TATA CONSULTANCY SERVICES
Experience certainty.

Security Testing – Test Execution Approach

17/02/2014

Vihang Shah

Gemology - ERU / ASU Testing Services

vihang.shah@tcs.com

knowmax
Our Intelligence Hub

Confidentiality Statement

Include the confidentiality statement within the box provided. This has to be legally approved

Confidentiality and Non-Disclosure Notice

The information contained in this document is confidential and proprietary to TATA Consultancy Services. This information may not be disclosed, duplicated or used for any other purposes. The information contained in this document may not be released in whole or in part outside TCS for any purpose without the express written permission of TATA Consultancy Services.

Tata Code of Conduct

We, in our dealings, are self-regulated by a Code of Conduct as enshrined in the Tata Code of Conduct. We request your support in helping us adhere to the Code in letter and spirit. We request that any violation or potential violation of the Code by any person be promptly brought to the notice of the Local Ethics Counselor or the Principal Ethics Counselor or the CEO of TCS. All communication received in this regard will be treated and kept as confidential.

Table of Content

List of Figures

1. Introduction

The Internet has revolutionised business operations across the globe. While IT infrastructure is the backbone, business applications (referred interchangeably with software hereafter) are the current growth engine for organisations. Applications provide an easy access through the Internet. However, protecting digital resources that are exposed on the network is challenging. Availability of a large number of Internet-based critical resources and services attract security threats. These days software security is gaining importance because applications with ubiquitous characteristics produce numerous threats for confidential information and user's privacy.

TCS is committed towards ensuring the security and privacy of customers and has therefore developed a Software Security Assurance (SSA) process. This is part of TCS's overall risk management programme to embed security in the software development lifecycle (SDLC). SSA reduces the possibility of requirement oversights, design flaws, implementation bugs and configuration mistakes during the SDLC.

The objective of application Security testing is to check application's ability to:

- Filter Resist most attacks
- Resist most attacks
- Tolerate attacks that cannot be resisted
- Recover within a specified time with minimum damage and
- Generate a trail to identify source and path of attack

2. Scope

Application Security assessment of GIA NGGS applications is in scope of the current engagement and Black Box security testing is covered.

2.1 In Scope

The TCS Application Security Testing process is based on OWASP (Open Web Application Security Project) Top 10 guidelines and TCS SSA Guidelines.

Black Box Security testing covered the non-functional aspects of the following major areas:

- Authentication Testing
- Authorization Testing
- Data Security and Data Validation Testing
- Testing for secure audit logging
- Testing for Client side attacks
- Insecure Cryptography
- Testing for Configuration Management

Test cases were designed for performing the tests mentioned above and issues were logged in Oracle Test Manager tool.

2.2 Security Testing Phases and Solution Approach

Black Box Security testing approach consists of four separate phases:

- Understanding
- Analysis
- Executions
- Reporting

- Understanding and Analysis phases are to plan the execution to achieve focused and maximum coverage depending on the risk category of features available in the application.
- Manual test execution done to uncover vulnerabilities, filtering of false positives and checking for any remaining areas.
- Reports are shared in two different formats, one in summary format to bring all stakeholders to a common understanding on existing vulnerabilities and a detailed report which would help developers fix the issue across the application.

2.3 Test Execution Approach

- Generic test cases were prepared based on the modules of the application. Test cases were divided based on Authentication, Authorization, Session Management, Data Security, Data Validation, Client Side Attack and Buffer Overflow.

- List of certain Questionnaires were made and were forwarded to the Development team. Answers/Feedback was documented in execution results and recommendations were provided wherever applicable.

- Defects found during testing were logged in OTM (Oracle Test Manager). Through this tool synchronisation was maintained between Development and Testing team. This was the basic approach followed while testing.

- To ensure comprehensive test coverage:

  - All the test cases were executed at least once and selected test cases were executed for the required number of iterations during regression phase and retesting of defects.
  - Risk Based Testing (RBT) was carried out for covering the high priority test cases, when there was limited time available for testing.

- Reports were maintained on daily basis in order to keep a record and status of execution, so that we can get a clear picture of Security Testing progress. Reports were scanned and analysed thoroughly on daily basis which was really helpful for us. Reports were made manually as well as through report feature in OTM.

3. Security tool

Security tool is a fake antivirus programme that gives misleading security threats on an application and security notifications that makes an application assume it's infected by malware. Security tools are perpetuated through the use of Trojans on the internet. Security tools are necessary as they protect an application from Security threats.

Some of the commonly used tools are:

- Paros Proxy
- OWASP ZAP(Zed Attack Proxy Project)
- BeEF(The Browser Exploitation Framework Project)
- Burp Suite
- PE Studio
- OWASP Xenotix
- Lynis The Hardening Unix Tool
- Recon-Ng The Web Reconnaissance Framework
- Suricata The Network IDS/IPS
- WPScan Word Press Security Tool

We have used PAROS Proxy tool as it is an open source tool and security testing can be done easily.

3.1 PAROS

Paros is a valuable testing tool for your security and vulnerability testing. Paros can be used to spider/crawl your entire site, and then execute canned vulnerability scanner tests. But Paros goes beyond that, it comes with a built in utility that can proxy traffic. This Paros Proxy utility can be used to tamper or manipulate any http or https traffic on the fly. This makes some of the more interesting security types of testing. It will help you isolate potential areas of security concern and then manual attempt to perform the type of testing you desire.

We need to change certain proxy settings (In Internet Explorer) for PAROS in order to use it in our application. In order to use Paros we need to perform following steps. Navigate-> Tools-> Internet options -> Connections -> LAN settings -> Advanced ->In HTTP textbox write local host instead of proxy.tcs.com for all, and if required change the port no(only if port no is not available). Now uncheck the Bypass proxy server for local address. In Paros tool we need to perform following steps. Navigate Tools-> Options-> Local Proxy. Here Address and Port size should be same as Internet Explorer.

In Paros tool we need to perform following steps. Navigate Tools-> Options-> Local Proxy. Here Address and Port size should be same as Internet Explorer.

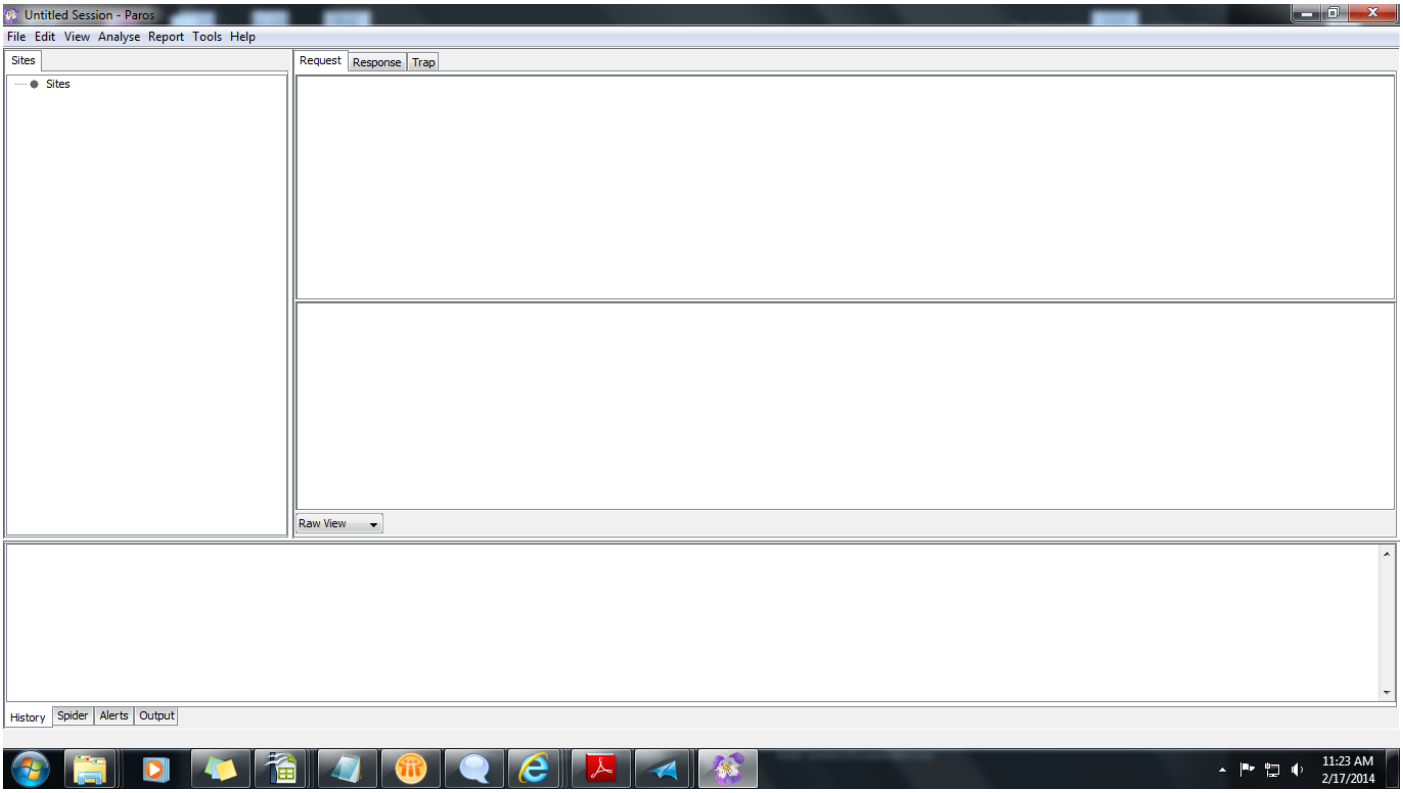


Figure 1: Paros Overview

4. OWASP (The Open Web Application Security Project)

The Open Web Application Security Project (OWASP) is an open community dedicated to enabling organizations to develop, purchase, and maintain applications that can be trusted. Following are top 10 OWASP security risks.

4.1 Injection

Injection flaws, such as SQL, OS, and LDAP injection occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing unauthorized data. In Injection we enter negative values in any field using Paros so as to check whether the application is working fine or not.

4.2 Broken Authentication and Session Management

Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, session tokens, or exploit other implementation flaws to assume other users' identities. In application we press F12 and navigate to Cache->View Cache information and cookie should be send with secure attribute.

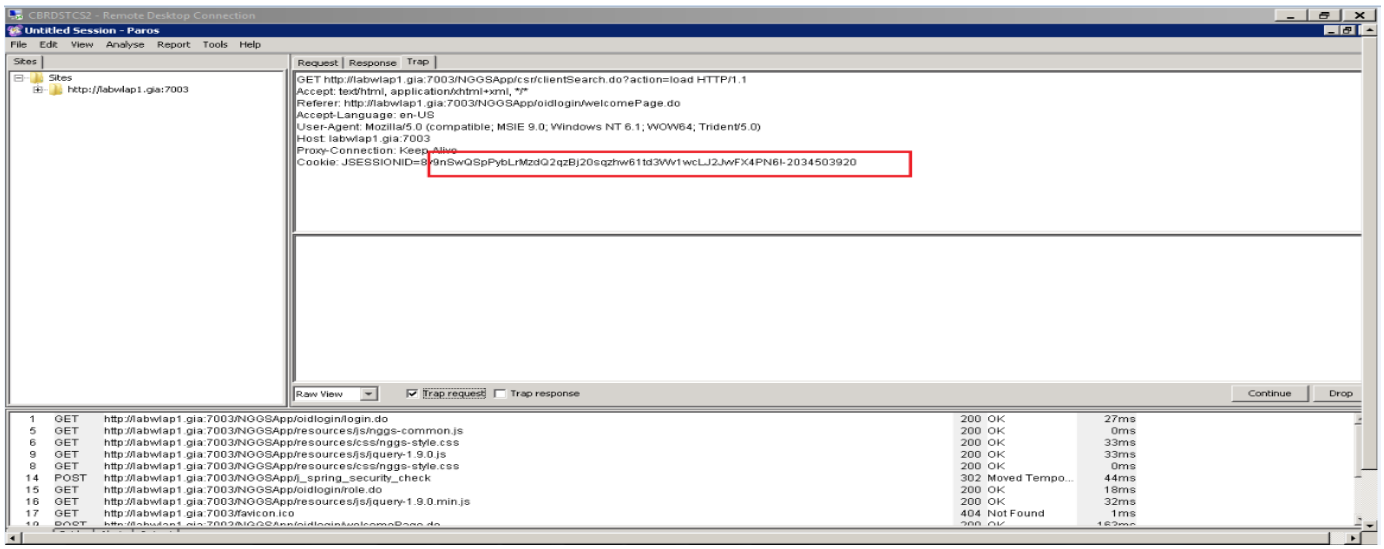


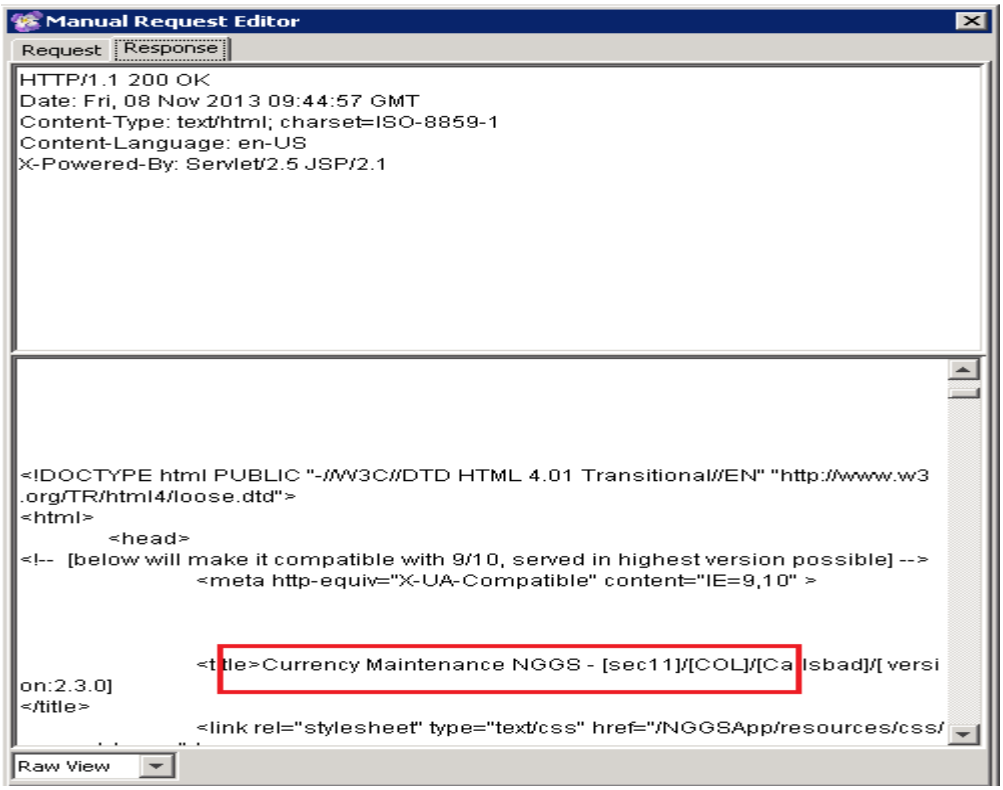
Figure 2: Broken Authentication and Session Management

4.3 Cross-Site Scripting (XSS)

XSS flaws occur whenever an application takes untrusted data and sends it to a web browser without proper validation or escaping. XSS allows attackers to execute scripts in the victim's browser which can hijack user sessions, deface web sites, or redirect the user to malicious sites. In Cross Site Scripting we need to enter various scripts in order to hack the screen.

4.4 Insecure Direct Object references

A direct object reference occurs when a developer exposes a reference to an internal implementation object, such as a file, directory, or database key. Without an access control check or other protection, attackers can manipulate these references to access unauthorized data.



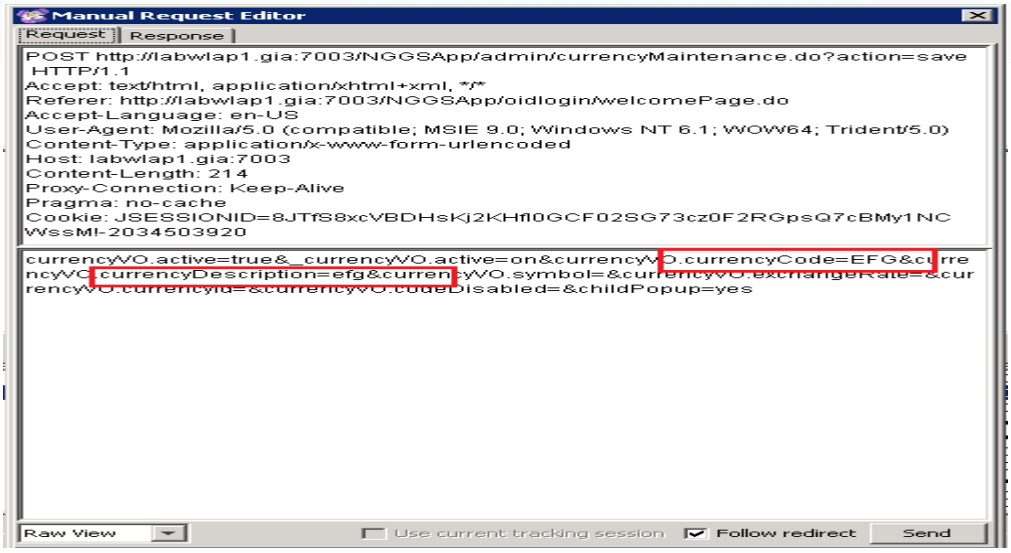


Figure 3: Insecure Direct Object Reference

4.5 Security Misconfiguration

Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. All these settings should be defined, implemented, and maintained as many are not shipped with secure defaults. This includes keeping all software up to date.

4.6 Sensitive Data Exposure

Many web applications do not properly protect sensitive data, such as credit cards, tax ids, and authentication credentials. Attackers may steal or modify such weakly protected data to conduct identity theft, credit card fraud, or other crimes. Sensitive data deserves extra protection such as encryption at rest or in transit, as well as special precautions when exchanged with the browser. In Paros we check the Trap Request checkbox in Paros, enter the User Name and Password in application and click on Login. Observe the request intercepted in Paros. We need to check that Application should use a proper secure communication channel between browser and web server which does not transmit the credentials in plaintext.

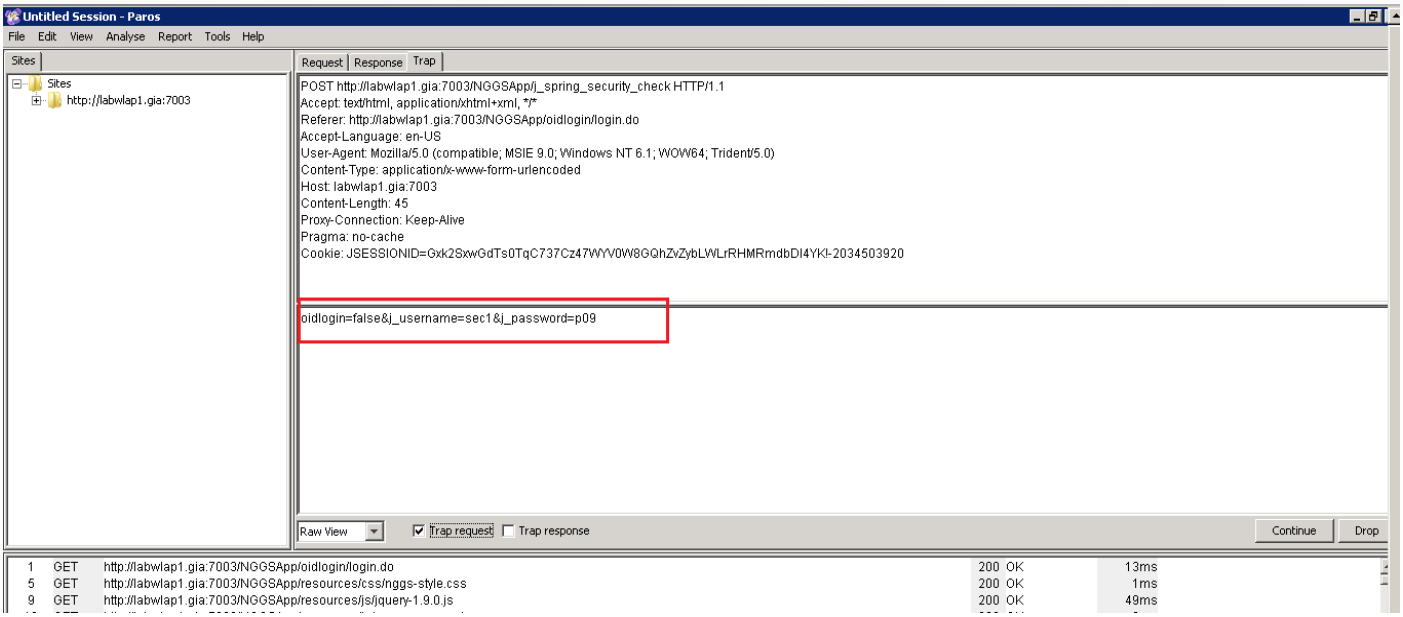


Figure 4: Sensitive Data Exposure

4.7 Missing Function Level across Control

Virtually all web applications verify function level access rights before making that functionality visible in the UI. However, applications need to perform the same access control checks on the server when each function is accessed. If requests are not verified, attackers will be able to forge requests in order to access unauthorized functionality.

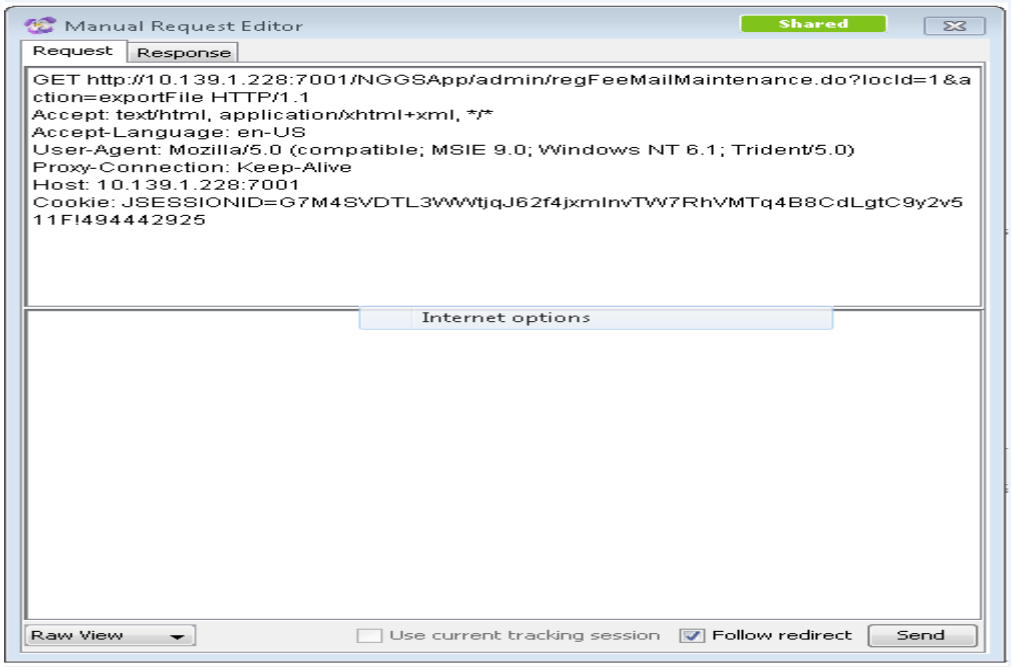


Figure 5: Missing Function Level across control

4.8 Cross-Site Request Forgery (CSRF)

A CSRF attack forces a logged-on victim's browser to send a forged HTTP request, including the victim's session cookie and any other automatically included authentication information, to a vulnerable web application. This allows the attacker to force the victim's browser to generate requests the vulnerable application thinks are legitimate requests from the victim. Intercept the request and get the URL through Paros. Enter the HTML Code in a separate notepad and save it with .html extension. Open the HTML file and click on the link. Screen will get opened; enter the details and trap the request. In Paros enter negative value in any field. Output should remain same and no changes should be done.

4.9 Using Components with Known Vulnerabilities

Vulnerable components, such as libraries, frameworks, and other software modules almost always run with full privilege. So, if exploited, they can cause serious data loss or server takeover. Applications using these vulnerable components may undermine their defences and enable a range of possible attacks and impacts.

4.10 Unvalidated Redirects and Forwards

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

Thank You

Contact

For more information, contact **vihang.shah@tcs.com**

About Tata Consultancy Services (TCS)

Tata Consultancy Services is an IT services, consulting and business solutions organization that delivers real results to global business, ensuring a level of certainty no other firm can match. TCS offers a consulting-led, integrated portfolio of IT and IT-enabled infrastructure, engineering and assurance services. This is delivered through its unique Global Network Delivery Model™, recognized as the benchmark of excellence in software development. A part of the Tata Group, India's largest industrial conglomerate, TCS has a global footprint and is listed on the National Stock Exchange and Bombay Stock Exchange in India.

For more information, visit us at **www.tcs.com**.

IT Services
Business Solutions
Consulting