LIYANA ARACHCHIGE RANIL

SHORT NOTES/ NETWORK HACKS/Other TERMS

1. **IP ADDRESS SPOOFING** – Change the IP address of TCP packets to pretend the source from which the packet originated from. This is commonly used in Denial Of service attacks. The attacker is not interested in reply packets from the destination. IP address spoofing usually can be avoided by Packet Filtering known and INGRESS FILTERING. It filters all the packets whose ORIGINATING IP address is from within the internal network.

2. **MAN IN THE MIDDLE ATTACK** – An active eavesdropping where the attacker makes independent connections with the victims and relay messages between them. The parties will think that they are having a private connection between them but truly the MIM is controlling the conversation

3. **PHISHING** – Email Phishing, Web Site Phishing, emails or web sites claiming that they are legitimate entities and makes you to believe to use your usernames and passwords for actions. Phishing is mostly done by duplicating a site to look like a legitimate site and making users to visit the site

4. **XSS** – Cross Site Scripting, Computer security vulnerability that enables malicious attacker to inject client side scripts into web pages viewed by others. An exploited cross site scripting vulnerability can be used by attackers to by pass access controls such as SAME ORIGIN POLICY.

5. **SQL INJECTION** – A code injection technique to exploit security vulnerability occurring in the DB layer of the application

6. **DNS SPOOFING** – Resolving the IP address of a DNS entry to a FAKE IP ADDRESS

7. **DNS HIJACKING or DNS REDIRECTION** – Redirecting DNS resolution to another DNS server which could be a rouge or bogus server. This could ultimately take the users to unintended web sites and make them believe that they are in the correct site

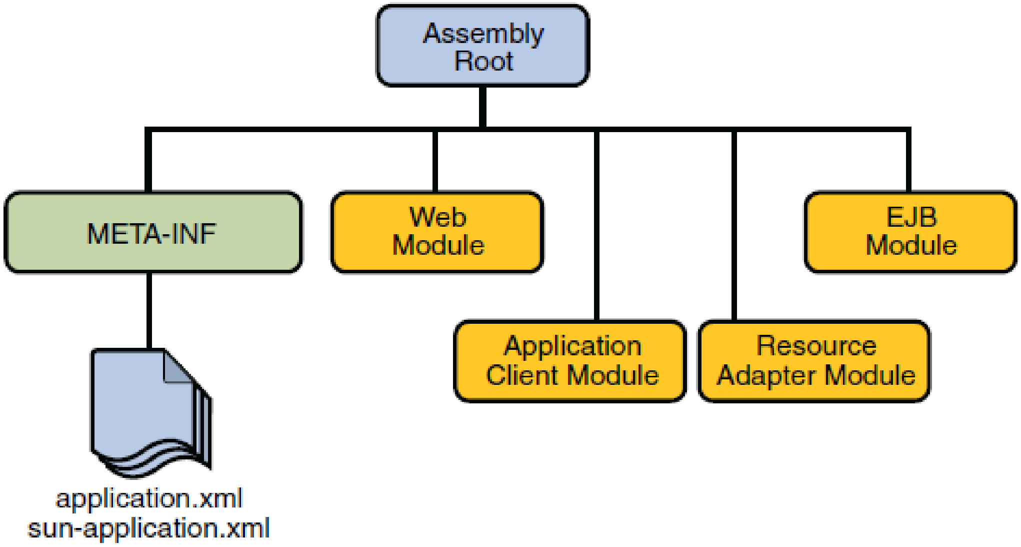
8. **DNS CACHE POINSONING** – When DNS server's cache UNAUTHENTICATED data for performance reasons DNS POISONING can happen. The DNS resolution will no longer be correct

9. **OFF BOARD SERVERS** – An off board server is a server which is communicating with a Legacy system using legacy systems protocol and communicating with the outside world using industry protocols

10. **SCREEN SCRAPERS** – When a legacy system does not have proper documentation or the source code is not available then the last option is to have a TERMINAL EMULATOR, or a SCREEN SCRAPER. This would use the UI in the legacy system and copy its values to the new one

11. **EAR** – A JAVA EE application is delivered in an EAR (Enterprise Archive) file. This is a standard JAR file with .ear extension. An EAR file contains JAVA EE modules and deployment descriptors.

FIGURE 1–6 EAR File Structure



12.

13. PROTOCOL

a. **HTTP** –

i. The most important protocol in the World Wide Web

ii. A request / response protocol

iii. HTTP communication usually happens on TCP/IP

iv. HTTP 1.0 and HTTP 1.1 has a primary difference on how it handles connections

v. In HTTP 1.0 every time you make a request there is a new connection while in HTTP 1.1 the connection that is made previously can still be used persistently for the other requests as well. This reduces network traffic while the web service is relieved from constant connections

vi. HTTP simple in implementation and simple to extend

vii. Most of the firewall would not block this traffic

viii. Stateless in nature

ix. Use cookies , URL rewriting , or HTTPS for state maintenance

x. Inherently insecure since information is passed as plain text

xi. HTTP is CONNECTION-BASED

b. **HTTPS** –

i. This is HTTP over SSL / TLS

ii. Secure , State is maintained and usually fire walls allow this traffic

iii. Slower and Expensive in terms of resource usage

iv. HTTPS is CONNECTION-BASED

c. **IIOP** –

i. Internet Inter ORB Protocol , allows communication among heterogeneous object oriented systems

ii. Starting with CORBA 2.3, pass by VALUE is supported. Hence applications can transfer objects from the server to the client by copying. Prior to version 2.3 CORBA only supported remote REFERENCES.

iii. Java Application can use IIOP in two ways , either with RMI-IIOP or IDL / IIOP (Business as usual)

iv. RMI/IIOP is simpler in terms of coding and implementing

v. RMI/IIOP uses CORBAS interoperability with RMI’s ease of programming

vi. No DEFAULT port exist for IIOP

vii. PORT are DYNAMICALLY assigned when object server binds to a name

viii. IIOP supports TUNNELING over HTTP to overcome firewall rules

ix. Interoperable standards , wide range of services are pros of this protocol

x. Performance is not that good for this

xi. IIOP is CONNECTION BASED

d. JRMP

i. Java Remote Method Protocol

ii. Connection based and Stateful protocol

iii. JRMP supports NAMING SERVICE which runs on the default port 1099

iv. The Object Servers are dynamically assigned port numbers as in IIOP

v. JRMP clients have built in support for HTTP tunneling

vi. A J2EE application serve is required to give support for RMI/IIOP protocol

vii. JRMP is simple and performance is higher

viii. Crons are , JRMP is not supported by Firewalls , hence need tunneling with a RMIServlet,The only available service is the Naming Service not like in CORBA, Works only in JAVA environment

ix. JRMP is CONNECTION-BASED