Web servers, applications, and vulnerabilities

ISSES

Co-funded by the
Erasmus+ Programme
of the European Union

Web application

- Runs on a remote server
- Accessed through a client
- Popular because they provide flexibility and power
- Apps can offer their unique services to a specific platform, or they can be platform independent
- Support for mobile computing
- How to attack and compromise them

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Who interacts with a web server

- Server Administrators
- Network Administrators
- End Users
- Application Administrator
- Application Developer

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Web Servers

- Software package that is designed to deliver files and content over HTTP
- Response to requests that come from clients in software form
- Differentiated by
  - Operating system support, server-side technologies, security models, client support, development tools, and many more factors
- Examples:
  - Internet information server (IIS)
  - Apache

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Apache Web Server

- The most popular technology of its type in the world
- Estimated 60 percent of web servers on the internet running the software
- Supports a large number of features (open source)
- Modules include the following:
  - Authentication, SSL support, TLS support, proxy support, URL rewriter, HTTP request filtering, python and perl support, PHP, compression support, intrusion detection, enhanced logging

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Internet Information Server (IIS)

- Microsoft's web server
- Modules that can be added to IIS include these:
  - Process management, Server-side language, Support for legacy technologies (mainly for IIS 6.0 users), Protocol listeners, Security support, Certificate support, Authentication support, Database support
- Protocol listeners
- HTTP listener - part of the HTTP.sys module

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Web Applications

- Software that is installed on top of a web server
  - Designed to respond to requests, process information, store information, and scale in response to demand
- Three variations:
  - Browser based
  - Client based
  - Mobile apps

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Client - Server

- Server application
  - Hosted on a web server
  - Accessed remotely via
    - A web browser or
    - A web-enabled application
- Client application
  - Performs minimal processing of information
  - Optimized to present the information to the user
- Information
  - stored on the server,
  - some small portions residing on the client
    - such as metadata

ISSES

Co-funded by the
Erasmus+ Programme
of the European Union

Platforms and technologies

- Some web applications are locked to a specific platform
  - Mobile OS dependent
  - Different versions of the client for different platforms
- Server-side technologies
  - Active server pages (ASP), ASP.NET, PHP, etc.
- Client-side technologies
  - Dynamic HTML (DHTML), HTML 5, javascript

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Web application details

- Consists of layers:
  - Presentation Layer
  - Logic Layer
  - Data Layer
- HTTPS
  - HTTP employing encryption mechanisms
- Use of an underlying web server technology
  - IIS, apache, etc
- Cookie
  - A file stored on a client system that is used as a token by applications

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Cookies

- Necessary for the functioning
  - Applications that rely on the ability to maintain state information across stateless protocols such as HTTP
- A potential liability
  - A commonly exercised attack method for malicious users who employ them
- Primary purpose - maintaining state information
  - Example online store

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Web Servers and Applications Vulnerabilities

- Many of the vulnerabilities similar to other environments
- Some vulnerabilities unique to this environment
- Some of the vulnerabilities:
  - Flawed web design
  - Buffer overflow
  - Error messages
  - …

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Flawed Web Design

- <form method="post" action="../../cgi-bin/formMail.pl“>
  <!--Regular FormMail options-->
  <input type=hidden name="recipient" value=someone@some.com>
  …
  <input type=hidden name="servername" value=https://payments.some.com>
  …
  <input type=hidden name="orderconfirmation" value="orders@some.com">
- <FORM ACTION =http://1.1.1.1/cgi-bin/order.pl" method="post“>
  <input type=hidden name="price" value=“500.00“>
  <input type=hidden name="prod_id" value=“12345“>
  QUANTITY: <input type=text name="quant" size=2 maxlength=2 value=1>

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Buffer Overflow

- Occurs when an application, process, or program attempts to put more data in a buffer than it was designed to hold
- A programmer creates a buffer in code but does not put restrictions on it
- Data overflows into the buffers it was not intended for
- The result can be corrupted or overwritten data

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Error Messages

- can reveal a lot of information about a server and a web application
- can be configured or suppressed as necessary

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Common Attack Methods

- Some of the common attack methods:
  - Insecure Logon Systems
  - Scripting Errors
  - Session Management Issues
  - Encryption Weaknesses
  - Directory Traversal Attacks
  - …

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Insecure Logon Systems

- Web applications require some sort of authentication or login process
- It is essential that it be handled safely and securely
- The incorrect or improper entry of information does not reveal data

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Scripting Errors

- Methods to exploit scripting languages:
  - Upload bombing
  - Default scripts
  - Poorly written scripts
  - …

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Session Management Issues

- The connection that a client has with the server application
- Some vulnerabilities of this type include the following:
  - Long-lived sessions
  - Logout features
  - Insecure or weak session identifiers
  - Poor or no password change controls
  - Inclusion of unprotected information in cookies

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Protecting Cookies

- List of the attributes that can be set on a per-cookie basis, which makes them safer to use:
  - Secure
  - Httponly
  - Expires
  - …

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Encryption Weaknesses

- When considering encryption and its impact on the application, focus on these areas of concern:
  - Weak Ciphers
  - Vulnerable Software

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Directory Traversal Attacks

- Example of a GET HTTP request URL:
  - `http://somesite.com/show.asp?view=history.html`
- Attacker can craft a custom URL:

`http://somesite.com/show.asp?view=../../../../../Windows/system.ini`

- Directory traversal attack countermeasures:
  - Running modern web server software
  - Enabling filtering of user input to the web server

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Testing Web Applications

- Burp Suite
  - Proxy
  - Spider
  - Scanner
  - Intruder
  - Repeater
  - Sequencer

ISSES
Co-funded by the Erasmus+ Programme of the European Union

Software development caused vulnerabilities

- How to include security in software development lifecycle
- How to detect vulnerabilities in production environment software products
- Some significant attempts in software vulnerabilities classification made by:
  - *CWE (Common Weakness Enumeration)* and
  - *OWASP* (*Open Web Application Security Project*)

ISSES
Co-funded by the Erasmus+ Programme of the European Union

OWASP Top 10

2017.

- A1 – Injection
- A2 – Broken Authentication
- A3 – Sensitive Data Exposure
- A4 – XML External Entities
- A5 – Broken Access Control
- A6 – Security Misconfiguration
- A7 – Cross-Site Scripting (XSS)
- A8 – Insecure Deserialization
- A9 – Using Known Vulnerable Components
- A10 – Insufficient Logging & Monitoring

2021.

- A1 – Broken Access Control
- A2 – Cryptographic Failures (new name)
- A3 – Injection (includes XSS)
- A4 – Insecure Design (new category)
- A5 – Security Misconfiguration (includes XXE)
- A6 – Vulnerable and Outdated Components (new name)
- A7 – Identification and Authentication Failures (new name)
- A8 – Software and Data Integrity Failures (includes Insecure Deserialization)
- A9 – Security Logging & Monitoring Failures (new name)
- A10 – Server-Side Request Forgery (new category)

ISSES
Co-funded by the Erasmus+ Programme of the European Union

OWASP 2021. top 10

- Broken Access Control
  - `http://example.com/app/getappInfo`
    `http://example.com/app/admin_getappInfo`
- Cryptographic Failures
  - An application encrypts credit card numbers in a database using automatic database encryption. However, this data is automatically decrypted when retrieved, allowing a SQL injection flaw to retrieve credit card numbers in clear text.
- Injection
  - `String query = "SELECT * FROM accounts WHERE custID='" + request.getParameter("id") + "'";`
  - `http://example.com/app/accountView?id=' or '1'='1`

ISSES
Co-funded by the Erasmus+ Programme of the European Union

OWASP 2021. top 10

- Insecure Design
- Security Misconfiguration
  - The application server comes with sample applications not removed from the production server. These sample applications have known security flaws attackers use to compromise the server. Suppose one of these applications is the admin console, and default accounts weren't changed. In that case, the attacker logs in with default passwords and takes over.
- Vulnerable and Outdated Components
  - There are automated tools to help attackers find unpatched or misconfigured systems.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

OWASP 2021. top 10

- Identification and Authentication Failures
  - Application session timeouts aren't set correctly. A user uses a public computer to access an application. Instead of selecting "logout," the user simply closes the browser tab and walks away. An attacker uses the same browser an hour later, and the user is still authenticated.
- Software and Data Integrity Failures
  - Many home routers, set-top boxes, device firmware, and others do not verify updates via signed firmware. Unsigned firmware is a growing target for attackers and is expected to only get worse. This is a major concern as many times there is no mechanism to remediate other than to fix in a future version and wait for previous versions to age out.

ISSES
Co-funded by the Erasmus+ Programme of the European Union

OWASP 2021. top 10

- Security Logging and Monitoring Failures
  - A major European airline suffered a GDPR reportable breach. The breach was reportedly caused by payment application security vulnerabilities exploited by attackers, who harvested more than 400,000 customer payment records. The airline was fined 20 million pounds as a result by the privacy regulator.
- Server-Side Request Forgery (SSRF)
  - Attackers can use SSRF to attack systems protected behind web application firewalls, firewalls, or network ACLs

ISSES
Co-funded by the Erasmus+ Programme of the European Union

NOTICE FOR STUDENTS

- Topics of the Advanced Network and System Security course involve the study of various mechanisms that violate information security and make intrusions into computer systems and networks.
- The application of these mechanisms when executed towards the systems of individuals and legal entities, which are not familiar with them and are not consentient with the activities on checking vulnerability and testing intrusions into their systems, is punishable under the Criminal Law of the Republic of Serbia (Articles 298 to 304a).
- Students enrolled at the Advanced Network and System Security course may use these methods for study purposes only within the closed laboratory environment provided for teaching the Advanced Network and System Security course.
- Students may not imply that they are in any way encouraged by the teachers or that they are recommended to use these methods toward other systems of the School of Electrical Engineering or the systems of any third party entity or individual.
- Any eventual activity that a student would undertake using these methods and mechanisms according to systems that are not within the laboratory on the course is the sole responsibility of the student.

ISSES
Co-funded by the Erasmus+ Programme of the European Union