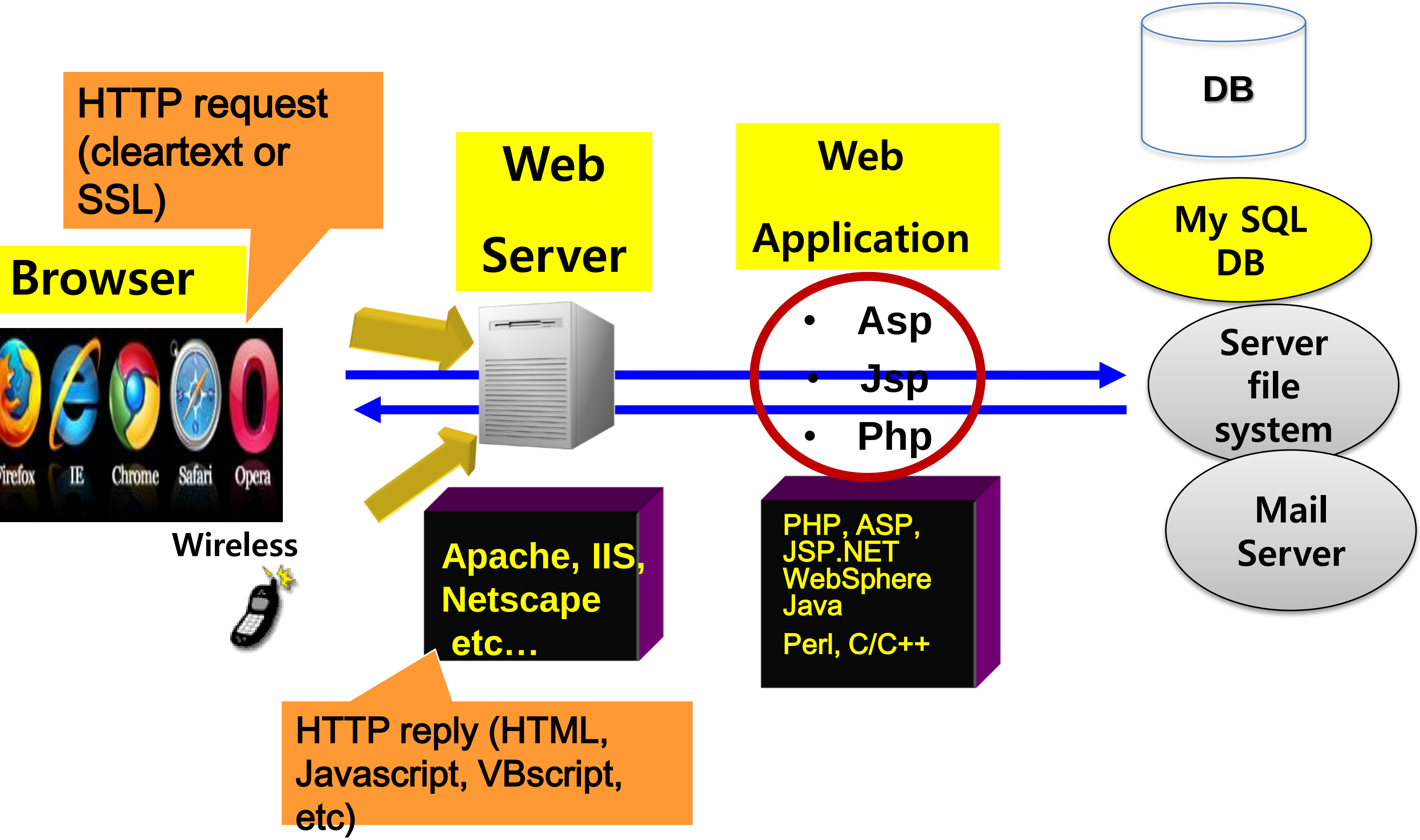
Web security measures overall

Web architectures



HTTP request (cleartext or SSL)

Browser

Wireless

Web Server

Apache, IIS, Netscape etc…

Web Application

- Asp
- Jsp
- Php

PHP, ASP, JSP.NET
WebSphere
Java
Perl, C/C++

DB

My SQL DB

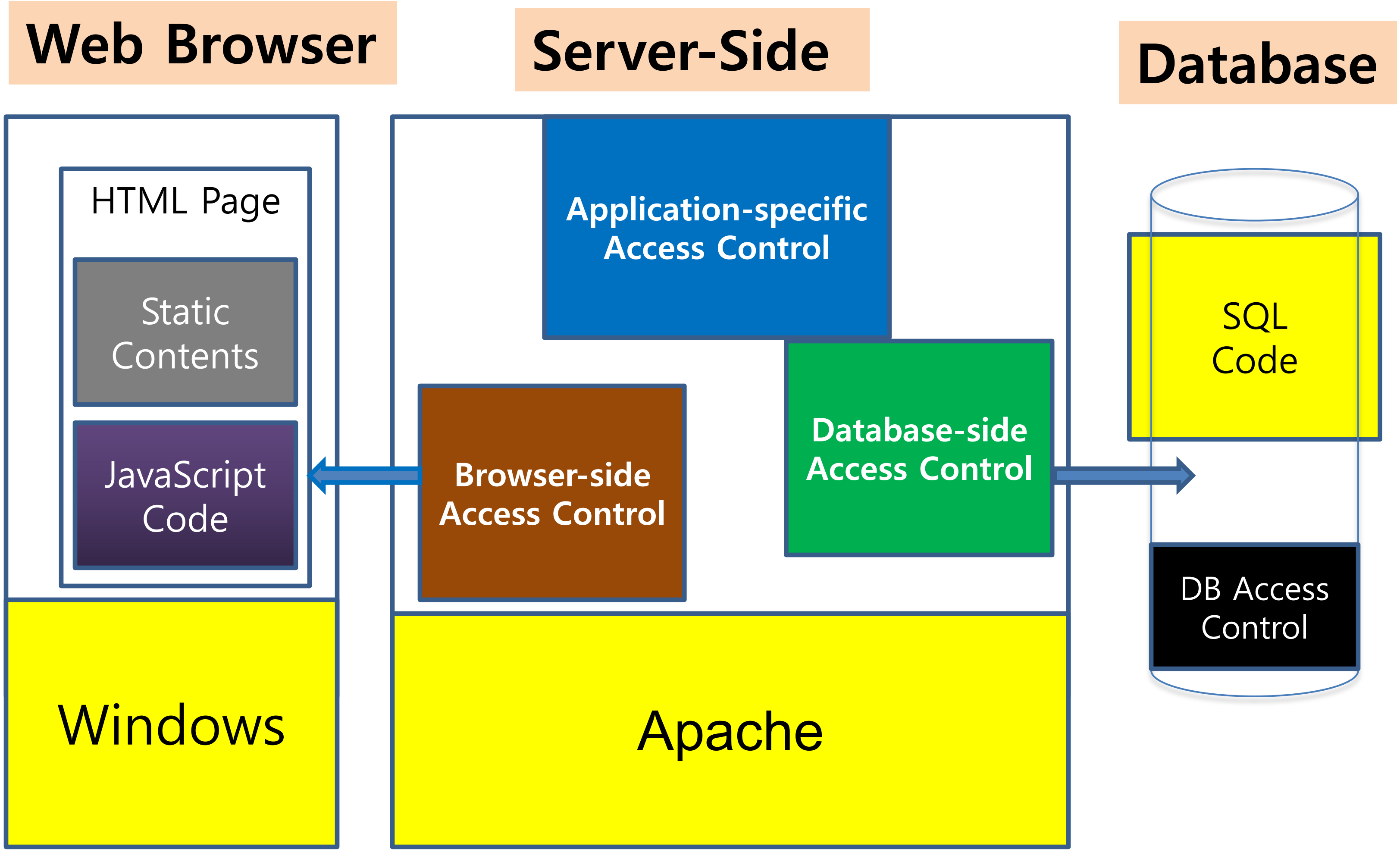
Server file system

Mail Server

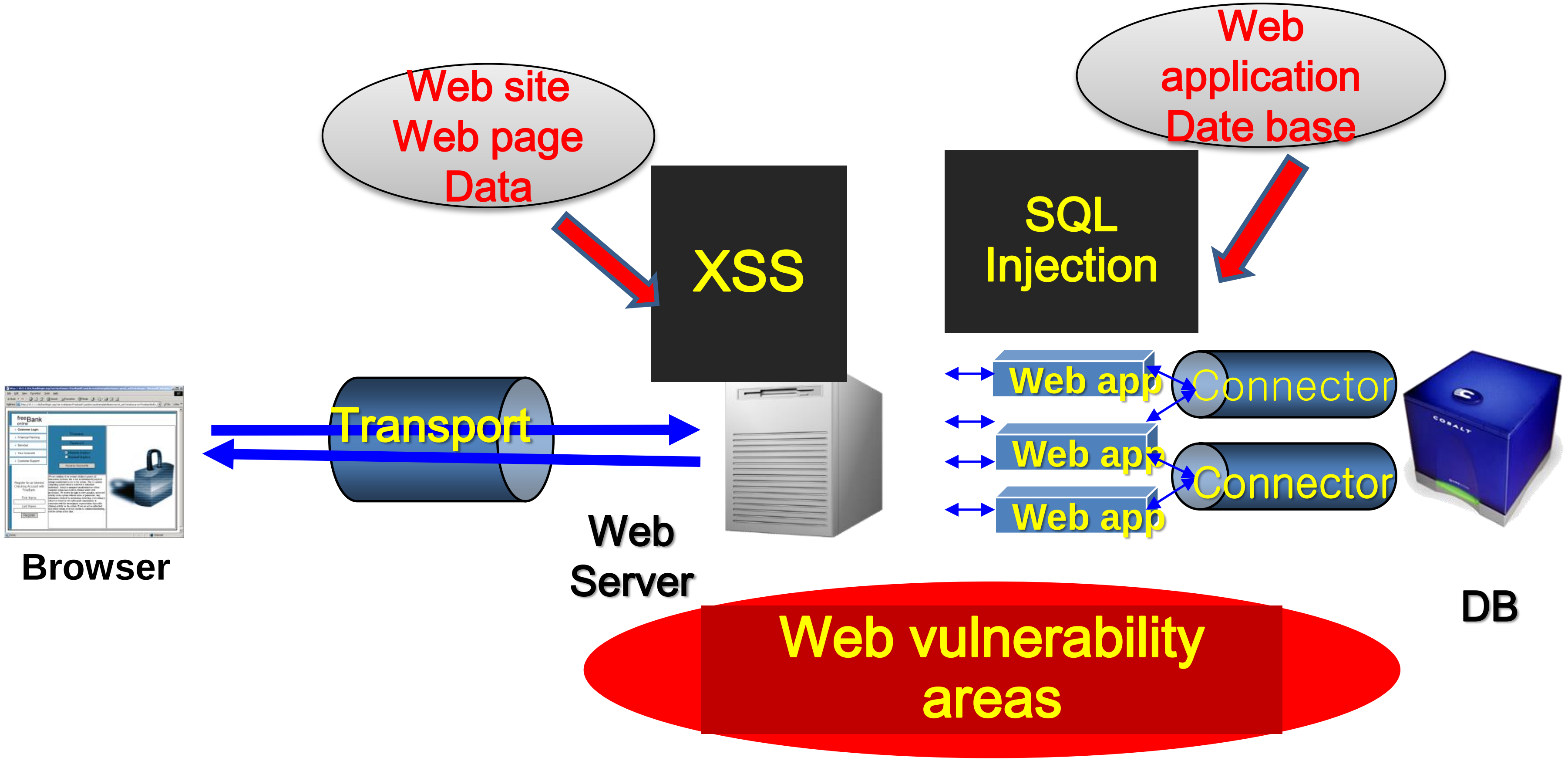
HTTP reply (HTML, Javascript, VBscript, etc)

complex architectures, multiple platforms, multiple protocols

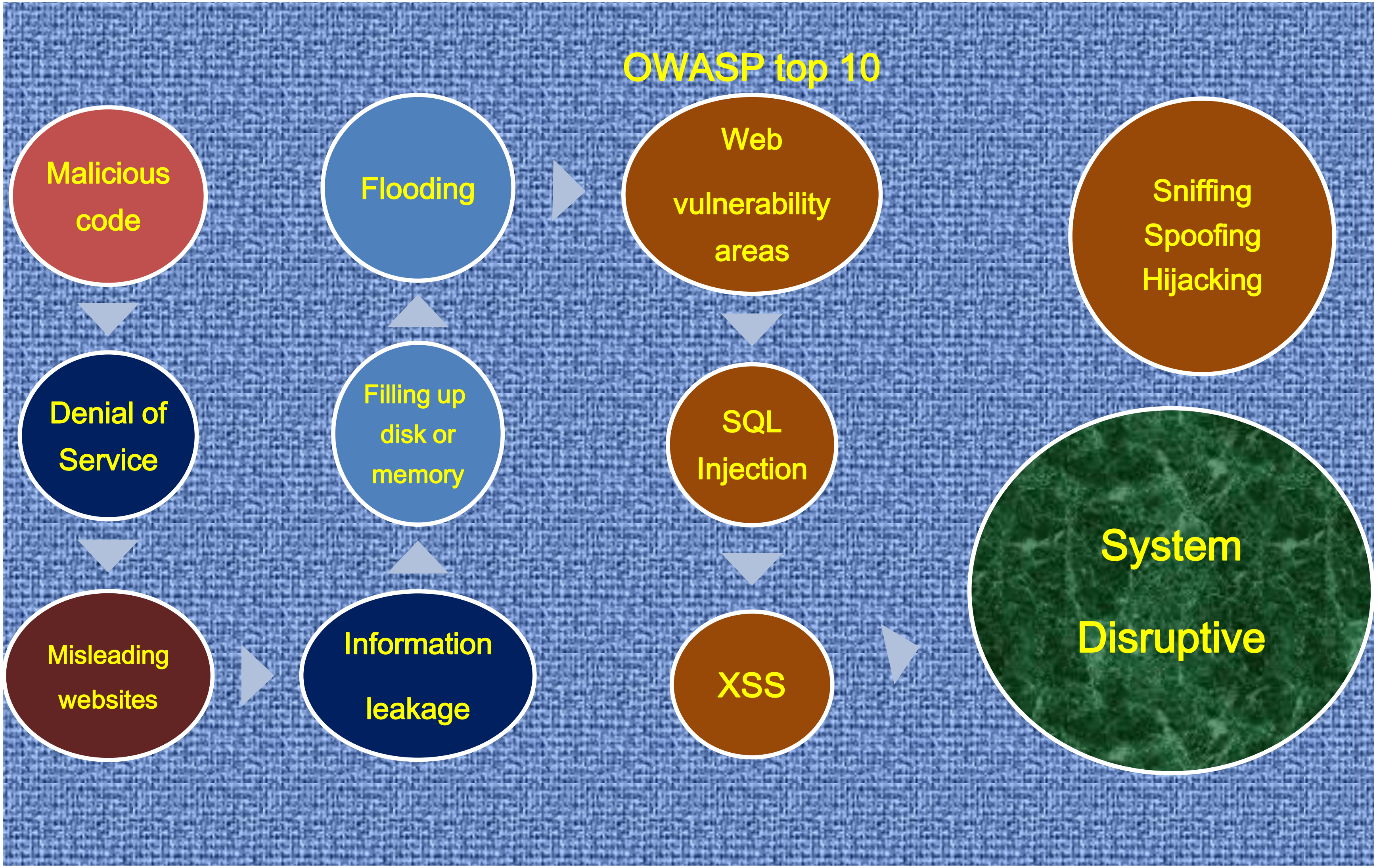
Application-Specific Logic



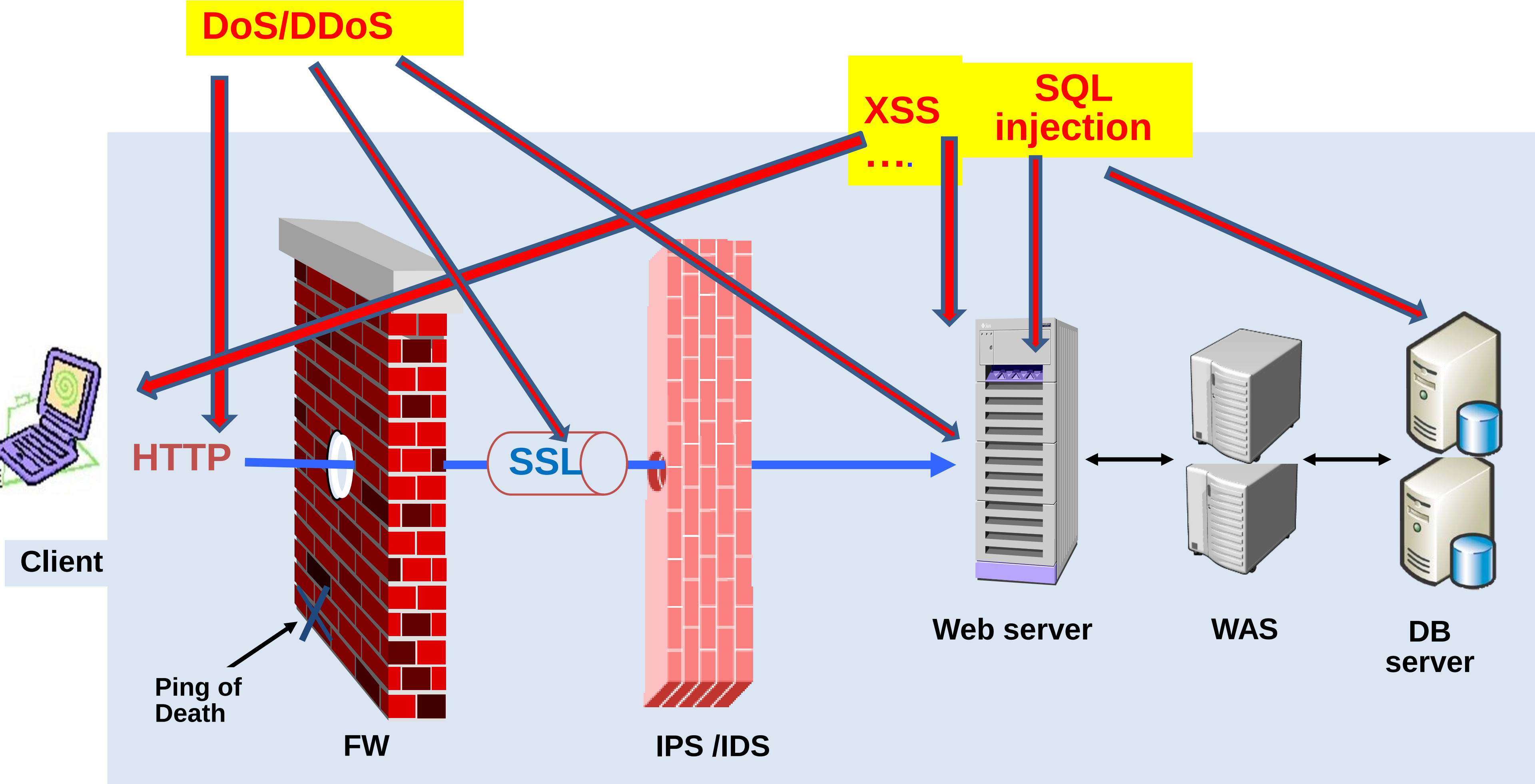
Common threats to web



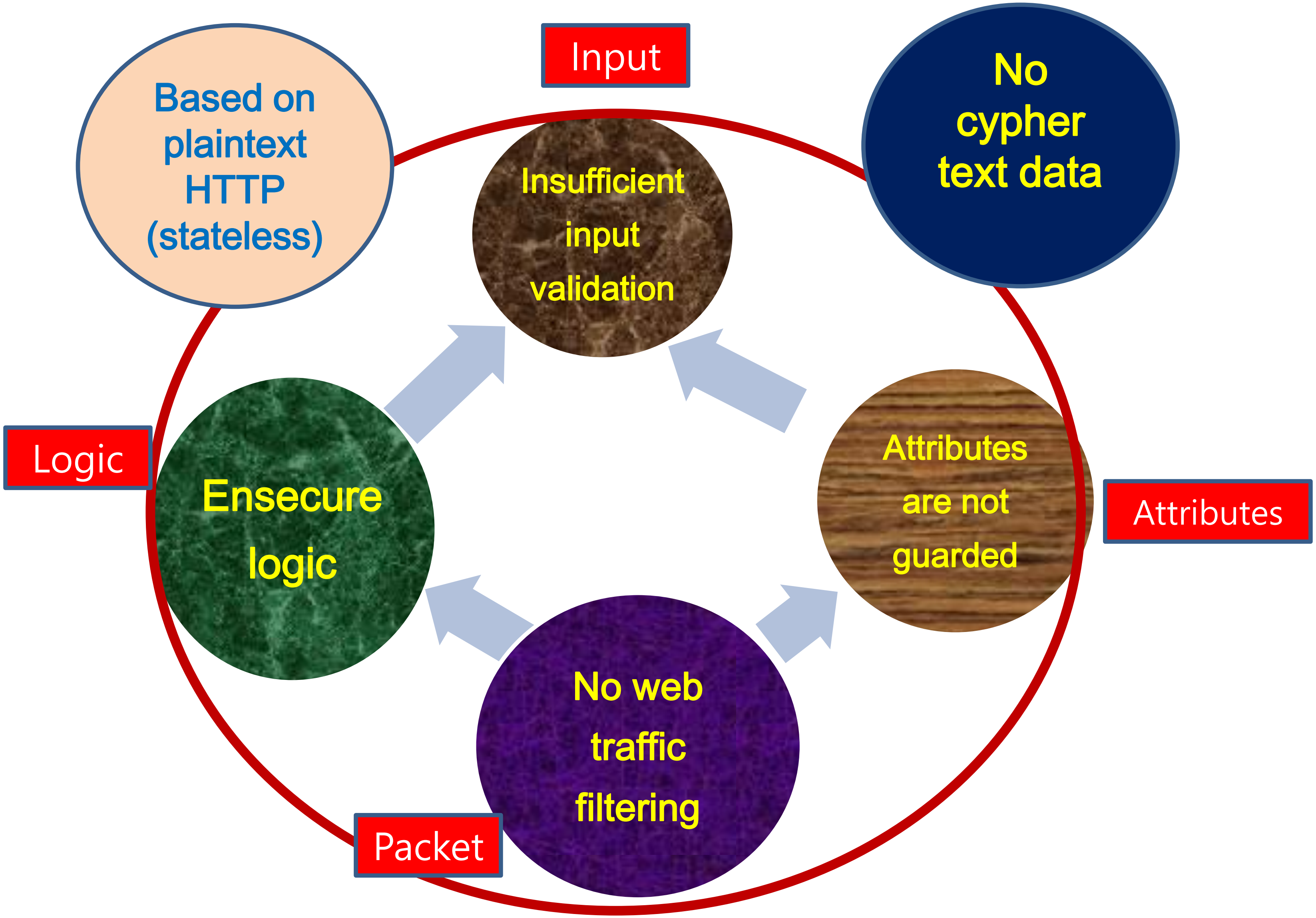
Threats to web include



Respong against Web hacking



Web sites have at least one serious vulnerability



Based on plaintext HTTP (stateless)

Input

Insufficient input validation

No cypher text data

Logic

Ensecure logic

Attributes are not guarded

Attributes

No web traffic filtering

Packet

Reasons of web vulnerabilities

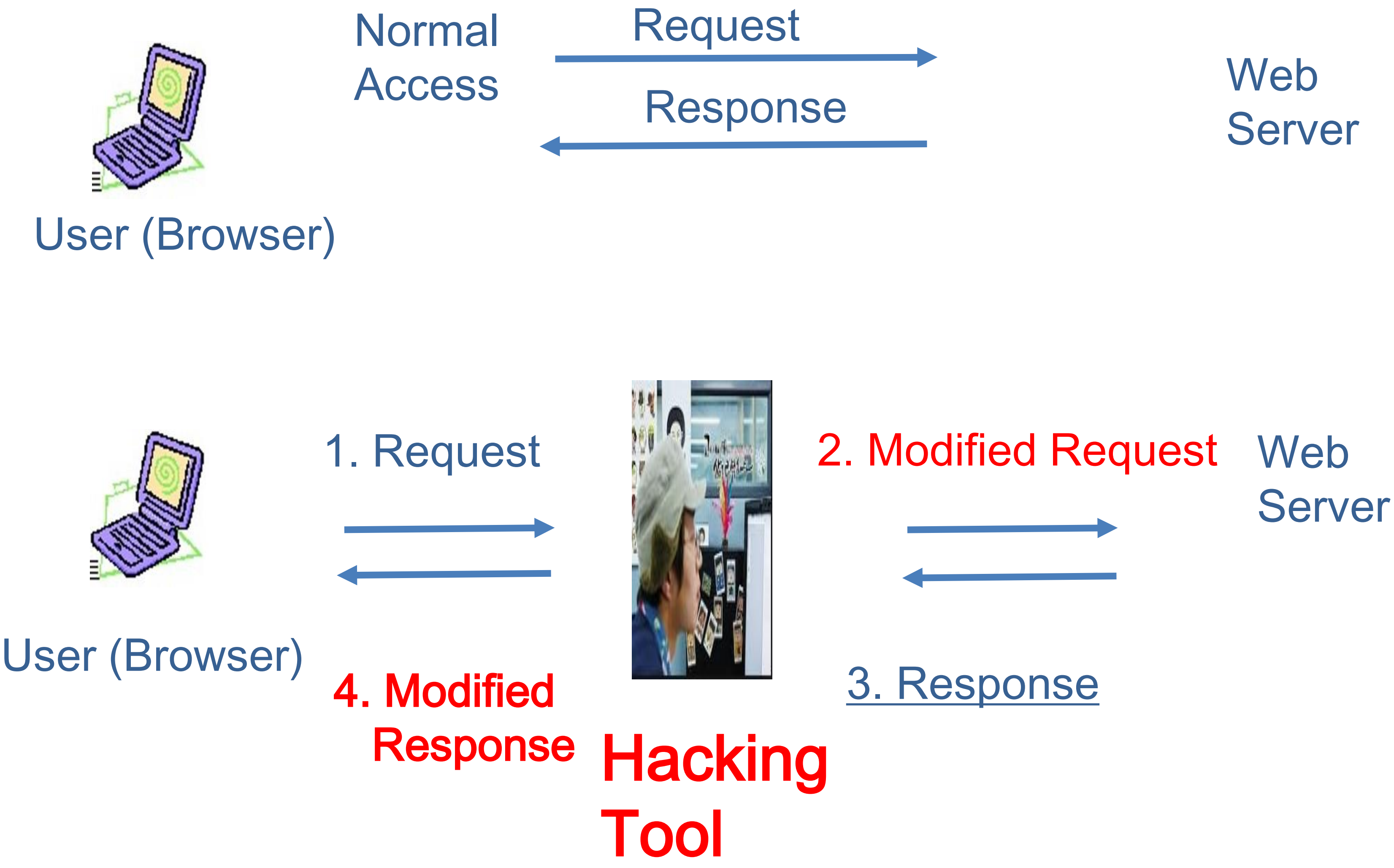
- Data => Insufficient input validation
- => Plaintext data input from client
- Logic => Poor web application logic
- => No vulnerability check
- Attributes => Web application attributes, Query String are insufficient input validation.

- Traffic => Transmission data no encryption

-> LAN section (from PC- to EXT.router)
-> Wireless section (PC- to access point)
-> Internetwork section (from EXT.router – to EXT.router)

- HTTP => No cypher text from client to web server
- Server => No authenticaon from client to web server

HTTP Intercept



Normal Access

Request

Response

Web Server

User (Browser)

1. Request

2. Modified Request

Web Server

User (Browser)

4. Modified Response

3. Response

Hacking Tool

Web Security Threats
include

- Information leakage
- Misleading websites
- Malicious code
- Denial of Service
- Killing of user threads
- Flooding machine with bogus requests
- Filling up disk or memory
- Isolating machine by DNS attacks
- Disruptive
- OWASP top 10 vulnerability areas

Web Threats
Acquisition Root Right, Data, etc.

User Authenticate → Cookie & Session Data

File Upload → Server Side Script file

File Download → System Backup file

SQL Injection → Private Member Data

Cross-Site Scripting (XSS) → Bulletin Board

Injection

Who is OWASP® Foundation?

- **The Open Web Application Security Project® (OWASP) is a nonprofit foundation that works to improve the security of software.**
- **Through community-led open-source software projects, hundreds of local chapters worldwide, tens of thousands of members, and leading educational and training conferences, the OWASP Foundation is the source for developers and technologists to secure the web.**
- **Tools and Resources**
- **Community and Networking, Education & Training**

https://owasp.org/

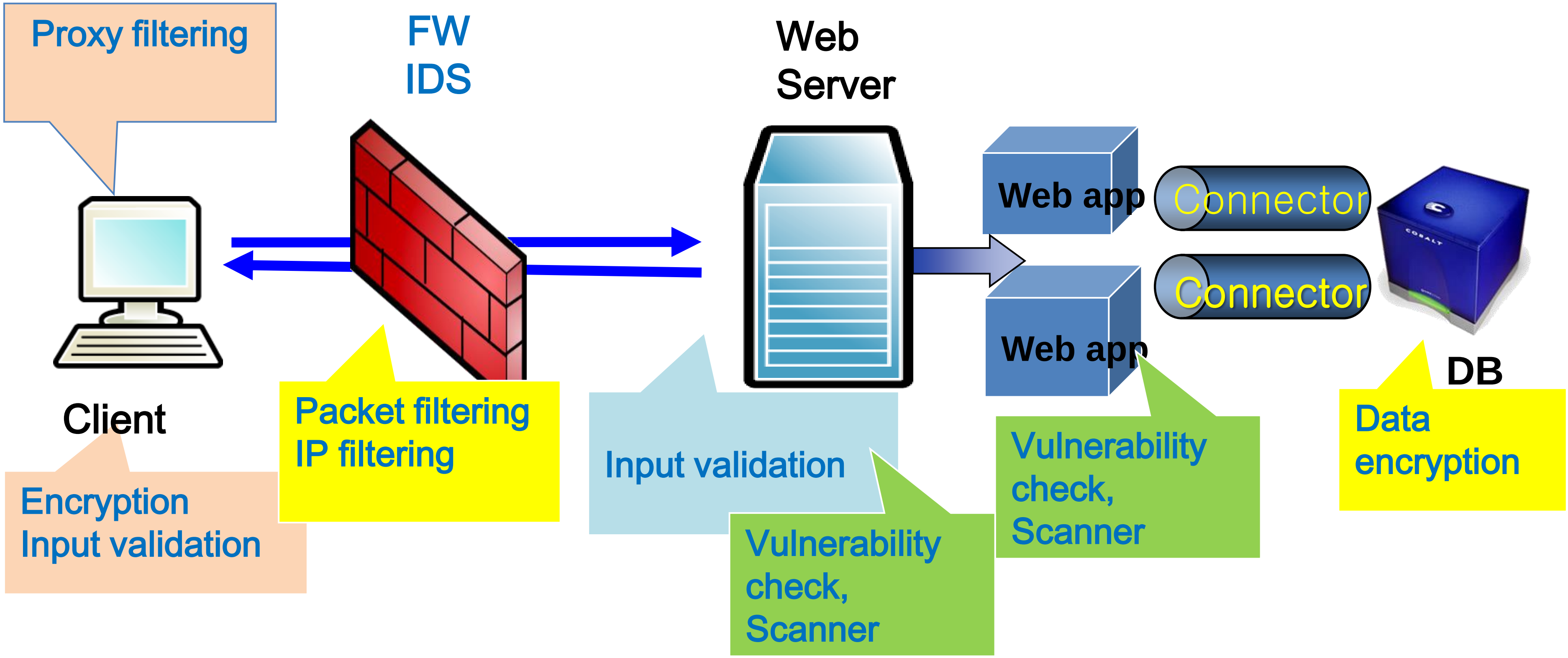
OWASP top 10 Web vulnerabilities

#1 SQL Injection
#2 Broken Authentication and Session Managemen
#3 XSS: Cross-Site Scripting
#4 Insecure direct object reference
#5 Security misconfiguration
#6 Sensitive data exposure
#7 Missing function level access control
#8 Cross-site request forgery
#9 Using components with known vulnerabilities
#10 Invalidated redirects and forwards

Web security measures

Summingup web security architectures

Client Web Server DB



Web security measures

HTTP => HTTP cypher => HTTPS

Server authentication from client => SSL

1. Data => Input data validation/check
2. Program => Secure coding
3. Traffic => Web traffic filtering /encryption
4. Attributes => Web application attributes checking => Query String
5. Weakpoint => Vulnerability check

Summing up of web security methods

Securing goals	Solution	Where	Method
Input validation	Application	Router, client , servers each	BLT file Checking app
Mid gate filtering	Tool Proxy filtering	-Between Client and servers -On the Client	Burp Suite Paros WebScarab Fiddler
Encryption	Encryption	Moving data from client- to servers	HTTPS SSL
Vulnerability check	Tool	Application Program	Nikto Nessus Acunetix
		Web system Client and servers Each	

Vulnerability check with scanner

Input validation

Start URL inspection

Form field consistency

Buffer overflow detection

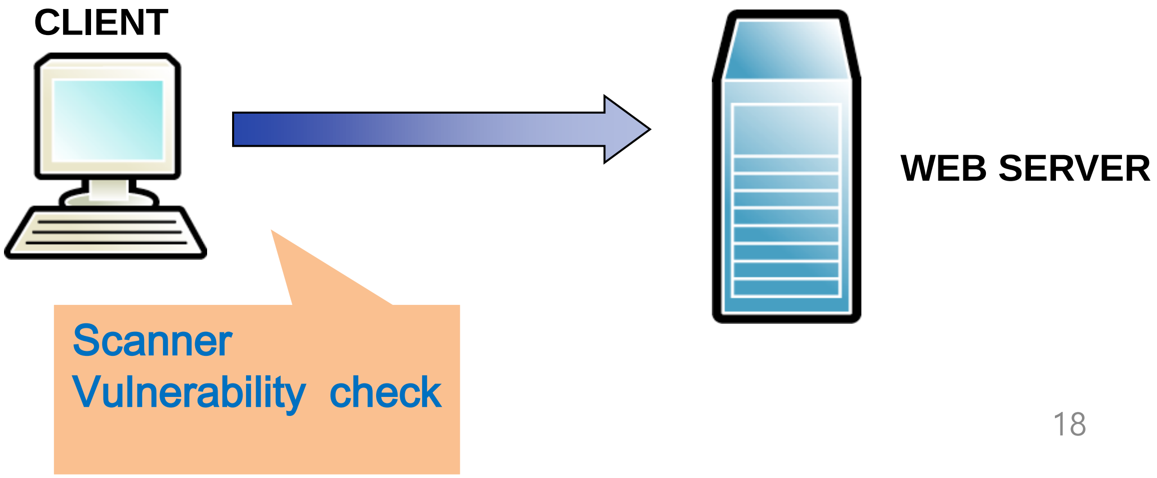
SQL inspection

Cookie consistency

Shell code detection

Forceful Browsing detection

Detect vulnerabilities periodically on web server through scanning



Scanner
Vulnerability check

Server security

- DB operation with least privileged user
- Remove unused stored procedures and built-
  n functions or control permissions
- Modify query authority according to purpose
- Access Control of Common System Objects
- Allow access only to trusted networks, server
  Error message exposure block