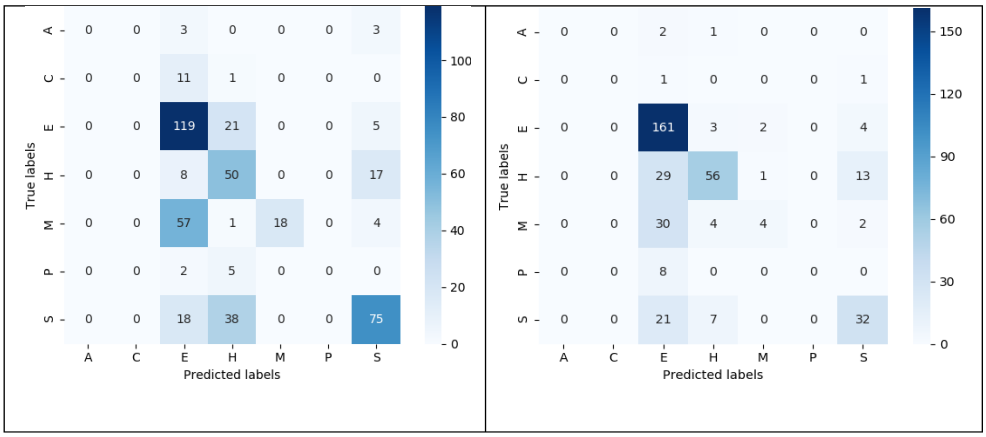


Fig. 8. Classification des enregistrements de 2015 basée sur les données de 2014 (à gauche) et de 2016 basée sur les données de 2015 (à droite). Les chiffres indiquent le nombre d'enregistrements.

4.2 Discussion

Comme nous l'avons montré dans la section précédente, la qualité des résultats de classification peut être résumée comme moyenne. Nous avons pu atteindre 70-80% d'étiquetage correct pour les classes d'appareils les plus peuplées. Dans certains cas, la classification tombe en dessous de 50%. L'utilisation de plus de données d'entraînement, c'est-à-dire le retour en arrière, n'améliore pas toujours la qualité de la classification, au contraire, dans la plupart des cas, elle la réduit.

Pour résumer les résultats de classification pour l'ensemble de la période 2011-2019(Q1) : La **figure 9** montre les valeurs de la mesure *F1* pondérée par le support (le nombre d'instances vraies pour chaque étiquette). En raison de la pondération, cela montre la qualité de la classification pour toutes les classes. Comme on le voit, le score *F1* pondéré varie entre 50% et 72%.

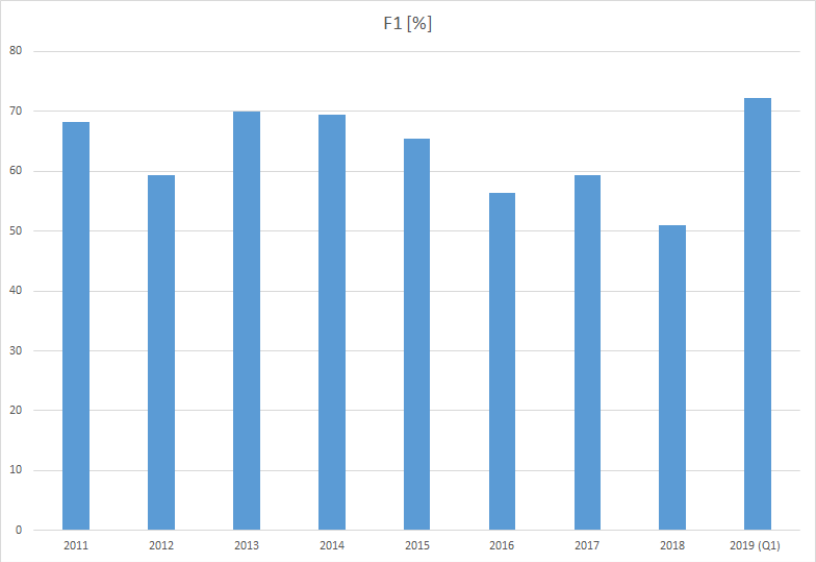


Fig. 9. Score F1 pour les enregistrements de 2011 à 2019(Q1).

5 Résumé

Nous avons proposé une classification des données de vulnérabilité liées aux dispositifs IoT à partir de la base de données publique CVE/NVD. Nous avons divisé les enregistrements de vulnérabilité en 7 catégories distinctes : Domicile et SOHO, SCADA, Entreprise et mise en réseau, Appareils mobiles, Dé- vices de PC et autres appareils non domestiques. Les échantillons de la base de données classés manuellement ont été utilisés pour entraîner un classificateur SVM afin de prédire les catégories de "nouvelles" vulnérabilités.

L'objectif du classificateur automatique est de prédire, et (si possible) dans les étapes suivantes - de prévenir et d'atténuer les menaces résultant des nouvelles vulnérabilités. Ce n'est pas une tâche triviale à exécuter à la main étant donné la taille de la base de données et le taux de croissance de celle-ci

- Lorsqu'une nouvelle vulnérabilité ou un nouvel exploit est découvert, il est souvent essentiel d'en connaître la portée par des moyens automatiques, aussi rapidement que possible.

Nous avons atteint des taux de précision et de rappel de classification de 70-80% pour les catégories fortement peuplées et d'environ 50% ou moins pour les catégories moins nombreuses. Ces résultats ne sont pas idéaux et, en pratique, ils nécessiteraient une intervention humaine supplémentaire (vérification et éventuellement reclassification). D'un autre côté, les classificateurs SVM ont prouvé à de nombreuses reprises qu'ils constituaient un mécanisme précis pour la classification de données textuelles. Dans notre cas, le problème réside dans les données elles-mêmes - ni le contenu de CVE ni celui de CPE ne fournissent suffisamment de données spécifiques pour que le SVM puisse discerner les catégories d'enregistrements. Nous pouvons conclure que l'ontologie de la vulnérabilité devrait être étendue pour fournir ces informations supplémentaires. Des conclusions similaires, bien qu'elles ne soient pas directement liées à la sécurité de l'IdO, ont été tirées par d'autres chercheurs - par exemple, dans [36], les auteurs proposent une ontologie unifiée de la cybersécurité qui incorpore et intègre des données hétérogènes et des schémas de connaissances.

de différents systèmes de cybersécurité, y compris des données sur les produits et les fournisseurs de produits.

Enfin, il convient également de mentionner que la méthode que nous utilisons n'est pas nécessairement limitée à la base de données CVE, de nombreuses autres bases de données de vulnérabilités en ligne existent [37], qui sont gérées par des entreprises (par exemple, Microsoft Security Advisories, TippingPoint Zero Day Initiative, etc.), des CERT nationaux ou des forums de professionnels (par exemple, BugTraq, Exploit DB, et d'autres). Les informations provenant de diverses sources peuvent être intégrées et classées par la méthode que nous avons proposée dans cet article. Cela devrait augmenter la précision de la classification et constitue un sujet de nos recherches futures.

1Références

1. Atzori, L., Iera, A., Morabito, G. : The internet of things : A survey. Comput. Netw., vol. 54, no. 15, pp. 2787 - 2805 (2010)
2. Page d'accueil RFC de l'IETF, IP Mobility Support for IPv4, Revised https://tools.ietf.org/html/rfc5944, dernier accès 2017/05/10
3. Le 7e programme-cadre a financé la recherche et le développement technologique européens de 2007 à 2013 ; Internet des objets et systèmes d'entreprise de l'Internet du futur ; http://cordis.europa.eu/fp7/ict/enet/projects_en.html, dernier accès le 10 mai 2017.
4. Modèle de référence architectural pour l'IdO - (ARM). Livret d'introduction. Récupéré de : http://iotforum.org/wp-content/uploads/2014/09/120613-IoT-A-ARM-Book-Introduction-v7.pdf, dernier accès 2017/05/10
5. Da Xu, L., He, W., Li, S. Internet of things in industries : Une enquête. IEEE Transactions on industrial informatics, *10*(4), p. 2233-2243 (2014).
6. Jalali, R., El-Khatib, K., McGregor, C. : Architecture de ville intelligente pour les services de niveau communautaire à travers l'internet des objets. In Intelligence in Next Generation Networks (ICIN), 18e conférence internationale sur, p. 108-113 (2015).
7. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, M. : Internet of things : A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*, *17*(4), 2347-2376 (2015).
8. Antonakakis, M., April, T., Bailey, M., Bernhard, M., Bursztein, E., Cochran, J., ... & Kumar, D. : Understanding the mirai botnet. Dans le 26e USENIX Security Symposium, USENIX Security 17, p. 1093-1110 (2017).
9. Ling, Z., Liu, K., Xu, Y., Gao, C., Jin, Y., Zou, C... & Zhao, W. : IoT security : an end-to-end view and case study, arXiv preprint arXiv:1805.05853. (2018)
10. S. à Silicon Lab, " Iot security vulnerability database ", http://www.hardwaresecurity.org/iot/database, août 2017.
11. Arias, O., Wurm, J., Hoang, K., & Jin, Y. : Privacy and security in internet of things and wearable devices. IEEE Transactions on Multi-Scale Computing Systems, 1(2), 99-109. (2015)
12. Wurm, J., Hoang, K., Arias, O., Sadeghi, A. R., & Jin, Y. : Analyse de sécurité sur les appareils IoT grand public et industriels. In 2016 21st Asia and South Pacific Design Automation Confer- ence (ASP-DAC) (pp. 519-524). IEEE. (2016)
13. Hernandez, G., Arias, O., Buentello, D., & Jin, Y. : Smart nest thermostat : Un espion intelligent dans votre maison. Black Hat USA, 1-8. (2015)

14. Balasch, J., Gierlichs, B., & Verbauwhede, I. : An In-depth and Black-box Characterization of the Effects of Clock Glitches on 8-bit MCUs. Dans 2011 Workshop on Fault Diagnosis and Tolerance in Cryptography (pp. 105-114). IEEE. (2011)
15. 28 Chen, D. D., Woo, M., Brumley, D., & Egele, M. : Towards Automated Dynamic Analysis for Linux-based Embedded Firmware. Dans NDSS (pp. 1-16). (2016)
16. /DEV/TTYS0, " Hacking the d-link dsp-w215 smart plug ", http://www.devttys0.com/2014/05/ hacking-the-d-link-dsp-w215-smart-plug/. (2014)
17. " Sécurité critique flaw : glibc stack-based buffer overflow in getaddrinfo() (cve-2015-7547) ", https://access.redhat.com/articles/2161461, (2015).
18. Smith, M. : " Security holes in the 3 most popular smart home hubs and honeywell tuxedo touch ", http://www.networkworld.com/article/2952718/microsoftsubnet/security-holes-in-the-3-most-popular-smart-home-hubsand-honeywell-tuxedo-touch.html, 2015.
19. Rahman, M., Carbunar, B., & Banik, M. : Fit and vulnerable : Attaques et défenses pour un dispositif de surveillance de la santé. arXiv preprint arXiv:1304.5672. (2013).
20. Obermaier, J., & Hutle, M. : Analyzing the security and privacy of cloud-based video surveillance systems. In Proceedings of the 2nd ACM International Workshop on IoT Privacy, Trust, and Security (pp. 22-28). ACM. (2016).
21. N. Dhanjani, "Security evaluation of the philips hue personal wireless lighting system," http://www.dhanjani.com/docs/Hacking%20Lighbulbs%20Hue%20Dhanjani%202013.pdf, (2013)
22. Durumeric, Z., Li, F., Kasten, J., Amann, J., Beekman, J., Payer, M., ... & Halderman, J. A. : The matter of heartbleed. In Proceedings of the 2014 conference on internet measurement conference (pp. 475-488). ACM. (2014)
23. Aviram, N., Schinzel, S., Somorovsky, J., Heninger, N., Dankel, M., Steube, J., ... & Käsper : DROWN : Breaking TLS Using SSLv2. In 25th USENIX Security Symposium (USENIX Security 16) (pp. 689-706). (2016)
24. Wang, J. A., Guo, M. : Vulnerability categorization using Bayesian networks. Dans les actes du sixième atelier annuel sur la recherche en cybersécurité et en intelligence informatique (pp. 1-4). (2010).
25. Na, S., Kim, T., Kim H. : "A study on the classification of common vulnerabilities and exposures using Naïve Bayes," in Proc. Int. Conf. Broadband Wireless Comput., Commun. Appl. Cham, Suisse : Springer, pp. 657-662. (2016)
26. Neuhaus, S., & Zimmermann, T. : Security trend analysis with cve topic models. Dans 2010 IEEE 21st International Symposium on Software Reliability Engineering (pp. 111-120). IEEE. (2010)
27. Huang, G., Li, Y., Wang, Q., Ren, J., Cheng, Y. et Zhao, X. : Méthode de classification automatique de la vulnérabilité des logiciels basée sur un réseau neuronal profond. IEEE Access, 7, 28291- 28298. (2019).
28. MITRE, base de données CVE Common Vulnerabilities and Exposures, https://cve.mitre.org/, dernière consultation : 02.01.2020 (2020)
29. MITRE, CWE Common Weakness Enumeration, A Community-Developed List of Software Weakness Types, https://cwe.mitre.org/about/index.html, dernier accès : 02.01.2020. (2020)
30. NIST, Security Content Automation Protocol v 1.3, https://csrc.nist.gov/projects/security-content-automation-protocol/, dernière visite le 02.01.2020. (2020)
31. NIST, Official Common Platform Enumeration (CPE) Dictionary, https://csrc.nist.gov/projects/security-content-automation-protocol/, dernière visite le 02.01.2020. (2020)
32. Vapnik, V. : Théorie de l'apprentissage statistique. John Wiley & Sons, New York, NY. (1998)

33. Liu, Z., Lv, X., Liu, K., & Shi, S. . Étude sur le SVM comparé aux autres méthodes de classification de texte. Dans Second International Workshop on Education Technology and Computer Sci- ence (Vol. 1, pp. 219-222). IEEE. (2010)
34. NLTK, Natural Language Toolkit, https://www.nltk.org/, dernier accès : 02.01.2020
35. Scikit-learn, Machine learning in Python, https://scikit-learn.org/stable/, dernier accès : 02.01.2020
36. Syed, Z., Padia, A., Finin, T., Mathews, L., & Joshi, A. : UCO : A unified cybersecurity on- tology. In Workshops at the Thirtieth AAAI Conference on Artificial Intelligence. (2016).
37. Liste des ressources de la base de données sur les vulnérabilités, https://www.yeahhub.com/list-vulnerability-data- bases-resources-2018-compilation/, dernier accès : 02.01.20120