

2025.09.12

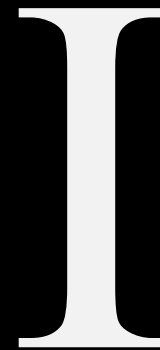
Web Application Tech

가디언 웹 보안 세미나 1

Guardian 2025

HTTP/HTTPS

Requests



HTTP

Request

```
GET /auth/488/YourDetails.ashx?uid=129 HTTP/1.1
Accept: application/x-ms-application, image/jpeg, application/xaml+xml,
image/gif, image/pjpeg, application/x-ms-xbap, application/x-shockwave-
flash, */*
Referer: https://mdsec.net/auth/488/Home.ashx
Accept-Language: en-GB
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64;
Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR
3.0.30729; .NET4.0C; InfoPath.3; .NET4.0E; FDM; .NET CLR 1.1.4322)
Accept-Encoding: gzip, deflate
Host: mdsec.net
Connection: Keep-Alive
Cookie: SessionId=5B70C71F3FD4968935CDB6682E545476
```

Hyper Text Transfer Protocol

개쩌는 텍스트 전송 규약

HTTP

Request

GET /auth/488/YourDetails.ashx?uid=129 HTTP/1.1

Accept: application/x-ms-application, image/jpeg, application/xaml+xml, image/gif, image/pjpeg, application/x-ms-xbap, application/x-shockwave-flash, */*

Referer: https://mdsec.net/auth/488/Home.ashx

Accept-Language: en-GB

User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1; WOW64; Trident/4.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; .NET4.0C; InfoPath.3; .NET4.0E; FDM; .NET CLR 1.1.4322)

Accept-Encoding: gzip, deflate

Host: mdsec.net

Connection: Keep-Alive

Cookie: SessionId=5B70C71F3FD4968935CDB6682E545476

HTTP

Response

```
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 09:23:32 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Set-Cookie: tracking=tI8rk7joMx44S2Uu85nSWc
X-AspNet-Version: 2.0.50727
Cache-Control: no-cache
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 1067
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html xmlns="http://
www.w3.org/1999/xhtml" ><head><title>Your details</title>
...
```

HTTP

Response

```
HTTP/1.1 200 OK
Date: Tue, 19 Apr 2011 09:23:32 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Set-Cookie: tracking=tI8rk7joMx44S2Uu85nSWc
X-AspNet-Version: 2.0.50727
Cache-Control: no-cache
Pragma: no-cache
Expires: Thu, 01 Jan 1970 00:00:00 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 1067
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://
www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html xmlns="http://
www.w3.org/1999/xhtml" ><head><title>Your details</title>
...
```

HTTP

Methods

GET

데이터 주세요

Note.

GET, POST는 거의 저 용도 고정
PUT/DELETE는 안쓰는 곳도 많음

POST

뭐 해주세요

Note.

이것들 말고도 HEAD, TRACE 등
많지만, 거의 안 쓰임

PUT/DELETE

자료 수정 / 삭제할래요

URL

scheme://<user>:<password>@<host>:<port>/<url-path>

https://google.com/

https 프로토콜로 youtube 호스트로 접속

ftp://user@host/foo/bar.txt

Ftp란 프로토콜로 user로 host에 접속하여 foo path 아래 bar.txt 접근

https://google.com@naver.com

IPV6 http://[::1]/index.html

HTTP

Headers – Request

Accept

받을 데이터 양식

HOST

URL의 HOST 이름

Authorization

Auth 관련 데이터

Referer

Request의 원래 출처지

Cookie

쿠키 (곧 설명)

User-Agent

내가 누군지

HTTP

Headers – Response

Location

어디로 가주세요

Expires

캐싱 기간

Server

서버이름

COOP

이 페이지가 다른 도메인(origin)을 참조 가능?

Set-Cookie

쿠키 발급

ACAO

다른 도메인이 서버 자원을 호출 가능?

Cross Origin Opener Policy. 나중에 다룸
Access Control Allow Origin. 나중에 다룸

HTTP

Headers

Q. 다 외워야 하나요?

A. 아니지만 굉장히 자주 보게 된다

강조한 것만 알아두면 문제 없음

HTTP

Cookie

Set-Cookie

Set-Cookie: sessionId=a1; HttpOnly; Path=/; Secure; SameSite=Lax

HttpOnly/Secure

Js는 쿠키접근 불가 / https만 허용

Domain / path

Domain/path 아래에서만 유효

HTTP

Status Codes

1xx: hold on

2xx: here you go

3xx: go away

4xx: you fucked up

5xx: I fucked up

-via @abt_programming

HTTP

Status Codes

| | |
|--------------------|----------|
| 1xx: hold on | 나중에 줄게 |
| 2xx: here you go | 정상동작 |
| 3xx: go away | Redirect |
| 4xx: you fucked up | 잘못된 접근 |
| 5xx: I fucked up | 잘못된 서버 |

-via @abt_programming

HTTPS

HTTP Secure

HTTP에 SSL이 추가된 구조

Secure Socket Layer

DNS 스푸핑이 있더라도 비교적 안전

HTTPS

간단한 동작 방식

1. SSL key 교환

컴퓨터 어딘가 `sslkeylogfile` 이 있음

2. 암호화 후 통신

모든 HTTP 를 암호화

HTTPS

간단한 동작 방식

Wireshark · Follow TCP Stream (tcp.stream eq 4) · Wi-Fi

| | | | | |
|----------|-------------------------|-------------------------|-------------------------|-----------|
| 00000000 | 16 03 | 01 04 de 01 00 04 | da 03 03 16 68 36 75 45 |h6uE |
| 00000010 | 2e c9 c1 e6 be 6f d0 e6 | b1 da 60 f1 78 df f6 56 |o..`..x..V | |
| 00000020 | 5b ee 7f f7 4c dc 1f 5e | e6 8f 52 20 70 d8 0e 83 | [...L..^..R p... | |
| 00000030 | c8 f6 bf 80 66 24 93 63 | 49 fa d4 34 25 9d 7c 72 | ...f\$.c I..4%. r | |
| 00000040 | c8 23 11 90 87 d3 af ce | 2c 12 a1 f3 00 22 13 01 | .#.....,"... | |
| 00000050 | 13 03 13 02 c0 2b c0 2f | cc a9 cc a8 c0 2c c0 30 |+./.....,0 | |
| 00000060 | c0 0a c0 09 c0 13 c0 14 | 00 9c 00 9d 00 2f 00 35 |/..5 | |
| 00000070 | 01 00 04 6f 00 00 00 21 | 00 1f 00 00 1c 6f 67 61 | ...o...!.....oga | |
| 00000080 | 64 73 2d 70 61 2e 63 6c | 69 65 6e 74 73 36 2e 67 | ds-pa.cl ients6.g | |
| 00000090 | 6f 6f 67 6c 65 2e 63 6f | 6d 00 17 00 00 ff 01 00 | oogle.co m..... | |
| 000000A0 | 01 00 00 0a 00 0e 00 0c | 00 1d 00 17 00 18 00 19 | | |
| 000000B0 | 01 00 01 01 00 0b 00 02 | 01 00 00 10 00 0e 00 0c | | |
| 000000C0 | 02 68 32 08 68 74 74 70 | 2f 31 2e 31 00 05 00 05 | .h2.http /1.1... | |
| 000000D0 | 01 00 00 00 00 00 22 00 | 0a 00 08 04 03 05 03 06 |" | |
| 000000E0 | 03 02 03 00 12 00 00 00 | 33 00 6b 00 69 00 1d 00 |3.k.i... | |
| 000000F0 | 20 ae 5f e0 1a 53 11 1c | 6b 3d 0e a8 0a bf be 05 | ...S.. k=..... | |
| 00000100 | 3b 47 6d d8 48 52 1e a0 | 4f e3 62 82 04 fd 2d fe | ;Gm.HR.. O.b...- | |
| 00000110 | 11 00 17 00 41 04 ae 7d | f8 63 96 19 95 80 e2 00 | ...A..} ..c..... | |
| 00000120 | 8c 6b 87 b0 12 df 89 9e | bd 5e bc ab 15 39 b7 8c | .k.....^..9.. | |
| 00000130 | aa fe 4e fe c8 59 08 5c | 1f db 13 a2 66 94 66 00 | ..N..Y.\f.f. | |
| 00000140 | e5 b7 bc 35 0a 78 59 94 | b2 03 49 55 bd 1d 0e ea | ...5.xY. ..IU.... | |
| 00000150 | 35 d7 7f 6e 37 2e 00 2a | 00 00 00 2b 00 05 04 03 | 5..n7..* ...+.... | |
| 00000160 | 04 03 03 00 0d 00 18 00 | 16 04 03 05 03 06 03 08 | | |
| 00000170 | 04 08 05 08 06 04 01 05 | 01 06 01 02 03 02 01 00 | | |
| 00000180 | 2d 00 02 01 01 00 1c 00 | 02 40 01 fe 0d 02 39 00 | -..... .@....9. | |
| 00000190 | 00 01 00 03 75 00 20 7f | c2 3a a2 5a 4f 2a 4f ab |u. . ..ZO*O. | |
| 000001A0 | 2c 82 e4 03 cb 71 66 7b | 2a 89 37 5d 49 b5 20 93 | ,....qf{ *.7]I. . | |
| 000001B0 | 42 e6 2a 66 bc f7 6b 02 | 0f 3a c3 b8 09 6a cd 44 | B.*f..k.i.D | |

10 client pkts, 7 server pkts, 4 turns.

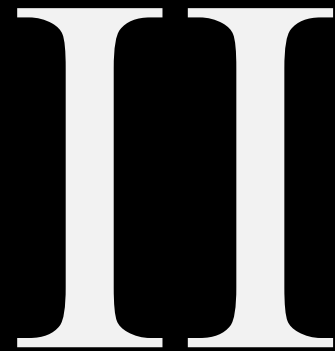
Entire conversation (7244 bytes) Show as Hex Dump No delta times Stream 4

Find: Case sensitive Find Next

Filter Out This Stream Print Save as... Back Close Help

BuildingBlocks

Gateway / WA / Containers



Nginx

Web Server + reverse proxy + ...

Web Server

정적인 파일들 (이미지, HTML, CSS, JS)을 빠르게 부려주는 역할

Reverse Proxy

들어오는 요청을 받아서 뒤쪽의 여러 서버에 나눠주는 Load Balancer 역할

Gateway

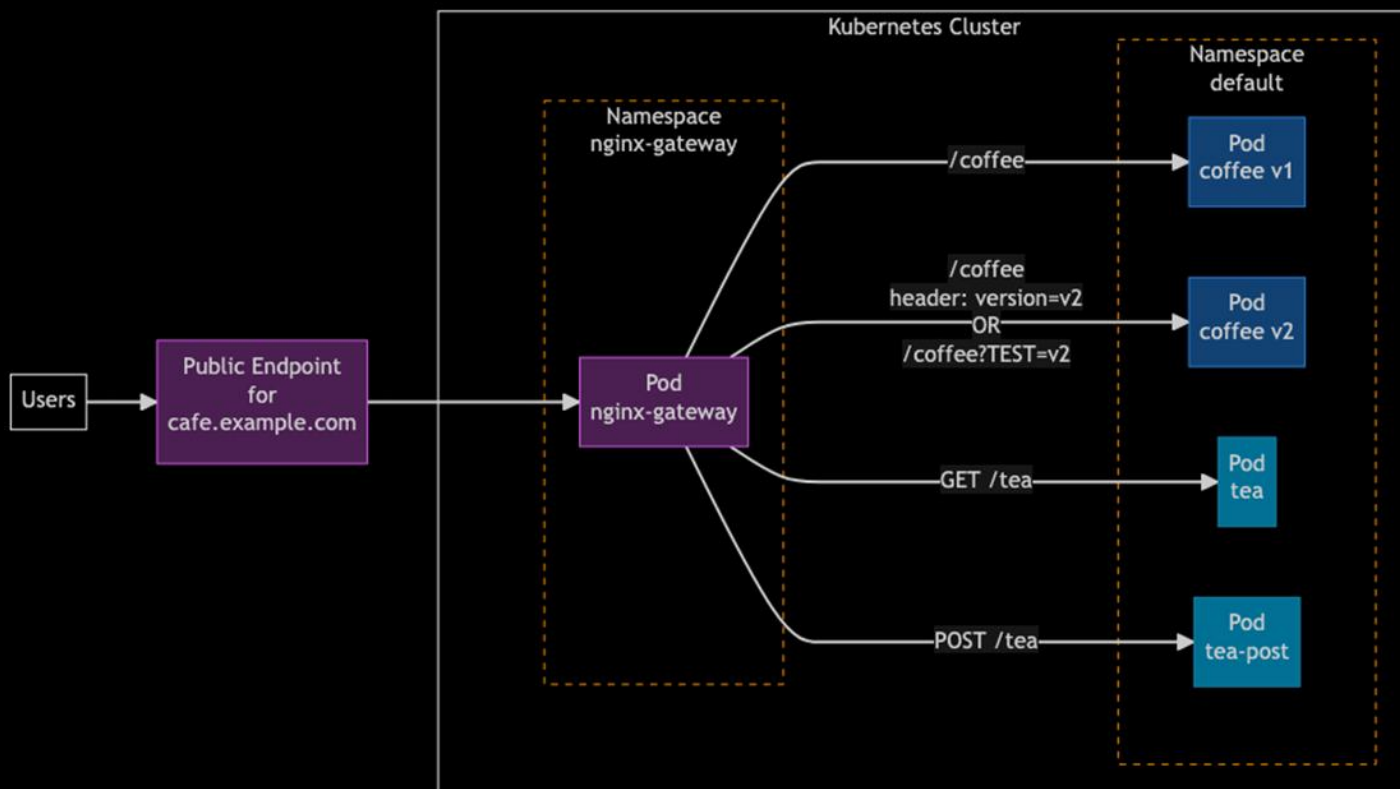
SSL(https 암호화), 캐싱, 압축, 보안 필터링 등 미리 처리

[Engine X]

Nginx 말고 Apache[아파치]도 많이 쓴다

Nginx

Diagram



FrontEnd

Client-Side + Server-Side Rendered

HTML, CSS, JS를 전달하는 곳

웹 3요소

UI/UX

브라우저에서 실행

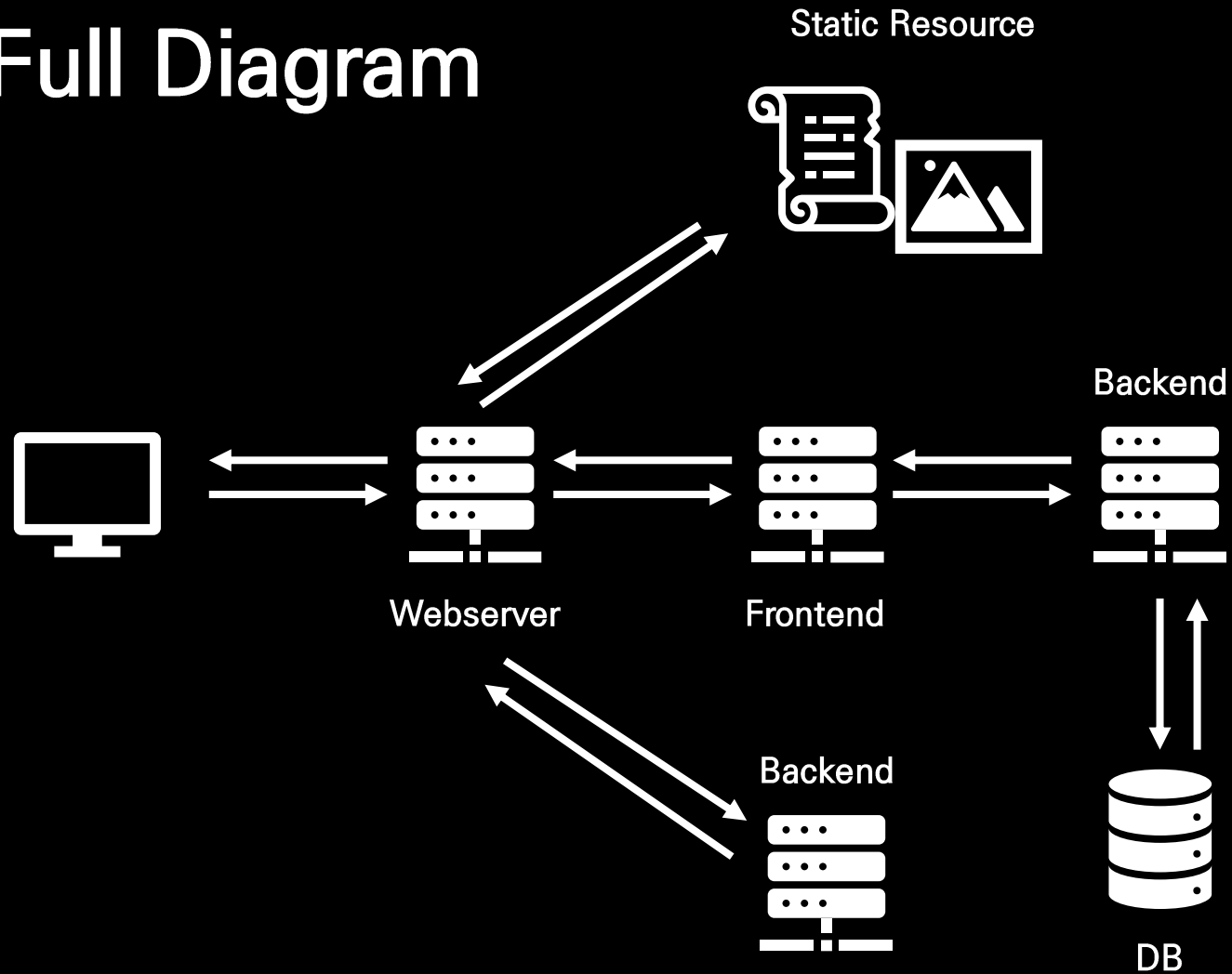
BackEnd

Server Logic

서버 로직 처리

DB 연동

Full Diagram



1cm-1cm 떨어진 주식 12px

1cm-1cm 떨어진 주식 12px

사용 언어? 프레임워크?

js, python, java, php, ...

어차피 비슷하기에 그때그때 문법을 알아두자.

DB

SQLs (mySQL, pSQL, SQLite), noSQLs (MongoDB, redis)

Next, express, flask, spring, ...

구조가 다 거기서 거기

1cm-1cm 떨어진 주석 12px

1cm-1cm 떨어진 주석 12px

Container

Docker

VM 같은거

정확히는 가상화 기술 종류에 Containers와 VM이 있는 것. VM이 더 무거움

어디서든 동일한 환경 제공

OS level 의 가상화라서 OS가 달라도 괜찮음

OS 가상화이기에 하드웨어가 다르면 작동 안 함

ex - CISC에서 빌드한 컨테이너를 AMD64에서 사용

Docker

Container

어떤 OS에서 동작할거예요?

이후 비어있는 윈도우/리눅스/이것저것 생성

만든 OS에서 어떤 파일들을 쓸거예요?

“비어있는” OS

만든 OS의 어떤 포트를 쓸거예요?

“비어있는” OS

“.dockerfile” 이라는 이름의 파일로 작성한다

```
1  # Use an LTS base image for stability and security updates
2  FROM node:18-alpine AS base
3
4  # Create and use non-root user for better security
5  RUN addgroup -S app && adduser -S app -G app
6  WORKDIR /app
7
8  # Install only production dependencies using cached layer
9  # 1) copy package manifests first to leverage build cache
10 COPY package*.json ./
11 RUN npm ci --omit=dev
12
13 # 2) copy the rest of the source code
14 COPY . .
15
16 # Document the port the app listens on
17 EXPOSE 3000
18
19 # Drop privileges
20 USER app
21
22 # Run the server
23 CMD ["node", "server.js"]
24
```

요?

```
1  # Build an image from Dockerfile in the current directory
2  docker build -t myapp:1.0 .
3
4  # Run a container (publish host port 3000 -> container 3000)
5  docker run -d --name myapp -p 3000:3000 myapp:1.0
6
7  # See running containers
8  docker ps
9
10 # Tail logs (Ctrl+C to stop tailing)
11 docker logs -f myapp
12
13 # Shell into a running container (useful for debugging)
14 docker exec -it myapp sh    # or bash if available
15
16 # Stop and remove the container
17 docker stop myapp && docker rm myapp
18
19 # Remove dangling images and reclaim space
20 docker image prune -f
21 docker system df
22
```

Docker

Container

어떤 OS에서 동작할거예요?

이후 비어있는 윈도우/리눅스/이것저것 생성

만든 OS에서 어떤 파일들을 쓸거예요?

“비어있는” OS

만든 OS의 어떤 포트를 쓸거예요?

“비어있는” OS

“.dockerfile” 이라는 이름의 파일로 작성한다

Docker-compose

Containers

Dockerfile 여러 개 한 번에 관리

일단 여러 개를 하나로 묶어준다

만든 OS에서 어떤 파일들을 어떻게 쓸거예요?

.dockerfile '들' 실행할 때 매번 cd 치기 귀찮음

만든 OS의 어떤 포트를 어떻게 쓸거예요?

.dockerfile '들' 실행할 때 -p 8080:3000 치기 귀찮음

“docker-compose.yml” 이라는 이름의 파일로 작성한다

D
Con

```
1  version: "3.9"
2  services:
3    web:
4      build: .
5      ports:
6        - "8080:3000"      # host 8080 → container 3000
7      volumes:
8        - ./data:/app/data
9      environment:
10       - NODE_ENV=production
11      depends_on:
12        - db                # db first
13
14    db:
15      image: postgres:16
16      environment:
17        - POSTGRES_USER=appuser
18        - POSTGRES_PASSWORD=secret
19        - POSTGRES_DB=mydb
20      volumes:
21        - dbdata:/var/lib/postgresql/data
22
23  volumes:
24    dbdata:
```

“docker-co

에요?

?

Docker-compose

Containers

만약 `.docker-compose.yml` 이 보인다?

아물따 `docker compose up -d`

종료하고 싶으면?

Gpt한테 물어보세요 (`docker compose down || docker stop ~~`)

“`docker-compose.yml`” 이라는 이름의 파일로 작성한다

Guardian 2025

2025.09.12

질문받습니다
질문이 있다면 하십시오

Guardian 2025

2.5cm-3.5cm 떨어진 제목 36px

제목 하단의 부제목 18px

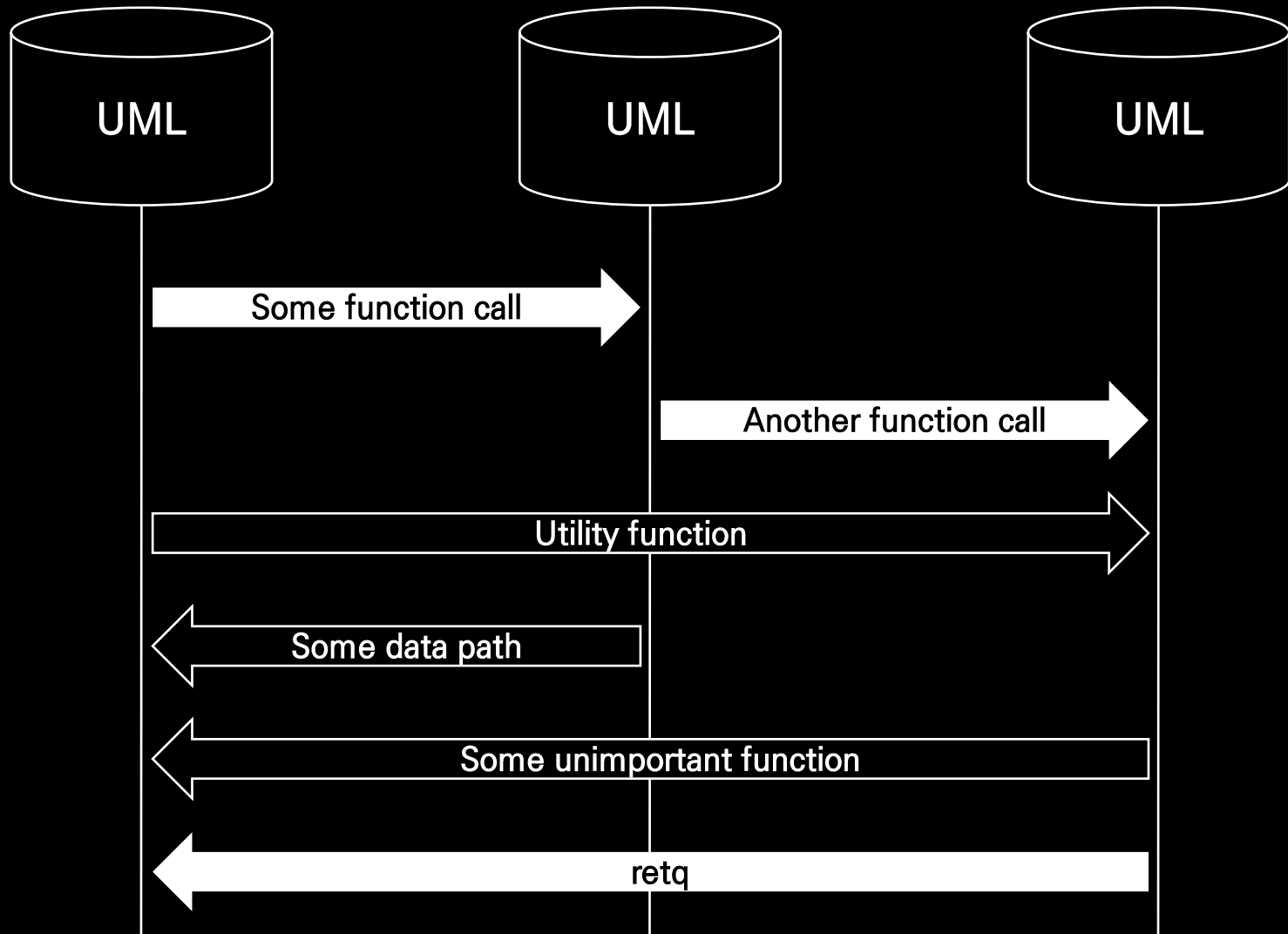
3.5cm 떨어진 내용 1 32px

| | |
|-------------------------|----------------------------|
| Git init | Git add . |
| Git status | Git reset . |
| Git add text.txt | Git commit -m "add README" |
| Git add . | Git log --oneline -n 3 |
| Git commit | Git commit -a -m "hello" |
| Ctrl+C | |
| Git commit -m "genesis" | |
| Git log | |
| Git log --oneline | |

1cm-1cm 떨어진 주석 12px

1cm-1cm 떨어진 주석 12px

중심에서 0.3cm 떨어진 소속 18px



중심에서 0.3cm 떨어진 소속 18px