Projet de fin d'études

Département de génie logiciel et des TI

Surveillance de systèmes complexe avec Splunk

Étudiants :

Olivier Grégoire	GREO10049507	Jean-Baptiste Charvet	CHAJ03119219
Nour El-Moussaoui	ELMN10069308	Stéphane Boyer	BOYS18067407
Wajdi Bounouara	BOUW05059404	Massaki Archambault	ARCM10069508

Professeur superviseur :

Alain April

Superviseurs représentants de Matricis :

Baptiste Cabrera

Cédric Mélançon

Montréal, 7 août 2019, session d'été

Suivi de version

Version	Modification	Date
0.1	Version initiale	2019-08-01
0.2	Relecture de Alain APRIL	2019-08-01
0.3	Correction suite aux commentaires de la version 0.2	2019-08-04
0.4	Compléter certaines parties manquantes	2019-08-04
1.0	Corrections avant remise finale	2019-08-06

Liste des figures

Liste des abréviations, sigles et acronymes

Termes	**Description**
PFE	Projet de Fin d'Étude
IoT	Internet des objets
IIoT	IoT dans le secteur industriel
VPN	Réseau privé virtuel
GCVP	gestion du cycle de vie des produits, ou GCVP
SQL	langage de requête structurée
RDB	Base de données relationnelle
DevOps	Développement logiciel (dev) et de l'administration des infrastructures informatiques (ops),
VM	Machine virtuelle (virtual machine)
JVM	Java Virtual Machine
PGI	Progiciel de Gestion Intégré (ERP en anglais)
GRC	Gestion de la Relation Client (CRM en anglais)
API	Application Programming Interface
ACV	Analyse de Cycle de Vie
SPL	Language de requête de Splunk (“Search Processing Language” en anglais)

Remerciements

La réalisation de ce projet n'est pas que le résultat du travail de notre équipe, mais aussi des conseils et de l'aide apportée par des acteurs que nous tenons à remercier convenablement.

Cédric Mélançon ainsi que Baptiste Cabrera, en tant que représentants de Matricis, ont fourni un suivi et une aide précieuse dans ce projet. C'est grâce à leur confiance et leurs encouragements que ce projet a atteint ses objectifs. De même, nous remercions Matricis qui a mis à disposition les moyens permettant de faire ce projet (outil Splunk, infrastructure, etc.) et de l'avoir proposé, tout simplement.

De même, nous remercions Dr. Alain April du département de Génie logiciel à l'École de Technologie Supérieure pour ses conseils quant à la gestion d'un projet de fin d'études.

Enfin, nous nous remercions mutuellement d'avoir fait preuve de proactivité et d'esprit d'équipe dans ce projet qui au final, a réuni beaucoup de monde!

1. Présentation du projet

1.1 Présentation de Matricis

Fondée en 1999, Matricis se spécialise en intégration d'applications d'entreprise, à l'intégration de données et à la mise en place d'architecture orientée services. L'expertise de Matricis s'étend aussi à l'Internet des Objets (IoT), à l'efficacité opérationnelle, à l'intelligence d'affaires et à la gestion des processus d'affaires.

1.2 Contexte et problématique

Le milieu industriel connaît les débuts d'une nouvelle révolution, celle de la convergence de la gestion des opérations d'une entreprise (c.-à-d. finance, marketing, ventes, etc.), de sa production (c.-à-d. chaînes de production en usine) et de son environnement numérique (c.-à-d. PGI et GRC). Concrètement, cette quatrième révolution industrielle[1] consiste à connecter les capteurs et contrôleurs des chaînes de production à l'infonuagique dans le but de simplifier et centraliser leur gestion, permettre aux acheteurs du produit fini de le personnaliser directement, et ce, sans passer par une communication avec les gestionnaires de la chaîne et une modification de l'agencement de celle-ci. Cette communication directe avec la chaîne de production peut par exemple être réalisée avec un outil de GRC.

Cette révolution n'est actuellement qu'en phase de recherche et développement. Les outils qui lui sont nécessaires sont déjà disponibles, mais ne sont pas encore interconnectés et fonctionnels à grande échelle. Le passage à ce nouveau paradigme de l'industrie 4.0 est donc l'élément-clé de l'actualité du secteur de la production automatisée.
En fait, l'utilisation en temps réel et de façon centralisée les données disponibles dans une usine permettra de répondre à plusieurs enjeux :

- La santé et sécurité des travailleurs
  Cela se traduit par exemple par la mesure permanente du volume sonore. Cela permet de connaître le niveau d'exposition au son et de déclencher une alerte dès que le seuil de dangerosité est atteint. De même, des déclenchements d'alerte lors d'une concentration en particules dans l'air ambiant ont une forte valeur pour cet enjeu.
- Le respect de la législation du travail
  La législation du travail a pour rôle de protéger les travailleurs. Pour l'exemple du bruit, le seuil de dangerosité est de 8h d'exposition par jour à 80dB, 4h à 83dB, etc[2]. Au Canada, c'est la norme CSA Z107.58[3] qui s'applique. Le respect de cette norme sans moyen de mesure en temps réel et permanent n'est pas garanti : il est aisé d'éteindre les machines bruyantes lors d'un contrôle de l'inspection du travail[4].
- La prise en compte du cadre socioculturel
  L'absence de données concernant la santé et sécurité des travailleurs (exposition à des dangers : bruits, radiations, air contaminé, etc.) rend difficile une véritable mise en

[1] https://fr.wikipedia.org/wiki/Industrie_4.0
[2] Institut national (français) de recherche et de sécurité pour la prévention des accidents du travail et des maladies professionnelles, dossier sur le bruit : www.inrs.fr/dms/inrs/GenerationPDF/accueil/risques/bruit/Bruit.pdf
[3] https://www.canada.ca/fr/sante-canada/services/publications/securite-et-risque-pour-sante/avis-bruit-machines-milieu-travail.html
[4] https://www.canada.ca/fr/emploi-developpement-social/services/sante-securite/rapports/inspection.html

place d'une gestion d'usine où la protection efficace des travailleurs est une culture d'entreprise. Par extension, protéger les travailleurs permet de limiter le nombre d'occurrences de maladies professionnelles à l'impact lourd sur la société, d'un point de vue humain.

- L'économie
  Les maladies professionnelles ont un véritable coût pécuniaire pour la société : c'est elle qui prend en charge les patients et dédommage ceux qui ont une incapacité de travail. Ainsi, le programme de Canadien de prévention sur les lieux de travail représente une économie de frais de santé au pays. Pour être efficace, cet effort de prévention a besoin de données fiables qui représentent une estimation précise de la réalité.
- L'environnement
  Similairement, la mesure et l'enregistrement des impacts environnementaux est la première étape de leur gestion. Pour limiter ou supprimer un impact d'un indicateur ACV[5], il faut donc être capable de le mesurer et de traiter les données mesurées.

Dans ce contexte, ce projet de fin d'études se concentre sur la gestion des données générées par les chaînes de production : "Surveillance de systèmes complexes avec Splunk". La question de recherche est **Comment utiliser, de façon intelligente, les données générées par une chaîne de production industrielle?**

Pour répondre à ce problème, deux logiciels sont principalement utilisés, il s'agit de l'outil "Splunk"[6] et du produit de GCVP "Thingworx"[7]. Splunk est un logiciel permettant de surveiller efficacement des données et de créer une logique d'affaires à partir de celles-ci. Il est donc possible, à partir de ce logiciel, de réaliser un système d'alerte en temps réel, un système de rapports ou encore un tableau de bord. Quant à lui, Thingworx permet de connecter capteurs et contrôleurs des actionneurs. Les données récupérées par ThingWorx sont utilisées afin d'alimenter en données Splunk.

En résumé, ce projet de fin d'études vise à expérimenter l'importation et le traitement de données de capteurs et contrôleurs d'usine à l'aide de Splunk.

1.3 Objectifs

Le projet est séparé en trois parties. La première partie vise l'exploration, la deuxième l'interfaçage et la troisième partie, l'exploitation des données. Les tâches suivantes sont exécutées dans un ordre qui ne correspond pas nécessairement à ce classement.

1.3.1 Phase d'exploration

Les sources de données du client sont identifiées précisément (c.-à-d. TomCat, SQLServer et Thingworx), tout comme leur contenu : la signification de chacun des champs. De même, les standards de gestion de chaque type de données sont étudiés.

1.3.2 Phases d'interfaçage

Dans cette phase, les outils pour exporter les données des sources vers Splunk (exemple : Telegraf, Forwarder Splunk, etc.) sont identifiés et testés.

---

[5] Page ix, liste d'indicateur ACV : https://www.hydroquebec.com/data/developpement-durable/pdf/ACV-filieres-energie-thermique-sommaire.pdf
Source, Centre international de référence sur le cycle de vie des produits, procédés et services (CIRAIG) : http://www.ciraig.org/fr/index.php

[6] https://www.splunk.com/

[7] https://www.ptc.com/en/products/iiot/thingworx-platform

Une fois que les outils les plus adaptés à notre cas d'utilisation seront identifiés, nous allons procéder à l'installation et à la configuration des outils d'exportation sur des machines virtuelles qui gèrent les sources de données.

Le "Forwarder Splunk" sert à exporter, de façon automatisée, des données vers une instance d'un indexeur appelé "Indexer" dans l'outil. Un "indexer" de Splunk sert à classer les données, quel que soit leurs types, selon une logique à définir (modèle de données), afin d'être exploitables par les différentes applications et extensions de Splunk.

En particulier, une de ces applications est le module de recherche et investigation. Il sert à créer des tableaux de données de recherches préenregistrées ainsi qu'un système d'alerte pour certains évènements à définir.

1.3.3 Phase d'exploitation

Une fois les données récoltées par Splunk, il sera possible d'exécuter la phase d'exploitation. Celle-ci contiendra notamment la définition des évènements qui doivent créer des alertes et définir leur politique de notification. À cette étape, il faudra définir quelles informations seront extraites des données et quelle logique d'affaires appliquer pour les classer. Finalement, une rencontre avec de futurs utilisateurs sera planifiée afin de mieux définir leurs besoins ainsi que le contenu de tableaux de bord à réaliser.

2. Méthodologie et organisation du travail

Notre équipe a choisi une méthodologie Agile-SCRUM afin de livrer une partie fonctionnelle du produit au client toutes les deux semaines (c.-à-d. un Sprint). La durée d'un Sprint peut varier dans le cadre de ce projet vu qu'il n'y a pas de jalons définis précisément par le client. Cette méthodologie Agile permet de s'assurer de la satisfaction des besoins du client (c.-à-d. l'entreprise Matricis) en obtenant un retour régulier de sa part. Après chaque Sprint, une démonstration du progrès, c'est-à-dire la preuve de concept, sera discutée avec le client, afin de faire valider les fonctionnalités réalisées et d'en faire l'amélioration, au besoin. Un atelier hebdomadaire (réunion SCRUM) sera également effectué afin de faire le suivi des tâches, de définir les prochaines étapes de travail et s'assurer de la synchronisation entre les membres de l'équipe. L'équipe aura à sa disposition les outils nécessaires de gestion et de collaboration, tel que : Slack, Google Drive et Azure DevOps afin de gérer ce projet selon les objectifs et délais établis.

2.1 Composition de l'équipe et rôles

Prénom	Entreprise	Rôle(s)	Responsabilités
Jean-Baptiste Charvet	ÉTS	DevOPS Thingworx	- Analyser des besoins d'affaires - Développer - Tester les livrables (Planification et test) - Présenter les tâches réalisées à la fin du Sprint
Nour El-Moussaoui	ÉTS	Gestionnaire de projet/DevOPS SQL Server	- Gouvernance - Analyser des besoins d'affaires - Développer - Présenter les tâches réalisées à la fin du Sprint
Olivier Grégoire	ÉTS	DevOPS Thingworx	- Analyser des besoins d'affaires - Développer - Présenter les tâches réalisées à la fin du Sprint - S'assurer du suivi des bonnes pratiques SCRUM
Stéphane Boyer	ÉTS	DevOPS Tomcat	- Analyser des besoins d'affaires - Développer - Tester les livrables (Planification et test) - Présenter les tâches réalisées à la fin du Sprint
Wajdi Bounouara	ÉTS	DevOPS SQL Server	- Analyser des besoins d'affaires - Développer - Tester les livrables (Planification et test) - Présenter les tâches réalisées à la fin du Sprint

Massaki Archambault	ÉTS	Lead technique/DevOPS Tomcat	-Supervision de l'aspect technique du projet - Analyse des besoins d'affaires - Développement - Présentation des tâches réalisées à la fin du sprint
Baptiste Cabrera	Matricis	Product Owner Splunk/ Scrum Master	- Gérer le Backlog - S'assurer que les besoins sont bien complétés
Cédric Mélançon	Matricis	Product Owner Thingworx	- Gérer le Backlog - S'assurer que les besoins sont bien complétés
Dr Alain April	ÉTS	Professeur attitré	- Accompagner les étudiants - Évaluer le projet

2.2 Livrables

Nom de l'artefact	Remis à	Description
Forwarder Splunk	Matricis	Installation d'une instance de "Forwarder" sur le serveur hébergeant Thingworx Un "Forwarder" est un outil permettant de faire de la collecte de données et l'envoi de celle-ci vers l'indexeur de données de Splunk. Les instances des autres éléments de l'architecture de Splunk nous sont fournies par Matricis : l'indexeur et le "search head".
Dépôts git de fichiers de configuration du Forwarder	Matricis	Configuration du Forwarder pour importer : ● les données applicatives de Thingworx ● les données de l'instance de TomCat avec laquelle est hébergée le service Thingworx. Ces logs, contiennent des métriques de la jvm, ainsi que les "access logs" à Thingworx. ● les données de l'instance de SQLserver de Thingworx, qui contient les entrées des différents objets connectés à Thingworx.
Tableaux de bord Splunk ("Dashboards")	Matricis	Vue d'ensemble des données recueillies permettant au client de faire le suivi et de prendre des décisions en fonction des besoins de celui-ci.
Système d'alerte Splunk	Matricis	Ensemble de règles d'alerte (par exemple : "Plus de 50 échecs de connexion d'un utilisateur ont été détectés en une minute") et règles de transmission de l'information

		(exemple : rapport envoyé par courriel ou notification sur une instance web de Splunk)
Wiki du projet	Matricis	Ensemble d'informations recueilli tout au long du projet. Le squelette de celui-ci est présenté en annexe 3.
Plan de projet	Alain April	Document réalisé au début du projet qui détail les objectifs du projet et comment celui-ci est organisé
Matériel de présentation orale	Alain April, Matricis	Diapositives de la présentation orale.
Évaluation des pairs	Alain April,	Évaluation du rendement des coéquipiers
Rapport technique	Alain April, Matricis	Document détaillant l'ensemble du projet

2.3 Risques

Risque	Impact	Probabilité	Mitigation / atténuation
Manque de régularité dans la disponibilité des membres (exemple : période d'intras et de finaux, obligations professionnelles, etc.)	important	élevé	Travail en binôme ("*extreme-*programming"), les chances d'un manque de disponibilité simultanée des deux personnes sont limitées : des cours différents sont suivis par les membres.
Manque d'expertise de la part des membres du projet	moyen	élevé	Effort important mis sur le travail de documentation sur les outils : en particulier Splunk et Thingworx
Dépassement des limitations de la licence Splunk (en termes de la quantité de données ingérées et nombre de requêtes exécutées avec Splunk). Cela mène à un blocage de la licence, donc une impossibilité d'utiliser Splunk.	élevé	faible	Réaction rapide en cas dépassement ponctuel des limites : un message d'erreur est fourni. Cesser les opérations en cours et contacter le service client.
Implication des utilisateurs : on a besoin d'eux pour définir les objectifs du projet	élevé	faible	Réunion hebdomadaire entre eux.

Expérience du gestionnaire de projet avec ce type de projet	moyen	élevé	Demande régulière de retour de la part de l'enseignant

2.4 Techniques et outils

L'un des principaux outils utilisés par l'équipe pour ce projet est la plateforme Azure DevOps. Celle-ci permet de réaliser la gestion de la base de connaissance du projet sous le format d'un Wiki (voir squelette en annexe 3). Il nous permet aussi de faire la gestion de nos « stories » ainsi que nos Sprints à l'aide d'une approche Kanban. Pour finir, celui-ci va aussi aider à réaliser le stockage centralisé des versions du code source à l'aide de l'outil Git.

En ce qui concerne la réalisation du plan de projet et le rapport de projet, nous avons utilisé l'outil collaboratif Google Drive. Cela permet à toute l'équipe de travailler facilement sur le même document en simultané.

Pour ce qui est attrait à la communication, l'équipe a fait appel aux outils Slack, Skype entreprise ainsi que le Kanban est le wiki de Azure DevOps. Le détail de l'utilisation de ces outils est expliqué dans la prochaine partie.

2.5 Communications

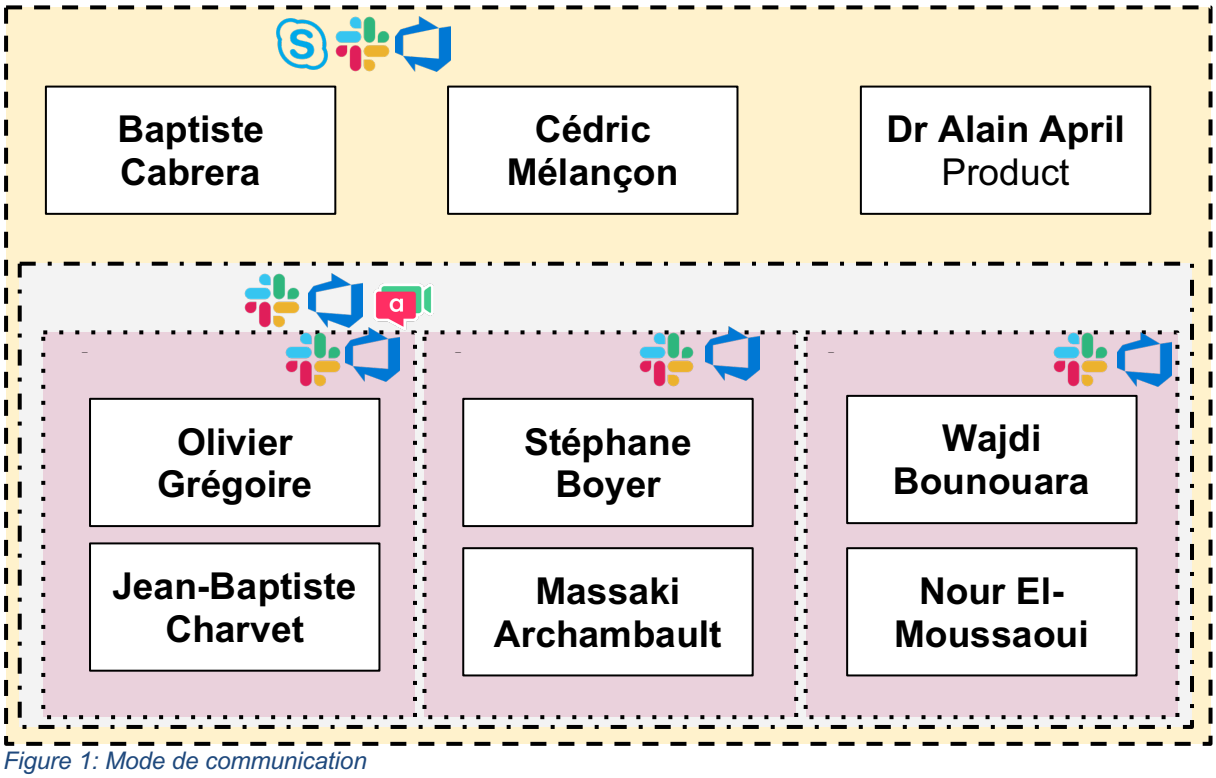


Figure 1: Mode de communication

Afin de réaliser une communication efficace, plusieurs canaux de communication ont été utilisés.
Pour pouvoir communiquer avec le comité de direction au quotidien, l'outil utilisé est slack (permettant de parler de façon asynchrone). À cela s'ajoute Azure DevOps qui a été connecté avec le Slack afin d'envoyer toutes les modifications qui sont réalisées sur les stories

présentes dans le Kanban. Cela permet à l'ensemble des membres du comité de direction de voir en direct toutes les évolutions sur le projet depuis un seul endroit. L'outil Skype entreprise a aussi été utilisé afin de réaliser nos rencontres SCRUM hebdomadaires. Cette rencontre permet de faire le point sur les différentes avancées et les bloquants. Le compte rendu de chacune de ces rencontres est présent dans l'annexe 1.
Afin de réaliser les communications en lien avec le projet, les membres du comité de projet ont utilisé un slack de projet séparé de celui utilisé dans le cadre du comité de direction afin d'éviter d'envoyer des messages qui seraient inutiles pour certains de ses membres. Azure DevOps est aussi utilisé à ce niveau-là afin de faire de réaliser le suivi des différentes stories sur le Kanban et pour partager de l'information entre les équipes à l'aide du wiki.
Chacune des équipes dispose aussi d'un channel privé dans le Slack de projet afin de pouvoir communiquer sur leur partie du projet sans avoir à communiquer avec tous les membres du comité de projet.

3. Présentation de la solution

3.1 Liste des possibilités

Q4-I2 Formuler les solutions: Décrivez les options de solutions et celle qui a été préférée: Formuler diverses solutions pour un problème de génie des technologies de l'information complexe

3.1.1 Les fonctionnalités à remplir

i/ Collecter les données

Il existe un grand nombre de type et de génération de machines industrielles, chacune dotée de capteurs et actionneurs différents (signaux analogiques, numériques, dont certains au protocole propriétaire, etc.). Leur collecte représente un enjeu en soi.

ii/ Uniformiser et visualiser les données

La collecte de données génère des données aux formats différents. Il faut donc uniformiser ces formats afin de pouvoir les traiter et visualiser avec un unique outil.

iii/ Créer une logique d'affaire en fonction les données

Une fois que les données sont visualisable et exploitable, il faut créer une logique d'affaires pour répondre aux enjeux présentés en introduction. Cette logique consiste à créer des actions qui se déclenchent selon des paramètres détectés dans les données : des alertes, des dashboards, etc.

3.1.2 Implémentation de toutes les fonctionnalités : Intelligence Industrielle[8]

Cette stratégie est celle de la startup montréalaise “Intelligence industrielle”, cette stratégie présente les **avantages** suivants :

- Seul le minimum de fonctionnalités est implémenté. Cela permet d’avoir un interface légère et simple à utiliser.
- Possibilité de personnalisation parfaitement adaptée à la stratégie de l’entreprise qui fournit un service en industrie 4.0.
- Pas de frais de licences d’outils propriétaires.

Cependant, cette solution amène des **contraintes** :

- Le coût en effort d’implémentation et maintenance des produits correspondants aux solutions. Créer une telle solution ne permet pas d’entrer instantanément dans le marché, il faut prendre en compte le temps de développement de la solution avec les clients.
- La hausse exponentielle de la complexité et de la gestion compatibilité entre les versions des solutions.
- Il est difficile de communiquer avec les machines dont les protocoles de communication propriétaire et liés à un produit logiciel vendu avec la machine.
- Le besoin de développer des connecteurs aux outils déjà existants que les usagers utilisent : PGI, GRC, etc.

3.1.3 Usage d’outils existants

C’est la solution qui a été sélectionnée pour ce travail. Elle est donc présentée plus précisément dans sa description. Les **avantages** et **contraintes** sont les exacts inverses du choix de l’implémentation manuelle de toutes les fonctionnalités présentées dans la partie précédente.

i/ Splunk[9]

Splunk est un outil qui peut remplir plusieurs rôles : la collecte, l’uniformisation et visualisation, la création d’une logique d’affaires. Dans le cadre de ce travail, il est utilisé pour toutes ces fonctions.

ii/ Telegraf[10]

Telegraf est la composante qui se charge de la collection de **métriques**. À la différence d'un **log**, une métrique est généralement une valeur numérique qui décrit l'état d'un système à un moment donné. Telegraf est un projet open source, maintenu par InfluxData et disponible sur GitHub.

[8] http://intelligenceindustrielle.com/fr/accueil/
[9] https://www.splunk.com/
[10] https://github.com/influxdata/telegraf

iii/ Thingworx[11]

C'est un outil IIoT, IoT adapté au contexte industriel, qui permet de se connecter aux machines dans une usine. La complexité de cette connexion est donc entièrement gérée par Thingworx.

3.2 Description de la solution choisie

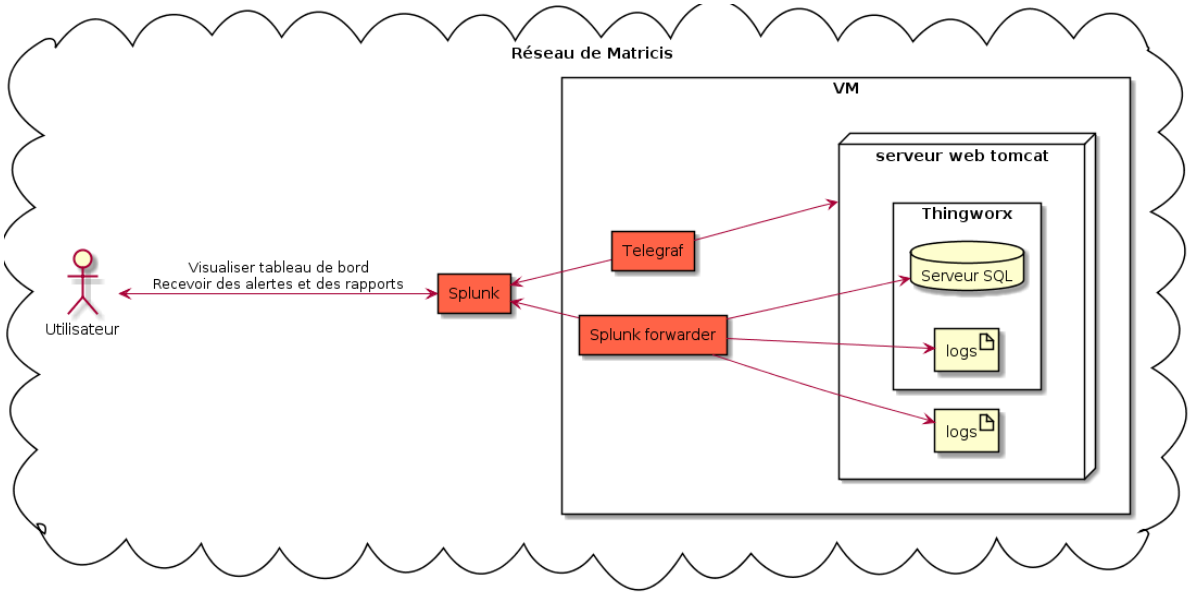


Figure 2: Vue globale de la solution

Au final l'équipe a fait le choix d'utiliser *Telegraf* et *Splunk forwarder* parce que chacune de ces technologies offrent des avantages différents.
Telegraf a été utilisé pour récupérer les métriques sur le serveur Windows directement. En ce qui concerne les autres parties, l'utilisation de Splunk forwarder était plus adapté puisque des fichiers logs sont générés directement par les différents logiciels.

3.2.1 Phase d'exploration

i/ Thingworx

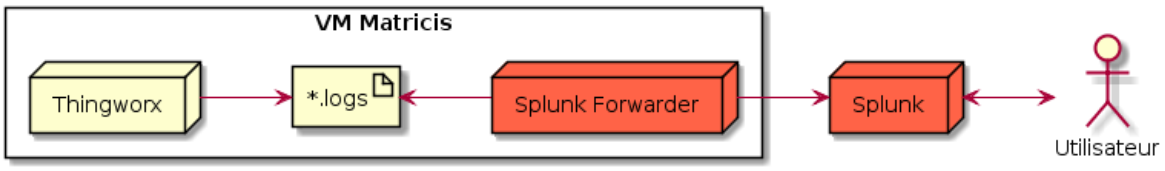


Figure 3: Architecture pour réaliser la récupération des données Thingworx

[11] https://www.ptc.com/en/products/iiot/thingworx-platform

Thingworx enregistre son activité sur plusieurs fichiers de logs à l'endroit où il est installé. Pour récupérer les données, les logs présents sur la VM, Splunk "universal forwarder"[12] est utilisé. Son installation et utilisation est décrite dans la section interfaçage.

ii/ SQL Server

Microsoft SQL Server est la base de données relationnelle utilisée par Thingworx pour assurer son bon fonctionnement. La base de données est hébergée sur une machine serveur de Matricis, et supporte l'instance de Thingworx. SQL Server possède des fichiers de logs générés automatiquement.

iii/ Tomcat

Tomcat est une implémentation open source des technologies java qui permettent de créer des services web, soient Java Servlet, Java Server Page, Java Expression Language et Java WebSocket. Une alternative courante, mais propriétaire, de Tomcat est Oracle WebLogic.

Dans l'architecture du système, Tomcat est une plateforme de type serveur web sur laquelle Thingworx est installé. Notamment, la JVM est gérée par Tomcat et les logs d'accès sont également gérés par lui.

iv/ Eventvwr

Eventvwr est le service de collecte de logs et d'évènements de Windows. Il permet, entre autres, d'enregistrer les tentatives de connexion sur le serveur via le service de bureau à distance ou de partage de fichier, le démarrage ou l'arrêt d'un service, l'exécution de tâches planifiées, etc. Windows expose également une API qui permet à n'importe quelle application d'enregistrer ses logs de manière centralisée et standard, bien que typiquement les applications qui ne sont pas développées par Microsoft n'utilisent pas ce système.

3.2.2 Phase d'interfaçage

i/ Splunk Universal Forwarder

Un forwarder Splunk est un outil permettant de collecter des données depuis n'importe qu'elle source et des les envoyer vers un autre forwarder ou bien une instance de Splunk.

Nous avons choisi d'utiliser un forwarder universel pour l'avantage qu'il présente sur l'autre forwarder disponible, le "heavy forwarder".

Un "heavy forwarder" permet de créer une logique de collecte de données : choix de destination (indexer, autre forwarder) en fonction du type de source ou évènement. Cette fonctionnalité n'est pas utile dans le contexte de ce PFE : nous ne gérons qu'une seule instance de Splunk et nous n'avons besoin que d'un seul forwarder comme les données

[12] https://docs.splunk.com/Documentation/Forwarder/7.3.0/Forwarder/Abouttheuniversalforwarder

proviennent de la même VM. De plus, il demande plus de ressource matérielle que les autres forwarders.

À noter qu'il existe une version obsolète du forwarder universel, le "light forwarder"[13].
Pour résumer, ce forwarder présente les qualités d'être léger et facilement scalable.

ii/ Telegraf

À l'instar du forwarder qui collecte des données et les envoie à l'instance de Splunk, Telegraf est un outil open source de collecte de métriques et les envoie nativement à Splunk. Il permet aussi d'agréger et créer une logique d'affaire avec ces métriques avant leur envoi : calculer des valeurs (valeurs extrémales, moyennes, médianes et autres traitements statistiques).

Telegraf est basé sur un système de plugins qui lui permet de collecter des données de diverses sources, par exemple, la JVM, l'état de la mémoire, l'état du CPU, l'utilisation du disque, des données sur SQL Server, etc. Ceci le rend beaucoup plus puissant que le système de collection de métrique du Splunk Universal Forwarder et nous l'avons choisi pour cette raison.

Originalement, Telegraf a été développé par InfluxData[14] pour l'ingestion de donnée dans InfluxDB et ne fait pas partie de la pile technologique standard de Splunk. Néanmoins, un plug-in a été écrit pour ajouter le support pour le protocole HEC de Splunk.

iii/ Splunk DBConnect

Splunk dispose d'un module permettant une connexion directe à des bases de données afin d'en extraire des événements, à l'aide de requêtes SQL lancées de manière régulière. Ce module a été utilisé pour extraire les métriques d'utilisation de la base de données de Thingworx, qui utilise SQL Server. Le module utilise un connecteur standard JDBC pour se connecter à tous types de bases de données. Voici ci-dessous un exemple d'url JDBC utilisé pour la connexion à la base de données:

```
jdbc:sqlserver://10.98.79.58:1433;databaseName=thingworx;selectMethod=cursor
```

Une fois la connexion mise en place, le Splunk DBConnect propose une interface pour utiliser une requête SQL et extraire des données, tout en pré visualisant les résultats.

[13] https://docs.splunk.com/Splexicon:Lightforwarder
[14] https://www.influxdata.com/

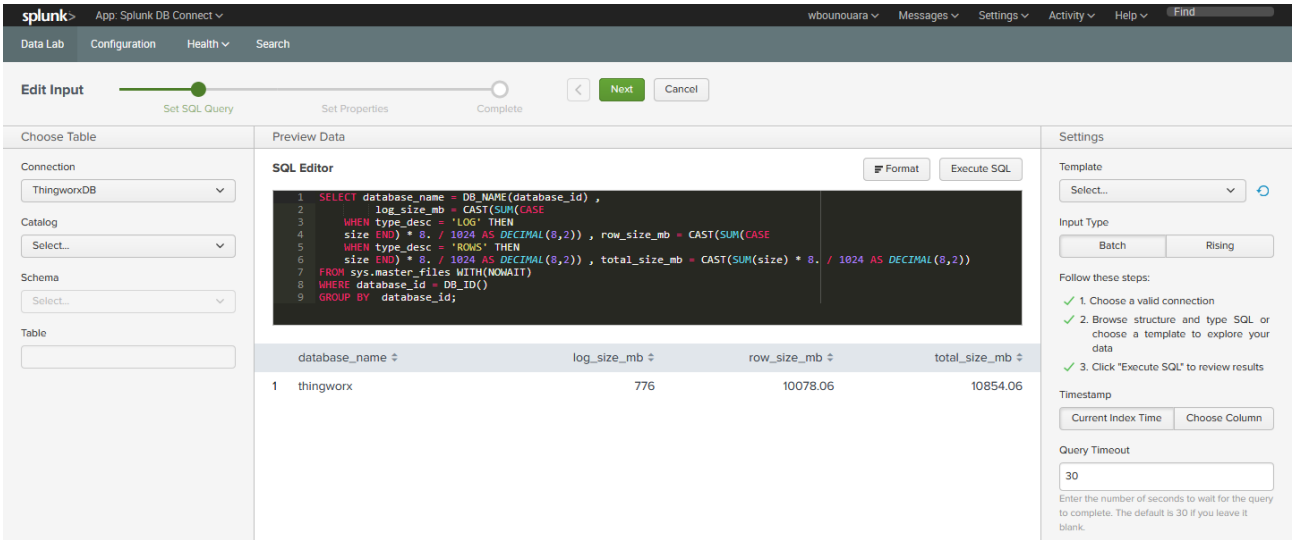


Figure 4: Interface de sélection des champs à importer

Il est possible, une fois la requête construite, de paramétrer le nombre de résultats maximums à récupérer, ainsi que la fréquence d'exécution de la requête.

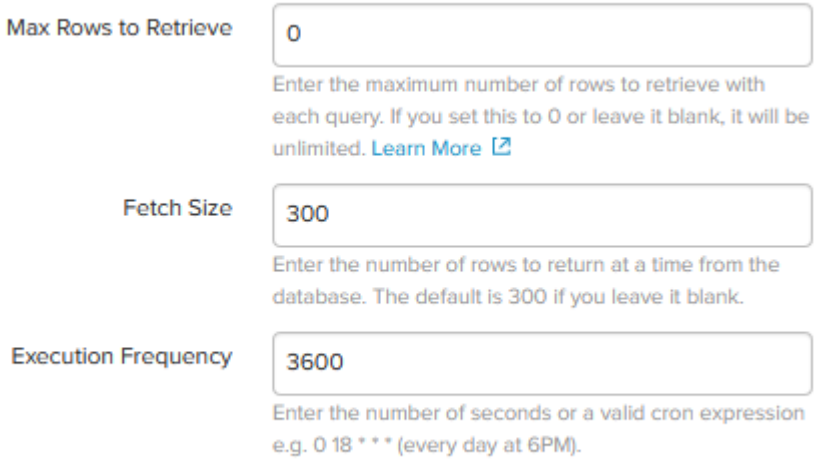


Figure 5: Paramètres de la requête à exécuter

Finalement, on observe les événements récupérés.

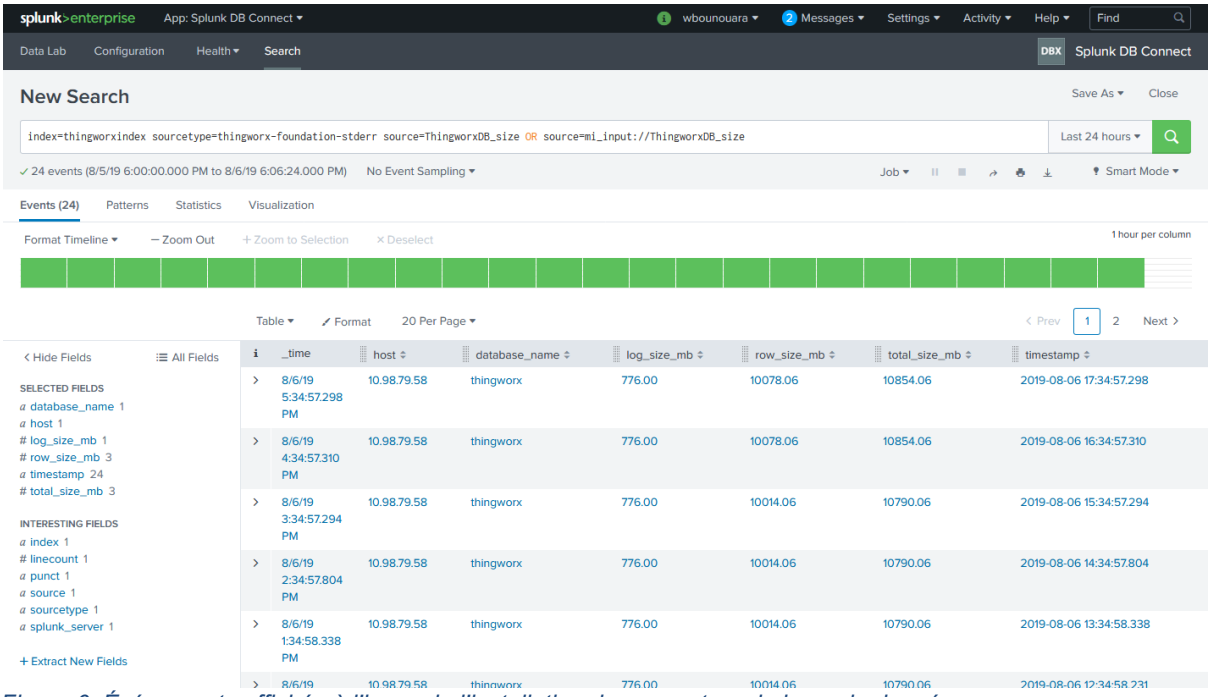


Figure 6: Événements affichés à l'issue de l'installation du connecteur de base de données

iv/ Extraction des données des logs de type log4j

Tomcat et Thingworx génèrent des logs avec le cadriciel log4j, qui permet d'obtenir un format standardisé de log avec des champs standards pour le temps d'un log ainsi que son niveau d'importance (FATAL ERROR WARN INFO DEBUG TRACE)[15]. De plus, ce cadriciel permet de générer des champs personnalisés adaptés au besoin de l'outil, Tomcat et Thingworx dans ce cas. Pour cette raison, Splunk ne propose pas d'outil natif d'extraction de ces champs et une stratégie d'extraction de ses valeurs a été choisie.
Les champs à extraire :

- Champs généraux : messageType (loglevel), origin, instance, user, session, thread, message
- Champs spécialisés aux objets connectés à Thingworx : thingName, thingTemperature, thingDescription, thingLastConnection, thingReportingLastEvaluation, thingTags, thingTemplate

Pour les extraire, des regex sont utilisées.
Exemple : [O:\s+(?P<origin>()|(\w+.){0,5}\w+)]

3.2.3 Phase d'exploitation des données

Maintenant que les données sont collectées et normalisées dans Splunk, il faut chercher comment elles peuvent être utilisées. Splunk permet de les utiliser de trois manières différentes.
La première utilisation qui a été faite des données est la création d'alertes. Celles-ci permettent de détecter en temps réel si des valeurs sont anormales et de faire des actions automatiquement.

[15] https://en.wikipedia.org/wiki/Java_logging_framework

La deuxième utilisation est la création est les tableaux de bord. Ceux-ci permettent d'avoir une vision d'ensemble du système en temps réel.
Pour finir la dernière utilisation qui peut être faite des données est la création de rapports. Ils permettent de compiler toutes les données sur une période statique de temps (trimestre, période d'exercice d'une entreprise, etc.). Cette dernière façon d'utiliser les données n'a pas été utilisée dans ce projet parce que l'objectif de ce projet est plus d'étudier les données produites pas Thingworx en temps réel pour pouvoir réagir en cas de problème plutôt que de réaliser des rapports statiques.

i/ Thingworx

Maintenant que Splunk récupère les fichiers log de Thingworx, il faut pouvoir générer des données qui ont du sens afin de pouvoir tester l'ensemble. Cette problématique a été présentée à Cédric Mélançon qui a réalisé une application en ligne de commande qui communique avec Thingworx en utilisant son API.

Elle permet de simuler un objet qui envoie des températures aléatoires au serveur Thingworx. Elle permet aussi de simuler une connexion et déconnexion d'un objet à Thingworx.

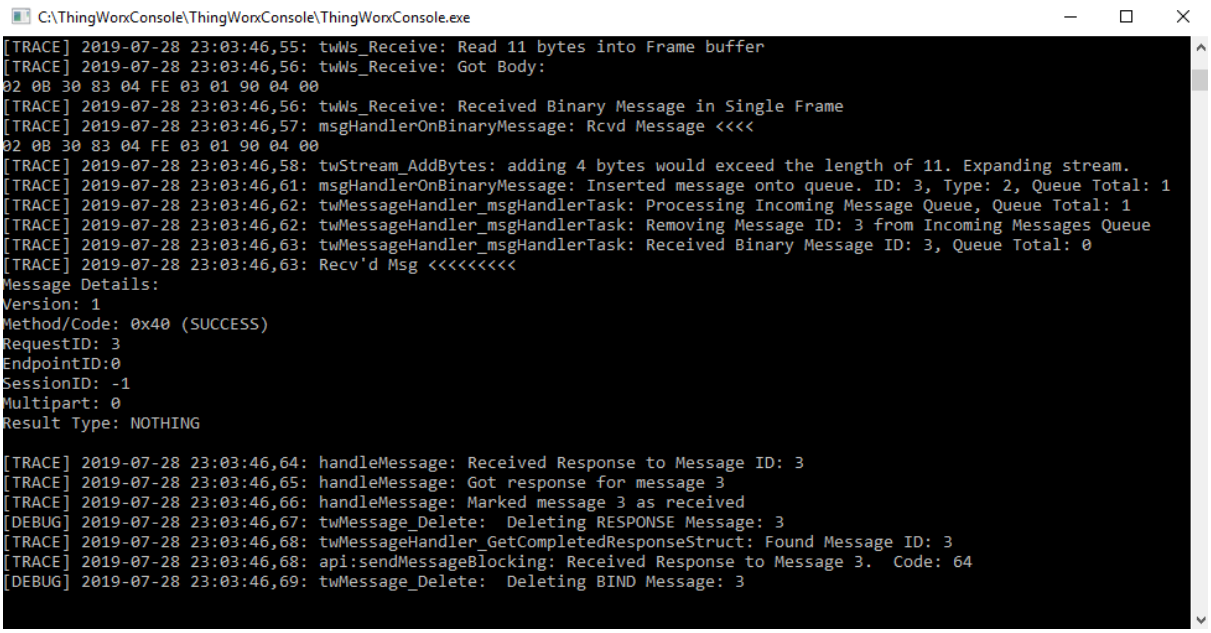


Figure 7: Application de simulation d'objet Thingworx réalisé par Cédric Mélançon

Pour l'utiliser, il faut commencer par connecter un nouvel objet en faisant appel à la commande "Connect". Une fois que la connexion est établie, la commande "Send" envoie la température aléatoire. La dernière commande est "Disconnect" et permet de fermer la communication entre l'objet et Thingworx.

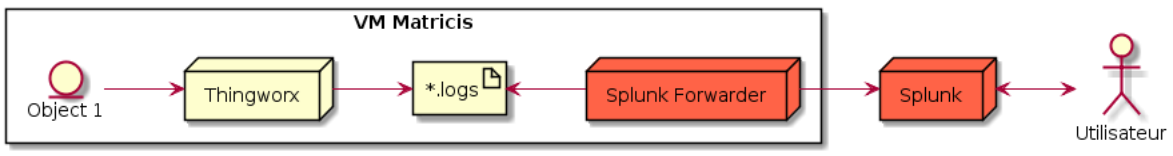


Figure 8: Architecture pour réaliser la récupération des données liés à l'objet simulé

a/ Alertes

Des détails d'implémentation sont disponibles en supplément dans les annexes. Les alertes 1 à 5 correspondent à l'état de l'application Thingworx. Les alertes 5 à 7 correspondent à l'état d'objets connectés à Thingworx.

Définition objet : une machine industrielle (bras robotisé, presse hydraulique, machine à extrusion, etc.) connectée à Thingworx. Pour cette POC, les machines sont "mockées" par une application pour simuler une machine constituée d'un unique capteur de température.

A1 : trop d'échec de connexion des utilisateurs

- niveau de sévérité : moyen
- Description : Plus de 3 échecs de connexion d'utilisateur en 5 min
- **Valeur pour utilisateur final** : Pour le gestionnaire de Thingworx, il est important de savoir que les utilisateurs échouent à se connecter.

A2 : compte utilisateur bloqué après des tentatives infructueuses de connexion

- niveau de sévérité : élevé
- Après 3 mauvais mots de passe, un compte utilisateur est bloqué et demande un reset de la part d'un administrateur
- **Valeur pour utilisateur final** : Pour le gestionnaire de Thingworx, recevoir une alerte pour un blocage de compte lui permet de le débloquer rapidement après avoir vérifié avec l'usager bloqué la raison du blocage.

A3 : Alerte attaque DDOS

- niveau de sévérité : critique
- Plus de 100 requêtes WEB par minute à la page de login de Thingworx
- **Valeur pour utilisateur final** : Pour réagir à une telle attaque, un sysadmin a besoin d'en être conscient, il peut également lier à cette alerte l'exécution d'un script pour réagir à l'attaque.

A4 : échec de connexion d'objets

- niveau de sévérité : moyen
- Trop d'échecs de connexion d'objets à Thingworx en 5min
- **Valeur pour utilisateur final** : Pour le gestionnaire de Thingworx, il est important de savoir la raison de l'inactivité de certains objets, une raison peut être une perte de connexion et un échec de reconnexion.

A5 : Objet inactif

- niveau de sévérité : high
- Pas de donnée envoyée depuis plus de 5 min

- **Valeur pour utilisateur final** : Entendu lors de la conférence d'Intelligence Industrielle[16] de mi-juillet pour le cours ENT202[17] au Centech[18] : "En moyenne, les machines sont utilisées seulement 35% du temps disponible (maintenance, machine oubliée dans un coin, etc.). Beaucoup d'acteurs du milieu industriel font l'erreur d'acheter plus de machines pour augmenter la production au lieu d'augmenter le temps d'usage de chacune."

A6 : Changement de température rapide

- niveau de sévérité : critique
- **Valeur pour utilisateur final** : Cela permet de détecter des incendies, un capteur défectueux ou un risque de casse imminent.

A7 : Dépassement seuil de température

- niveau de sévérité : critique
- **Valeur pour utilisateur final** : idem A6

b/ Dashboard

D1 : état d'une usine

Ce dashboard donne une vision d'ensemble de l'état d'une usine sur l'intervalle de temps sélectionnable. Le nombre d'objets connectés et inactifs y est clairement mis en avant. De même, on voit la variation de température au cours de l'intervalle de temps de l'objet de test "PFE_Splunk_TestDevice", ainsi que son évolution au cours du temps.

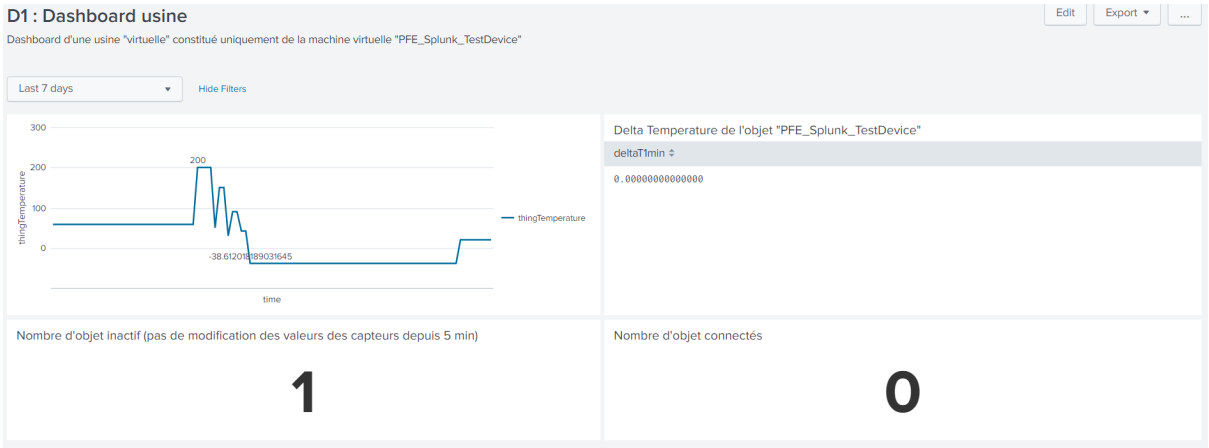


Figure 9: Dashboard représentant l'état d'une usine

D2 : état de Thingworx

[16] http://intelligenceindustrielle.com/fr/accueil/
[17] https://www.etsmtl.ca/etudes/cours/ENT202
[18] https://centech.co/

Quant à lui, ce dashboard présente l'état de l'application Thingworx sur l'intervalle de temps sélectionnable : le nombre de connection effectué par jours, le nombre de connexion effectuées ar utilisateurs, la liste des utilisateurs qui ont été bloqué, etc.

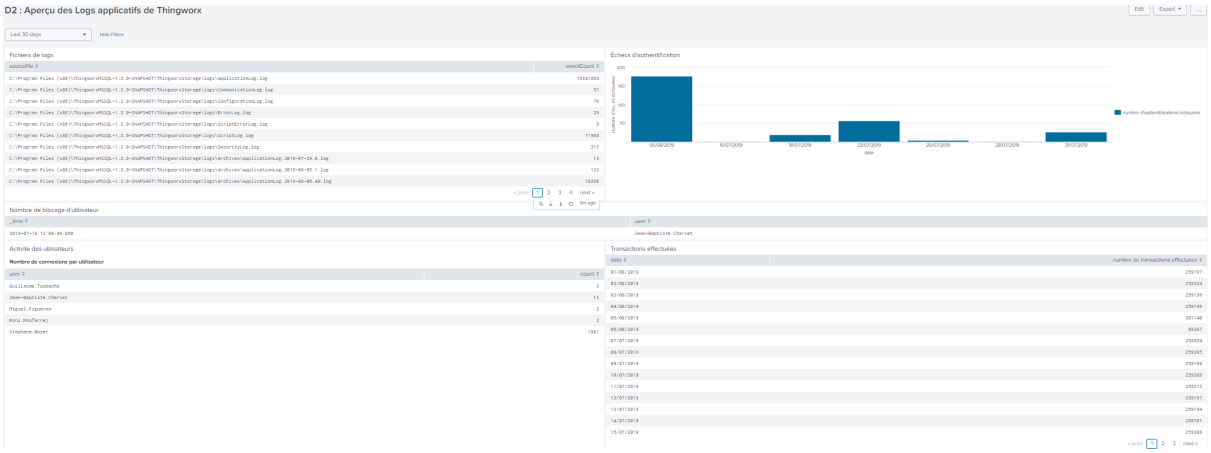


Figure 10: Dashboard représentant l'état applicatif de Thingworx

c/ Rapport

Ils consistent à convertir les dashboard en "dashboards statique" pour un intervalle de temps prédéfini. Par exemple, on peut générer un rapport par dashboard par mois, trimestre, période d'exercice, etc.

ii/ Server SQL

En ce qui concerne la portion SQL Server , les logs d'erreurs SQL sont extraits à l'aide du Splunk Forwarder à partir du dossier C:\Program Files\Microsoft SQL Server\MSSQL*n*.MSSQLSERVER\MSSQL\Log (ou *n* est la version). Dans ce dossier , on retrouve principalement :

- Le fichier ERRORLOG qui contient les erreurs liées à la plateforme (rotation dans ERRORLOG.1, ERRORLOG.2, …)
- Le fichier SQLAGENT.OUT qui contient tous les logs liés à l'agent (rotation dans SQLAGENT.1, SQLAGENT.2, …)

Ces erreurs automatiquement générées sont ensuite indexées dans Splunk sous un index appelé "sql" qui sera utilisé afin de produire les visualisations (Alertes , rapports ou tableaux de bord) pertinentes dans le cadre d'une preuve de concept.

Alertes

A1 : Espace disque plein dans le schéma thingworx

- niveau de sévérité : Élevé.
- Une alerte est déclenchée en temps réel lorsqu'il manque de l'espace disque dans le schéma tw dans la base de donnée SQL.
- L'alerte est envoyée à l'utilisateur concerné dans Splunk.

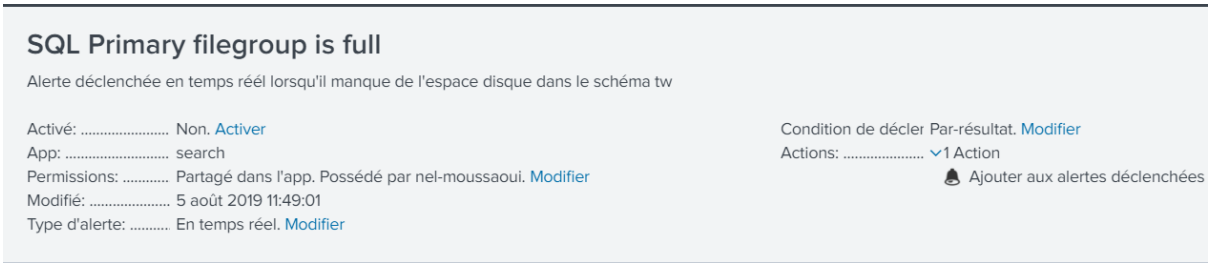


Figure 11: Alerte déclenchée lorsque l'espace disque est plein

A2 : Mot de passe tw admin dans SQL est expiré

- Niveau de sévérité : Élevé
- Une alerte est déclenchée en temps réel lorsque le mot de passe du compte twadmin est expiré
- Dans le cas échéant, l'alerte est envoyée au responsable du compte

Figure 12: Alerte déclenchée lorsqu'un mot de passe est expiré

Tableau de bord

Le tableau de bord configuré dans le cadre de ce projet illustre en temps réel l'évolution des erreurs SQL répertoriés par type sous forme de graphique temporel et tableau standard. On retrouve en ordre les éléments de visualisation suivants :

D1 : Logs d'erreurs SQL

- Évolution du log d'erreur : Graphique temporel qui illustre l'évolution des logs d'erreurs dans le temps par type.
- SQL errors: Tableau qui se rafraîchit toutes les 30 secondes pour illustrer les dernières erreurs du log SQL
- Nombre de mots de passe expiré: Nombre de mots de passes twadmin (compte admin du schéma Thingworx SQL) expiré cette semaine.

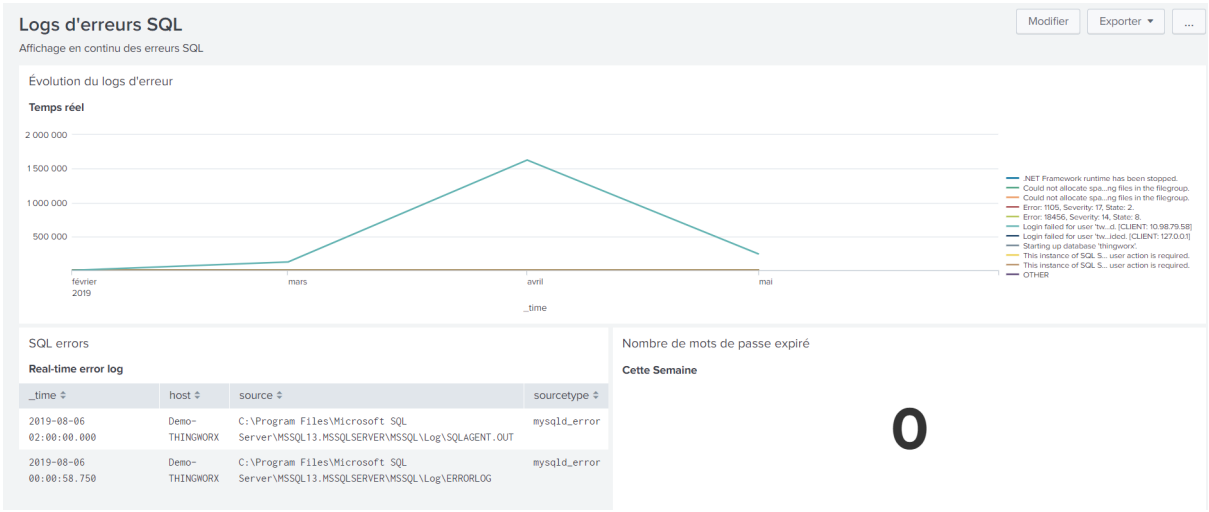


Figure 13: tableau de bord représentant les logs d'erreurs SQL en temps réel

Rapport

R1: Log d'erreur SQL

- *SQL error log* - Planification d'envoi d'un rapport du log d'erreur complet - 1 fois par mois aux utilisateurs concernés.

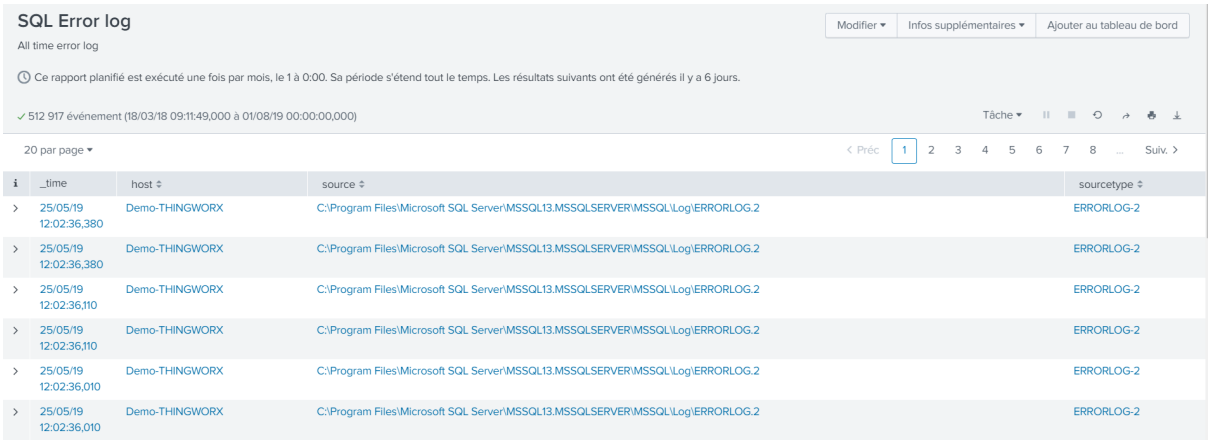


Figure 14: Rapport du log d'erreur complet planifié une fois par mois

R2: Erreur SQL par type

- nom: *Erreur SQL par type* - non planifié pour le moment.
- Sélecteur de temps disponible pour planifier l'envoi de ce rapport, mais aussi pour changer le taux de rafraîchissement du graphique.

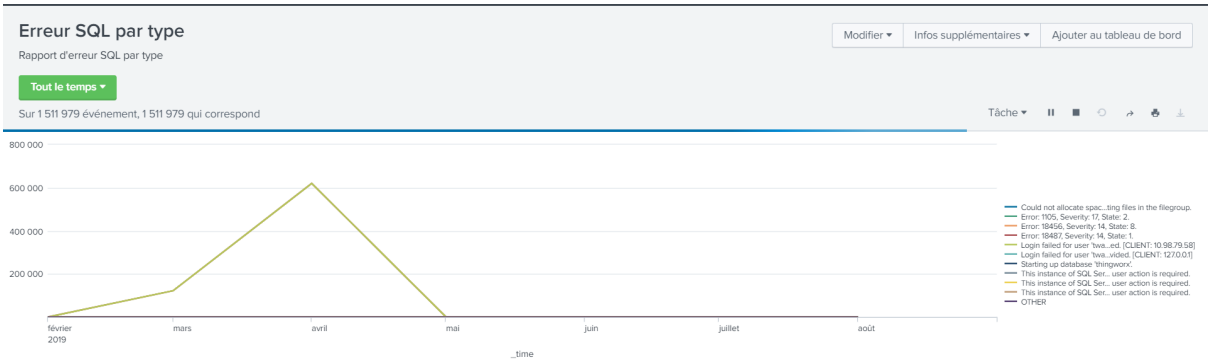


Figure 15: Rapport représentant les erreurs SQL par type sur un graphique temporel

iii/ Tomcat

Les alertes, tableaux de bord et rapports pour Tomcat se rapportent généralement à la performance ou la disponibilité de Thingworx. Si Thingworx est inaccessible ou la performance de celui-ci est dégradé, ces outils permettent d'identifier le problème.

Alertes

A1: Temps de réponse du site

- Niveau de sévérité: Haute
- Cette alerte est lancée si le site prend beaucoup de temps à répondre.
- Permet de savoir si les performances du site sont dégradées, ce qui peut indiquer un autre problème sous-jacent.

A2: Disponibilité du site

- Niveau de sévérité: Critique
- Cette alerte est lancée si l'accès au site est impossible depuis le serveur Splunk.
- Permet de savoir si le site est accessible pour les utilisateurs.

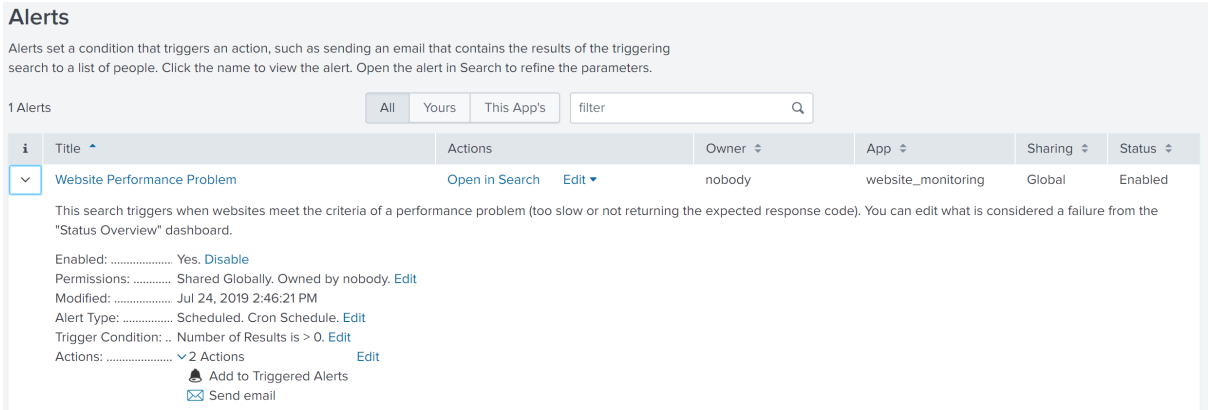


Figure 16: Alerte de disponibilité du site web

Tableau de bord

- **D1: Tomcat Thingworx Status**

Ce dashboard permet d'afficher l'état du fonctionnement du site web. Il affiche le statut du site, le dernier temps de réponse du site et la moyenne des temps réponses enregistrées ainsi que les extrêmes et permet un retour dans le temps des temps réponses du site. Il permet en un coup d'oeil de connaître l'état du site Thingworx.

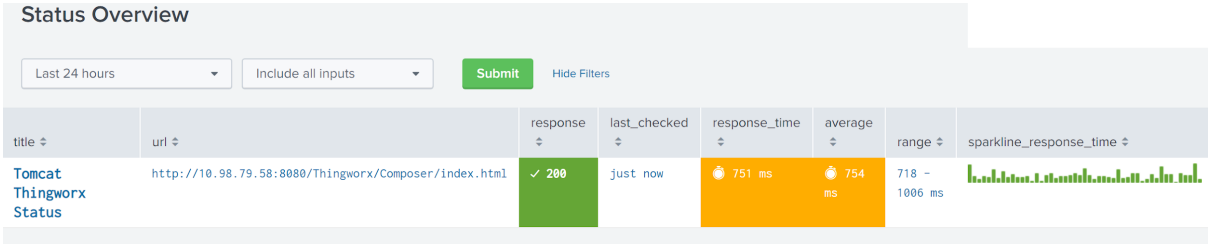


Figure 17: D1: Tomcat Thingworx Status

Rapport

R1: Logs d'erreur du site

Ce rapport donne une vue d'ensemble des logs générés par Tomcat et donne un graphe du nombre de logs par sévérité. Ceci permet d'identifier une anomalie dans le flux de logs en provenance de Tomcat d'un coup d'œil.

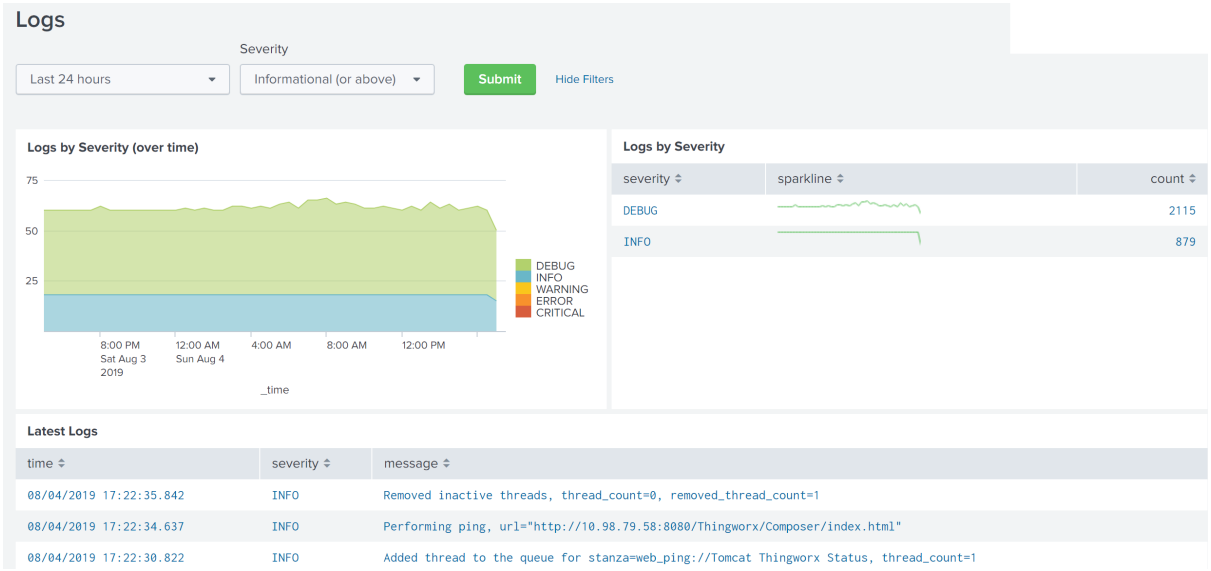


Figure 18: Rapport des logs d'erreur du site

R2: Status History

Ce rapport affiche le temps moyen de réponse du site, le temps le plus élevé, son niveau de disponibilité et le nombre de non-fonctionnalités du site pour les dernières 24 heures et il est mis à jour de façon continue. Ceci permet d'avoir une vue d'ensemble sur la disponibilité et les performances de Thingworx.

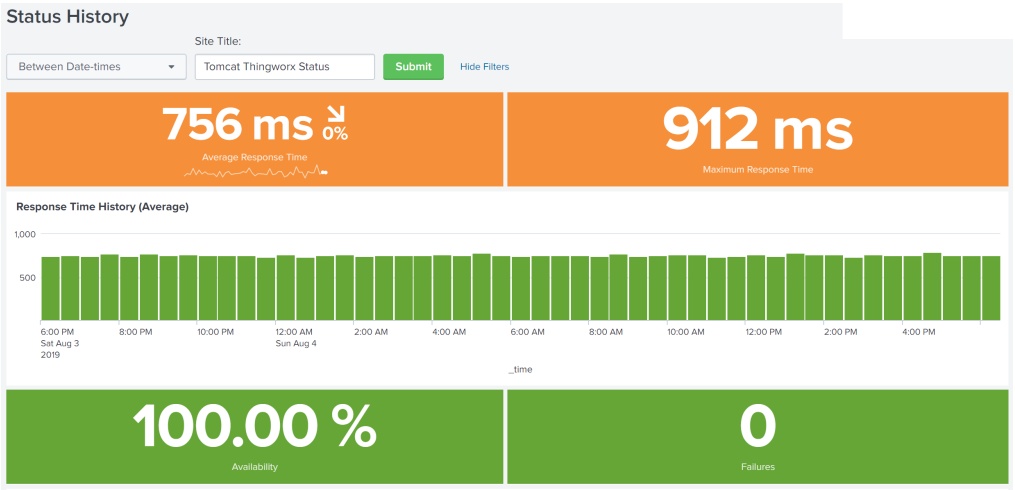


Figure 19: Statut du site web

3.3.3 Prise en compte des enjeux sociétaux pendant la conception de la solution

Dans la présentation du contexte du problème traité, les enjeux résolubles avec la solution ont été présentés. Ici sont présentés les enjeux pris en compte pendant la conception de la solution.

i/ Les enjeux économiques, environnementales et de santé des membres de l'équipe

En optimisation **la performance du traitement de données** de la solution, une économie des ressources matérielles utilisées a été réalisée. Concrètement, les analyses en temps réels des données (alertes et indexages) ont été ajustées et limitées afin d'éviter de surcharger la VM. Cela permet d'éviter d'avoir besoin d'augmenter les ressources de la VM, donc d'augmenter son coût opérationnel, directement lié à sa consommation électrique. En fait, ce problème est récurent en big data : plus le nombre d'opérations de traitement de données effectuées concomitamment est grand, plus le besoin en ressource et donc le coût opérationnel augmente.

À plus petite échelle, nous avons **augmenté le thermostat des climatiseurs** de la bibliothèque en salle de réunion, afin de ne pas provoquer de choc thermique les jours de forte chaleur. En effet, la climatisation n'est pas censée créer un écart de température de plus de 7 degrés Celsius avec l'extérieur[19] pour éviter des risques sur la santé. Cela permet également de baisser la consommation électrique de l'école et par conséquent son impact environnemental.

De même, nous n'avons pas seulement **évité des déplacements véhiculés** aux membres les plus éloignés de l'école pour leur gagner du temps, mais aussi pour limiter l'impact environnemental de notre travail.

Enfin, l'impact le plus conséquent de notre travail est au niveau des possibilités amenées par notre solution. Rendre les techniques de l'industrie 4.0 accessible aux usines permet d'**optimiser leurs chaînes de production**. Cette optimisation signifie pouvoir produire plus sans moyens supplémentaires (pas de coût opérationnel et environnemental

[19] Étude de l'association QueChoisir : https://www.santemagazine.fr/actualites/actualites-sante/climatiseur-gare-au-choc-thermique-425953

supplémentaire), produire autant avec moins de ressources (diminuer le coût opérationnel et environnemental).

Améliorations et suite du projet

Limites du système et solutions

Durant le développement de ce projet, nous nous sommes heurtés à la limite du nombre de recherches concourantes de Splunk, qui était configuré à un maximum de 3 sur notre instance. Ceci nous empêchait d'activer toutes les alertes et d'afficher tous les dashboards en même temps.

La solution est d'augmenter le nombre maximum de recherches en temps réel dans la configuration de splunk. De plus il est possible d'ajouter un Search Heads dédié aux alertes, de dimensionner ses ressources matérielles afin de pouvoir exécuter les recherches en temps réel effectuées par les alertes. Ceci a été documenté dans le wiki du projet dans Azure Devops et pourrait être appliqué dans un développement futur.

Améliorations suggérées

Pour la durée de notre projet, il n'y a jamais eu de capteurs réels de connectés au système, uniquement un thermomètre simulé. Pour que le système développé soit vraiment utile, il faudrait connecter de vrais appareils et ajuster les alertes, tableau de bord et rapports pour les intégrer.

Il serait également intéressant d'explorer la capabilité de Splunk d'exécuter automatiquement des scripts au déclenchement d'une alerte. Ceci permettrait d'ajouter une capacité de réparation automatique au système. Par exemple, si Thingworx ne répond plus, le service Tomcat pourrait automatiquement être redémarré pour rétablir le service.

Conclusion

Pour conclure, l'équipe Splunk s'entends pour affirmer que ce projet est un succès en termes d'atteinte de ses objectifs principaux (Voir annexe 7). Partie d'une base de connaissance minime sur les fonctionnalités de l'outil, l'équipe s'est mobilisé dès le départ afin de suivre la formation disponible sur Splunk. L'apprentissage et la découverte de ce logiciel s'est donc fait par expérimentation. Par la suite, diverses preuves de concept ont été formulés et présentés à la demande de Matricis afin d'exploiter et démontrer les capacités de Splunk une fois intégré dans un contexte industriel. En tant qu'équipe Agile, la priorité principale était de maintenir une bonne relation avec Matricis en remettant les livrables demandés à la suite de l'explicitation des besoins et requis d'affaires qu'a formulé le client.

Sur une autre note, il y a plusieurs leçons apprises ainsi que des améliorations qui peuvent être apportés dans le cadre de projet. En ce qui concerne les leçons apprises, les membres de l'équipe ont été mis dans un contexte réel de l'industrie actuel ou un ingénieur est apporté à faire l'intégration d'un nouvel outil (SPLUNK) au sein de l'entreprise sans nécessairement avoir les connaissances techniques sur celui-ci. Ceci dit, plusieurs aspects sont à prendre en considération par un ingénieur dans un tel projet comme la communication adéquate avec les parties prenantes, la bonne formulation et explicitation des besoins et exigences, la gestion des contraintes et des limitations de l'outil et surtout la définition de la portée d'un tel projet.

Annexes

Annexe 1: Comptes-rendus des rencontres SCRUM

Compte-rendu 2019-07-25

Avancée et Enjeux :

- @Massaki : Trim de l'index Tomcat, compactage des données (+doc), aide de @Nour pour regex
- @Jean-Baptiste + @Olivier : regex pour extraction des champs, récupération de données des devices, alertes
- @Nour: ingestion des log d'erreur SQL, rapport, alertes et dashboard
- @Wajdi : dashboards et alertes sur les métriques des BDs
- @Stéphane : onboarding de "website monitoring" pour monitoring du site et expérience client, création d'alertes

Objectifs :

- @Stéphane : Raffinage des alertes
- @Massaki : N'avancera pas (final + job), reste dispo pour aide
- @Jean-Baptiste + @Olivier : continuer sur alertes et dashboards, communication avec @Wajdi et @Nour pour identification de certaines données
- @Wajdi + @Nour : Raffinage et optimisation des alertes et dashbaords
- @Baptiste : voir config smtp, suppression des anciens logs (identifiés par @Massaki)

Autres :

- Préparation d'un démo de présentation (Doodle par @Olivier)
- Est-ce que nous conservons notre prochaine rencontre, est-ce que nous la transformons en retro, … ?

Compte-rendu 2019-07-18

Avancée et Enjeux :

- @Massaki : Ajout des log de l'event viewer vers Splunk
- @Jean-Baptiste : Documentation, extraction des champs des fichiers de logs type log4j
- @Olivier : Ébauche du rapport et sync avec @Jean-Baptiste

- @Nour: Ingestion des logs SQL Server dans Splunk, tests de création de dashboards et alertes, extraction des champs (bloquant à investiguer sur l'utilisation de champs dans la recherche), documentation
- @Wajdi : Génération de graphes sur les métriques des BDs (bloquant à investiguer sur l'utilisation dans les dashboards)
- @Stéphane : Exploration alertes et dashboards pour Tomcat

Objectifs :

- @Stéphane : Continuer les alertes et dashboards
- @Massaki : Réduire les buckets, optimisation, assister @Stéphane pour les alertes et dashboards
- @Jean-Baptiste + @Olivier : Finalisation du rapport, et alertes et dashboards
- @Wajdi + @Nour : Continuer extraction des champs, génération dashboards et alertes

Autres :

- Laissez savoir @Baptiste dès que vous aurez le lieu de votre présentation le 7 aout en 10 et 11h

Compte-rendu 2019-07-11

Avancée et Enjeux :

- @Massaki : Documentation (Telegraf, Tomcat, Splunk UF), commencer import des logs Windows dans Splunk pour correlation
- @Jean-Baptiste : Identifications des données à importer faite et documentée, normalisation des champs commencée (log4j et regex dans Splunk), exploration des données simulées de devices dans Thnigworx
- @Olivier : Suivi avec Alain pour date de présentation, demande de commentaire sur ce qui était déjà fait dans le rapport + corrections suite aux commentaires, exploration des données simulées de devices dans Thnigworx
- @Nour: Atelier de revue des US et assignation, définition du scope technique et ré-enlignement, travail avec @Wajdi sur Sql Server
- @Wajdi : Ingestion des logs et métadonnées relatives aux BD Sql
- @Stéphane : Pas d'avancée

Objectifs :

- @Stéphane : Communication avec @Massaki pour rattrapage et avancer
- @Massaki : Exploration des alertes et dashboards, transférer les données Telegraf dans un index distinct
- @Jean-Baptiste + @Olivier : Finaliser la normalisation des champs, continuer avec les données de devices, référence des indexes

- @Wajdi + @Nour : Créer des graphs sur l'évolution des méta des BDs dans le temps, présentation de données ingérées et dashboards, continuer sur l'ingestion des logs

Autres :
Laissez savoir @Baptiste dès que vous aurez le lieu de votre présentation le 7 août en 10 et 11h

Compte-rendu 2019-07-04

Avancée et Enjeux :

- @Massaki : Peu d'avancée comme prévu (déménagement + intra)
- @Jean-Baptiste : Ingestion de logs de Thingworx, création de dashboard et alertes (démo)
- @Olivier : Analyse pour le rapport
- @Nour: Peu d'avancée
- @Wajdi : Tests de requêtes pour infos de BD
- @Stéphane : Etude de la config de Tomcat et UF

Objectifs :

- @Stéphane : Communication avec @Massaki pour rattrapage et avancer
- @Massaki : Documentation
- @Jean-Baptiste + @Olivier : Avancer les données de Thingworx dans Splunk, normaliser les champs et intégrer des états de devices
- @Olivier : Essayer d'obtenir une plage horaire plus précise pour la présentation le 7 aout
- @Wajdi + @Nour : Procéder a l'ingestion des données SQL et créer dashboard
- @Baptiste : Regarder s'il y a moyen d'obtenir les logs de SQL via requêtes pour aider @Wajdi
- @Cédric : Générer des activités de devices pertinentes (notamment perte de connexion)

Autres :
Deuxième atelier ce lundi

Compte-rendu 2019-06-27

Avancée et Enjeux :

- @Massaki : Demo de metriques et de dashboard dans Splunk sur le serveur web de Thingworx (Tomcat)
- @Jean-Baptiste + @Olivier + @Nour: Finalisation et rendu du plan de projet
- @Wajdi : Identification de requêtes pour les métriques clés de SQL Server + rattrapage de @Nour
- @Stéphane : Etude de la config de Tomcat et UF

Objectifs :

- @Stéphane : Communication avec @Massaki pour rattrapage et avancer
- @Massaki : Documentation
- @Jean-Baptiste + @Olivier : Plagier se baser sur le travail de @Massaki pour les données de Thingworx dans Splunk
- @Olivier : Définir squelette du rapport
- @Wajdi + @Nour : Déterminer le comment pour l'ingestion des métriques de SQL Server dans Splunk (avec aide de @Massaki) + ingestion si possible
- @Baptiste : Définir les critères de réussite/objectifs (avec @Cédric)

Impediment :
@Massaki n'aura pas internet pour la semaine mais sera joignable sur Slack.

Compte-rendu 2019-06-20

Avancée et Enjeux :

- @Stéphane Pas d'avancee
- @Massaki avancée sur UF, Telegraf et Tomcat Manager, logs rendus TomCat dans Splunk, enjeu de connexion (va entrer en contact avec @Jean-Baptiste)
- @Olivier avancée sur plan de projet
- @Jean-Baptiste enjeu de connexion (perte de temps), s'est penché sur la doc des UF
- @Wajdi pas de données pertinente dans les tables Thingworx
- @Nour mise en place des créneau de workshop

Objectifs pour la semaine prochaine :

- @Stéphane va se mettre up to date avec @Massaki
- @Massaki ingestion des données Telegraf dans Splunk
- @Olivier et @Jean-Baptiste avancée sur UF (contact avec @Massaki)
- @Wajdi récupération des meta de SqlServer (taille BD, …)
- @Nour premier workshop, rattrapage et sync avec @Wajdi

Impediment :
@Massaki sera peu dispo dans la semaine a venir et déménage la semaine suivant.

Compte-rendu 2019-06-13

Avancée et Enjeux :

- @Baptiste va faire un suivi pour les accès admin aux serveurs, je vous reviens sous 48h max
- @Jean-Baptiste (excusé) n'a pas d'avancée à communiquer (intras), pas de bloquant
- @Wajdi Connexion à la BD fonctionnelle, a identifiée et répertoriée les tables, et a setup DBConnect

    - @Cédric, pourras-tu donner un coup de main pour identifier les tables avec de l'info pertinente ? Merci
- @Massaki Tentative de mise en place de Telegraph et UF, bloqué par manque de droits (en cours de résolution par @Baptiste)
- @Olivier idem que @Jean-Baptiste, bloquant en attente de retour pour le plan de projet
    - @Alain, serait-il possible d'avoir un retour au courriel de @Nour ? Merci
- @Nour n'a pas d'avancée à communiquer (intras + job), pas de bloquant
- @Stéphane (absent), pourrais-tu nous faire un suivi ? Merci

Objectifs pour la semaine prochaine :

- @Jean-Baptiste + @Olivier : Faire une conf pour le forwarder, et ensuite exporter automatiquement les logs Thingworx vers Splunk
- @Wajdi : PoC pour import d'une table dans un index de Splunk avec DBConnect
- @Massaki : avancée sur Telegraph + UF dès qu'accès débloqués
- @Stéphane, pourrais-tu nous parler de tes objectifs pour la semaine prochaine ? Merci

Autres :

Plan de projet en attente de réponses d'@Alain
@Nour propose une rencontre plus orientée équipe pour décider des avancées et solutions sur les prochaines taches (type grooming)

- @Baptiste peut se rendre disponible si vous avez besoin d'inputs ou d'aide pour l'organisation, sinon ouvert a ce que vous fassiez cela entre vous
- @Massaki va lancer la discussion sur ce point sur #Slack pour récupérer vos avis/besoins/envie

Compte-rendu 2019-06-06

Avancée et Enjeux :

- @Wajdi n'arrivait pas à se connecter sur Sql
    - Penser que le domaine est TEST (test.local)
    - Essayer d'utiliser SSMS (SqlServer Management Studio)
    - Voir avec @Cédric au besoin
- @Jean-Baptiste avait des enjeux de connexion (problème flux http ?)
    - Discussion et pistes de résolution
- @Massaki a travaillé sur UF et l'ingestion des logs Tomcat + Telegraph
- @Jean-Baptiste et @Olivier ont travaillé sur les specs du plan de projet
- @Stéphane (excusé) a continué sur la formation et regarder les logs Tomcat
- @Nour, peux-tu nous faire un CR de ton avancée svp ?

A venir :

- @Jean-Baptiste et @Olivier ont comme objectif de créer des dahboards pour Thingworx
- @Massaki va installer Telegraph et finaliser l'ingestion des logs Tomcat

- @Wajdi va travailler sur DbConnect

Impediments/autres :

- Semaine d'intra, moins de dispo pour leprojet
- Erreur de débutant à éviter dans Spluk pour l'ingestion de logs/résultats Sql : Pensez à spécifier le champs pour le timestamp, sinon toutes les données seront ingérées en date du jour de la première ingestion

Compte-rendu 2019-05-23

Avancée et Enjeux :

- Environnement Splunk prêt
- L'équipe avance bien sur le Fundamental 1 de Splunk
- Eclaircissement sur le mode de fonctionnement
    - C'est l'équipe qui drive et avance sur les work items
    - Un responsable par PBI qui gère avec l'équipe (tasks, évaluation, suivi, …), potentiellement un backup
    - Proposition d'@Olivier : Baser les Story Points sur le temps estimé dans un premier temps (1pt = 1j ?), tout le monde est OK avec ça ?
- PBI pris en charge
    - User Story 3481: @Massaki
    - User Story 3482: @Wajdi
    - User Story 3489: @Jean-Baptiste
- @Massaki a encore des enjeux de connexion au VPN, il va regarder de son côté et contactera @Baptiste si besoin de support pour une mise en relation avec notre équipe de support

TODO :

- Baptiste
    - Créer les usagers Splunk et communiquer les informations de connexion (avant la fin de la semaine, worst case scenario, lundi)
    - Créer une feature ou les besoins entourant le travail de l'équipe pourront être traités (#3623)
- Cédric
    - Créer le repo Git
    - Voir si on peut générer des échantillons de chaque type de logs dans Thingworx
    - Vérifier si la team a accès au(x) serveur(s), donner les accès le cas échéant
    - Voir avec les autres équipes qui travaillent sur Thingworx si on peut définir une plage de temps journalière de maintenance pour ajustement des configs si nécessaire (1h par jour ? libre les weekends ?)21h – 22h

Opportunités :

- Baptiste
    - Webhook Azure DevOps / Slack

Annexe 2 : Liste des fonctionnalités

Work Item Type	ID	Title	State	Comment Count	Resolved Date
Feature	3480	Collecte de données	Closed	0	Tue Aug 06 2019 13:03:23 GMT-0400 (heure d'été de l'Est)
Feature	3496	Stockage de données	Closed	0	Tue Aug 06 2019 13:03:25 GMT-0400 (heure d'été de l'Est)
Feature	3506	Présentation de données	Closed	0	Tue Aug 06 2019 13:03:27 GMT-0400 (heure d'été de l'Est)

Annexe 3 : Liste des stories

Work Item Type	ID	Title	Assigned To	State	Comment Count	Resolved Date
User Story	4019	Créer un dashboard pour les métriques de Thingworx	Wajdi Bounouara	Closed	1	Mon Jul 29 2019 19:58:39 GMT-0400 (heure d'été de l'Est)
User Story	4020	Créer une alerte pour le dépassement du seuil de la taille de la BD Thingworx	Wajdi Bounouara	Closed	1	Wed Jul 24 2019 03:50:01 GMT-0400 (heure d'été de l'Est)
User Story	3482	Données/logs SqlServer	Wajdi Bounouara	Closed	1	Mon Jul 29 2019 19:58:33 GMT-0400 (heure d'été de l'Est)
User Story	3880	Créations de dashboards Tomcat	Stephane Boyer	Closed	1	Mon Aug 05 2019 11:42:12 GMT-0400 (heure d'été de l'Est)

User Story	3879	Créations d'alertes Tomcat	Stephane Boyer	Closed	4	Mon Aug 05 2019 11:42:10 GMT-0400 (heure d'été de l'Est)
User Story	3890	Analyser l'application de Cédric	Olivier Gregoire	Closed	4	Tue Jul 23 2019 19:40:41 GMT-0400 (heure d'été de l'Est)
User Story	3885	Déterminer le format du rapport final	Olivier Gregoire	Closed	0	Wed Jul 17 2019 19:15:25 GMT-0400 (heure d'été de l'Est)
User Story	4002	Rédaction rapport final	Olivier Gregoire	Closed	0	Tue Aug 06 2019 11:40:22 GMT-0400 (heure d'été de l'Est)
User Story	4003	Présentation oral	Olivier Gregoire	Closed	0	Tue Aug 06 2019 11:44:52 GMT-0400 (heure d'été de l'Est)
User Story	3507	Identifier les données pour la surveillance des composants Thingworx	Olivier Gregoire	Closed	1	Tue Jul 09 2019 19:25:44 GMT-0400 (heure d'été de l'Est)
User Story	3509	Identifier les données pour l'expérience client	Nour El-Moussaoui	Closed	2	Thu Jul 11 2019 22:21:29 GMT-0400 (heure d'été de l'Est)
User Story	3996	Extraction et normalisation des champs pour les logs SQL	Nour El-Moussaoui	Closed	2	Tue Jul 23 2019 22:06:14 GMT-0400 (heure d'été de l'Est)
User Story	3884	Créations de rapports SQL	Nour El-Moussaoui	Closed	3	Thu Jul 25 2019 14:47:50 GMT-0400 (heure d'été de l'Est)
User Story	3881	Créations de dashboards SQL	Nour El-Moussaoui	Closed	2	Thu Jul 25 2019 14:41:00 GMT-0400 (heure d'été de l'Est)
User Story	4033	Documentation Wiki des dashboards,alertes et rapport de logs SQL	Nour El-Moussaoui	Closed	1	Thu Jul 25 2019 15:25:34 GMT-0400 (heure d'été de l'Est)

User Story	3878	Créations d'alertes SQL	Nour El-Moussaoui	Closed	3	Thu Jul 25 2019 14:47:45 GMT-0400 (heure d'été de l'Est)
User Story	3874	Identifier les données pertinentes des serveurs SQL	Nour El-Moussaoui	Closed	0	Mon Jul 08 2019 19:16:40 GMT-0400 (heure d'été de l'Est)
User Story	3873	Données serveurs	Massaki Archambault	Closed	3	Thu Jul 18 2019 11:02:39 GMT-0400 (heure d'été de l'Est)
User Story	3707	Installer un splunk forwarder	Massaki Archambault	Closed	1	Sun Jun 16 2019 17:45:05 GMT-0400 (heure d'été de l'Est)
User Story	3481	Données/logs Tomcat	Massaki Archambault	Closed	5	Mon Jun 17 2019 23:42:30 GMT-0400 (heure d'été de l'Est)
User Story	3508	Identifier les données pour la surveillance de la connexion des devices	Jean-Baptiste Charvet	Closed	1	Mon Jul 29 2019 18:48:15 GMT-0400 (heure d'été de l'Est)
User Story	3497	Normalisation des champs pour les logs Thingworx	Jean-Baptiste Charvet	Closed	2	Tue Jul 23 2019 18:05:21 GMT-0400 (heure d'été de l'Est)
User Story	3493	Données Thingworx	Jean-Baptiste Charvet	Closed	1	Thu Jul 04 2019 08:35:54 GMT-0400 (heure d'été de l'Est)
User Story	3489	Identifier les autres données de Thingworx à ingérer	Jean-Baptiste Charvet	Closed	0	Wed Jul 10 2019 10:41:47 GMT-0400 (heure d'été de l'Est)
User Story	3511	Créations de dashboards TW	Jean-Baptiste Charvet	Closed	0	Thu Aug 01 2019 11:56:42 GMT-0400 (heure d'été de l'Est)
User Story	3510	Créations d'alertes TW	Jean-Baptiste Charvet	Closed	1	Fri Aug 02 2019 08:55:46 GMT-0400 (heure d'été de l'Est)

User Story	4037	Extraction d'informations d'un objet connecté à thingworx	Jean-Baptiste Charvet	Closed	0	Mon Jul 29 2019 16:54:00 GMT-0400 (heure d'été de l'Est)
Bug	4052	Toutes les alertes sont bloquées.	Jean-Baptiste Charvet	Closed	4	Tue Aug 06 2019 11:44:33 GMT-0400 (heure d'été de l'Est)

Annexe 4 : Squelette du wiki

Services & Composantes *Cette partie contient toutes les informations sur chacun des composants du projet qui peuvent être utile pour chacun des membres de l'équipe*

- SQL
- Tomcat
- Thingworx
- Splunk
- Telegraf

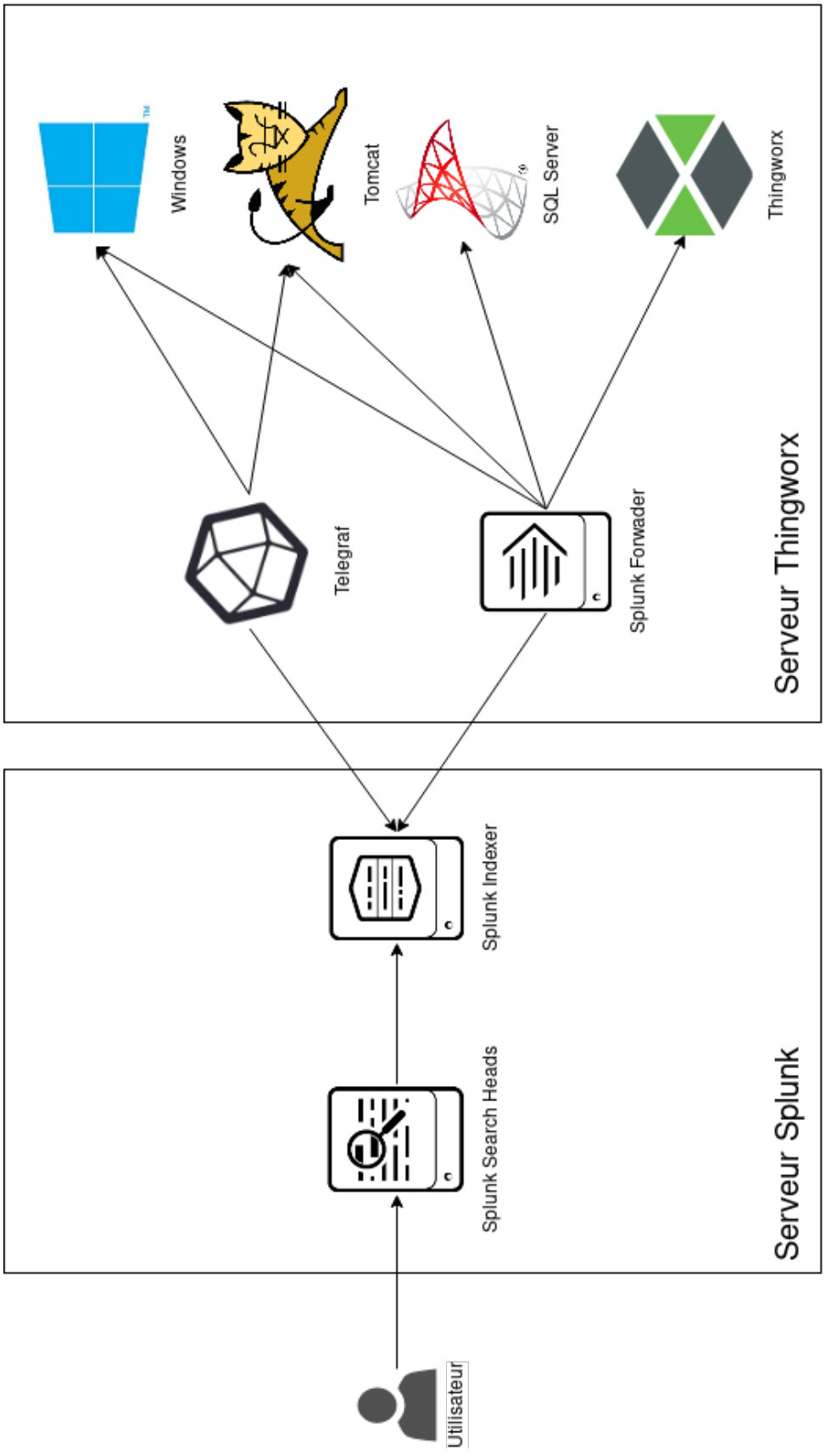
Alertes, dashboard et rapport *Liste des idées avec l'avancement de chacune d'elles*

- Tomcat
    - Alertes
    - Dashboards
    - Rapports
- Thingworx
    - Alertes
    - Dashboards
    - Rapports
- SQL server
    - Alertes
    - Dashboards
    - Rapports

Liste questions sur le projet *Page contenant toutes les questions que nous nous posons ainsi que leurs réponses une fois que nous les avons*

Compte rendu réunion

Annexe 5: Architecture du système



Annexe 6 : Exemples de requêtes

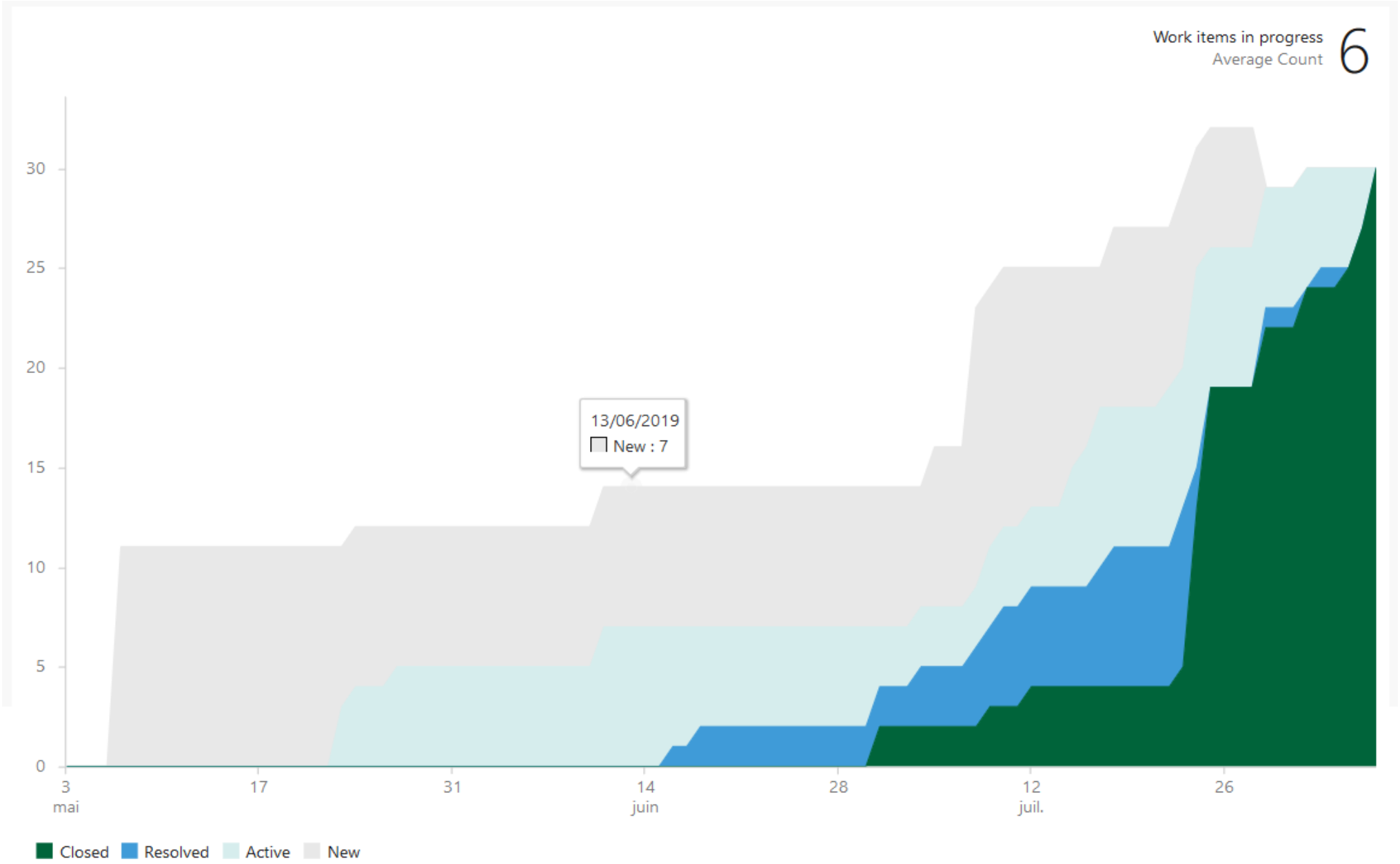
A1 : trop d'échec de connexion des utilisateurs

index="thingworxindex" sourcetype="log4j" "Authentication failed: Please make sure the credentials are correct." source="C:\\Program Files (x86)\\ThingworxMSSQL-1.2.0-SNAPSHOT\\ThingworxStorage\\logs\\SecurityLog.log"

A2 : compte utilisateur bloqué après des tentatives infructueuse de connexion

index="thingworxindex" source="C:\\Program Files (x86)\\ThingworxMSSQL-1.2.0-SNAPSHOT\\ThingworxStorage\\logs\\SecurityLog.log" origin="S.c.t.s.a.AuthenticationUtilities" "User LOCKED after * login attempts for * within * minutes"

Annexe 7: Évolution de la résolution des user stories à réaliser


Work items in progress
Average Count
6
30
25
20
15
10
5
0
13/06/2019
New : 7
3
mai
17
31
14
juin
28
12
juil.
26
Closed
Resolved
Active
New