

Comptes-rendus hebdomadaires

Nom: **Petit**

Prénom: **Gwénaél**

Entreprise: **WorldLine**

Résumé de la mission : Mission de prise en main de l'environnement de travail consistant à mettre à jour un script permettant d'afficher l'état des différents processus sur un serveur.

Semaine du 11/03 au 15/03

Ce que j'ai fait:

- Le lundi et mardi matin ont été consacrés à la configuration du poste de travail. J'ai dû installer et configurer des logiciels tels que Putty, MySQL Workbench et Eclipse. J'ai aussi pu prendre en main des outils utilisés par l'entreprise, comme Confluence et Jira.
- Durant les premiers jours, mon objectif était de comprendre le fonctionnement et les caractéristiques du produit TOM (Trade Order Management) sur lequel je vais travailler. Pour cela, j'ai consulté la documentation relative au projet, et le Product Manager m'a fait une présentation détaillée du produit ainsi que de l'organisation de l'entreprise pour ce projet.
- À partir de mercredi, j'ai pu commencer à travailler sur ma première mission, qui consiste à mettre à jour un service permettant d'afficher l'état des services sur un serveur. Voir [détail mission](#). Cette mission m'a permis de prendre en main OPS, un outil interne de **Worldline** qui permet de configurer et de déployer des services sur des serveurs.

Ce que j'ai appris:

- J'ai découvert plus en détail le fonctionnement d'une équipe informatique au sein d'une entreprise de cette envergure. Le projet rassemble plusieurs équipes, chacune comprenant un certain nombre de collaborateurs, ce qui nécessite une communication efficace, prenant ainsi plus de temps que je ne l'imaginais dans le travail d'un développeur.
- J'ai appris à configurer et déployer un service avec OPS et découvert comment l'entreprise utilise gitlab pour ses déploiements.
- Cette semaine était une semaine d'adaptation et au delà des compétences techniques, l'apprentissage concernait surtout la prise en main de l'environnement de travail et l'adaptation à

l'entreprise. J'ai par exemple participé à des réunions journalières avec mon équipe pour partager nos avancées.

Difficultés rencontrés:

- Le produit TOM est assez complexe et concerne un domaine pour lequel j'ai encore peu de connaissances (la finance). J'ai donc passé une partie de mon temps à lire de la documentation pour comprendre son fonctionnement et son utilité.
- Il m'a fallu un peu de temps pour comprendre comment le déploiement fonctionnait, en effet, lors du premier déploiement, j'ai eu différents problèmes concernant les chemins d'accès, que j'avais mal configuré.

Programme pour la semaine prochaine:

- Lundi sera consacré au déploiement du service sur tous les serveurs.
- J'ai une réunion lundi pour en savoir plus sur ma prochaine mission.

Détail Mission:

Résultat actuel du script actuel:

```
[DEV:www@tdtbr002s:/MIDDLE/bnperto]$ stat_serv
stat_serv

Statuts des process TOM sur tdtbr002s:

FLUSHORDER.....[NORMALLY STARTED]....Démarrage : 00:01:01
EXPIRATION_BNP.....[NORMALLY STARTED]....Démarrage : 00:02:01
MAR_BP2S.....[NORMALLY STARTED]....Démarrage : 00:05:00
MAR_CAT.....[NORMALLY STARTED]....Démarrage : 00:05:00
MAR_FORTIS.....[NORMALLY STARTED]....Démarrage : 00:05:00
MAR_default.....[NORMALLY STARTED]....Démarrage : 00:05:00
POSTTRADE/.....[NORMALLY STARTED]....Démarrage : 00:05:00
POSTTRADESENDER/.....[NORMALLY STARTED]....Démarrage : 00:05:00
loggerviewer_fr.....[NORMALLY STARTED]....Démarrage : 00:05:00
```

La modification demandée consiste à afficher des noms plus clairs (pour l'instant, seul le nom renvoyé lorsqu'on exécute "ps -f" est affiché) et à afficher la date du dernier lancement du serveur.

Les informations nécessaires pour un service sont récupérées à partir d'un fichier CSV. On y récupère leur nom, l'heure de démarrage, l'heure d'arrêt, les jours de démarrage et les jours d'arrêt.

Un script permettant de récupérer ces informations a déjà été développé par une autre équipe, il faut donc configurer notre service pour utiliser leur script.

3 étapes:

- Mettre à jour le csv (rajouter une colonne avec un label)
- Mettre à jour un fichier `utils` qui définit des fonctions nécessaires au script (renommer ces fonctions)
- Configurer le service à l'aide d'un logiciel interne appelé OPS.

1.

La mise à jour du CSV consiste juste à rajouter une colonne et les réorganiser.

2.

Le fichier `utils` contenant plus de 300 fonctions, il serait trop long de toutes les renommer une par une, surtout qu'il suffit de leur ajouter un préfixe `"cop_"`.

On utilise donc commande `sed`, dont voici un exemple.

```
sed -i "s/^r/cop_r/g" utils
```

3.

La configuration du service sur OPS est la partie la plus complexe, puisque je n'avais jamais utilisé OPS avant.

Il faut donc ajouter une ressource dans le module `"Monitoring"`, qu'on appelle `"monitoring-test"`, puis ajouter les serveurs sur lesquels on souhaite la déployer.

Ensuite, on définit notre configuration en se basant sur le template de la configuration faite par l'équipe qui a développé les scripts utilisés puis on adapte les variables.

Configuration:

Search...	🔍	Σ Expressions	📖 Documentation	🔴 Advanced
Global				
destination directory path	?	↻	/MIDDLE/bnpprto/monitoring-test	
Template Variables				
Defining key/value	+			
Download Sources	+			
conf	⊖			
Type	↻	git		▼
git repository url	↻	https://gitlab.kazan.myworldline.com/bnpprto-ts-deployment/bnpprto-ts-deployment-monitoring-tools		
git repository subdirectory	?	none		
git username	↻	bnpprto-ts-deployment-technical-user		
git access token	↻		
git revision/branch/tag	↻	\${version_name}		
destination directory relative path	↻	conf		
etc	⊖			
Type	↻	git		▼
git repository url	↻	https://gitlab.kazan.myworldline.com/wts-product/tools/monitoring.git		
git repository subdirectory	?	/\ empty /\		
git username	↻	wts-product-technical_user		
git access token	↻		
git revision/branch/tag	↻	1.2		
destination directory relative path	↻	etc		

Conclusion

Une fois le déploiement terminé, on peut essayer la commande qui donne le résultat suivant.

```
[DEV:www@tdtbr002s:/MIDDLE/bnpprto]$ cop_stat_serv
cop_stat_serv

Statuts des process TOM sur tdtbr002s:

FLUSHORDER.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:01:02
EXPIRATION_BNP.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:02:02
MAR_BP2S.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:01
MAR_CAT.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:01
MAR_FORTIS.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:01
MAR_DEFAULT.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:02
POSTRADe.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:02
POSTTRADESENDER.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:02
LOGGERVIEWER.....[NORMALLY STARTED]....Démarrage : 2024-03-14 00:05:01
```

Un service d'envoi de mails utilise celui que l'on vient de mettre à jour, il faut donc vérifier qu'il fonctionne toujours (c'est le cas).

On déploie ensuite sur tous les serveurs et la mission est terminée.