



सुशासन

School of Good Governance And Policy Analysis

A Workshop on Website Quality, Accessibility and Security

April 2, 2009

Security Guidelines for websites & Web- enabled Applications

Chandra Prakash Gupta,
Scientist – “B”,
NIC



Topics to cover

- What is Information Security
- Types of attacks
- OWASP (Open Web Application Security Project) : Top 10 vulnerabilities.
- Category of Websites & possible vulnerabilities



What is Information Security

The concepts, techniques, technical measures, and administrative measures used to protect information assets from deliberate or inadvertent unauthorized acquisition, damage, disclosure, manipulation, modification, loss, or use



Some types of attacks

- Denial of Service (DoS) Attacks
- Website Defacement
- Viruses and Worms
- Unauthorized Access
- Malicious Code and Trojans



OWASP : Top 10 vulnerabilities

- Input validation Vulnerabilities
- Cross-Site Scripting (XSS) Flaws
- Broken Access Control
- Broken Authentication and Session Management
- Buffer Overflow
- Injection Flaws
- Error Handling Problems
- Insecure Storage
- Denial of Service
- Insecure Configuration Management



Category of Websites

- Websites can be broadly categorized into:
 - 1) Static Web sites
 - 2) Dynamic Web sites



Category of Websites

1. Static Web sites can be further categorized into:
 - A. Pure static web site.
 - B. Static web site where the contents are rendered.
 - C. Static web site having a feedback form



Category of Websites

2. Dynamic Web sites can be further categorized into:
 - A. Primarily Static web sites with one feedback application
 - B. Primarily Static web site with one application for a query module
 - C. Sites having number of applications and open to all
 - D. Sites having applications for a closed user group
 - i. Only authenticated users can access the functional modules
 - ii. Only authenticated users having the privilege can access the modules for which they are authorized

Websites & Possible vulnerabilities

1.A Pure Static Web Sites :


- Hosted simple Read only permission.

1.B Static Web sites with static content generating script Files :

- Hosted with Script Only or Read and Execute permission.

1.C Static Web sites with feedback form:


- Apply client-side validation on form elements.
- Use matching pattern field in feedback form page.
- Hosted with Read only permissions.
- The form handling program/application on the foreign URL remains to be audited for security vulnerabilities.



Websites & Possible vulnerabilities

1.C Static Web sites with feedback form:

- Input Validations vulnerabilities.
- Cross site scripting vulnerabilities.
- The form handling program/application on the foreign URL remains to be audited for security vulnerabilities.



Websites & Possible vulnerabilities

2.A Primarily Static site with one Feedback application:

- Input Validations vulnerabilities
- Cross site scripting vulnerabilities
- Buffer overflow
- Error Handling vulnerabilities
- Denial of Service vulnerabilities




Websites & Possible vulnerabilities

2.B Primarily Static site with one query module:

2.C Sites having number of applications and open to all:

- Input Validations vulnerabilities
- Cross site scripting vulnerabilities
- SQL injection vulnerabilities
- Buffer overflow
- Error Handling vulnerabilities
- Denial of Service vulnerabilities



Websites & Possible vulnerabilities

2.D Sites having applications for a closed user group:

- Input Validations vulnerabilities
- Cross site scripting vulnerabilities
- SQL injection vulnerabilities
- Buffer overflow
- Error Handling vulnerabilities
- Denial of Service vulnerabilities
- Broken access control
- Broken account and session management
- Insecure storage



Vulnerabilities & Solutions

- **Input validation Vulnerabilities**

Information from web requests is not validated before being used by a web application.

Attackers can use these flaws to attack backside components through a web application.

- **Solution:**

Perform Input Validations.



Vulnerabilities & Solutions

- **Cross-Site Scripting (XSS) Flaws**

The web application can be used as a mechanism to transport an attack to an end user's browser.

A successful attack can disclose the end user's session token, attack the local machine or spoof content to fool the user

- **Solution:**

Apply strong input validation to be applied on the variables .The application must provide strong Input/Output validation checks across all the fields in the feedback form page.



Vulnerabilities & Solutions

■ **Broken Access Control**

Web sites allow certain users to access or perform certain functions while disallowing others access to these functions. This is termed as Access Control or Authorization.

- **Forced Browsing past access control checks**
- **Insecure Ids**

■ **Solution:**

Apply strong authentication and session management logic in addition to access control on each of these pages.



Vulnerabilities & Solutions

- **Broken Authentication and Session Management**

The applications suffer from certain issues such as the session life cycle being not managed or being faulty. Also the credentials used for the purposes of authentication are not properly managed with the result that the credentials can be exploited by a malicious user to gain privileged access

- **Solution:**

Generate a unique session id for each login.

Invalidate the session on the server application when Sign Out option is clicked by the user.

The session may also be invalidated after a certain time of inactivity.



Vulnerabilities & Solutions

- **Buffer Overflow**

In these kind of attacks, the attacker sends an arbitrarily large input to the application. By sending carefully crafted input in the web application, the attacker can execute arbitrary code and corrupt the execution stack of the application and causing the application to execute the code that was inserted in the input—effectively taking over the machine.

- **Solution:**

Size checking on all such inputs is required.



Vulnerabilities & Solutions

- **Injection Flaws**

In these kind of attacks, the attacker sends an arbitrarily large input to the application. By sending carefully crafted input in the web application, the attacker can execute arbitrary code and corrupt the execution stack of the application and causing the application to execute the code that was inserted in the input—effectively taking over the machine.

- **Solution:**

Perform input validations for user input and perform roles-user authorisation checks to allow access to the application.



Vulnerabilities & Solutions

- **Error Handling Problems**

Error conditions that occur during normal operation are not handled properly. If an attacker can cause errors to occur that the web application does not handle, they can gain detailed system information, deny service, and cause security mechanisms to fail, or crash the server.

- **Solution:**

Apply error handling into the application.



Vulnerabilities & Solutions

- **Insecure Storage**

This vulnerability deals with improper protection of sensitive information like password, or any other sensitive information.

- **Solution:**

The solution to the above is to implement the salted MD5 technique.



Vulnerabilities & Solutions

- **Denial of Service**

Web applications are susceptible to denial of service attacks. Once an attacker can consume all or some of the required resources, they can prevent legitimate users from accessing the system. Attackers can cause the user accounts to be locked or even the entire application to fail.

- **Solution:**

Use pattern matching input.

Vulnerabilities & Solutions

- **Insecure Configuration Management**
 - **Un-patched flaws in the server s/w**
 - **Server s/w flaws or misconfiguration that allow directory listing or directory traversal**
 - **Improper file and directory permission**
 - **Unnecessary services enabled including remote administration**
 - **Default a/cs with default passwords**



■ Thank You

National Informatics Centre (NIC),
Madhya Pradesh State Centre, Bhopal.