HTML

한글 폰트 설치 : sudo apt install fonts-nanum*

백문이불여일견

百聞而不如一見

백문이불여일견

백 번 듣는 것이 한 번 보는 것보다 못하다

백문이불여일견

백 번 듣는 것이 한 번 보는 것보다 못하다

백문이불여일타

백 번 듣는 것이 한 번 코딩해 보는 것보다 못하다

HTML

HTML 개요

- HyperText Markup Language 의 약자
  - HyperText : 문서와 문서가 링크로 연결됨
  - Markup : 태그로 이루어짐
  - Language : 언어
- 웹 페이지를 만들기 위한 언어 (확장자 .html)
- 웹 브라우저에서 동작하는 언어
- HTML은 일련의 요소(element)로 구성, 요소는 태그(tag)로 표시됨
- 브라우저는 태그를 표시하지 않고, 태그를 사용하여 페이지 렌더링
  - 렌더링 : 서버로 부터 파일을 받아 브라우저에서 출력하는 과정

<내용> : tag


```
<!DOCTYPE html>
<html>
    <head>
        <title>Title</title>
    </head>
    <body>
        <h1>Heading</h1>
        <p>paragraph</p>
    </body>
</html>
```

HTML

HTML 태그(tag)

- HTML 태그는 홑화살괄호 < > 로 둘러싸인 요소 이름
- 일반적으로 와 같이 쌍으로 사용
  - 앞의 는 시작 태그
  - 끝의 는 종료 태그
- 시작 태그만 있고 태그가 없는 태그도 존재
  - <img>,
, <hr> 등과 같은 태그로, 빈 태그(empty tag)라고 함

```
<!DOCTYPE html>
<html>
    <head>
        <title>Title</title>
    </head>
    <body>
        <h1>Heading</h1>
        <p>paragraph</p>
    </body>
</html>
```

HTML

HTML 요소(element)

- 시작 태그와 종료 태그로 구성되며, 내용은 사이에 삽입됨
- 시작 태그와 종료 태그까지의 모든 것을 뜻함
- 여러 속성(attribute) 을 가질 수 있으며, 속성은 요소에 대한 추가 정보를 제공

```
<a href="http://www.boanproject.com/">boanproject</a>
```

a	: 태그 이름
<a>	: 시작 태그
	: 종료 태그
href	: a 태그의 속성 이름
http://boanproject.com/	: 속성 값
google	: 내용

HTML

HTML 속성(attributes)

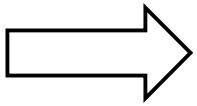
- HTML 요소에 대한 추가 정보를 제공
- 모든 HTML 요소는 속성을 가질 수 있음
- 속성은 항상 시작 태그에 지정되어 사용됨
- 일반적으로 이름/값의 쌍으로 제공되어 사용 (name="value")

```
<a href="http://www.boanproject.com/">boanproject</a>

a                          : 태그 이름
<a>                        : 시작 태그
</a>                       : 종료 태그
href                       : a 태그의 속성 이름
http://boanproject.com/    : 속성 값
google                     : 내용
```

HTML 기본 구조

```
<!DOCTYPE html>
<html>
    <head>
        <title>Page Title</title>
    </head>
    <body>
        <h1>Heading</h1>
        <p>paragraph</p>
    </body>
</html>
```



<!DOCTYPE html>
이 문서를 HTML5로 정의

<html>
HTML 페이지의 최상단(root) 요소

<head>
문서의 메타데이터를 정의
(문서에 대한 제목, 스크립트 등)

<body>
눈으로 보이는 페이지 내용(content)

<h1> ~ </h6>
제목(heading)을 지정

<p>
단락(paragraph) 지정

HTML 실습

- 제목
  - <h1>Heading 1</h1>
  - <h2>Heading 2</h2>
  - <h3>Heading 3</h3>

- 단락
  - Paragraph
  - Pparagraph2

- 줄바꿈
  -
br tag 1
  -
br tag 2

HTML 실습

- 주석
  - 프로그래밍에 있어 내용을 메모하는 목적으로 쓰임
  - 개발자가 작성한 코드에 대한 이해를 돕는 설명이나 디버깅을 위해 작성한 구문

  - <!-- 주석내용 -->

HTML 실습

링크

- This is a link

-
  <h2>링크 클릭</h2>

HTML 실습

이미지

- src 속성 : 이미지 소스 파일 이름 지정
- width 및 height 속성 : 이미지의 너비와 높이를 지정

- <img src="이미지 주소">
- <img src=" 이미지 주소" width="500" height="300">

HTML 실습

HTML 엔티티

- HTML에서 예약된 문자(예약어)
- HTML 예약어를 의미 그대로 사용하기 위해 별도로 만든 문자셋을 엔티티라고 함
- HTML에서 특수문자를 읽으면 문자로 인식하지 않고 태그로 인식한다. 따라서 문자로 인식하여 표현하기 위한 것
- https://dev.w3.org/html5/html-author/charref

```
<p><p>태그는 단락을 표현</p>

<p>&lt;p&gt;태그는 단락을 표현 </p>
```

변환 대상	변환 값	변환 대상	변환 값
<	<	)	)
>	>	(	(
#	#	&	&
"	"	'	'
/	/		

HTML 실습

- form 태그
  - 사용자 입력을 수집하는 데 사용
  - 수집한 정보를 서버로 전송
  - 로그인, 회원가입, 글 작성 등의 기능에서 서버로 전송할 때, 사용하는 정보를 입력하는 것

```
<form>

</form>
```

HTML 실습

form > input 요소

- input 요소는 가장 중요한 form 요소 중 하나
- type 속성에 따라 여러가지 방법으로 표시됨
- 텍스트 필드(text), 체크박스(checkbox), 전송 버튼(submit) 등과 같은 다양한 유형의 input 요소가 존재

```
<form>
        First name : <input type="text" id="fname" name="fname"><br>
        Last name : <input type="text" id="lname" name="lname"><br>
        Size : <input type="text" maxlength="10" id="max" name="max">
</form>

<!--
maxlength 속성 : 입력 값 길이 제한 가능

name 속성 : 중복된 이름으로 사용 가능하며, 데이터가 전송될 때 파라미터 이름으로 사용

id 속성 : 중복된 이름으로 사용 불가능하며, 주로 자바스크립트에서 다루기 위해 사용
-->
```

HTML 실습

- form > input 요소 > Submit Button
  - form 데이터를 form-handler에 제출하기 위한 버튼을 정의(type="submit")
  - form-handler는 입력 데이터를 처리하기 위한 서버의 페이지를 뜻함
  - form-handler는 form의 action 속성에 지정됨
  - method는 HTTP 요청 방식을 의미하고, GET과 POST로 나뉨

```
<form action="/login_page.php" method="POST">
        ID : <br>
        <input type="text" name="id" value="John"><br>
        Password : <br>
        <input type="password" name="passwd"><br><br>
        <input type="submit" value="Submit">
</form>

<!--
value 속성 : 입력 필드에 나타나는 초기 데이터를 의미
-->
```

HTML 실습

- form > input 요소 type 속성 hidden
  - 사용자에게 보이지 않는 숨겨진 입력 필드 생성
  - 사용자가 변경할 필요가 없는 정보 또는 변경하면 안되는 데이터를 함께 보낼 때 유용하게 사용됨
  - 중요한 정보가 기본 초기 값으로 입력되거나, 변경되면 안되는 데이터를 임의로 변경하여 전송한 경우 서버에서 처리된다면 취약!

```
<form action="/login.php" method="POST">
        아이디 : <input type="text" name="user_id"><br>
        비밀번호 : <input type="password" name="user_pw"><br>
        <input type="hidden" id="token" name="token" value="aa2!#64kl">
        <input type="submit">
</form>
```

HTML 실습

- form > input 요소 타입

```
<input type="button">
<input type="checkbox">
<input type="date">
<input type="email">
<input type="file">
<input type="hidden">
<input type="image">
<input type="month">
<input type="number">
<input type="password">
<input type="radio">
<input type="search">
<input type="submit">
<input type="tel">
<input type="text">
...
```

숙제

HTML

HTML 연습문제

1) h1 태그를 사용하여 "HTML Hello" 문자 출력 코드 작성
2) 네이버 홈페이지로 이동하는 하이퍼 링크 코드 작성
3) 구글 로고 이미지를 출력하는 코드 작성
4) "3)번 항목으로 만든 이미지"에 구글 홈페이지에 대한 링크가 적용되도록 코드 작성
5) 사용자가 id와 passwd를 입력하고, 입력한 값이 "/admin.php"로 전송되는 코드 작성(POST 메서드)

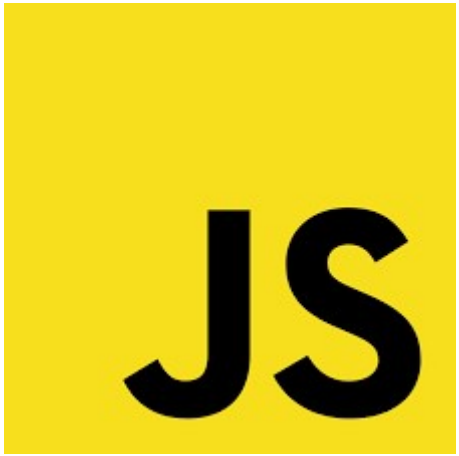
tcpschool : www.tcpschool.com/html/intro

생활코딩 : https://opentutorials.org/course/3084

JavaScript

JavaScript 개요

- **자바스크립트는 객체 기반의 스크립트 프로그래밍 언어**
  - 객체 기반 : 자바 언어와 다르게 클래스가 필요 없이 호출과 동시에 객체가 바로 생성됨
  - 스크립트 프로그래밍 언어 : 컴파일하지 않고도 실행할 수 있는 프로그래밍 언어
- **대부분 웹 페이지를 동적으로 다룰 때 사용**
  - HTML은 정적으로 무언가를 변경할 수 없음
- **참고로 "자바(Java)"와 "자바스크립트(JavaScript)"는 전혀 다른 언어**



JavaScript 적용법

내부 자바스크립트 코드

- <script> 태그를 사용하여 HTML 문서에 삽입
- HTML에서 <script> </script> 태그를 만들어 자바스크립트 코드를 삽입

외부 자바스크립트 코드

- 외부 파일로 자바스크립트 파일(.js)을 생성하여 HTML 문서에 삽입
- HTML 태그와 JS 코드를 분리할 수 있는 장점

```
<!DOCTYPE html>
<html>
<body>
        <script> alert('Hello World!');</script>
        <script src="test.js"></script>
</body>
</html>
```

```
// test.js
alert('Hello World!!!!!');
```

JavaScript 실습

기본 문법

- 변수 선언
- 값 할당
- 값 계산
- 주석

```
<script>
        let x, y;
        x = 2;
        y = 3;
        x = 3 + 4;
        y = x * 5;
        // y = x * y * 5 + 2;
        document.write(x);
        document.write(y);
</script>
```

JavaScript 실습

- 기본 타입
  - Boolean : true와 fasle 두 가지 값을 가짐
  - Null : null 한 가지 값을 가짐
  - Undefined : 값을 할당하지 않은 변수가 가지는 값
  - Number : 숫자를 표현할 때 사용
  - String : 문자(텍스트) 데이터를 표현할 때 사용

과제!

```
<script>
        let x = 10;
        let y = 15;
        const z = "boanproject";
        const p = true;
        document.write(typeof(x)); //number
        document.write(typeof(y)); //number
        document.write(typeof(z)); //string
        document.write(typeof(p)); //blooean
</script>
```

JavaScript 실습

문자열(String)

- String.charAt(index) 함수는 주어진 문자열에서 특정 index에 위치하는 문자를 반환
- String.fromCharCode(n1, n2, n3 …, nX) 함수는 유니코드 값을 문자로 변환

```
<script>
	const x = "boanproject";
	let y = x+" "+"sk";
	document.write(y);

	document.write("<br><br>");

	let z = y.charAt(5);
	document.write(z);

	document.write("<br><br>");

	document.write(String.fromCharCode(65,97));
</script>
```

JavaScript 실습

함수

- 특정 작업을 수행하도록 설계된 코드 블록
- 함수는 function 키워드, 이름 괄호 ( )로 정의
- 함수가 실행할 코드는 중괄호 { } 안에 정의

```
<script>

    function myFunction(a, b) {
        return a * b;
    }

    let x = myFunction(3, 5);
    document.write(x);

</script>
```

JavaScript 실습

함수

- 특정 작업을 수행하도록 설계된 코드 블록
- 함수는 function 키워드, 이름 괄호 ( )로 정의
- 함수가 실행할 코드는 중괄호 { } 안에 정의

```
function addNum(x, y) {
    return x + y;
}

document.write(addNum(3, 5));

alert(addNum(3, 5));
```

JavaScript 실습

- 함수를 사용하는 이유
  - 코드를 재사용 (코드를 한 번 정의하고 여러 번 사용 가능)
  - 다른 인자(아규먼트)로 동일한 코드를 여러 번 사용하여 다른 결과 생성 가능

```
<script>
        function toCelsius(f) {
          return (3/5) * (f-32);
        }
        document.write(toCelsius(77));
</script>
```

JavaScript 연습문제

1) name 이라는 변수를 선언하고 문자열 security를 지정

2) x 라는 변수를 선언하고 숫자 100을 지정

3) y 라는 변수를 선언하고 숫자 50을 지정

4) z 라는 변수를 선언하고 x와 y를 더한 결과를 지정하고, document.write를 사용하여 z 변수 값을 출력

JavaScript 실습

조건문(if)

- 주어진 조건에 만족하면 주어진 코드가 실행되고, 거짓이면 아무것도 실행되지 않음

```
<script>
	let  x = 10, y = 20;
	 if (x == y) {
		document.write("x == y");
	 }
	 if (x < y) {
	   document.write("x < y");
	 }
	 if (x > y)
		document.write("x > y");
</script>
```

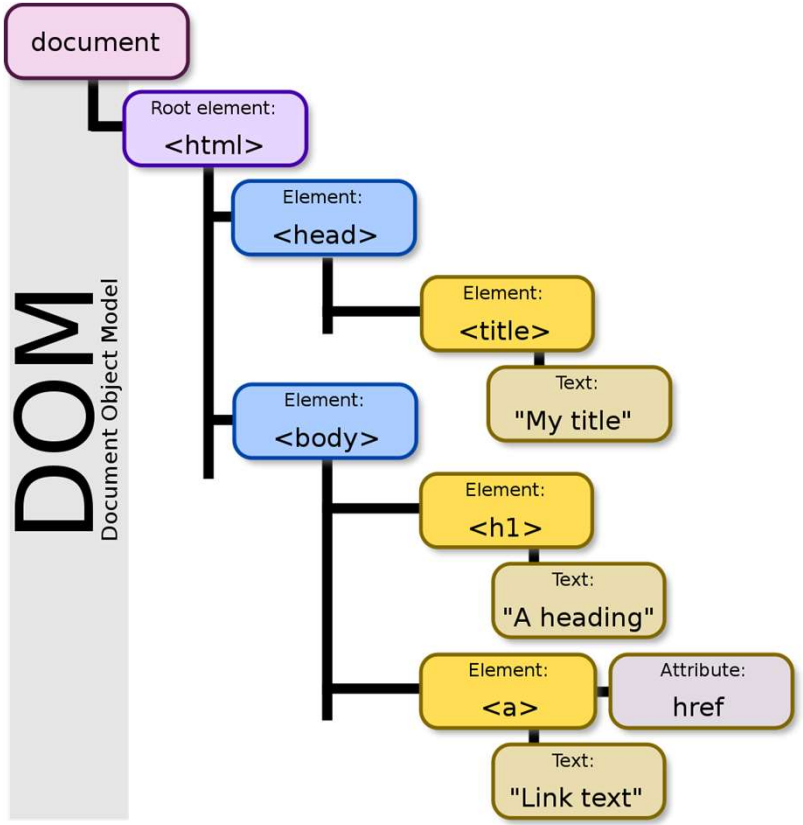
JavaScript 실습

- 조건문(else-if)
  - 하나의 조건 안에 추가적인 조건이 필요한 경우 사용됨

```
let  x = 10, y = 20;
if (x == y) {
    document.write("x == y");
} else if (x < y) {
    document.write("x < y");
} else {
    document.write("x > y");
}
```

JavaScript 실습

- DOM
  - 문서 객체 모델(Document Object Model)
  - XML, HTML 문서의 각 항목을 계층으로 표현하여 생성, 변형, 삭제할 수 있도록 돕는 인터페이스
  - Document 객체를 사용하여 HTML 요소에 접근



https://en.wikipedia.org/wiki/Document_Object_Model

JavaScript 실습

Document 객체

- 웹 페이지에 존재하는 HTML 요소에 접근하고자 할 때는 반드시 Document 객체부터 시작
- Document 객체는 HTML 요소와 관련된 작업을 도와주는 다양한 메소드를 제공

메서드	설명
document.getElementById(id)	요소(element) id로 요소(element) 찾기
element.innerHTML	요소(element)의 내부 HTML 변경
document.write("text")	주어진 텍스트를 HTML 페이지에 출력
document.cookie	문서의 쿠키(cookie)를 반환
document.domain	문서 서버의 도메인 이름을 반환
document.URL	문서의 전체 URL 주소를 반환

```
<p id="boanproject"></p>

<script>
    document.getElementById("boanproject").innerHTML = "boanproject getElement";
</script>
```

JavaScript 실습

이벤트 핸들러

- 웹 브라우저가 알려주는 HTML 요소에 대한 사건의 발생을 의미
- HTML 태그에 속성으로 이벤트 리스너를 등록

```
<p onclick="alert('String Click')">String Click!</p>

<h1 onclick="alert('Click')">Click on this text!</h1>

<button onclick=alert(document.cookie)>Button</button>
```

JavaScript 연습문제

1) x가 y보다 크면 "x > y"를 alert하고, 그렇지 않으면 "x < y"를 alert하도록 코드 작성

2) 버튼 요소를 클릭하면 문서의 쿠키(cookie)를 alert하는 코드 작성

3) attack_code 이름을 가진 함수를 생성하고, 함수 내용에는 alert을 사용하여 XSS 문자열을 출력하는 코드 작성

4) 위에서 작성한 attack_code 함수 호출

웹 기본 지식

https://portswigger.net/web-security/cross-site-scripting/cheat-sheet

웹 기본 지식

World Wide Web

- 1989년 3월 CERN(유럽 입자 물리학 연구소)의 팀 버너스 리 박스 등의 제안으로 시작되어 연구
- 인터넷에 연결된 컴퓨터를 통해 사람들이 정보를 공유할 수 있는 전 세계적인 정보 공간
- 간단히 Web, WWW, W3라고 부름

웹 기본 지식

초기의 웹

- 단순히 텍스트로 구성되고, 하이퍼텍스트를 이용해 다른 페이지로 이동
- 1990년 11월에 CERN에서 세계 최초의 웹 서버와 웹 브라우저 개발
- 대한민국 최초의 홈페이지는 1993년에 개발됨

웹 기본 지식

하이퍼텍스트 (Hypertext)

- 1965년 '테드 넬슨'이라는 컴퓨터 과학자가 처음으로 고안한 개념
- 한 문서에서 다른 문서로 즉시 접근할 수 있는 텍스트(링크)
- Hyper와 Text를 합성하여 만든 컴퓨터 및 인터넷 관련 용어
  - Hyper : 건너 편의, 초월, 과도한
  - Text : 문자

웹 기본 지식

웹 브라우저

- 웹 브라우저 또는 브라우저라고 부름
- 웹 서버와의 통신을 통해 요청과 응답을 주고 받음
- HTML, JAVASCRIPT, CSS 등의 클라이언트 언어를 해석하여 그래픽 사용자 인터페이스(GUI)를 제공
- 사용자가 입력한 URL을 이용해 서버에 자원을 요청하고, 서버로부터 응답을 받아 해석 후 사용자에게 출력

웹 기본 지식

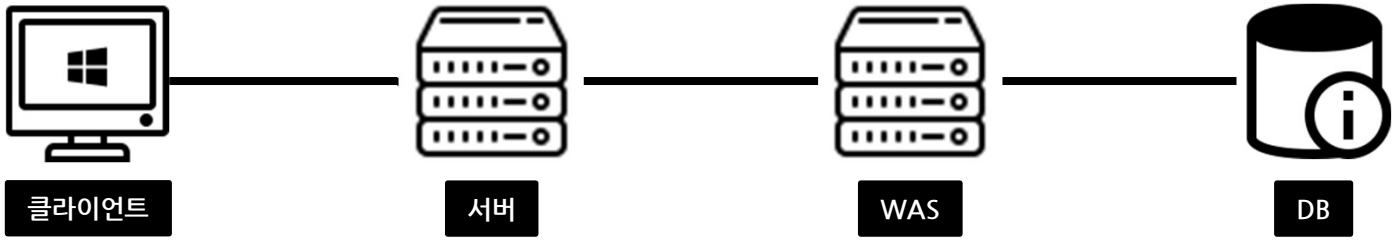
웹 브라우저

- 1990년 11월에 CERN에서 세계 최초의 웹 브라우저 개발 "WWW"
- 1994년 12월에 넷스케이프에서 "넷스케이프 내비게이터 1.0" 출시
- 1995년 마이크로소프트에서 "인터넷 익스플로러 1.0과 2.0" 출시
- 2003년 애플 "사파리" 출시
- 2004년 모질라 "파이어폭스" 출시
- 2008년 9월에 구글에서 "크롬" 출시

웹 기본 지식

웹 기본 구조

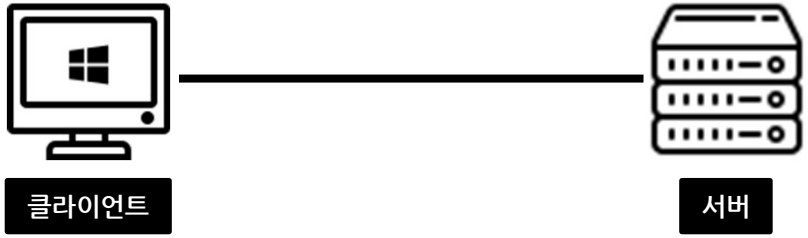
- 웹은 기본적으로 클라이언트(Client)와 서버(Server) 구조로 이루어짐
- 클라이언트는 서비스를 받는 부분으로 "웹 브라우저"
- 서버는 서비스를 제공하는 부분으로 "웹 서버"
  - 일반 서버는 **정적 컨텐츠**를 제공
  - WAS(Web Application Server)는 **동적 컨텐츠**를 제공



웹 통신

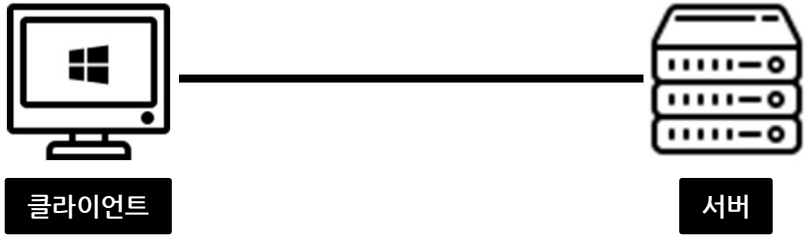
HTTP, URL, HTML

- HTTP : 통신을 지원해주는 역할
- URL : 클라이언트가 서버에 자원 요청
- HTML, CSS … : 클라이언트가 서버에 요청한 자원에 대한 결과



웹 통신

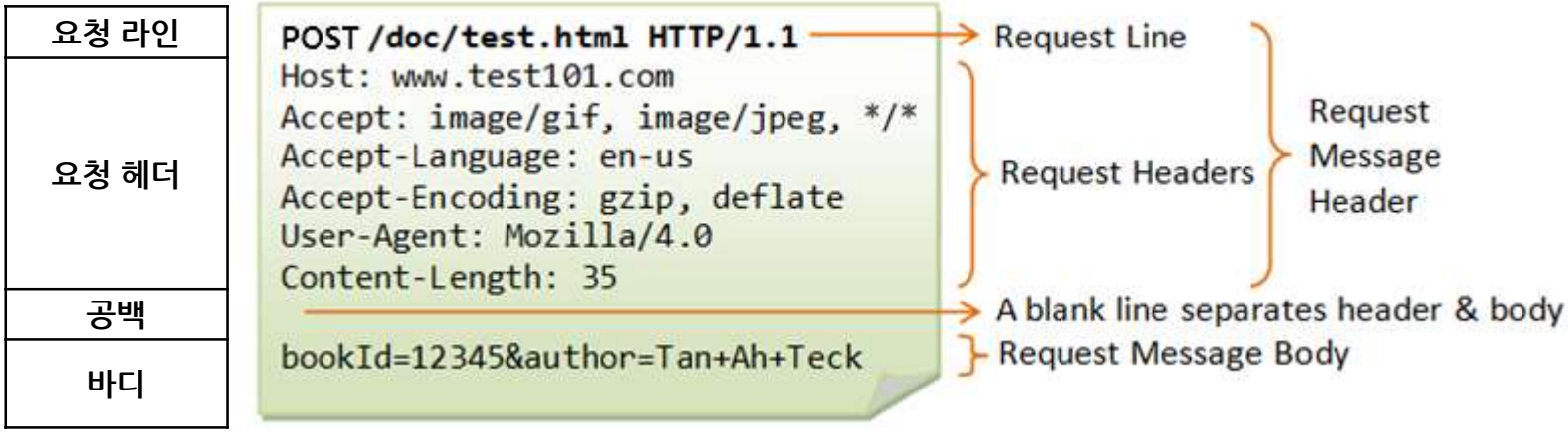
- 클라이언트가 서버에 요청하면 서버는 요청에 대한 응답을 보내줌
- HTTP Request Message
- HTTP Response Message

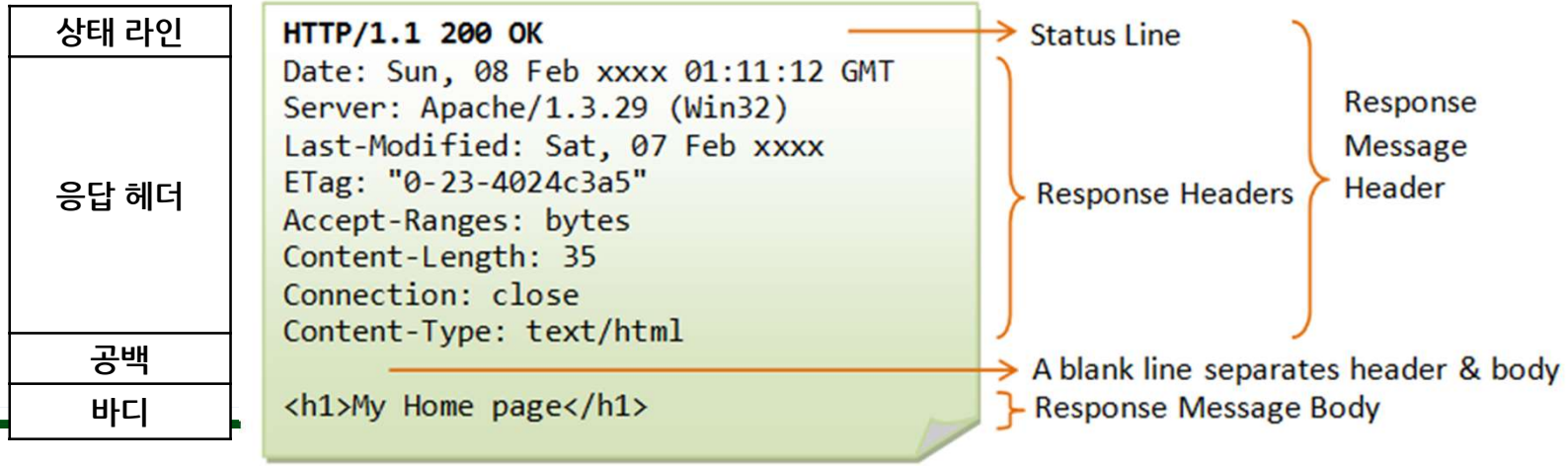


웹 기본 지식

웹 통신 & 메시지 구조

- "클라이언트"가 "서버"에 요청하면 "서버"는 요청에 대한 응답을 보내줌
  - HTTP Request Message
  - HTTP Response Message





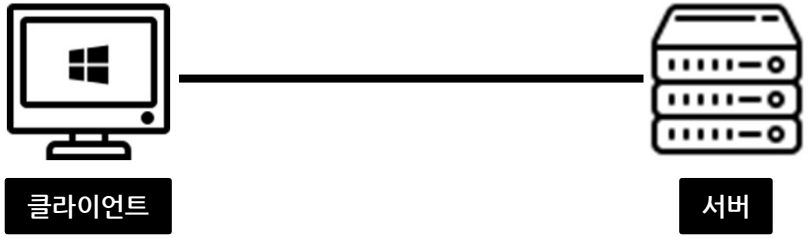
https://www3.ntu.edu.sg/home/ehchua/programming/webprogramming/http_basics.html

웹 기본 지식

URL (Uniform Resource Locator)

- 사전 : 인터넷에서, 어느 사이트에 접속하기 위해서 입력해야 하는, 주소를 포함한 일련의 문자.
- 네트워크 상에서 자원이 어디 있는지를 알려주기 위한 규약
- 서버에 있는 자원을 요청할 때 사용

http://www.boanproject.com/security/test.png

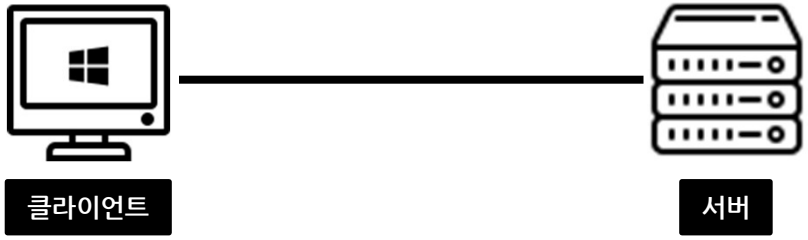


웹 기본 지식

URL 구조

- http
  - 스키마(scheme) 부분으로 사용할 프로토콜을 명시
- www.boanproject.com
  - 호스트(host) 부분으로 서버 주소(위치)를 명시하고 포트 사용 가능
- security
  - 접근하는 자원에 대한 서버 디렉터리
- test.png
  - 서버에 요청하는 자원 이름

http://www.boanproject.com/security/test.png



URL 예약 문자

- 예약 문자는 특별한 의미를 가짐

예약 문자	설명
/	URL 경로에서 경로 세그먼트를 구분
?	파라미터의 시작을 구분
=	파라미터 이름과 값을 구분
&	파라미터 구분
+	공백(Space)
:	호스트에서 스키마를 구분하고 포트에서 호스트를 구분
#	URL의 마지막 부분을 구분하여 서버로 전달
! $ * , ; @ [ ] ( )	…

URL 인코딩

- URL에 문자를 표현하는 방식
- 데이터 전송에 있어서 손실을 막기위해 사용함 (브라우저 특성에 따라 문자 처리방식이 다르기 때문)
- 웹 브라우저를 사용할 경우 브라우저에서 자동으로 URL 인코딩이 됨

문자	인코딩
/	%2F
?	%3F
=	%3D
&	%26
+	%2B
:	%3a
#	%23
! $ * , ; @ [ ] ( )	…

Burp Suite 소개 및 환경 구성

Burp Suite

Burp Suite

- Burp Suite는 웹 취약점 진단를 위한 보안 테스팅 도구
- 프록시 기능 및 취약점 스캔 기능을 제공
- 버프 스위트는 사용자 브라우저와 대상 애플리케이션 사이에서 프록시 서버 역할을 함

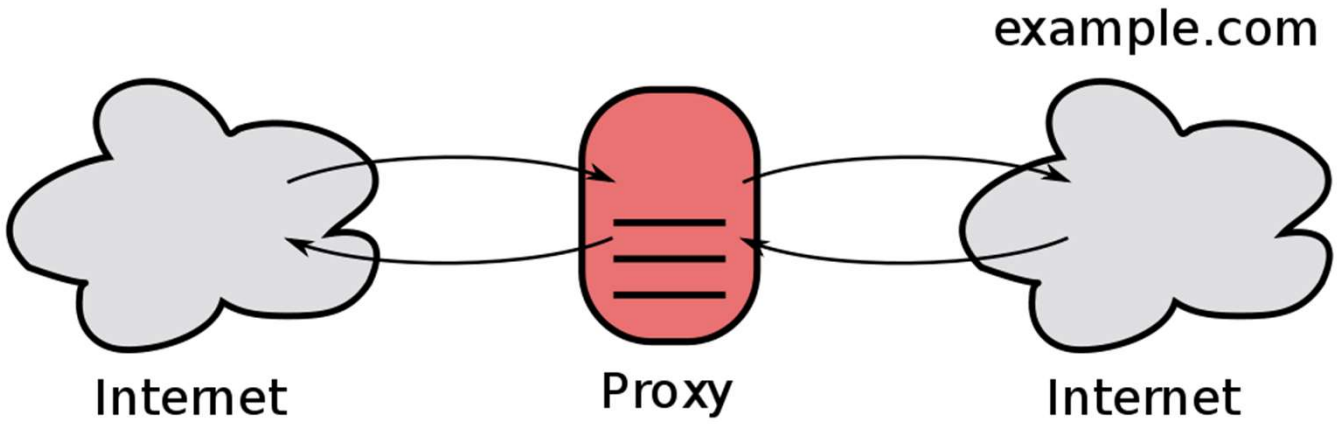
Burp Suite
Community Edition

Burp Suite
Professional

Dafydd Stuttard - PortSwigger founder and Chief Swig - ask me anything

프록시 서버

- 클라이언트가 자신을 통해 다른 네트워크 서비스에 간접적으로 접속할 수 있게 하는 것
- 서버와 클라이언트 사이의 중계기로, 대리로 통신을 수행하는 것을 "프록시"
- 프록시 서버 중 일부는 프록시 서버에 요청된 내용들을 캐시를 이용하여 저장
  - 원격 서버에 접속하여 데이터를 가져올 필요가 없어 전송 시간 절약 가능
  - 외부와 연결하지 않기 때문에 트래픽을 줄여 네트워크 병목 현상을 방지하는 효과
- 통신을 할 때 데이터 흐름이 프록시 서버를 거치므로 유해사이트 차단이나 외부침입 방지에 사용



https://en.wikipedia.org/wiki/Proxy_server

Burp Suite

소개

- Burp Suite는 웹 애플리케이션 보안 테스트를 위한 포괄적인 도구 모음
- 전세계 15,000개 이상의 조직에서 60,000명 이상의 보안 전문가가 선택한 도구
- Dafydd Stuttard(다피드 스터타드)가 2003년 6월에 실제 트림(burping) 소리와 함께 Burp의 첫 번째 버전을 작성하여 Proxy, Sock, Spider 및 Repeater를 포함한 Burp Suite v1.0 출시
- 2022년 7월 기준 최신 버전은 "2022.6.1"

Burp Suite
Community Edition

Burp Suite
Professional

Dafydd Stuttard - PortSwigger founder and Chief Swig - ask me anything

Burp Suite

제품 종류

- **Burp Suite Community Edition - 무료**
  - 무료 버전이므로 약간의 제한이 있지만, 학습하는 입장에서 충분히 테스트 가능
- **Burp Suite Professional - $399 (1년 구독)**
  - 자동화된 기능과 무료버전에서의 제한적인 기능 사용 가능
- **Burp Suite Enterprise Edition - $6,995 이상 (1년 구독)**
  - 개인 전문가보다는 큰 기업 단위에서 여러 자산을 동시에 스캔 또는 통계 기능을 사용할 때 추천

Burp Suite

제품 종류 - 대표적인 기능

- **Burp Suite Community Edition - 무료**
  - HTTP(s) / WebSockets 프록시 및 기록
  - 필수 도구 - Repeater, Decoder, Sequencer, and Comparer.
  - Burp Intruder (데모)
- **Burp Suite Professional - $399 (1년 구독)**
  - Community Edition의 모든 기능
  - 프로젝트 파일 (작업 저장)
  - Burp Intruder (전체 버전)
  - 웹 취약점 스캐너
  - Pro 전용 BApp 확장
  - 검색 기능
  - 자동 및 수동 OAST 테스트 (Burp Collaborator)
- **Burp Suite Enterprise Edition - $6,995 이상 (1년 구독)**
  - Burp Suite Professional의 모든 기능
  - 많은 수의 웹 애플리케이션에서 동시 스캔 실행 가능
  - CI/CD 플랫폼과 통합
  - 조직 전체 또는 일부의 보안 상태를 확인하기 위한 대시보드
  - 역할 기반 액세스 제어 및 SSO

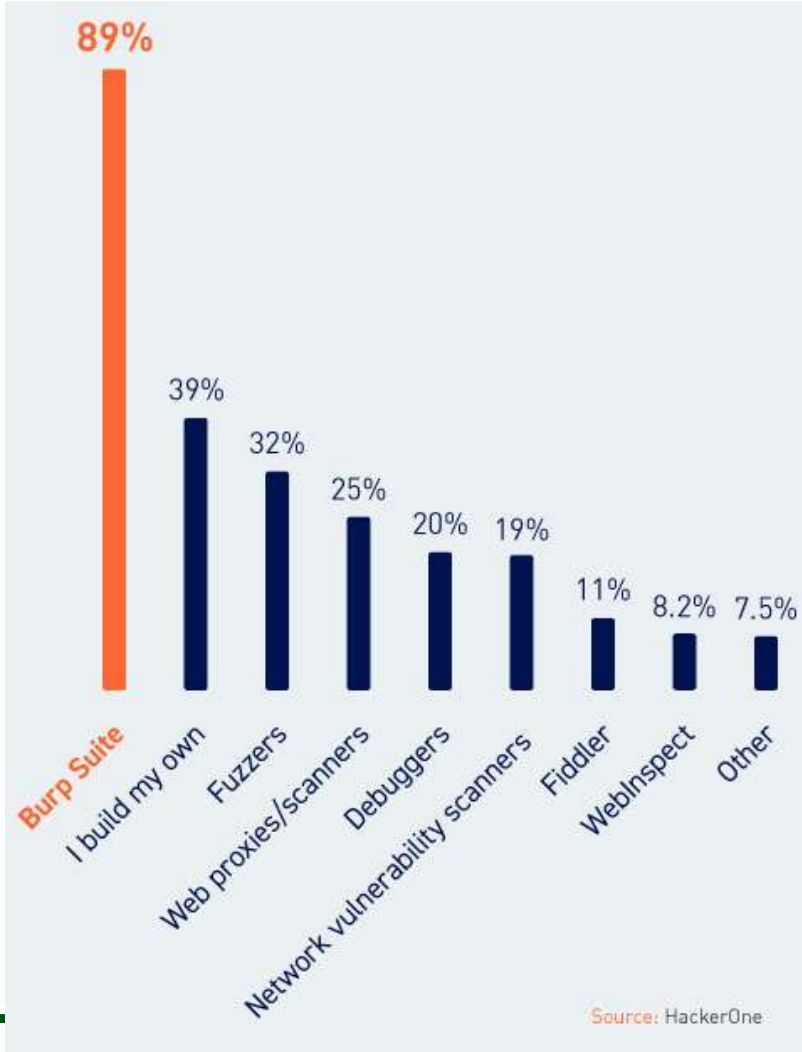
Burp Suite

통계 - HackerOne

- **해킹할 때 가장 도움이 되는 소프트웨어, 하드웨어 또는 도구**
  - Burp Suite 인기는 해커의 88% 이상이 사용
  - 자체 도구를 제작하는 비율도 38% 이상 큰 비중을 차지

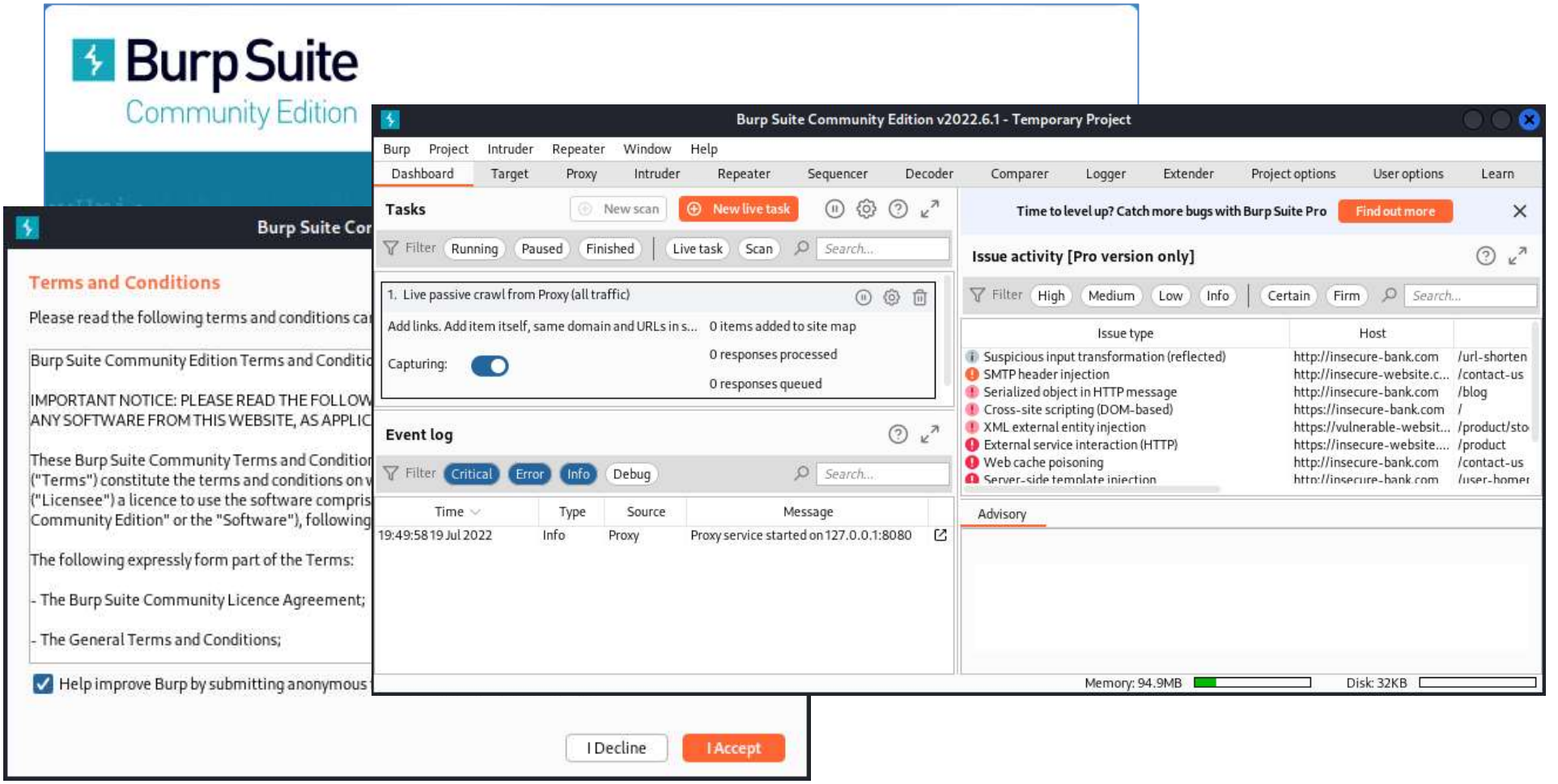
PortSwigger

hackerone



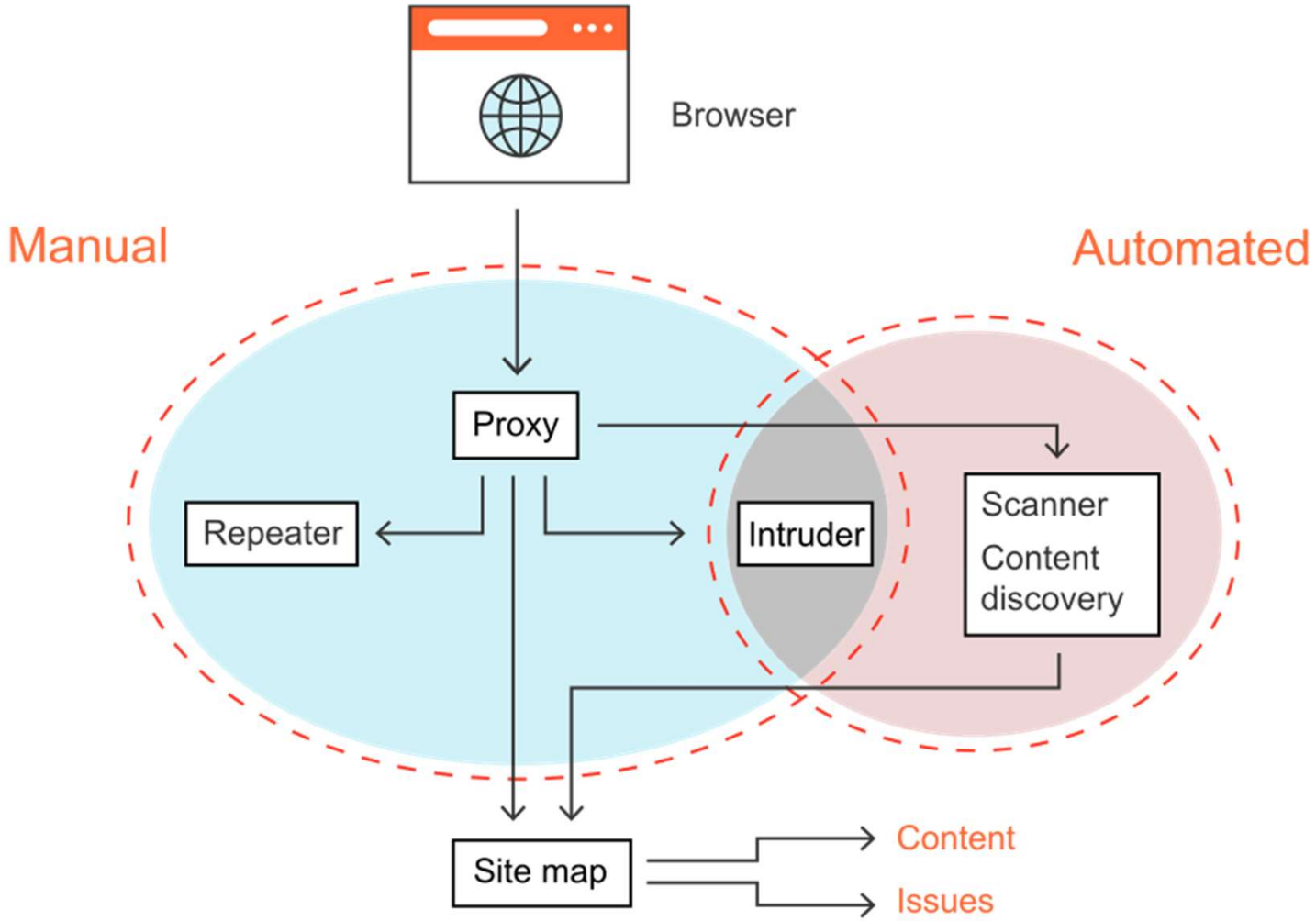
Burp Suite

실행



Burp Suite

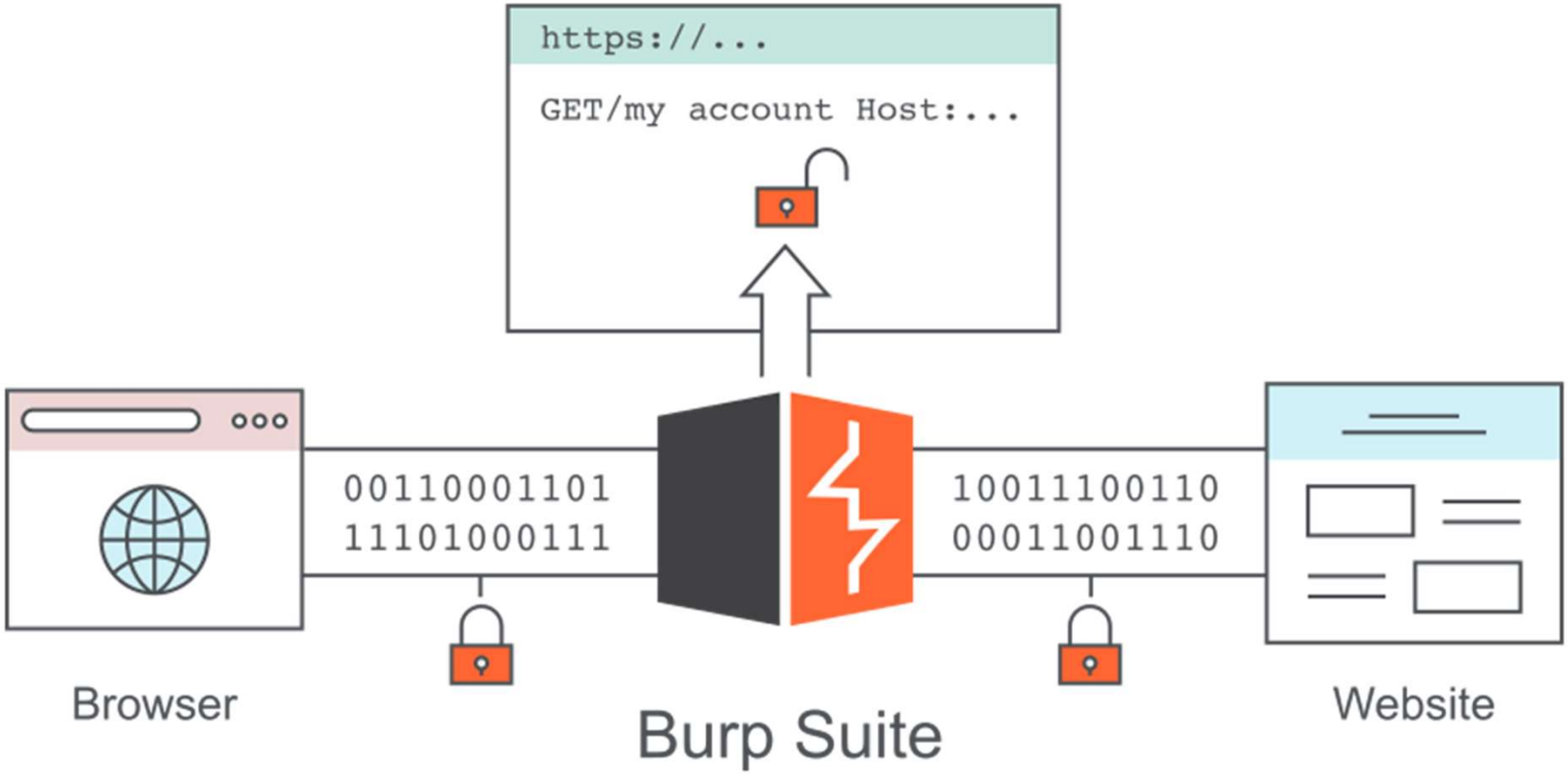
구성 요소



Burp Suite

MitM HTTP 프록시

- MitM은 man-in-the-middle로 중간자를 의미함



Burp Suite

Proxy - 대표적인 기능

- (실습) bee-box 대상으로 통신 확인
- Intercept
  - 브라우저와 대상 웹 서버 간에 전달되는 모든 요청 및 응답을 Intercept(가로채기), 검사, 수정 가능
- HTTP history
  - 프록시를 통과한 모든 메시지의 전체 기록
- Options
  - 프록시 관련 옵션 설정