



Go Agent v0.1.9

배포일: 2022-04-27

베타 - 버그 수정

panic 상황에서 웹 트랜잭션 정보가 수집 안되는 오류 수정

- net/http
- github.com/labstack/echo
- github.com/gorilla/mux
- github.com/go-gonic/gin

! `recover()` 를 통해서 오류 정보를 수집하지만, 다시 동일한 panic를 발생시킵니다.

Go

```
defer func() {  
    // panic  
  
    x := recover()  
    var err error = nil  
    if x != nil {  
        err = fmt.Errorf("Panic: %v", x)  
  
        // panic 정보 수집  
        trace.Error(ctx, err)  
        err = nil  
    }  
  
    ...  
  
    // 다시 panic 발생  
  
    if x != nil {  
        panic(x)  
    }  
}
```

베타 - 업데이트

http status code 수집

400 이상의 status code는 오류메세지로 수집합니다.

github.com/labstack/echo

echo/v4 지원

기존 echo/whatapecho 패키지 대신에 echo/v4/whatapecho를 사용합니다.

Go

```
import (  
    "github.com/labstack/echo"  
  
    "github.com/whatap/go-api/trace"  
    "github.com/whatap/go-api/instrumentation/github.com/labstack/echo/v4/whatapecho"  
)
```

echo의 HTTPErrorHandler Wrapper 추가

HTTPErrorHandler를 통해서 오류 정보를 추가로 수집합니다.

`echo.Context.Error()` 함수가 호출될 때 오류 정보를 수집합니다. Recover 미들웨어가 처리한 panic 정보를 수집합니다.

whatapecho 미들웨어가 마지막으로 실행 되면 HTTPErrorHandler 정보를 최대한 많이 수집할 수 있습니다. 미들웨어 등록을 최상단에서 설정하고, `Pre()` 함수를 통해 등록합니다.

Go

```
func main() {  
    ...
```

```
e := echo.New()
//echo DefaultHTTPErrorHandler wrapper
e.HTTPErrorHandler = whatapecho.WrapHTTPErrorHandler(e.DefaultHTTPErrorHandler)
e.Pre(whatapecho.Middleware())
e.Use(middleware.Recover())
...
}
```

로그 출력

debug 옵션을 설정하면 모니터링 관련된 로그를 출력합니다.

```
debug=true
```

트랜잭션, DB 커넥션, Sql 실행, Http 외부 호출에 대한 로그를 추가했습니다.

Http Method를 지정해서 수집을 제외할 수 있는 설정 추가

지정한 uri 에 대해서 특정 HTTP Method로 요청되는 트랜잭션의 수집을 제외합니다. 콤마(,) 구분으로 여러개의 uri, Method를 등록할 수 있습니다. 대소문자 구분은 없습니다.

- ignore_http_method_urls

```
#Default : ""#
```

```
#Type : String#
```

- ignore_http_method

```
#Default : ""#
```

```
#Type : String#
```

```
ignore_http_method_urls=/index, /sql/select
ignore_http_method=options, connect
```