

TUGAS

UAS KEAMANAN INFORMASI



Oleh :

Barnaba Habibi
1310652050

PROGRAM STUDI TEKNIK INFORMATIKA - SORE
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH JEMBER
2015

KONSEP PEMROGRAMAN

Mari kita mulai dengan memahami beberapa konsep pemrograman dasar . Seperti komputer telah menjadi lebih kuat dan di mana-mana , proses dan metode yang digunakan untuk membuat perangkat lunak komputer telah tumbuh dan berubah . Berikut adalah beberapa contoh dari konsep pemrograman :

- a) Kode mesin , kode sumber , dan perakitan
Kode mesin tergantung CPU ; itu adalah serangkaian 1s dan 0s yang menerjemahkan instruksi yang dimengerti oleh CPU . Source code adalah instruksi komputer 63 bahasa pemrograman yang ditulis dalam teks yang harus diterjemahkan ke dalam kode mesin sebelum eksekusi oleh CPU .
- b) Compiler , interpreter , dan bytecode
Compiler mengambil kode sumber , seperti C atau Basic , dan kompilasi ke dalam kode mesin . Bahasa ditafsirkan berbeda dari bahasa dikompilasi : Kode ditafsirkan dikompilasi dengan cepat setiap kali program dijalankan . Kode bytecode , seperti Java bytecode , juga ditafsirkan
- c) Jenis perangkat lunak dirilis untuk umum

Ada beberapa jenis perangkat lunak yang akan dirilis ke publik , seperti :

- Perangkat lunak open dan closed sumber
Software sumber tertutup adalah perangkat lunak biasanya dirilis dalam bentuk executable : kode sumber dirahasiakan . Perangkat lunak open source menerbitkan kode sumber terbuka . Perangkat lunak berpemilik adalah perangkat lunak yang tunduk pada perlindungan kekayaan intelektual seperti paten atau hak cipta . Contoh : Windows dan Linux
- Free Software , Shareware , dan crippleware
Freeware adalah software yang gratis untuk digunakan . Shareware adalah software proprietary yang berfungsi penuh yang dapat awalnya digunakan secara gratis . Jika pengguna terus menggunakan Shareware untuk jangka waktu tertentu yang ditentukan oleh lisensi (seperti 30 hari) , lisensi Shareware biasanya membutuhkan pembayaran
Crippleware adalah software proprietary sebagian fungsi , sering dengan fitur kunci dinonaktifkan . Pengguna biasanya diperlukan untuk melakukan pembayaran untuk membuka fungsionalitas penuh .

METODE PENGEMBANGAN APLIKASI

Adalah kerangka yang digunakan untuk struktur, rencana, dan mengontrol proses pengembangan sistem informasi. Berikut ini adalah beberapa contoh dari metode pengembangan aplikasi.

- Waterfall Model
Waterfall Model adalah aplikasi model pengembangan linear yang menggunakan rigidphases; ketika salah satu fase berakhir, berikutnya dimulai. Langkah terjadi secara berurutan, dan theunmodified Waterfall Model tidak memungkinkan pengembang untuk kembali ke langkah sebelumnya.

- Spiral

Model spiral adalah model proses pembangkit risiko-didorong untuk proyek-proyek perangkat lunak. Berdasarkan pola risiko unik dari proyek tertentu, model spiral

panduan tim untuk mengadopsi elemen dari satu atau lebih model proses, seperti tambahan, air terjun, atau prototyping evolusi.

- Pengembangan Agile Software

Pengembangan perangkat lunak Agile adalah sekelompok metode pengembangan perangkat lunak di mana persyaratan dan solusi berkembang melalui kolaborasi antara diri-mengorganisir, tim lintas-fungsional. Mempromosikan perencanaan adaptif, perkembangan evolusioner, pengiriman awal, perbaikan terus-menerus, dan mendorong respon yang cepat dan fleksibel terhadap perubahan

- Extreme Programming

Extreme Programming (XP) adalah metode pengembangan Agile yang menggunakan pasang programmer yang bekerja di luar spesifikasi rinci

- Rapid Application Development

Rapid Application Development (RAD) cepat mengembangkan perangkat lunak melalui penggunaan prototipe, database "boneka" GUI, back-end, dan banyak lagi. Tujuan dari RAD cepat memenuhi kebutuhan bisnis dari sistem; permasalahan teknis yang sekunder.

- SDLC

System Development Life Cycle (SDLC), juga disebut sebagai pengembangan aplikasi siklus hidup, adalah istilah yang digunakan dalam rekayasa sistem, sistem informasi dan rekayasa perangkat lunak untuk menggambarkan proses perencanaan, menciptakan, pengujian, dan menggunakan sistem informasi

OBJECT-ORIENTED PROGRAMMING

Adalah model bahasa pemrograman yang diselenggarakan di sekitar objek dan bukan tindakan dan data daripada logika . Secara historis , program telah dilihat sebagai prosedur logis yang mengambil input data , proses, dan menghasilkan output data .

- Cornerstone Object-Oriented Programming concepts

Cornerstone Berorientasi Objek konsep Pemrograman meliputi objek , metode , pesan , warisan , delegasi , polimorfisme , dan polyinstantiation . Kami akan menggunakan contoh objek yang disebut " Addy " untuk menggambarkan konsep landasan . Addy adalah obyek yang menambahkan dua bilangan bulat ; itu adalah sebuah benda yang sangat sederhana, tetapi memiliki kompleksitas yang cukup untuk menjelaskan konsep OOP inti . Addy mewarisi pemahaman tentang angka dan matematika dari kelas induk nya (kelas disebut operator matematika) . Berikut adalah ringkasan dari konsep pemrograman berorientasi objek ini digambarkan oleh Addy :

- Object→Addy
 - Class →Mathematical Operators
 - Method→Addition
 - Inheritance→Addy inherits an understanding of numbers and math from his parent class mathematical operators: the programmer simply needs to program Addy to support the method of addition
 - Example input message→1+2
 - Example output message→3
 - Polymorphism→Addy can change behavior based on the context of the input, overloading the + to perform addition or concatenation, depending on the context
 - Polyinstantiation→Two Addy objects (secret and top secret), with different data.
- Object Request Brokers

Adalah Program yang memungkinkan panggilan dibuat dari satu komputer ke komputer lain melalui jaringan komputer , memberikan transparansi lokasi melalui panggilan prosedur remote . ORB mempromosikan interoperabilitas sistem objek terdistribusi , memungkinkan sistem tersebut akan dibangun dengan mengumpulkan benda-benda dari vendor yang berbeda , sementara bagian yang berbeda berkomunikasi satu sama lain melalui ORB .

- COM and DCOM
- Adalah teknologi proprietary Microsoft untuk komunikasi antara komponen perangkat lunak yang didistribusikan di seluruh jaringan komputer . DCOM , yang awalnya disebut " Jaringan OLE " , memperluas COM Microsoft , dan menyediakan substrat komunikasi bawah infrastruktur COM + server aplikasi Microsoft .
- Penambahan " D " untuk COM adalah karena penggunaan luas DCE / RPC (Distributed Computing Environment / Remote Procedure Calls) - lebih khusus Microsoft versi yang disempurnakan , yang dikenal sebagai MSRPC

Dalam hal ekstensi ditambahkan ke COM , DCOM harus memecahkan masalah

- Menyusun - serialisasi dan deserialisasi argumen dan mengembalikan nilai-nilai panggilan metode " atas kawat " .
- pengumpulan sampah Distributed - memastikan bahwa referensi yang dimiliki oleh klien antarmuka dilepaskan ketika, misalnya , proses klien jatuh , atau koneksi jaringan hilang .
- Ini harus menggabungkan Ratusan / Puluhan Ribuan benda yang diadakan di browser klien dengan transmisi tunggal untuk meminimalkan penggunaan bandwidth .

SOFTWARE VULNERABILITIES, TESTING, AND ASSURANCE

Setelah proyek ini berlangsung dan perangkat lunak yang telah diprogram, langkah selanjutnya adalah pengujian perangkat lunak, berfokus pada kerahasiaan, integritas, dan ketersediaan sistem, aplikasi, dan data diproses oleh aplikasi. Perhatian khusus harus diberikan pada penemuan kerentanan perangkat lunak yang dapat menyebabkan data atau sistem kompromi. Akhirnya, organisasi harus mampu untuk mengukur efektivitas proses pembuatan perangkat lunak mereka dan mengidentifikasi cara untuk memperbaikinya.

- Software vulnerabilities

Sebuah kerentanan perangkat lunak adalah lubang keamanan, kesalahan, atau kelemahan yang ditemukan dalam perangkat lunak atau sistem operasi (OS) yang dapat menyebabkan masalah keamanan. Sebuah contoh dari cacat perangkat lunak adalah buffer overflow.

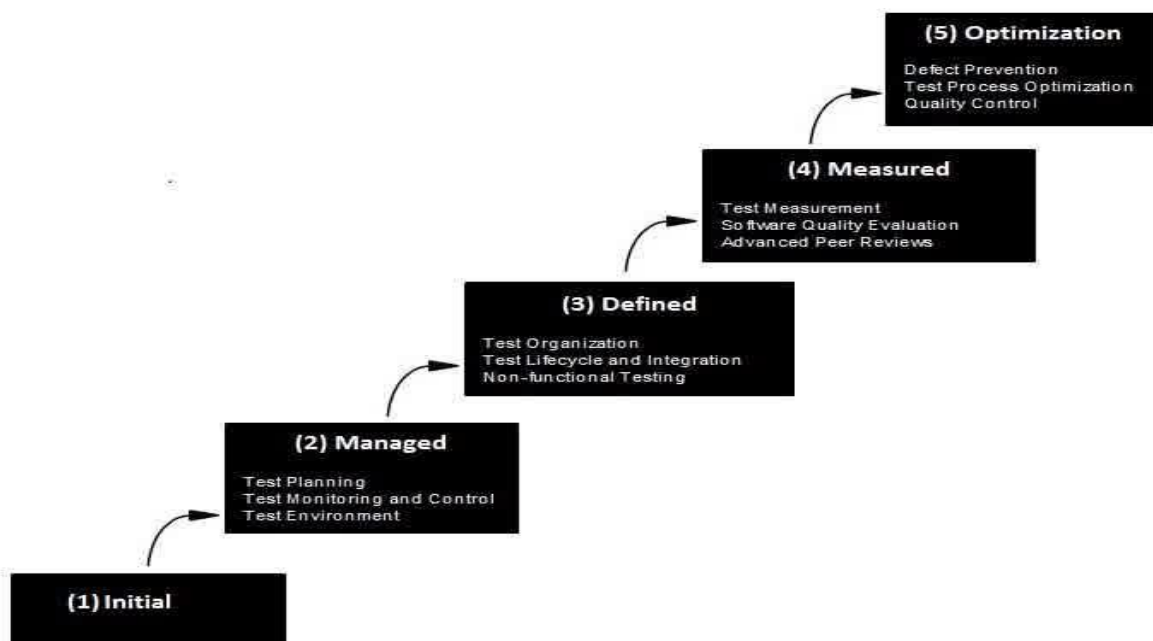
- Types of software vulnerabilities

Kebanyakan kerentanan keamanan perangkat lunak jatuh ke salah satu dari sekelompok kecil kategori:

- a) buffer overflows
- b) masukan unvalidated
- c) masalah akses kontrol
- d) kelemahan dalam otentikasi

- Software Capability Maturity Model

Software Engineering Institute (SEI) Capability Maturity Model (CMM) menentukan serangkaian meningkatnya tingkat organisasi pengembangan perangkat lunak . Semakin tinggi tingkat , semakin baik proses pengembangan perangkat lunak , maka mencapai setiap tingkat adalah proses yang mahal dan memakan waktu .



DATABASES

Database adalah kumpulan informasi yang terorganisasi sehingga dapat dengan mudah diakses , dikelola , dan diperbarui . Dalam satu tampilan , database dapat diklasifikasikan menurut jenis konten : bibliografi , teks lengkap , numerik , dan gambar .

- Relational databases

Adalah satu set kolektif beberapa set data yang diselenggarakan oleh tabel , catatan dan kolom . RDBs membangun hubungan yang jelas antara tabel database . Tabel berkomunikasi dan berbagi informasi , yang memfasilitasi searchability data, organisasi dan pelaporan . RDBs menggunakan Structured Query Language (SQL) , yang merupakan aplikasi pengguna standar yang menyediakan antarmuka pemrograman yang mudah bagi interaksi database . RDB berasal dari konsep fungsi matematika data pemetaan

- Database integrity

Integritas data mengacu menjaga dan menjamin akurasi dan konsistensi data melalui seluruh siklus hidup , dan merupakan aspek penting untuk desain , implementasi dan penggunaan sistem yang menyimpan , proses , atau mengambil data .

- Database replication and shadowing

Bayangan journal , atau database membayangi , memungkinkan komputer sekunder untuk menjaga " bayangan " salinan database yang dipilih sebagaimana yang telah diperbarui pada mesin utama . Dengan terus mentransfer informasi jurnal dari mesin utama untuk mesin sekunder , membayangi memungkinkan pemulihan untuk sistem yang biasanya hanya dalam beberapa transaksi dari database sumber. Anda dapat menggunakan bayangan untuk berbagai tujuan , masing-masing dengan mengatur sendiri pertimbangan penting tergantung pada lingkungan sistem Anda . Beberapa tujuan yang paling umum puas dengan bayangan meliputi berikut ini :

- Pemulihan bencana , penggunaan yang paling umum ; itu adalah sederhana dan murah .
- Baca - hanya melaporkan server dimana tugas pelaporan ad hoc dapat beroperasi pada data saat ini tanpa mempengaruhi produksi .
- replikasi rendah anggaran mana database direplikasi pada contoh bayangan menggunakan journal .
- Gagal atas dalam beberapa keadaan tertentu .