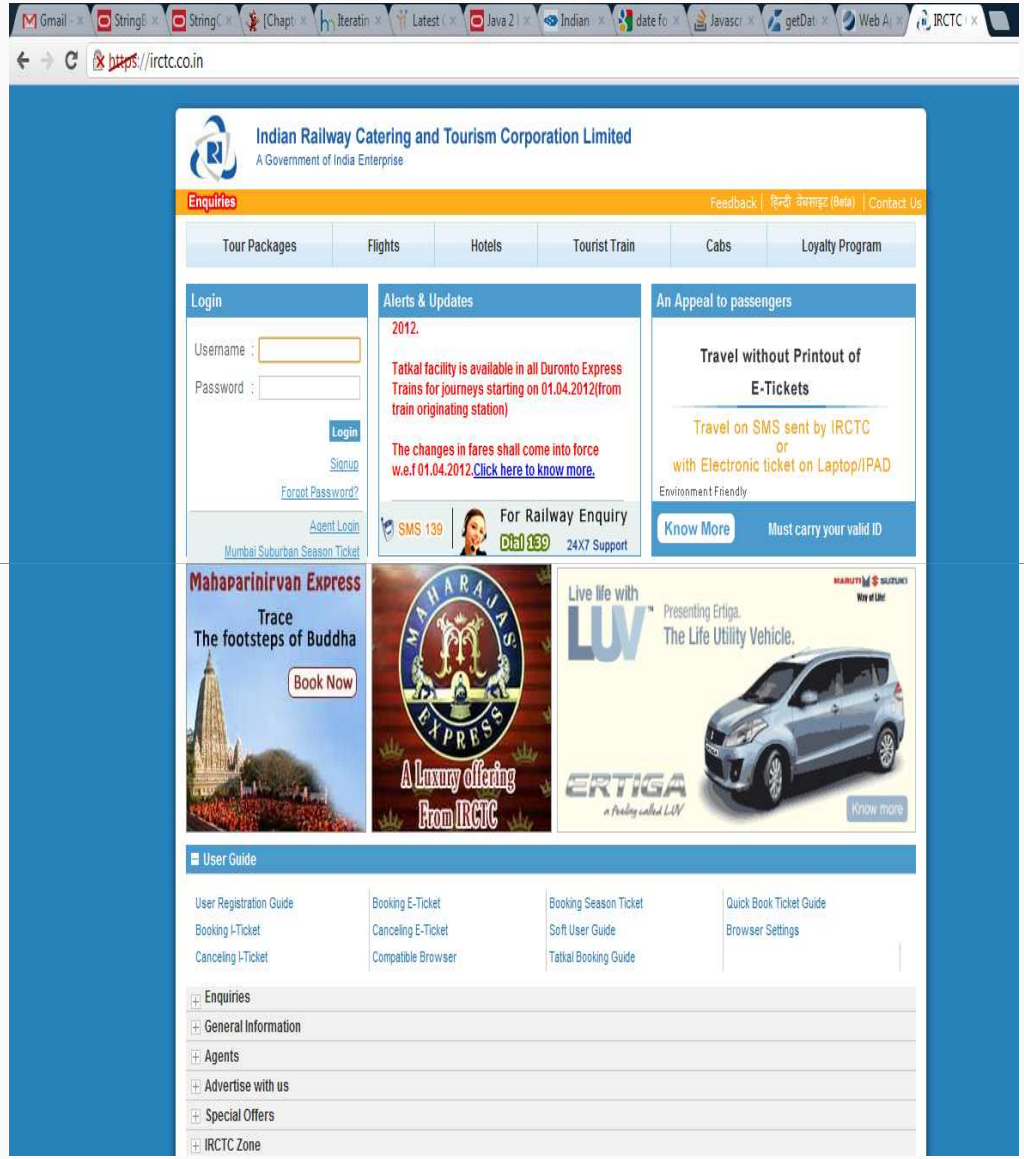
Web Application Security

PRESENTED BY:
Pankaj sharma
Scientist-c
Cert-in

What is a Web Application?

A web application or web service is a software application that is accessible using a web browser or HTTP(s) user agent.

Example of a Web Application is a website

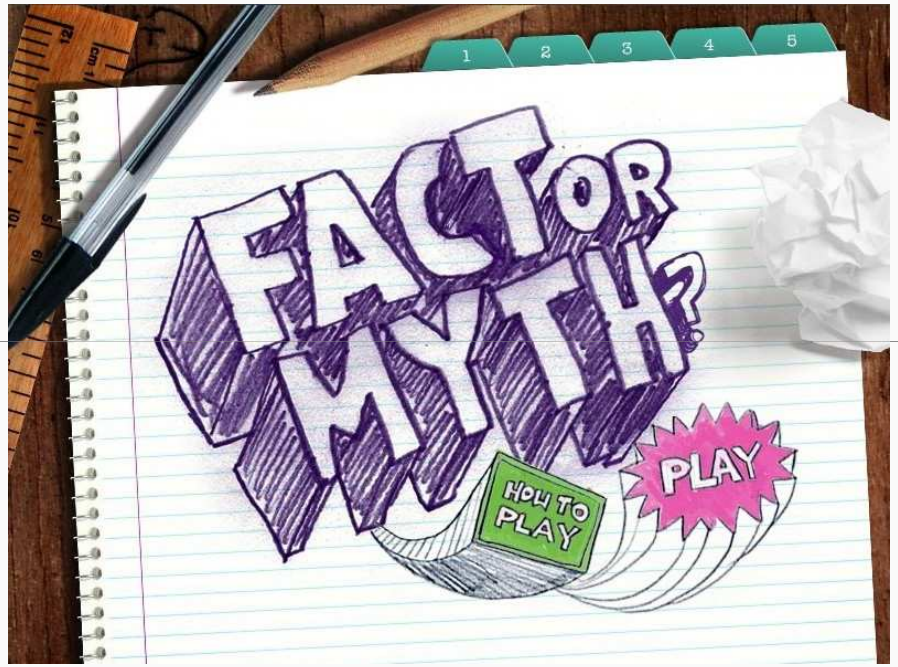


Overview

- Background
- Web application vulnerabilities
- Securing web apps
- Database Security
- Compromise and Countermeasures
- Conclusion

BACKGROUND - COMMON MYTHS

- The users will follow the rules while accessing a website or web application.
- The users will only send required input
- The Application has all enforcements
  – Client side Java scripts will take care of validation
  – Username and password is encoded on login page
- We use SSL(Secure Sockets Layer)
- Technology takes care of security
  – My firewall thwarts all attacks
  – My IDS/IPS can detect any attacks



Almost every web application is vulnerable!

- "8 out of 10 websites vulnerable to attack"— WhiteHat "security report "

- "75 percent of hack happen at the application." – Gartner "security at Application level"

- 64 percent developers are not confident in their ability to write secure applications."-Microsoft Developer Research

HOW TO SECURE WEB APPLICATIONS

Educate

- *Developers* – Software security best practices
- *Testers* – Methods for identifying vulnerabilities
- *Security Professionals* – Software development, Software coding best practices
- *Executives, System Owners, etc*. – Understanding the risk and why they should be concerned


OWASP TOP TEN 2011

- A1: SQL Injection
- A2: Cross-Site Scripting (XSS)
- A3: Broken Authentication and Session Management
- A4: Insecure Direct Object References
- A5: Cross-Site Request Forgery (CSRF)
- A6: Security Misconfiguration
- A7: Insecure Cryptographic Storage
- A8: Failure to Restrict URL Access
- A9: Insufficient Transport Layer Protection
- A10: Unvalidated Redirects and Forwards

The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are.

LET US UNDERSTAND A FEW.....

INJECTION

Code injection can be used by an attacker to introduce or "inject" code into a computer program to change the course of execution.

The results of a code injection attack can be disastrous.

For instance, code injection is used by some computer worms to propagate.

SQL injection is an often used technique to attack databases through a website. This is done by including portions of SQL statements in a web form entry field in an attempt to get the website to pass a newly formed rogue SQL command to the database.

SQL INJECTION

SQL Injection Attack

- SQL injection is an attack technique that tricks database to execute unintentional malicious commands.
- Most commonly executed by directly web forms, but can also be directed through URL hacking, request hacking using debugging tools, or using bots that emulates browsers and manipulate web requests.

OWASP Rank

- Ranks Top most Vulnerability in 2010
- Ranked 2nd Top in 2007

What I am Dealing With?

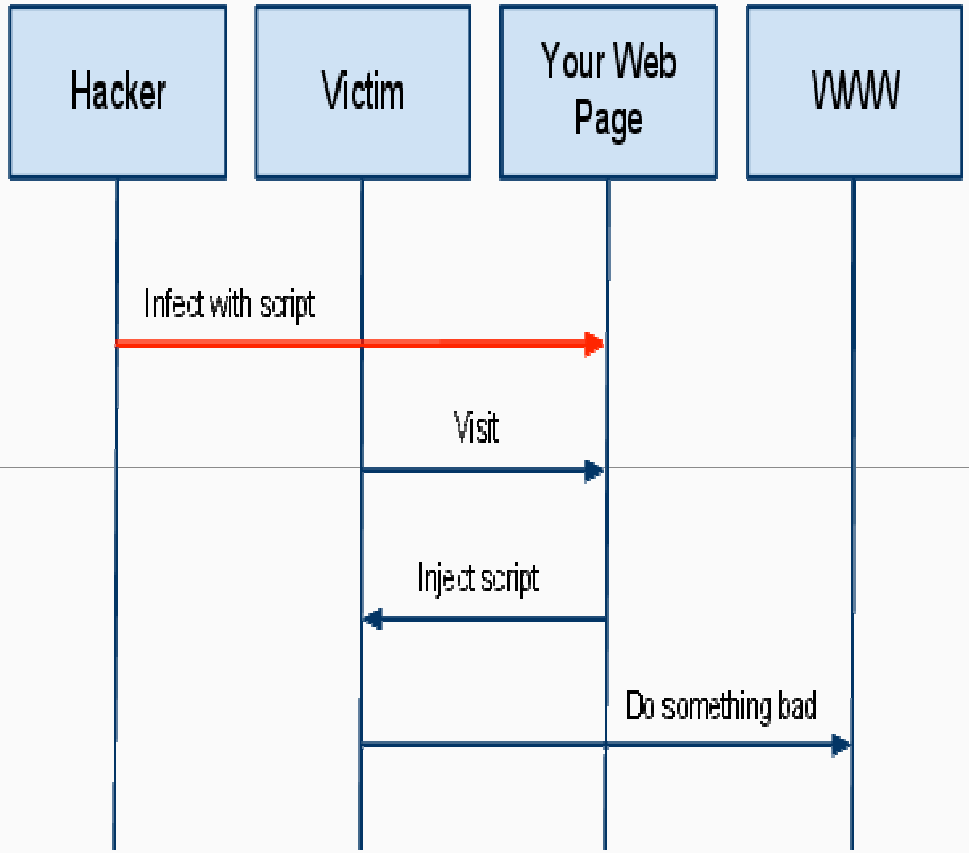
DEFENDING INJECTION

- Strong Input Validation
- Avoid dynamic queries
- Open low privilege connection to Database

CROSS SITE SCRIPTING(XSS)

Cross-site scripting holes are web-application vulnerabilities that allow attackers to bypass client-side security mechanisms normally imposed on web content by modern web browsers.

By finding ways of injecting malicious scripts into web pages, an attacker can gain elevated access-privileges to sensitive page content, session cookies, and a variety of other information maintained by the browser on behalf of the user.



A High Level View of a typical XSS Attack

DEFENDING XSS

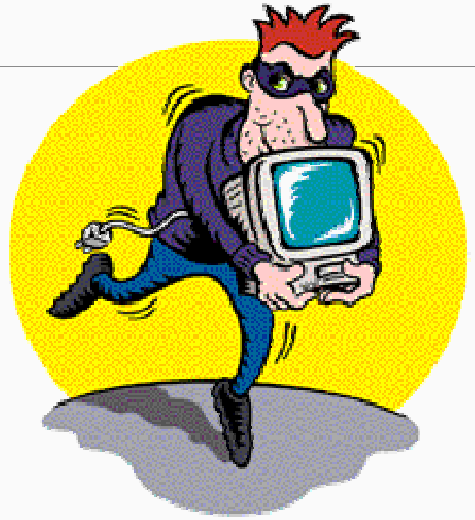
- Total server side validation
  – All headers, cookies, query strings, form fields and hidden fields
  – Positive filter

- Sanitize inputs & outputs
  – Prevents inserted scripts from being transmitted to users in an executable form
  – Define own static error message pages

WHAT CAN I DO AS A USER

- Logout of all sessions when done
- Do not select the “Remember me” Option
- Protect your cookies! Desktop Security
- Ensure you use SSL – when given choice of standard / secure login
- Patch your browser to be safe from some nasty Cross-site Scripting attacks
- Treat emails with Session ID info in URL’s just as securely as username/passwords

DATABASE SECURITY



WHAT IS DATABASE SECURITY

- The protection of the database against unauthorised access, either intentional or accidental

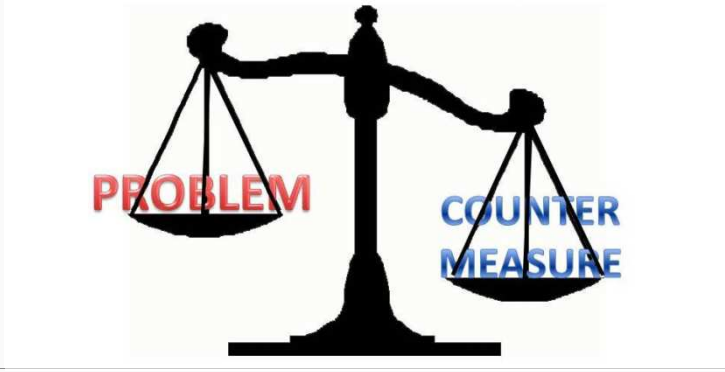
- Security **countermeasures** should combat **threats** and the **outcomes** of such threats

OUTCOMES OF SECURITY COMPROMISE

- Theft and fraud
- Loss of confidentiality
- Loss of privacy
- Loss of integrity
- Loss of availability

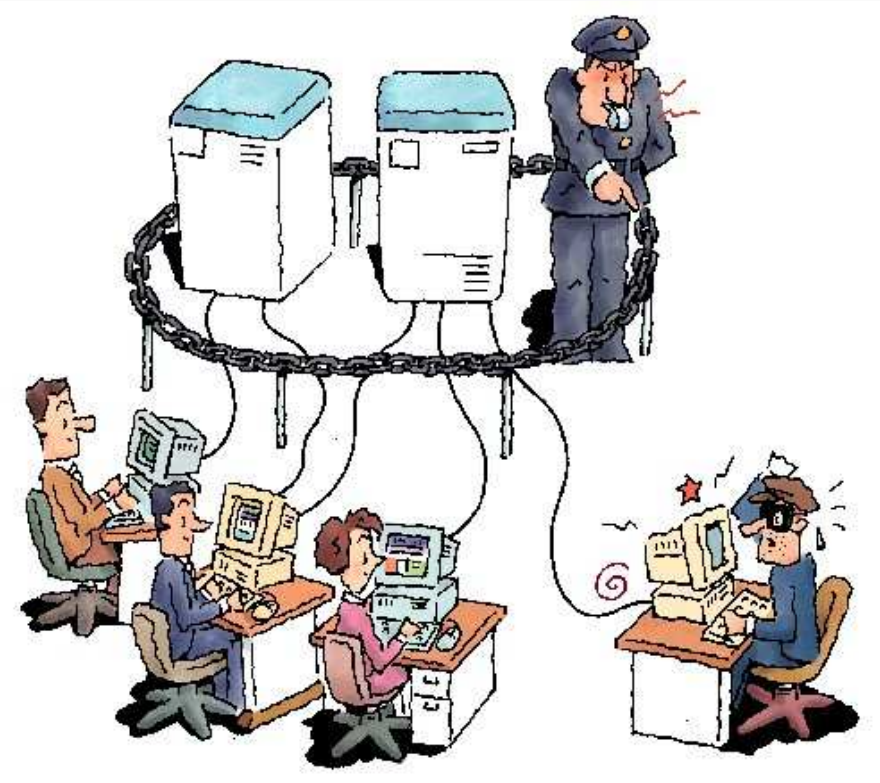
TYPE OF COUNTERMEASURES

- Authorization
- Views
- Backup and recovery
- Integrity
- Encryption
- RAID



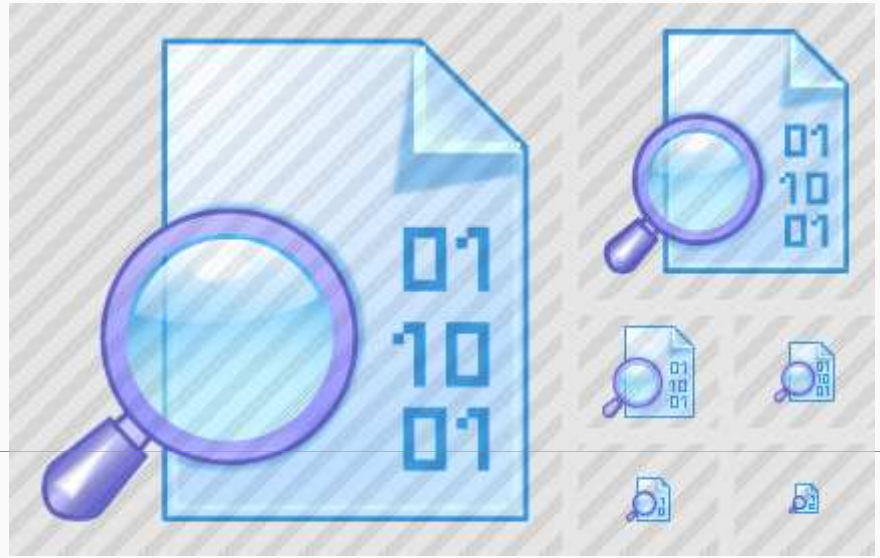
AUTHORIZATION

- Different authorization for different users
  - Accounts clerk
  - Accounts manager
  - End users



VIEWS

- A means of restricting user access to certain data
- Can restrict the viewing of specific attributes within a table
- More restrictive way of maintaining access control at a sub-table level



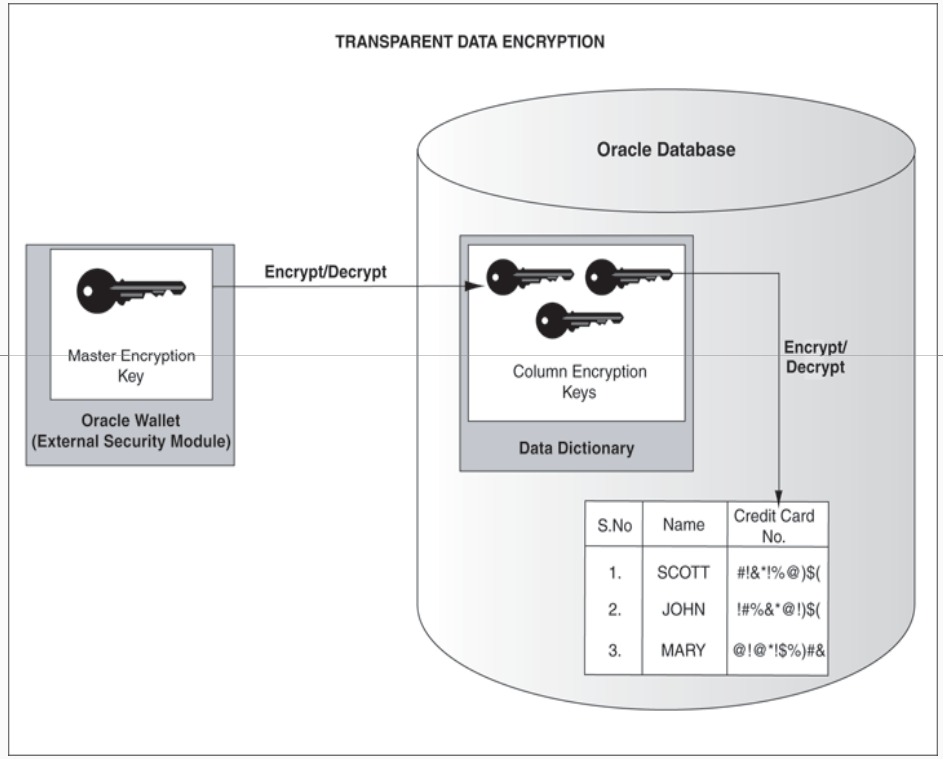
DATABASE BACKUP AND RECOVERY

The process of periodically taking a copy of the database and log file on to offline storage media.

DBMS should provide backup facilities to assist with the recovery of a database failure.

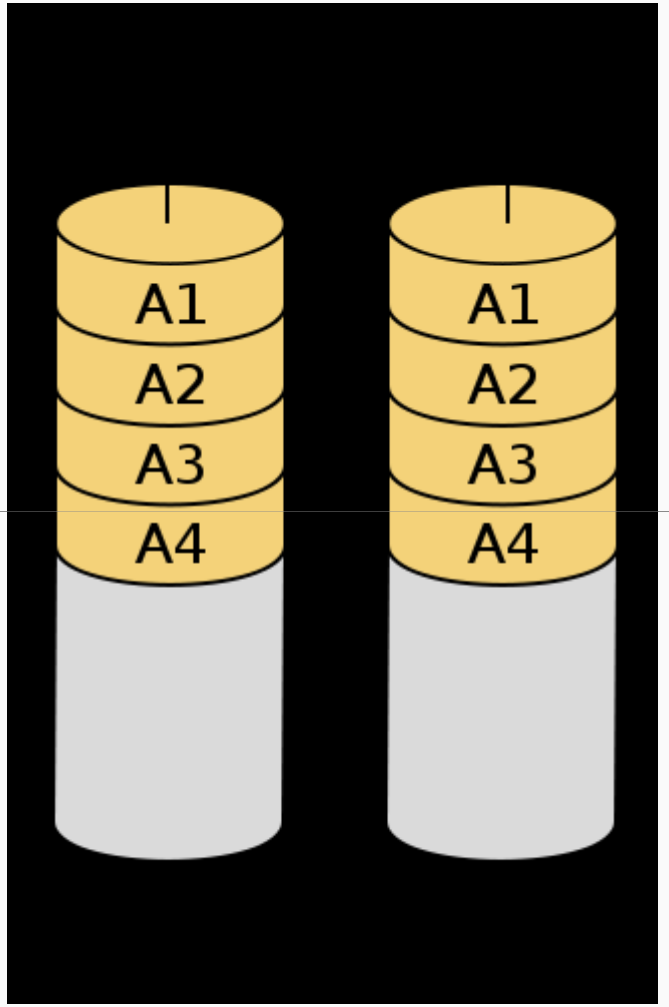
ENCRYPTION

- The encoding of data by algorithm such that the data is unreadable without the means to decode
- Symmetric key encryption
  - e.g. DES
- Asymmetric key encryption
  - e.g. RSA
- Sometimes used together in practice



RAID

- Redundant Array of Independent Disks
- Addresses similar issues to backup and recovery
- Copies data across an a bank of disks
  - Different RAID levels can be used addressing redundancy (mirroring), striping and parity
- Creates a more fault tolerant system than using a single disk



CONCLUSION

- Security is an essential aspect of databases in ensure data:
  - correct and complete
  - only accessible to those that should be allowed to see it
- The web model of databases adds another need for security in the communication of data


Thank you!