

가장 많은 고객이 선택한 클라우드 모니터링 서비스 1위, 와탭
당신의, IT 서비스의 성능을 극대화합니다.

 WhaTap

DOCS

MSSQL Agent Guide

WhaTap Support

Version 1.0



Table of Contents

MSSQL Agent Guide	1
1. Install	2
1.1. 에이전트 실행 및 모니터링 개요	2
1.2. 표준 설치	5
1.3. 에이전트 네이밍	8
1.4. 데이터베이스 서버 별 설정	9
1.5. 설치 에러 대응	9
1.6. Advanced Feature	10
1.7. 설치 환경	12
2. 수집 데이터 종류 및 수집 주기	13
2.1. Real Time monitoring	14
2.2. Active Session	15
2.3. Trends	15
2.4. SQL Stat	15
2.5. 데이터베이스 모니터링 설정	15

MSSQL Agent Guide

제목 : MSSQL Agent Guide

작성자 : WhaTap Support

이메일 : support@whatap.io

날짜 : 2019-03-29

버전 : 1.0

설명 : 본 문서는 WhaTap MSSQL Monitoring 에이전트에 대해 설명합니다.

Chapter 1. Install

에이전트 설치부터 실행, 업데이트 및 종료에 대해 설명하며, 나아가 설치 중 발생할 수 있는 오류에 대한 사례들을 설명합니다.

1.1. 에이전트 실행 및 모니터링 개요

와탭 DBX모니터링은 Query 기반으로 데이터베이스의 성능 데이터를 수집하여 성능 모니터링 서비스를 제공합니다. 데이터베이스 성능 관리는 데이터베이스의 성능을 실시간으로 감시하고 문제점을 진단, 분석하여 문제가 되는 SQL을 튜닝하고 적용하여 문제점이 해결되었는지 실시간으로 감시하는 작업입니다. 와탭 DBX는 이러한 일련의 작업이 순환 관리되도록 설계된 데이터베이스 성능 모니터링 솔루션입니다. 데이터베이스의 동작을 감시하고 진단 및 분석하여 가용성과 성능을 일정한 수준으로 유지하는데 도움이 되도록 구성되어 있습니다. 데이터베이스에 어떤 문제가 발생한 경우, 담당자에게 알람을 전송하여 상황을 인지하고, 수집된 데이터를 통하여 DB상황을 보다 정밀하게 분석하여 Root Cause를 명확하게 밝혀내는데 도움을 주는 솔루션입니다.

Public Cloud, Private Cloud 모두에 적용가능 하며, 쿼리 베이스로 데이터를 수집하기 때문에 모니터링 대상 서버의 Operating System에 무관합니다.

따라서, 와탭 데이터베이스 모니터링 서비스(DBX)는 모니터링 대상 데이터베이스 서버에 직접 에이전트를 설치하지 않고도 RDS(Amazon Relational Database Service) 포함한 어떤 데이터베이스라도 모니터링이 가능 합니다.

에이전트 구성

에이전트의 구성은 수집 서버, 에이전트, 그리고 수집서버로 이루어집니다.

- 수집서버
 - 에이전트가 수집한 데이터베이스 성능 데이터를 수집, 저장 및 통계 정보 추출하고 이를 사용자에게 효율적인 방법으로 제공합니다. 수집서버는 지역(Region)별로 설정이 가능합니다. 지역(Region)별로 수집서버의 주소가 다르게 할당 되므로 사용자가 선택한 지역(Region)에 따라 수집서버 주소는 다를 수 있습니다. 지역(Region) 선택은 프로젝트 생성시에 함께 설정합니다.
- 기본 에이전트
 - Query기반으로 성능 데이터를 수집하여 서버로 전송합니다. 모니터링 대상 데이터베이스서버에 직접 에이전트를 설치하지 않고도 모니터링이 가능 합니다.
- 부가에이전트(XOS Agent)
 - XOS 에이전트는 데이터베이스 서버의 프로세스 사용량을 모니터링 할 수 있는 부가 옵션 에이전트 입니다. 데이터베이스 서버의 프로세스 사용량을 모니터링 하고 싶은 경우, 데이터베이스 서버에 별도의 에이전트를 띄워 데이터를 수집 할 수 있습니다.
- 네트워크
 - 기본 에이전트: 와탭 모니터링 에이전트는 모니터링 정보를 수집하여 서버에 데이터 전송하기 위하여 외부통신(TCP)을 위한 6600 포트를 사용합니다.
 - 부가 에이전트: 에이전트가 설치된 서버와 데이터베이스 서버간 외부통신(UDP)을 위한 3002포트를 사용합니다. 내부 포트가 충돌이 나는 경우, dbx_port 옵션을 통해 포트 변경이 가능합니다.

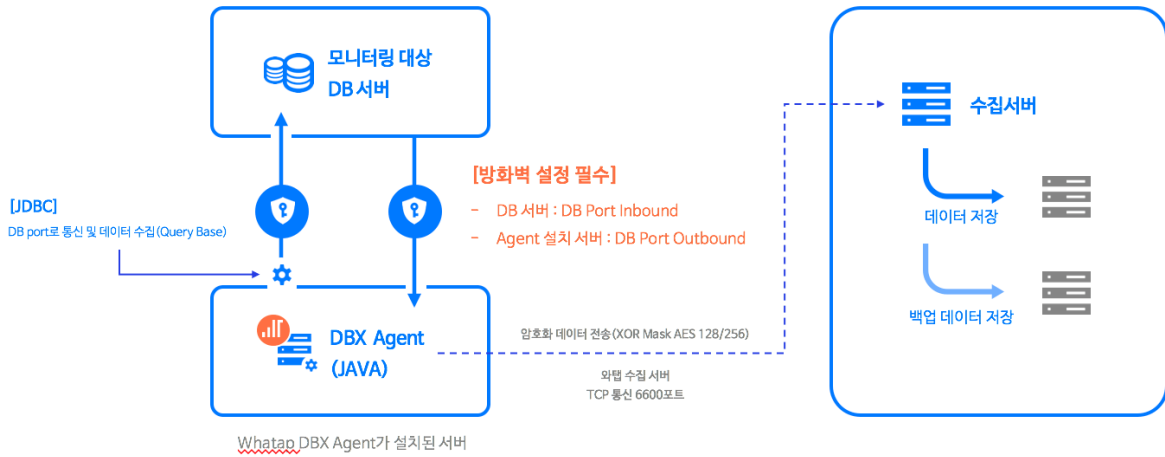


Figure 1. 에이전트 구성도

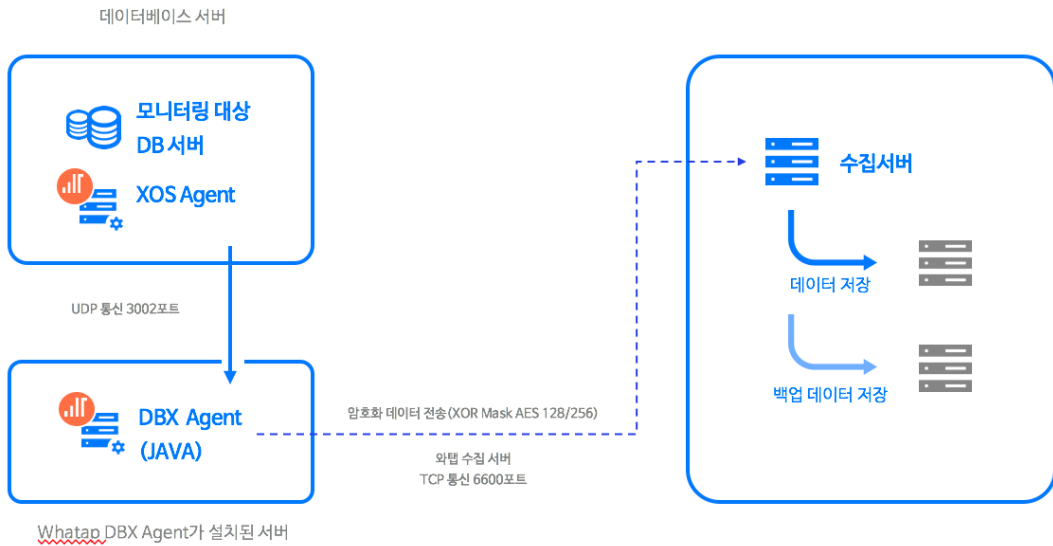


Figure 2. 부가에이전트(XOS Agent) 구성도

에이전트 구성 파일

Table 1. 에이전트 구성 파일의 종류

파일명	설명
whatap.conf	데이터베이스 서버의 데이터를 수집하는 수집서버의 주소와 서버의 프로젝트 라이선스 키가 입력되는 파일.
alert/alert.conf	수집하는 모니터링 항목에 임계치를 설정하는 파일. 임계치를 초과하는 경우 알림 이벤트가 발생한다.
scripts/	원격으로 sql 스크립트를 실행할수 있는 스크립트들을 모아둔 디렉토리.
ps.sh	프로세스 아이디를 읽어 오는 스크립트. 에이전트 프로세스를 종료할 때 해당 아이디를 참고 한다.
stop.sh	에이전트 프로세스를 종료할 때 사용하는 스크립트.

uid.sh (uid.bat)	데이터베이스 접속 정보를 조합하여 암호화된 uid 를 생성하기 위한 쉘 스크립트. db.user 파일이 생성된다. 최초 한번만 설정해두면, 그 이후부터는 암호화된 uid 를 통해 모니터링 대상 데이터베이스 서버로부터 데이터 수집이 이루어진다.
start.sh (start.bat)	에이전트 실행을 위한 쉘 스크립트. 에이전트가 실행되면 데이터베이스 서버의 모니터링 정보를 수집하기 시작한다.
startd.sh (startd.bat)	에이전트 실행을 위한 쉘 스크립트를 백그라운드로 실행 할 수 있는 스크립트.
whatap.agent.dbx- [VERSION].jar	(Tracer 프로그램) 데이터베이스 서버의 정보를 수집하고 수집된 정보를 서버로 전송하는 프로그램
jdbc	데이터베이스 서버 연결을 위해 참조 되는 라이브러리들을 모아두는 디렉토리. 에이전트와 데이터베이스 서버의 연결을 위한 라이브러리를 직접 다운로드 받아 java의 classpath 옵션에 경로를 지정하여 사용한다.
xos	데이터베이스 서버의 프로세스 사용량을 모니터링 할 수 있는 부가 옵션 에이전트가 포함된 디렉토리.
xos/xos.conf	데이터베이스 서버의 프로세스 사용량 데이터를 수집 및 데이터 전송을 위한 에이전트서버의 주소와 통신 포트가 입력되는 파일.

JDBC다운로드 안내

DBX에서 지원하는 데이터베이스 모니터링이 가능한 종류별로 다운로드 링크를 안내합니다.

Postgresql (지원 버전: 9.2 이상)

: <https://jdbc.postgresql.org/download.html>

EDB (지원 버전: 9.2 이상)

: <https://www.enterprisedb.com/advanced-downloads>

Oracle (지원 버전: 11g 이상)

- Oracle Database 11g Release 1 JDBC Drivers : <http://www.oracle.com/technetwork/database/enterprise-edition/jdbc-111060-084321.html>
- Oracle Database 12c Release 1 JDBC Driver : <http://www.oracle.com/technetwork/database/features/jdbc/jdbc-drivers-12c-download-1958347.html>

Mysql & MariaDB (지원 버전: 5.5 이상)

- Connector/J 8.0.11 : <https://dev.mysql.com/downloads/connector/j/>
- Connector/J 5.1.46 : <https://dev.mysql.com/downloads/connector/j/5.1.html>

Mssql (지원 버전: 2005 이상)

- Microsoft JDBC Driver 6.4 for SQL Server : <https://www.microsoft.com/en-us/download/details.aspx?id=56615>
- Microsoft JDBC Driver 6.2 for SQL Server

: <https://www.microsoft.com/en-us/download/details.aspx?id=55539>

1.2. 표준 설치

와탭 DBX 모니터링 서비스를 사용하기 위해서는 모니터링 대상 데이터베이스 서버와의 네트워크 통신이 가능한 서버에 와탭 에이전트를 설치 해야 합니다. 와탭 데이터베이스 모니터링 서비스는 모니터링 대상 데이터베이스 서버에 직접 에이전트를 설치하지 않고도 모니터링이 가능 합니다. 와탭 DBX 모니터링 에이전트 설치 방법은 whatap.io 사이트에서 압축된 에이전트 파일을 다운로드 받아 서버 임의의 위치에 압축을 풀어 실행하는 것만으로 설치가 완료됩니다.

1.2.1. 설치 절차

프로젝트 생성

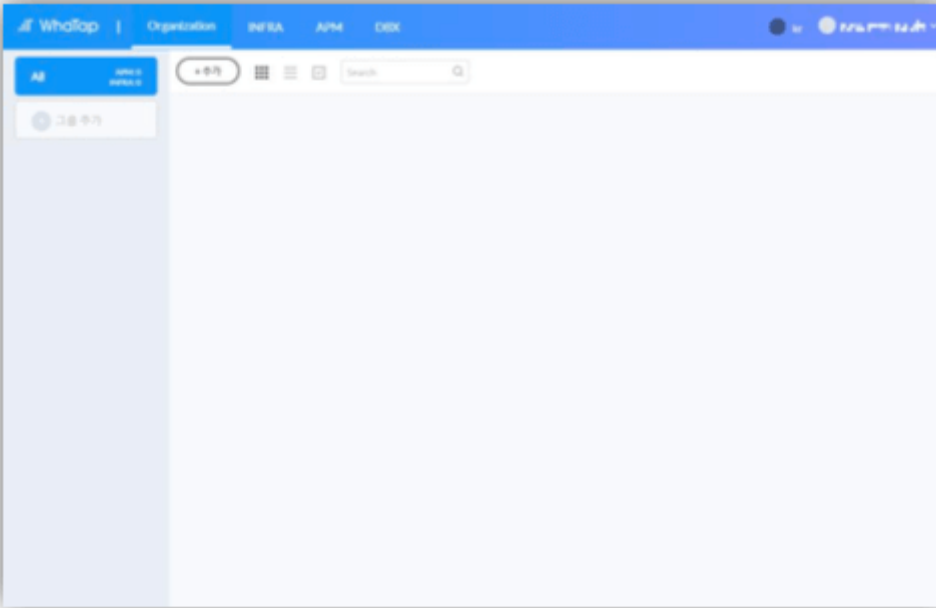


Figure 3. 프로젝트 생성 페이지

서버를 등록하기 위해 우선 프로젝트를 생성합니다. 추가 버튼을 선택하면 아래와 같이 프로젝트 생성 창이 나타납니다. 설치하려는 데이터베이스 아이콘을 선택한 뒤, 희망하는 프로젝트명과 데이터 서버 지역(Region), 소속하게 될 그룹을 선택한 뒤 프로젝트를 생성합니다.

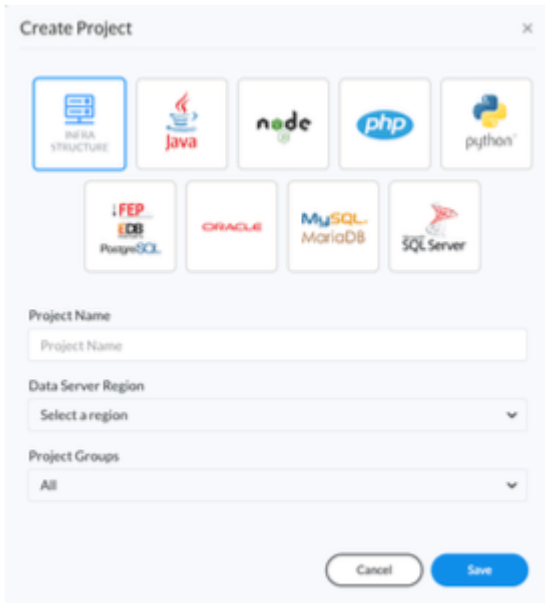


Figure 4. 프로젝트 이름 지정 및 수집 서버 지역 선택

라이선스 발급

- 1 JDK가 설치되어 있어야 합니다. 지원 버전: Java 6 이상
- 2 Oracle 지원 버전: 11gR2 이상



라이선스 발급

라이선스 키를 생성해 주세요.
압축을 해제한 폴더에 아래의 whatap.conf 설정 내용을 확인합니다.

```

1 210.122.38.128:6600 210.122.38.170:6600 tcp outbound

```

Whatap Server (210.122.38.128/210.122.38.170)도 데이터 전송을 위해 6600 포트가 열려 있어야 합니다. (Tcp Outbound)

[Figure - 3] 라이선스 발급 받기

Figure 5. 라이선스 발급 받기

프로젝트 관리화면에서는 우선적으로 라이선스를 발급 받습니다. 라이선스 키는 프로젝트별로 귀속되기 때문에, 유출되거나 배포되어서는 안됩니다. 반드시 본인 프로젝트에 서버를 등록할 때에만 이용하시기 바랍니다.

에이전트 다운로드



Figure 6. 에이전트 다운로드

라이선스를 발급 받은 후에는 ‘에이전트 파일 다운로드’ 버튼이 활성화 되었음을 확인할 수 있습니다. 해당 버튼을 눌러 와탭 에이전트 파일을 다운로드 받습니다. 다운로드가 완료되면 안에 있는 whatap.conf 파일의 설정을 확인하여 라이선스키와 데이터 수집 서버 주소가 정상적으로 들어가 있는지를 확인합니다.

아래 명령어를 통해 파일을 다운로드 받을 수 있습니다.

```
$ wget -O whatap.agent.database.tar.gz https://service.whatap.io/download/agent/[DB_TYPE]
```

아래 명령어를 통해 파일을 압축을 해제 할 수 있습니다.

```
$ tar -xzvf whatap.agent.database.tar.gz
```

정상적으로 들어가 있는게 확인이 되었다면 모니터링을 대상 데이터베이스 서버의 정보를 입력합니다.

```
license=[라이선스 키]
whatap.server.host=[수집서버 정보]
dbms=[DB 종류]
db=[DB_NAME(main)]
db_ip=[D_IP]
db_port=[D_PORT]
```

- wget 으로 직접 다운 받을 경우, whatap.conf 파일에 라이선스키와 데이터 수집 서버 주소가 정상적으로 들어가지 않습니다. 해당 방식으로 다운 받을 경우, 업로드 후 별도로 라이선스키와 데이터 수집 서버 주소를 넣어주시기 바랍니다.
- 에이전트는 설정파일에 있는 데이터베이스 서버 주소로 연결을 맺고 데이터 수집을 합니다. 따라서 데이터베이스 서버 연결에 필요한 포트가 차단되어 있으면 데이터 수집이 되지 않습니다.
- 에이전트는 수집 서버 주소로 데이터베이스 서버의 성능 정보를 전송합니다. 그러므로 방화벽에 수집 서버 IP 로의 TCP 아웃바운드 포트 (6600)이 차단되어 있으면 데이터 수집이 되지 않습니다.

UID 생성

데이터베이스 접속을 위하여 암호화된 uid 를 생성하기 위해 username 과 password 를 입력하고 shell script 를 실행합니다. 최초 한번만 설정해두면, 그 이후부터는 암호화된 uid 를 통해 모니터링 대상 데이터베이스 서버로부터 데이터 수집이 이루어집니다.

```
$ ./uid.sh [DB_USER] [DB_PASSWORD]

The uid is updated!!
```

에이전트 시작

whatap 폴더 밑의 shell script 로 에이전트를 실행합니다. 에이전트가 실행되면 데이터베이스의 모니터링 정보를 수집하기 시작합니다. 에이전트 로그를 확인하여 에이전트의 정상 기동 여부를 확인합니다.

```
$ ./start.sh

-      - - - - -      - - - - -
| | / / / / / _ _ _ / _ _ / _ _ _
| / / / / _ \ \ _ \ / / / / _ \ \ \
| _ / | _ / / / \ \ _ \ / / / \ \ \ /
| _ / | _ / / / \ \ _ \ / / / \ \ \ /
| _ / | _ / / / \ \ _ \ / / / \ \ \ /
| _ / | _ / / / \ \ _ \ / / / \ \ \ /
| _ / | _ / / / \ \ _ \ / / / \ \ \ /

Just Tap, Always Monitoring
WhaTap DBX version 1.3.9 20181120
```

에이전트 프로세스 확인

다음과 같은 명령어를 통하여 동작중인 와탭 DBX 에이전트 프로세스를 확인 할 수 있습니다.

```
$ ps -ef | grep dbx
```

로그 확인

서비스 재시작시 정상적으로 로그가 뜨는 화면을 통해 정상 구동 여부를 확인합니다. 에이전트 등록 후 수집과 관련된 로그는 \$WHATAP_HOME/logs 디렉토리에서 확인 할 수 있습니다. 에이전트가 정상적으로 동작하는 경우 해당 날짜에 다음과 같은 로그 파일을 확인 할 수 있습니다.

- whatap-DBX-[OID]-[DATE]
- whatap-boot-[DATE].log)

모니터링 확인

서버에서 정상적으로 로그가 올라온 것을 확인한 뒤, 콘솔에 정상적으로 등록되어 있는 여부를 확인하기 위해 해당 프로젝트의 ‘관리-에이전트 목록’ 메뉴에 올라온 해당 데이터베이스 서버의 명칭을 확인합니다.

에이전트 목록

아이디	이름(아이디)	수정	시작 시간	종료/비활성	상태	버전	DB IP	DB Port	DB Version	cpu_core(s)	정보
-1283227466	DBX-O-119-1521	수정	2018-10-30 13:58:05	2018-11-27 14:39:22	활성화	1.3.9 20181113	192.168.0.240	1521	10.2.0.2.0	1	more
-106493477	DBX-O-101-1521	수정	2018-10-26 14:24:25	2018-11-27 14:39:27	활성화	1.3.9 20181113	192.168.0.240	1521	12.2.0.1.0	4	more
2134080373	DBX-O-102-1521	수정	2018-10-26 14:42:30	2018-11-27 14:39:22	활성화	1.3.9 20181113	192.168.0.240	1521	18.0.0.0.0	4	more

- 데이터베이스명은 {type}-{ip2}-{ip3}-{port} 의 형태의 식별ID가 부여됩니다.

1.3. 에이전트 네이밍

와탭 모니터링 서비스에서 모니터링 대상을 포괄적인 의미로 오브젝트라는 명칭을 사용하며 에이전트와 혼용해서 사용하기도 합니다. 모니터링 대상 즉 오브젝트를 구분하기 위해서는 이름을 구분해야 합니다. 와탭은 오브젝트 이름과 오브젝트 아이디를 통해서 대상을 구분합니다.



오브젝트(에이전트)이름은 프로젝트 단위로 유일해야 합니다.



오브젝트 아이디 = CRC32(오브젝트 이름) 입니다.
오브젝트 아이디/이름이 바뀌면 이전 데이터와 연결되지 않습니다.

1.3.1. 이름 결정 방식

데이터베이스 서버로부터 추출한 정보를 활용하는 이유는 데이터베이스 서버 정지, 네트워크 단절 또는 에이전트 문제로 인한 수집 서버와 에이전트의 통신 단절 상태가 복구되었을 경우, 재 접속된 에이전트로부터 송신되는 정보가 기존 에이전트로부터 송신된 정보와의 연속성을 유지하기 위해서 입니다.

에이전트 이름 식별

와탭은 모니터링 정보 수집 대상인 데이터베이스 서버 식별을 위한 정보로 기본적으로 데이터베이스 서버로부터 수집한 정보를 활용합니다. 기본적으로 활용하는 정보는 데이터베이스 서버 종류, 데이터베이스 서버의 IP, 서비스 포트를 조합하여 데이터베이스 서버를 고유 식별자로 사용하게 되며 필요에 따라 사용자가 지정한 명칭을 사용하거나 패턴을 변경하여 사용하는 것도 가능합니다. 이때에는 꼭 고유한 값이어야만 합니다. 데이터베이스 서버로부터 추출한 정보를 활용하는 이유는 데이터베이스 서버 정지, 네트워크 단절 또는 에이전트 문제로 인한 수집 서버와 에이전트의 통신 단절 상태가 복구되었을 경우, 재 접속된 에이전트로부터 송신되는 정보가 기존 에이전트로부터 송신된 정보와의 연속성을 유지하기 위해서 입니다. 와탭이 데이터베이스 서버를 식별하기 위해 사용하는 기본 패턴은 다음과 같습니다.

기본 패턴에 대한 변경은 whatap.conf에서 설정에서 가능합니다.

object_name

Default : {type}-{ip2}-{ip3}-{process}

Type : String

명칭	설명
{type}	whatap.app_name 에 설정된 값을 사용합니다.
{ip#}	IP를 나누었을 때 #번째 자리를 사용합니다.
{process}	whatap.app_process_name 에 설정된 값을 사용합니다.
{hostname}	서버 호스트명을 사용합니다.

1.4. 데이터베이스 서버 별 설정

1.5. 설치 에러 대응

1.5.1. 방화벽 설정 확인

와탭 서버에 대한 TCP 아웃바운드 방화벽이 설정되어 있으면 모니터링 정보를 서버로 전송 할 수 없으므로 방화벽 차단을 해제해야 합니다.

방화벽 확인 방법 (telnet 서버IP 서버포트)

telnet 명령 수행 시 하기와 같은 접속 관련 정보가 표시되어야 정상입니다.

```
$ telnet 52.193.60.176 6600
Trying 52.193.60.176...
Connected to 52.193.60.176.
Escape character is '^]'.
```

- 수집 서버 정보는 에이전트 설치 메뉴에서 확인할 수 있습니다.

1.6. Advanced Feature

1.6.1. 데이터베이스 프로세스 모니터링을 하고 싶은 경우(XOS에이전트 설치)

Postgresql, Oracle에서는 데이터베이스 프로세스 모니터링을 할 수 있습니다. 데이터베이스 서버의 프로세스 사용량을 모니터링 하고 싶은 경우, 데이터베이스 서버에 별도의 XOS 에이전트 프로세스를 띄워 데이터를 수집 할 수 있습니다. 이경우, 진행중인 세션의 cpu usage값을 함께 확인 할 수 있습니다.

에이전트 서버에 있는 `whatap.conf` 파일에 다음 옵션을 설정합니다.

```
xos=[DBX_DB_SERVER_IP]
xos_port=[DBX_PORT] (default: 3002)
```

다음 `xos` 디렉토리를 데이터베이스 서버로 옮깁니다.

```
압축을 해제한 폴더/xos/
```

데이터베이스 서버에 있는 `xos.conf` 파일에 에이전트 서버 정보를 입력합니다.

Agent Server로 데이터 전송을 위해 `dbx_port`로 설정된 포트(default 3002)가 열려 있어야 합니다. (UDP Outbound)

```
dbx_ip=[DBX_AGENT_SERVER_IP]
dbx_port=[DBX_PORT] (default: 3002)
cpu_limit=0
mem_limit=10240
```

데이터베이스 서버에서 다음 프로세스를 실행 합니다.

```
$ ./start.sh
```

참고) background로 실행하고자하는 경우에는 다음 명령어로 실행합니다.

```
$ ./startd.sh
```

1.6.2. 서버명을 임의로 부여하여 관리하고 싶은 경우

에이전트는 데이터베이스 서버 종류와 데이터베이스 서버의 IP, 서비스 포트를 사용하여 자동으로 서버명을 부여합니다.

데이터베이스 인스턴스명 지정 옵션

기본 패턴에 대한 변경은 whatap.conf에서 설정에서 가능합니다.

- object_name default: {type}-{ip2}-{ip3}-{port}

데이터베이스 인스턴스명 패턴

프로젝트에 등록된 데이터베이스 서버별로 동일한 이름이 사용되면 안되므로 서버명을 고정으로 사용해서는 안됩니다. Internal ip address로 인해 서버 ip가 중복되는 경우는 데이터베이스 서버명을 그룹 단위로 패턴화하여 적용할 수 있습니다.

Table 2. 패턴 옵션

설정	설명
type	DBVX
ip#	ip를 .으로 나누었을 때 #번째 자리(0부터)
port	데이터베이스 포트
hostname	호스트 명

1.6.3. AES 256 암호화 적용

와탭 DBX 에이전트는 수집된 데이터를 암호화하여 서버로 전송합니다. 데이터의 중요도나 설정에 따라 이를 변경할 수 있습니다. 기본적으로 XOR 연산과 AES 알고리즘을 통한 암호화를 사용하며 평문을 128비트 단위로 나누어 암호화, 복호화를 수행하며, 사용자의 설정에 따라 256비트까지 확장할 수 있습니다.

설정

와탭 DBX 에이전트가 설치된 디렉토리에서 whatap.conf 파일에 아래와 같은 설정을 추가합니다.

cypher_level=256 설정을 추가합니다.

```
license=[라이선스 키]
whatap.server.host=52.78.209.94/52.78.224.235
cypher_level=256
```

재실행

설정을 추가한 후 에이전트를 재기동합니다.

1.6.4. 제약 사항

복수 리전을 단일 프로젝트로 선택 불가

whatap.io 사이트에서 프로젝트 생성 시, 지역(Region)은 중복 선택이 불가하며 복수의 지역(Region)을 활용하는 경우 별도의 프로젝트를 생성해야 합니다.

Cloud 환경과 같이 복수의 지역(Region)에 서버가 존재하는 경우, 네트워크 latency 등 성능 정보 수집 상의 제약 사항을 회피하기 위하여 지역(Region)단위로 수집 서버를 위치시켜야 합니다. 와탭은 고객 요구 사항에 부응하기 위하여 지역(Region)별로 수집 서버를 구축합니다.

에이전트 이름(ONAME)중복 사용 불가

와탭 DBX 모니터링은 에이전트의 식별을 위한 용도로 에이전트의 IP 주소와 Port 정보를 활용하기 때문에 사용자 환경의 모니터링 대상 데이터베이스 서버가 동일 IP, 동일 Port를 사용하는 경우 와탭 서버에서 해당 서버 인스턴스를 구분 할 수 없게 됩니다.

- 서버 인스턴스가 동적으로 확장되는 환경이 아니라면 '서버명을 임의로 부여하여 관리하고 싶은 경우'를 참조하여 데이터베이스 식별을 위한 명칭을 직접 지정하는 방식을 통해 우회 할 수 있습니다.
- 와탭에서는 현재 모니터링 대상 데이터베이스가 internal address 또는 네트워크 가상화로 인해 중복된 IP를 사용할 경우, '서버명을 임의로 부여하여 관리하고 싶은 경우'를 참고하여 별도의 데이터베이스명 패턴을 활용합니다.

1.7. 설치 환경

1.7.1. 운영체제

다음 운영체제 환경에서 에이전트를 실행 할 수 있습니다.

- CentOS 6 이상(64bit)
- Ubuntu 14 이상(64bit)

1.7.2. 지원하는 데이터베이스 종류

DBX서비스에서는 다음과 같은 데이터베이스들을 지원합니다. Java6이상 JDK가설치되어 있어야합니다.

데이터베이스	지원버전
Oracle	Oracle 10g 이상
PostgreSQL	Postgresql 9.2 이상 / EDB PAS 9.2 이상
MySQL	MySQL 5.5 이상 / MariaDB 5.5 이상
MSSQL	SQL Server 2005 이상

Chapter 2. 수집 데이터 종류 및 수집 주기

와탭의 DBX 에이전트는 다양한 데이터베이스들을 지원합니다. 성능분석을 위해 현재 진행중인 스탯 지표, 세션데이터를 수집하며 수집된 데이터들을 저장하여 특정 기간동안의 전체적인 추이를 분석 및 비교 기능을 제공하고 있습니다. 또한 특정 시점에 진행중이었던 세션을 파악하여 느린 Query를 분석할 수 있으며, 수집된 SQL들을 재 가공하여 SQL들의 통계데이터 분석이 가능합니다.

Table 3. 데이터베이스별 지원 버전

데이터베이스	지원버전
Oracle	Oracle 10g 이상
PostgreSQL	Postgresql 9.2 이상 / EDB PAS 9.2 이상
MySQL	MySQL 5.5 이상 / MariaDB 5.5 이상
MSSQL	SQL Server 2005 이상

Table 4. 데이터베이스 종류별 지원 기능

카테고리	데이터베이스	Postgresql	Oracle	Mysql	Mssql
Instance List	Instance List	O	O	O	O
Monitoring	Multi DataBase	O (일부 지표만 multi db stat 가능)	O (12c 이상이고 multi tenant 인 경우)	X	O
	Multi Instance	O	O	O	O
Trends	Trends	O	O	O	O
	Trends Compare	O	O	O	O
	Long Term Trends	O	O	O	O
SQL Stat	SQL Stat	O	O	O	O
Analysis	Lock Tree	O	O	O	O
	PQ Tree	지원 대상 아님	O	지원 대상 아님	지원 대상 아님
	Dead Lock	X	X	O	X
	DB Parameter	O	O	O	O
	Table Info	지원예정	지원예정	O	지원예정
	Table Space	지원예정	O	지원예정	지원예정
	SGA	지원 대상 아님	O	지원 대상 아님	지원 대상 아님
	Scripts	X	X	O	X
	Weekly/Monthly Counter	O	O	O	O
Event	Event List	O	O	O	O
	Event Stat	O	O	O	O
	Event Setting	O	O	O	O
	Notification	O	O	O	O
Report	Daily Report	O	O	O	O
	Weekly Compare Report	X	X	O	X
Management	Agent Setup	O	O	O	O
	Agent List	O	O	O	O
	Project Management	O	O	O	O

Table 5. 데이터 수집 주기

수집 주기	수집 항목
5초(실시간)	stat / event / active session list / lock tree / pq tree<oracle> / process list(xos)
5분	sql stat 누적값 전송
1시간	tablespace <oracle> / sga <oracle> / dead lock <mysql>
6시간	dbnamepack
24시간	table size <mysql> / db parameter / items

Table 6. 주요 공통 지표

active sessions	쿼리를 실행중인 세션 수
total sessions	모든 세션 수
execute count	sql 의 실행 갯수
lock wait sessions	lock 을 wait 하는 세션 수

2.1. Real Time monitoring

Public Cloud, Private Cloud 모두에 적용하기 위해 쿼리 베이스의 모니터링으로 제품을 구성했습니다. 5초 주기의 실시간 데이터 수집으로 데이터베이스의 성능을 다양한 관점에서 모니터링 합니다.

2.1.1. Multi-Server Monitoring

하나의 화면에서 동시에 여러 인스턴스를 그룹화 하여 모니터링 합니다. 관심이 있는 중요 지표들을 선택하여 전반적인 시스템 상황을 모니터링 가능합니다.

2.1.2. Instance List

에이전트가 설치된 데이터 베이스서버들의 목록을 보여줍니다. 주요 성능 지표 및 상태(Status)를 보여줄 수 있도록 구성되어 있습니다. 성능 지표는 5초 주기로 갱신되며 목록의 컬럼은 설정을 통하여 원하는 컬럼들을 정의할 수 있습니다.

2.1.3. Monitoring(Multi-Database)

실시간 모니터링은 일정 주기(5초)로 성능 지표들을 모니터링합니다. 현재 진행중인 액티브 세션, 락 정보, 프로세스 정보를 확인 할 수 있습니다.

2.1.4. Monitoring(Multi-Instance)

프로젝트에 등록된 모든 인스턴스들을 5초 주기로 실시간 모니터링 합니다. 또한 특정 인스턴스 별로 필터링하여 모니터링 할 수 있습니다.

2.1.5. Monitoring Cloud Watch 활용

아마존 RDS인 경우 Cloud Watch API를 통하여 AWS에서 제공하는 정보를 함께 볼 수 있습니다. 수집주기는 최소 1분 입니다.

2.2. Active Session

현재 진행중인 세션들의 정보(Query포함)를 확인 합니다. 10초이상 진행되고 있는 세션인 경우 노란색, 20초 이상의 세션인 경우 빨간색으로 표현됩니다. Query의 플랜정보를 함께 확인 할 수 있습니다.

2.2.1. Process

XOS에이전트를 설치 한 경우, 데이터베이스 서버의 프로세스에 대한 정보를 확인 할 수 있습니다.

2.3. Trends

수집된 정보를 바탕으로 특정 하루 동안의 전체적인 추이를 분석할 수 있는 기능을 제공하고 있습니다. 특정 시점에 진행중이었던 세션을 파악하여, 느렸던 Query를 분석할 수 있습니다.

서로 다른 일자에 대한 트렌드 데이터를 비교하여 데이터베이스의 성능 추이를 비교합니다.

하루 이상의 장기 트렌드를 분석합니다.

2.4. SQL Stat

하루 동안 수집된 데이터에서 Top N 쿼리를 보여주고 분석할 수 있는 기능을 제공하고 있습니다.

인스턴스별로 수집된 SQL을 DB, Schemaname, Machine 등으로 카테고리화 하여 상세한 조회가 가능합니다.

SQL PLAN 정보를 확인할 수 있습니다. MySQL, Postgres의 경우에는 데이터베이스 내에서 플랜 정보 변경에 대한 확인이 어렵지만 SQL통계기능을 통해 Top-N쿼리에 대한 플랜 정보를 자동으로 수집할 수 있습니다.

2.5. 데이터베이스 모니터링 설정

해당 가이드는 데이터베이스 모니터링을 할 때에 설정할 수 있는 옵션에 대해 설명합니다. (*) 표시가 있는 옵션은 애플리케이션을 재시작 하여야만 적용되는 옵션입니다.

2.5.1. 에이전트 네트워크 통신에 관한 설정

license

Default : NONE

Type : String

에이전트 설치시 서버로부터 부여받은 라이선스를 지정합니다. 라이선스에는 에이전트가 속한 프로젝트와 보안 통신을 위한 암호키를 포함하고 있습니다.

whatap.server.host

Default : 127.0.0.1,127.0.0.1

Type : String

에이전트가 수집한 데이터를 전송할 서버를 지정합니다. 수집서버 이중화로 2개 이상의 IP를 가진 경우 콤마(,)로 분리하여 지정할 수 있습니다. 지정된 IP 에는 수집서버 proxy 데몬이 리스닝 상태로 서비스 되어야 합니다.

whatap.server.port

Default : 6600

Type : String

수집서버 PORT 를 지정합니다. 포트는 하나만 지정할 수 있으므로 whatap_server_host 에 지정된 수집서버들은 동일 PORT 를 사용해야 합니다.

dbms

Defalut: None

Type : String

모니터링 할 데이터베이스의 종류를 지정합니다.

DB 의 종류는 다음중 한가지를 지정합니다.

oracle, postgresql, mysql, mssql

db_ip

Defalut: None

Type : String

모니터링 할 데이터베이스 서버의 ip 를 지정합니다.

db_port

Defalut: None

Type : Int

데이터베이스가 통신에 사용하는 포트를 지정합니다.

각 데이터베이스별 기본 포트는 다음과 같습니다.

oracle: 1521

postgresql: 5432

edb: 5444

mysql: 3306

mssql: 1433

#db

Defalut: None

Type : String

데이터베이스 접속에 필요한 db 명을 지정합니다.

oracle 의 경우 SID 를 지정합니다.

postgresql 과 edb 의 경우 지정하지 않으면, 기본값으로 postgres 의 db 를 사용합니다.

mssql 의 경우 지정하지 않으면, 기본값으로 master 의 db 를 사용합니다.

mysql 의 경우는 지정하지 않아도 됩니다.

2.5.2. AWS RDS Cloud Watch 사용에 대한 설정

aws_region

Default : None Type : String

AWS 리전을 지정합니다.

aws_access_key

Default : None

Type : String

AWS 액세스 키를 지정합니다.

aws_secret_key

Default : None

Type : String

AWS 시크릿 키를 지정합니다.

cloud_watch

Default : None

Type : String

cloud watch 모니터링의 유무를 지정합니다. 지정값은 on 또는 off 입니다.

cloud_watch_instance

Default : None

Type : String

cloud watch 인스턴스 명을 지정합니다.

cloud_watch_interval

Default : None

Type : Second

cloud watch 로 모니터링 값을 가져올 주기를 지정합니다. (단위: 초)

cloud_watch_metrics

Default : None

Type : String

cloud watch 로 가져올 모니터링 지표 이름을 지정합니다.

다음의 값을 지정할 수 있습니다.

cloud_watch_metrics=CPUUtilization,FreeableMemory,FreeStorageSpace,NetworkTransmitThroughput,NetworkReceiveThroughput

2.5.3. 애플리케이션 등록에 관한 설정

object_name

Default : {type}-{ip2}-{ip3}-{port}

Type : String

애플리케이션을 식별하는 에이전트 이름(ONAME) 체계입니다. ONAME을 토대로 OID가 생성됩니다.

- remark: 재기동 필요 (Apache 및 PHP-FPM)

명칭	설명
{type}	whatap.app_name 에 설정된 값을 사용합니다.
{ip#}	IP를 나누었을 때 #번째 자리를 사용합니다.
{port}	whatap.app_port_name 에 설정된 값을 사용합니다.
{hostname}	서버 호스트명을 사용합니다.

auto_ongame_enabled

Default : false

Type : Boolean

서버에 등록될 에이전트 이름(ongame)을 서버로부터 자동 부여 받는 기능을 활성화 합니다. 적용 시, -Dwhatap.name, -Dwhatap.ongame 옵션은 무시됩니다. 수집 서버와의 통신을 통해 ongame 을 부여 받은 이후, 에이전트의 일반적인 동작을 개시합니다.

auto_ongame_prefix

Default : agent

Type : String

에이전트 이름을 서버로부터 자동 부여할 때 에이전트 이름의 prefix, 보통 업무 명을 사용합니다. prefix 일련번호 1~) 부여됩니다.

auto_ongame_reset

Default : 0

Type : Int

서버로부터 새로운 에이전트 이름을 부여받기 위해서 수정합니다. 에이전트 이름을 자동 부여하면 `what.oname` 이라는 시스템 환경 변수에 셋트됩니다. 한번 셋트되면 자바 인스턴스가 재기동 될 때까지 유지 되는데 리셋을 원할 때 `auto.oname_reset` 값을 수정합니다. (현재 설정 값과 다른 값으로 변경하면 적용됩니다.)

2.5.4. 에이전트 데이터 수집에 관한 설정

multi_db_stat

Default: 1

Type : Int

multi db 의 stat 정보 수집 처리 유무를 지정합니다.

oracle (pluggable db 인 경우) 과 postgresql 에서만 유효합니다.

db instance 내의 각 db 별로 stat 을 수집할지의 여부를 지정하게 되며,

수집 부하를 줄이기 위해 0 으로 지정할 수 있습니다.

long_run_session_sec

Default: 10

Type : Second

지정한 시간보다 길게 수행되고 있는 세션을 long run 세션으로 처리합니다. (단위: 초)

deadlock_interval

Default: 1

Type : Int

mysql 에서 deadlock 정보를 가져오는 주기를 지정합니다. (단위: 분) deadlock 정보는 주기동안 발생했던 가장 마지막의 deadlock 정보만을 가져옵니다.

tables_hour

Default: 5

Type : Int

테이블 정보를 저장하기 위해 정보를 수집하는 시각을 지정합니다. 기본값으로 5 를 사용하며, 새벽 5시에 수집을 하게 됩니다.

tables_min_row

Default: 10000

Type : Int

테이블 정보 수집시 지정한 값보다 작은 row 수를 가진 테이블 정보는 수집하지 않습니다.

tables_exclude_sys

Default: false

Type : Boolean

테이블 정보 수집시 sys 소유인 테이블 정보의 수집 유무를 지정합니다.

true 로 지정하면 sys 소유의 테이블 정보는 수집하지 않습니다.

stat

Default: 하단참조

Type : String

oracle 모니터링 정보중, 수집처리할 stat 지표들을 콤마로 구분하여 지정합니다.

다음의 값을 기본값으로 사용합니다.

stat=logons current,user commits,session logical reads,CPU used by this session,DB time,enqueue waits,physical reads,physical writes,execute count,db block changes,opened cursors current,parse count (total),parse count (hard),sorts (memory)

event

Default: 하단참조

Type : String

oracle 모니터링 정보중, 수집처리할 event 지표들을 콤마로 구분하여 지정합니다.
다음의 값을 기본값으로 사용합니다.
event=db file sequential read,db file scattered read,enq: TX – row lock contention

exclude_background

Default: 3

Type : String

oracle 모니터링 정보중, 세션 갯수 및 세션 리스트 수집시 background 세션의 제외 유무를 지정합니다.

0 : 제외하지 않음.

1 : 세션 리스트 수집시 background 세션 제외.

2 : 세션 갯수 처리시 background 세션 제외.

3 : 세션 리스트와 갯수 처리시 background 세션 제외.

my_opt

Default: 0

Type : Int

mysql 의 data 디렉토리 권한에 따라 performance_schema 가 존재 및 사용 가능 상태임에도 불구하고 사용하지 못할 경우가 있으며, 이때에는 dbx 에이전트가 정상 동작하지 않습니다.

이때에 performance_schema 를 사용하지 않도록 하려면 1 을 지정합니다.

2.5.5. XOS에이전트 설정에 관한 설정

xos

Default: 0

Type : String

xos 에이전트 사용 여부를 지정합니다.

0 인 경우 xos 에이전트 수집 처리를 하지 않습니다.

1 또는 xos 에이전트의 ip 를 적어 지정합니다.

ip 를 지정할 경우, 지정한 ip 의 xos 에이전트로부터만 수집 처리를 합니다.

xos_port

Default: 3002

Type : String

xos 에이전트와 통신할 포트를 지정합니다.

xos 에이전트와는 udp 로 통신을 하게 되며,

방화벽등이 설정 되어 있을 경우 지정한 포트의 udp 통신이 될수 있도록 설정해 주어야 합니다.

xos_cpu_iowait

Default: 1

Type : Int

xos 로부터 수집한 cpu 사용률에 iowait 의 값을 포함할지를 지정합니다.

1로 지정할 경우 iowait 를 포함하고, 0 으로 지정시 iowait 를 포함하지 않습니다.

xos_mem_buffcache

Default: 0

Type : Int

xos 로부터 수집한 memory 사용률에 buffcache 의 값을 포함할지를 지정합니다.

1로 지정할 경우 buffcache 를 포함하고, 0 으로 지정시 buffcache 를 포함하지 않습니다.