Function	Objective	Risks	CIS Top 20 Reference	Test Steps	Expected result
Web site security	Customers or hackers are prevented from entering malicious javascript code at the TRA (Toll road authority) invoice input box. Also, they prevent SQL injection at the username and password console.	Entering malicious javascript code referred to as cross-site scripting in the input box can result in malware such as worms and viruses. SQL injection at the username and password console will result in the attacker getting hold of the admin username and password and compromising the web server.	Control 18: Application Software Security	**Utilize pentesting tools** such as Nessus and Netsparker to scan for website vulnerabilities such as cross-site scripting and SQL injections. Test for **proper input validation controls for HTML/Javascript and SQL.**	The pentesting tools find no website vulnerabilities. The input validation controls are in place.
WebLogic servers	The patch management process is effective. The WebLogic Server configuration data which are stored in clear text on the file system are not read by intruders	Uneffective patch managment will result in vulnerable operating systems and applications where intruders can compromise the systems. An unencrypted filesystem can result in an intruder defeating the application security established with WebLogic Server authentication and authorization.	Control 4: Vulnerability Assessment and Remediation	Develop a robust **patch management process with Weblogic Domain with Enterprise Manager Cloud Control.** Create and maintain an accurate inventory of assets and a schedule the installation of patches from the oracle support website which the Enterprise Manager will link to. You can also set rollback and validation steps to a patch plan you create. **Configure domain-wide admin port** which separates administration traffic from application traffic. The admin port requires SSL and reduces risk for transmitting server configuration information in plain text. **Configure weblogic domains in production mode** which uses more stringent security parameters than default development mode.	The patching management process checks and patches the system vulnerabilities on a regular schedule. The Weblogic server configuration data is transmitted with SSL and therefore is not in plain text.
Database security	The OLTP (online transaction processing) oracle database which the TRA transactions use is ACID compliant. ACID ensures that the transactions are atomic, consistent, isolated and durable.	Non-compliant ACID databases can lead to data loss and data inconsistencies.	Control 13: Data Protection	The testing for database ACID compliance involves four requirements (atomicity, consistency, isolation, and durability). 1. For **atomicity**, when the customer pays their invoice online using their bank account, the credit of the TRP invoice balance must follow the debit from the bank. If the debit fails then the whole transaction fails. In other words, everything must fail or everything must succeed. 2. For a transaction to be **consistent,** the TRP invoice for example cannot be overpaid and thus result in a negative balance. 3. For a transaction to be **isolated,** a customer paying an invoice for a particular date cannot affect the invoice balance on the other dates. 4. Finally, for a transaction to be **durable** , there should be no data loss in case of power outages or accidental server shutdowns.	The oracle database is ACID compliant. That is, it is atomic, consistent, isolated, and durable

Transaction processing	The input data entered in the input fields is validated to allow for correct alphnumeric data. on the website should prevent an intruder from entering malicious code. The input data should result in complete transactions to save the customer time.	An intruder can enter unvalidated input data as malicious code. The TRA can lose customer satisfaction and thus business if the web application does not complete the customer's transactions.	Control 13: Data Protection	The following data edits and controls for the transactions on the TRA web application should be tested: **1. Confirm that there is a sequence check.** For example, when an invoice is generated for a missed toll or toll violation the invoices should be numbered sequentially where a specific range implies a particular range. If the invoice number is out of range the record is discarded and implies that there is a error in the transaction. This way the customer does not get billed in error. Another example, is when the customer is a veteran. When this customer opens a new account and enter their veteran specialty license plate number , a sequence check will be placed next to the plate number and the tolls will be free. **2. Confirm that there is range check.** For example, the total purchase of the tolls in one given day should be in the range between 0 and 100. If the invoice is below 0 and more than 100 there is an error. You cannot have negative balance and it is not reasonable that total tolls can be more than 100. **3. Confirm that there is an existence check.** For example, for a new account when the customer enters their license plate number and state residence, a search will take place to confirm that the license plate number entered is registered. **4. Confirm that there is a completeness check.** When a new account is opened by the customer the customer will not be able to proceed unless the credit card number, expiration date, and CVV number is entered.These fields will have and asterick indicating required fields. **5. Confirm that there is a validity check**. For example, the credit card numbers, bank account numbers and invoice numbers must be numeric. If a letter or character is entered there will be a error popup box notifiying the customer to enter a numeric value. **6. Confirm that there is a duplicate check.** New transaction are matched to those previously input to ensure that they have not already been entered. For example, the current TRA invoice number is checked with past invoice numbers to confirm there are no duplicates and the customer pays the vendor twice.	Testing should be able to determine whether all the data checks are performing the way they should.
Remote access security	Remote access by the vendors at odd hours during the day are not prone to security attacks.	System attacks can lead to compromised servers and databases where Personal Identifiable Information (PII) is stolen.	Control 1: Inventory of Authorized and Unauthorized Devices	The following needs to be tested to ensure secure remote access for the vendor: **1. confirm the user has temporary access** and is restricted to certain internal hosts. **2. confirm that a site-to-site virtual private network** is configured where the vpn tunnels are established automatically ; modern VPN tunneling protocols such as L2TP/IPsec and SSL/TLS SSTP should be used. **3. confirm that each user from the vendor has their own username and password** to be validated by the LDAP server. **4. confirm that MFA (multifactor authentication)** is enabled. **5. confirm that administative access** for all users from the vendor is disabled **6. confirm that SSH (secure shell) or Microsoft RDP** (remote desktop protocol) for secure remote connections is used.	Testing should be able to determine whether remote access for the vendors is secure.
Proper separation of the PROD, DEV and TEST environments	Users in development do not have access to more sensitive data than they need to do their jobs. That is, users with access to DEV should not be allowed to make changes to PROD.This is referred to as "separation of duties" where the tasks and associated privileges for a specific security are spread amoung multiple people.	If the developer's usernames and passwords get compromised and they can update PROD, then the attacker will have access to PII such as credit card numbers and steal money.	Control 14: Controlled Access Based on the Need to Know	The following needs to be tested to ensure proper separation of PROD, DEV and TEST environments. **1. Confirm that three virtual machines (PROD, DEV, and TEST)** should be created on RHEL using VMware for example. In VMware a snapshot can be taken of PROD when powered on to be used for the DEV and TEST environments. **2. Confirm that users do not have "root" usernames nor "SYS" usernames.** Each username should be unique to the user who has administrative rights. **3. Confirm that Linux admin and Oracle admin groups are created** and that they contain users with the appropriate read, write and execute permissions for the filesystem.	Testing should be able to determine if there is proper "separation of duties" for the PROD, DEV, and TEST environments and the appropriate permissions for users in these environments.
Logging and Monitoring	Security logging and monitoring is being performed on database and application changes. In addition, logging and monitoring for the vendors is performed and prioritized as critical.	The security team will be unaware of an intruder trying to compromise the applications and databases if there is no logging and monitoring.	Control 16: Account Monitoring and Control	The following tests need to be performed to ensure that logging and monitoring is enabled for applications, databases, and vendors: **1. Confirm that Splunk Enterprise Security** (a security information and event management solution) is installed. **2. Confirm that User Behavior Analytics** is used to log and monitor what the vendors are doing. **3. Confirm that Security Domain and Risk Analysis dashboards** are used to track login attempts, track breach endpoints and track and categorize assets by risk.	The installaton of Splunk Enterprise Security should allow the security team to view logs and monitoring events for the applications,databases and vendors.