\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

166주차 - 소프트웨어 개발 방법론

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 소프트웨어 개발 방법론 종류

2. 구조적 방법론

1) 정형화된 분석 절차에 따라 요구사항 파악하여 문서화하는 처리 중심 방법론

2) 쉬운 이해 및 검증이 가능한 프로그램 코드 생성하는 것이 목적

3) 복잡한 문제를 다루기 위해 분할과 정복 원리 적용

4) 절차

타당성 검토 -> 계획 -> 요구사항 -> 설계 -> 구현 -> 시험 -> 운용/유지보수

3. 정보공학 방법론

1) 계획,분석,설계,구축에 정형화된 기법을 상호 연광성 있게 통합 적용하는 자료 중심 방법론

2) 대규모 정보 시스템 구축에 적합

3) 데이터베이스 설계를 위한 데이터 모델링으로 개체 관계도(ERD) 사용

4) 절차

계획 -> 분석 -> 설계 -> 구축

4. 객체지향 방법론

1) 개체를 객체로 만들어 조립해서 구현하는 방법

2) 구조적 기법의 문제의 해결책

3) 패키지 다이어그램,배치 다이어그램,상태 전이도

4) 절차

요구분석 -> 설계 -> 구현 -> 테스트 -> 인도

5. 컴포넌트 기반(CBD : Component Based Design) 방법론

1) 컴포넌트를 조합하여 하나의 새로운 어플리케이션 만듦

2) 컴포넌트 재사용, 시간과 노력 절감

3) 기능 추가 간단, 확장성 보장

4) 유지 보수 비용 최소화

5) 절차

개발준비->분석->설계->구현->테스트->전개->인도

6. 애자일 방법론

1) 요구사항에 유연하게 대응하도록 일정한 주기 반복

2) 소규모 프로젝트, 숙달된 개발자, 급변하는 요구사항

3) 종류 : 익스트림 프로그래밍, 스크럼, 칸반, 크리스탈

4) 절차

사용자스토리->계획,개발,테스트

7. 제품 계열 방법론

1) 특정 제품에 적용하고 싶은 공통된 기능 정의 개발

2) 임베디드 소프트웨어(특정 기능에 맞게 특화)

3) 영역공학 : 영역분석,영역설계,핵심자산 구현

응용공학 : 제품요구분석,제품설계,제품 구현

4) 둘의 연계를 위해 요구사항,아키텍처,조립 생산 필요

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

167주차 - S/W 공학의 발전적 추세

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 소프트웨어 재사용의 개요

1) 시간 비용 단축

2) 실패 위험 감소

3) 시스템 구축 방법에 대한 지식 공유

4) 시스템 명세,설계 코드 등 문서 공유

2. 소프트웨어 재사용 방법

1) 합성 중심(Composition-Based) : 블록(모듈)을 조립해 완성시키는 방법, 블록 구성 방법이라고도 함

2) 생성 중심(Generation-Based) : 추상화 형태로 쓰여진 명세를 구체화하여 프로그램 만듦, 패턴 구성 방법이라고도 함

3. 소프트웨어 재공학의 개요

1) 기능 추가하여 소프트웨어 성능 향상

2) 예방해서 위기 해결

3) 수명 연장, 개발 기간 단축

4) 오류 줄고, 비용 절감

5) 주요 활동

- 분석(Analysis) : 명세서 확인해 동작 이해, 재공학할 대상 선정

- 재구성(Restructuring) : 구조 향상을 위한 코드 재구성, 기능과 외적인 동작은 바뀌지 않음

- 역공학(Reverse Engineering) : 일반적인 개발 단계와는 반대로 기본 코드를 복구, 구성요소와 관계를 통해 설계도 추출

- 이식(Migration) : 다른 운영체제나 하드웨어 환경에서 사용 가능하도록 변환

4. CASE(Computer Aided Software Engineering)의 개요

1) 요구분석,설계,구현,검사및디버깅 전체 또는 일부를 자동화하는 것

2) CASE도구를 통해 관리되는 공통 모듈을 사용할 수 있어 재사용성 향상

3) 모듈 관리 자동 수행하므로 유지보수 간편

4) 주요 기능 : 소프트웨어생명주기 전단계 연결, 개발모형지원,그래픽지원,모델의 모순검사및오류검증,자료흐름도 작성

5) 원천 기술 : 구조적 기법,프로토타이핑,자동프로그래밍,정보저장소,분산처리

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

168주차 - 비용 산정 기법 - 상향식

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 상향식 비용 산정 기법의 개요

1) 세부 작업 단위 비용 산정 후 전체 비용 산정

2) LOC 기법, 개발 단계별 인월수 기법, 수학적 산정 기법

2. LOC(원시 코드 라인 수, source Line Of Code) 기법

1) 비관치, 낙관치, 기대치 측정하여 예측치 구함

- 비관치 : 가장 많이 측정된 코드

- 낙관치 : 가장 적게 측정된 코드

- 기대치 : 측정된 모든 코드 라인 수의 평균

2) 측정 쉽고 많이 사용됨

3) 공식

- 예측치 = (비관치 + 낙관치 + 4 × 기대치) / 6

- 노력(인월) = 개발 기간 × 투입 인원

= LOC / 1인당 월평균 생산 코드 라인 수

- 개발 비용 = 노력(인월) × 단위 비용(1인당 월평균 인건비)

- 개발 기간 = 노력(인월) / 투입 인원

- 생산성 = LOC / 노력(인원)

3. 개발 단계별 인월수(Effort Per Task) 기법

1) LOC 기법 보완, 각 기능 구현시키는 데 필요한 노력은 생명 주기 단계별로 산정

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

171주차 - 수학적 산정 기법

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 수학적 산정 기법의 개요

1) 상향식 비용 산정 기법으로, 경험적 추정 모형, 실험적 추정 모형이라고도 함

2) 개발 비용 산정의 자동화가 목표

3) 종류 : COCOMO 모형, Putnam 모형, 기능 점수(FP) 모형

2. COCOMO 모형 개요

1) 보헴이 제안, 원시 프로그램의 규모인 LOC에 의한 비용 산정 기법

2) LOC 예측 후 비용 산정 방정식에 대입

3) 중소규모 소프트웨어 프로젝트 비용 추정에 적합

4) 같은규모라도 성격에 따라 비용 다름

5) 노력(Man-Month)로 나타냄

3. COCOMO 개발 유형

1) 조직형(Organic Mode)

- 중소규모 5만(50KDSI) 라인 이하의 소프트웨어 개발

- 사무처리, 업무, 과학용 응용 소프트웨어 개발에 적합

- 비용 산정 공식

노력 = 2.4 × (KDSI)^1.05

개발 기간 = 2.5 × 노력^0.38

2) 반분리형(Semi-Detached Mode)

- 트랜잭션 처리 시스템이나 운영체제, 디비관리시스템 등 30만(300KDSI) 라인 이하 소프트웨어 개발

- 컴파일러, 인터프리터 같은 유틸리티 개발에 적합

- 비용 산정 공식

노력 = 3.0 × (KDSI)^1.12

개발 기간 = 2.5 × 노력^0.35

3) 내장형(Embedded Mode)

- 초대형 규모 트랜잭션 처리나 운영체제 등 30만(300KDSI)라인 이상 소프트웨어 개발

- 신호기 제어 시스템, 미사일 유도 시스템, 실시간 처리 시스템 등 시스템 프로그램 개발에 적합

- 비용 산정 공식

노력 = 3.6 × (KDSI)^1.20

개발 기간 = 2.5 × 노력^0.32

4. Putnam 모형

1) 소프트웨어 생명 주기 전 과정 동안 사용될 노력 분포 가정 모형

2) 생명주기예측모형

3) 시간에 따른 함수로 표현되는 Rayleigh-Norden 곡선의 노력 분포도를 기초로 함

4) 대형 프로젝트의 노력분포산정에 이용되는 기법

5) 산정 공식

개발 노력 = 원시코드라인수³ / (환경 상수³ × 개발기간⁴)

- 환경 상수 : 2000, 8000, 12000

5. 기능 점수(FP) 모형

1) 알브레히트 제안, 기능 증대 요인별 가중치 부여, 총기능점수 산출

2) 유용성과 간편성으로 최선의 평가를 받고 있음

3) 공식

기능 점수(FP) = 총 기능 점수 × [0.65 + (0.1 × 총 영향도)]

4) 기능별 가중치

단순 보통 복잡

- 자료입력 3 4 6

- 정보출력 4 5 7

- 명령어 3 4 5

- 데이터파일 7 10 15

- 필요한 외부 루틴과의 인터페이스 5 7 10

6. 자동화 추정 도구

1) SLIM : Rayleigh-Norden 곡선과 Putnam 예측 모델을 기초로 개발

2) ESTIMACS : FP모형을 기초로 개발

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

172주차 - 프로젝트 일정 계획

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 개요

1) 소작업 파악, 분배, 순서, 일정 정하는 것

2) 종류 : PERT/CPM, WBS, 간트 차트

2. PERT(Program Evaluation and Review Technique, 프로그램 평가 및 검토 기술)

1) 프로젝트에 필요한 전체 작업의 상호 관계를 표시하는 네트워크

2) 과거 경험이 없어서 소요 기간 예측이 어려운 소프트웨어에서 사용

3) 노드,간선으로 구성되며 원 노드에는 작업을 간선에는 낙관치,기대치,비관치 순서로 작성

4) 결정 경로, 작업에 대한 경계시간 , 작업 간의 상호 관련서

5) 공식

작업 예측치 = (비관치 + 4 × 기대치 + 낙관치) / 6

평방 편차 = [(비관치 - 낙관치) / 6]²

3. CPM(Critical Path Method, 임계 경로 기법)

1) 작업 나열, 소요 기간 예측

2) 노드와 간선으로 구성된 네트워크로 노드는 작업, 간선은 작업 사이의 전후 의존 관계 나타냄

3) 원형 노드 = 작업+소요기간, 박스 노드 = 이정표 + 예상 완료 시간

4) 각 작업 순서,의존관계,동시에 수행될 수 잇는지 한눈에 볼 수 있음

5) 경영층의 과학적 의사 결정 지원

6) 병행 작업 가능하도록 계획 가능, 자원 할당도 가능

7) 임계 경로는 최장 경로를 의미함

8) 일정 계획의 순서

- 프로젝트 규모 측정

- 각 단계에서 필요한 작업 분할

- 상호 의존관계를 CPM네트워크로 나타냄

- 일정 계획을 간트 차트로 나타냄

4. 간트 차트(=시간선 차트)

1) 각 작업이 언제 시작하고 종료되는지 일정을 막대 도표를 이용하여 표시

2) 중간목표미달성 시 이유,기간 예측 가능

3) 문제점이나 초과 지출도 관리 가능

4) 자원 배치 인원 계획에 유용하게 사용

5) CPM네트워크 데이터 바탕

6) 작업 경로 표시할수없음, 변화에 대한 적응성이 약함

7) 계획 수립 또는 수정 때 주관적 수치에 기울어지기 쉽다.

8) 이정표,작업 일정,작업기간,산출물로 구성

9) 수평 막대의 길이는 각 작업의 기간을 나타냄

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

173주차 - 소프트웨어 개발 방법론 결정

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 소프트웨어 개발 방법론 결정의 개요

1) 확정된 소프트웨어 생명 주기와 개발 방법론에 맞춰 개발,활동,작업,절차 등 정의하는 것

2) 프로젝트 관리

- 일정관리

- 비용관리

- 인력관리

- 위험관리

- 품질관리

2. 소프트웨어 개발 방법론 결정 절차

1) 프로젝트 관리와 재사용 현황 반영

2) 개발 단계별 작업 및 절차를 소프트웨어 생명 주기에 맞춰 수립

- 기본 생명 주기,지원 생명 주기, 조직 생명 주기별로 주요 프로세스 확인

- 개발 프로세스,개발생명주기,프로세스 모형 정리

3) 결정된 방법론의 단계별 매뉴얼 작성

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

174주차 - 소프트웨어 개발 표준

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. ISO/IEC 12207

1) 표준 소프트웨어 생명 주기 프로세스, 개발,운영,유지보수 등 체계적으로 관리하기 위한 표준 제공

2) 종류

- 기본 생명 주기 프로세스 : 획득,공급,개발,운영,유지보수

- 지원 생명 주기 프로세스 : 품질보증,검증,확인,활동검토,문서화,형상관리

- 조직 생명 주기 프로세스 : 관리,기반구조,훈련,개선

2. CMMI(Capability Maturity Model Integration, 능력 성숙도 통합 모델)

1) 조직 성숙도 평가 모델

2) 성숙도 종류

단계 프로세스 특징

- 초기 정의된 프로세스 없음 작업자 능력에 따라 성공 여부 결정

- 관리 규칙화된 프로세스 특정한 프로젝트 내의 프로세스 정의 및 수행

- 정의 표준화된 프로세스 조직의 표준 프로세스를 활용하여 업무 수행

- 정량적 관리 예측 가능한 프로세스 프로젝트를 정량적으로 관리 및 통제

- 최적화 지속적 개선 프로세스 프로세스 역량 향상을 위해 지속적인 프로세스 개선

3. SPICE(Software Process Improvement and Capability dEtermination)

1) 정보 시스템 분야에서 소프트웨어 품질 및 생산성 향상을 위해 프로세스 평가 및 개선

2) ISO/IEC 15504

3) 개발 기관 스스로 평가

4) 5개의 범주와 40개의 세부 프로세스로 구성됨

- 고객-공급자 프로세스

- 개발하여 고객에게 전달하는 것 지원

- 구성요소 : 인수,공급,요구도출,운영

- 프로세스 수 : 10개

- 공학 프로세스

- 명세화 구현 유지보수로 구성

- 구성요소 : 개발,유지보수

- 프로세스 수 : 9개

- 지원 프로세스

- 소프트웨어 생명 주기에서 다른 프로세스에 의해 이용되는 프로세스로 구성

- 구성요소 : 문서화,형상,품질보증,검증,확인,리뷰,감사,품질문제해결

- 프로세스 수 : 8개

- 관리 프로세스

- 소프트웨어 생명 주기에서 프로젝트 관리자에 의해 사용되는 프로세스로 구성

- 구성요소 : 관리,프로젝트 관리, 품질 및 위험관리

- 프로세스 수 : 4개

- 조직 프로세스

- 조직의 업무 목적 수립과 조직의 업무목표달성을 위한 프로세스로 구성

- 구성요소 : 조직배치,개선활동프로세스,인력관리,기반관리,측정도구,재사용

- 프로세스 수 : 9개

5) 프로세스 수행 능력 6레벨로 나뉨

- level 0 : 불완전 프로세스 구현 안됨 또는 목적 달성 못함

- level 1 : 수행 프로세스 수행되고 목적 달성

- level 2 : 관리 정의된 자원 한도 내 프로세스가 작업 산출물 인도

- level 3 : 확립 소프트웨어 공학 원칙에 기반해 정의된 프로세스 수행

- level 4 : 예측 프로세스가 목적달성을 위해 통제되고 양적인 측정을 통해 일관되게 수행

- level 5 : 최적화 프로세스 수행을 최적화하고 지속적인 개선을 통해 업무 목적 만족

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

175주차 - 소프트웨어 개발 방법론 테일러링

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 소프트웨어 개발 방법론 테일러링의 개요

1) 프로젝트 상황 및 특성에 맞게 정의된 방법론의 절차 사용기법 등을 수정 및 보완하는 작업

2) 관리적 측면에서 최단기간 안에 프젝 진행을 위해 사전 위험 식별,제거 작업

3) 기술적 측면에서 특성에 맞는 최적화된 기술, 도구 도입

4) 수행절차

프젝 특징 정의 -> 표준 프로세스 선정 및 검증 -> 상위 수준의 커스터마이징 -> 세부 커스터마이징 -> 테일러링 문서화

2. 고려사항

1) 내부적 기준

- 목표 환경 : 시스템 개발 환경과 유형이 다른 경우 테일러링 필요

- 요구사항 : 프젝 생명 주기 활동에서 우서적으로 고려할 요구사항이 다른 경우

- 프로젝트 규모 : 규모가 서로 다른 경우

- 보유 기술 : 프로세스,방법론,산출물,구성원 능력이 다른 경우

2) 외부적 기준

- 법적 제약사항 : IT Compliance 가 다른 경우

- 표준 품질 기준 : 분야별 표준 품질 기준이 다른 경우

3. 기법

1) 프젝 규모와 복잡도에 따른 기법 : 규모,난이도에 따라 상중하로 나눔

2) 프젝 구성원에 따른 기법 : 기술적 숙련도와 방법론의 이해 정도 확인

3) 팀내 방법론 지원에 따른 기법 : 수행 시 각 팀별로 방법론 담당 인력 배정

4) 자동화에 따른 기법 : 작업 부하를 줄이기 위해 중간 단계에서 산출물을 자동화 도구를 이용해 산출

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

176주차 - 소프트웨어 개발 방법론 프레임워크

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 소프트웨어 개발 프레임워크의 개요

1) 공통 구성 요소와 아키텍처를 일반화한 반제품 형태의 소프트웨어 시스템

2) 선행 사업자의 기술에 의존하지 않은 표준화된 개발 기반으로 사업자 종속성이 해소

3) 이미 내장된 클래스 라이브러리로 구현되어 있어 개발자는 이미 존재하는 부분을 확장 및 이용

4) 주요 기능 : 예외처리,트랜잭션처리,메모리공유,데이터소스관리,서비스관리,쿼리서비스,로깅서비스,사용자인증서비스

2. 프레임워크 특성

1) 모듈화 : 캡슐화를 통해 모듈화 강화

2) 재사용성 : 재사용 가능

3) 확장성 : 다형성을 통한 인터페이스 확장 가능

4) 제어의 역흐름 : 객체들의 제어를 프레임워크에 넘김으로써 생산성 향상

3. 스프링 프레임워크

1) 자바 플랫폼, 오픈 소스 경량형 애플리케이션 프레임워크

2) 동적 웹 사이트

3) 전자정부 프레임워크의 기반이 됨

4. 전자정부 프레임워크

1) 공공부문 정보화, 특정 업체의 종속성 배제, 사업별 공통 컴포넌트 중복 개발방지

5. 닷넷 프레임워크(.NET)

1) Windows 개발 및 실행 환경 제공, 마이크로소프트사에서 통합 인터넷 전략을 위해 개발

2) 코드 실행을 관리하는 CLR(Common Language Runtime)이라는 가상머신 상에서 작동

3) 메모리관리, 유형 및 메모리 안정성, 보안, 네트워크 작업 등 여러가지 서비스 제공

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

177주차 - 네트워크 관련 신기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 지능형 초연결망

1) 과학기술정보통신부 주관 4차 산업혁명 시대를 맞아 급격히 증가하는 데이터 트래픽을 효과적 수용하기 위한 사업

2) 국가 전체 망에 소프트웨어 정의 기술(SDE) 적용하는 방법, IoT,클라우드,빅데이터,5G 등 수용

3) 중장기 네트워크 발전 전략

2. 소프트웨어 정의 기술(SDE, SDx : Software-Defined Everything)

1) 자원을 가상화하여 제공하고, 중앙에서는 통합적으로 제어 가능

2) 관련 용어

- 소프트웨어 정의 네트워킹(SDN)

- 네트워크를 컴퓨터처럼 모델링하여 여러 사용자가 각 소프트웨어들로 네트워킹을 가상화하여 제어,관리

- 하드웨어 의존하는 네트워크 체계보다 효율적

- 기존 네트워크에 영향 없이 특정 서비스 전송 경로 수정을 통해 인터넷상 문제 해결

- 소프트웨어 정의 데이터(SDDC)

- 데이터 센터의 모든 자원을 가상화하여 인력 개입 없이 소프트웨어만으로 조작

- 소프트웨어 정의 스토리지(SDS)

- 물리적 데이터 스토리지를 가상화하여 여러 스토리지를 하나처럼, 하나의 스토리지를 여러 스토리지로 나눠 사용하는 기술

3. IoT(Internet of Things, 사물 인터넷)

1) 실세계와 가상세계의 다양한 사물을 인터넷으로 연결

2) 유비쿼터스(장소에 상관없이 네트워크 접속) 공간 구현

3) IoT 주요 길술 : 스마트 센싱, 유무선 통신 및 네트워크 인프라, 사물 인터넷 인터페이스, 사물 인터넷을 통한 서비스

4) IoT 기반 서비스는 개방형 아키텍처를 필요로 해서 정보 보안 기술이 필요

4. IoT 관련 용어

1) M2M(Machine to Machine, 사물 통신)

- 무선 통신을 이용한 기계 간 통신

- 변압기 원격 감시, 전기, 가스 등의 원격 검침, 무선 신용카드 조회기, 위치 추적 시스템 등 무선통합하여 사용

2) 메시 네트워크(Mesh Network)

- 차세대 이동통신,홈네트워킹,공공안전 등 특수 목적을 위한 네트워크

- 수십,수천개의 디바이스를 그물망으로 유기적으로 연결하여 안정적 네트워크 사용

3) 와이선(Wi-SUN)

- 스마트 그리드와 같은 장거리 무선 통신을 필요로 하는 IoT 서비스를 위한 저전력 장거리 통신 기술

- 스마트 그리드 : 에너지 효율성 높이는 지능형 전력망 시스템

- 저전력 장거리 기술(LPWA : Low-Power Wide Area)

- 낮은 지연속도, 메시 네트워크 기반 확장성 등 짧은 시간 동안 데이터 전송이 빈번한 검침 분야에 유용

4) UWB(Ultra WideBand, 초광대역)

- 짧은 거리에서 많은 양 데이터를 낮은 전력으로 전송, 무선 디지털 펄스라고 하고, 블루투스와 비교됨

- 땅속이나 벽 뒤로도 전송 가능

5) 피코넷(PICONET)

- 여러개의 독립된 통신장치가 블루투스 기술이나 UWB 통신 기술을 사용해 통신망 형성

- 주로 수십미터 이내의 좁은 공간에서 네트워크 형성, 정지 또는 이동 중에 있는 장치 모두 포함

6) USN(Ubiquitous Sensor Network, 유비쿼터스 센서 네트워크)

- 각종 센서로 다양한 정보를 무선으로 수집

- 필요한 모든 것에 RFID 태그 부착, 사물 뿐만 아니라 주변 환경정보까지 모든 데이터 관리

7) SON(Self Organizing Network, 자동 구성 네트워크)

- 주변 상황에 맞춰 스스로 망 구성

- 갑작스러운 사용자 증가,감소에 따라 자동으로 주변 셀과 협력해 셀 용량을 변화, 장애 시 자체적 치유도 가능

8) 저전력 블루투스 기술(BLE : Bluetooth Low Energy)

- 일반 블루투스처럼 2.4GHz 대역 사용하지만 연결 안된 대기 상태에서는 절전 모드 유지

- 낮은 전력으로 저용량 데이터 처리하는 시계,장난감,비컨 극소형 사물 인터넷에 적함

9) NFC(Near Field Communicationm, 근거리 무선 통신)

- 고주파(HF)를 이용한 근거리 무선 통신 기술, 아주 가까운 거리에서 양방향 통신 지원 RFID 기술의 일종

- 13.56MHz 이용해 10cm 내에서 최고 424Kbps 속도로 데이터 전송

5. 클라우드 컴퓨팅

1) 각종 컴퓨팅 자원을 중앙 컴퓨터에 두고 인터넷 기능을 가진 단말기로 언제어디서든 작업 수행할 수 있는 환경

2) 중앙 컴퓨터는 복수의 데이터 센터를 가상화 기술로 통합한 대형 데이터 센터, 컴퓨팅 자원 보유

3) 사용자는 키보드,모니터,마우스,통신포트만 갖추면 업무 수행 가능

4) 그리드 컴퓨팅은 수많은 컴퓨터를 하나의 컴퓨터처럼 묶어 분산 처리하는 방식, 기상 예측 등 대규모 연산에 사용

클라우드 컴퓨팅은 중앙의 대형 데이터 자원을 필요한 이들에게 필요한 순간에 빌려주는 방식

5) 관련 용어

- 모바일 클라우드 컴퓨팅(MCC)

- 클라우드 서비스를 이용해 소비자와 소비자파트너가 모바일 기기로 클라우드 컴퓨팅 인프라 구성해 자원 공유

- 인터클라우드 컴퓨팅

- 각기 다른 클라우드 서비스를 연동하거나 컴퓨팅 자원의 동적 할당이 가능하도록 여러 클라우드 서비스나 자원 연결

- 서비스 형태

- 대등 접속(Peering) : 서비스 제공자 간 직접 연계

- 연합(Federation) : 자원 공유를 기본으로 사용 요구량에 따른 동적 자원 할당, 논리적으로 하나의 서비스

- 중개(Intermediary) : 서비스 제공자 간 직간접적 연계 또는 단일 서비스 제공자를 통한 중개 서비스 제공

- 클라우드 기반 HSM(Hardware Security Module)

- 클라우드 기반으로 암호화 키 생성,저장,처리 등 작업 수행하는 보안기기

- 클라우드에 인증서 저장, 개별 기기에 인증서 저장 필요 없음

- 암호화 키 생성이 하드웨어적으로 구현, 소프트웨어적 키 생성이 가지는 보안 취약점 무시

- 파스-타(PaaS-TA)

- 소프트웨어 개발 환경을 제공하기 위해 개발한 개방형 클라우드 컴퓨팅 플랫폼

- 인프라 제어 및 관리 환경, 실행 환경, 개발 환경, 서비스 환경, 운영 환경으로 구성

6. 기타 용어

1) 징(Zing)

- 10cm 이내 2.5Gbps 속도의 데이터 전송 가능한 초고속 근접무선통신(NFC)

- 길거리 광고나 키오스크 접목하여 사용

2) NDN(Named Data Networking)

- 콘텐츠 자체 정보와 라우터 기능만으로 데이터 전송 수행

- 콘텐츠 중심 네트워킹(CCN) 과 같은 개념

- 해시테이블에 기반을 두는 P2P시스템과 같이 콘테츠 정보와 라우터만으로 목적지 확정

3) NGN(Next Generation Network, 차세대 통신망)

- ITU-T에서 개발하고 있는 유선망 기반의 차세대 통신망

- 이동통신에서 제공하는 완전한 이동성과 하나의 망으로 모든 정보와 서비스를 패킷으로 압축하여 전송

4) 올(all)-IP

- 기존 통신망을 모두 IP기반의 망으로 통합한 차세대 네트워크

5) WBAN(Wireless Body Area Network)

- 웨어러블 또는 몸에 심는 형태의 센서나 기기를 무선으로 연결하는 개인 영역 네트워크 기술

6) GIS(Geographic Information System, 지리 정보 시스템)

- 지리적인 자료를 처리할 수 있는 컴퓨터 응용 시스템, 위성을 이용해 모든 사물 위치 정보 제공

7) 애드 혹 네트워크(Ad-hoc Network)

- 재난 현장 같이 별도 고정 유선망 구축 불가일 때 모바일 호스트만을 이용해 구성한 네트워크

- 유선망,기지국 필요없고 호스트 이동에 제약 없어 빠른 망 구성과 저렴한 비용

8) 네트워크 슬라이싱(Network Slicing)

- 5G 핵심 기술 중 하나로 하나의 물리적 네트워크를 여러개의 가상 네트워크로 분리

- 각 네트워크가 고객 맞춤형 서비스 제공

9) 파장 분할 다중화(WDM : Wavelength Division Multiplexing)

- 광섬유를 이용한 통신 기술, 파장이 다른 광선은 서로 간섭일으키지 않음을 이용해 많은 신호를 보내 여러 단말기가 동시에 회선 사용

10) 개방형 링크드 데이터(LOD : Linked Open Data)

- Linked Data와 Open Data의 합성어, 누구나 사용할 수 있도록 웹상에 공개된 연계 데이터

- 웹상 존재 데이터를 개별 URI로 식별, 각 URI에 링크 정보 부여해서 상호 연결된 웹을 지향하는 모형

11) 스마트 그리드

- 정보 기술을 전력에 접목해 효율성을 높인 시스템, 전력 IT라고도 함

- 모든 통신, 정보 관련 애플리케이션 인프라를 하나의 시스템으로 통합 관리하여 에너지 관리 좋음

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

178주차 - 네트워크 구축

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 네트워크 설치 구조

1) 통신망은 정보를 전달하기 위해 통신 규약에 의해 연결한 통신 설비의 집합

2) 장치들의 물리적 위치에 따라 나뉨

2. 성형(Star, 중앙 집중형)

1) 중앙에 중앙 컴퓨터가 있고 이를 중심으로 단말 장치들이 연결된 중앙 집중식 네트워크 구성 형태

2) point-to-point 방식으로 회선 연결

3) 중앙 컴퓨터를 통해 데이터 교환

4) 단말장치 추가,제거 쉬움

5) 단말장치 고장은 영향 없지만 중앙장치 고장은 전체 영향 감

6) 교환 노드의 수가 가장 적다

3. 링형(Ring, 루프형)

1) 컴퓨터와 단말장치를 서로 이웃하는 것끼리 point-to-point 방식으로 연결

2) 분산 및 제어 가능

3) 단말장치 추가,제거 및 기밀보호가 어렵다

4) 단말장치에서 전송 지연 발생 가능

5) 중계기 수가 많아짐

6) 단방향 또는 양방향 전송, 단방향의 경우 컴퓨터,단말,회선 중 하나라도 고장나면 전체 통신망에 영향

4. 버스형(Bus)

1) 한 개의 통신 회선에 여러 대의 단말장치 연결

2) 물리적 구조 간단, 단말장치 추가/제거 쉬움

3) 단말장치 고장나도 전체 영향 없음

4) 기밀 보장 어렵고 통신 회선 길이에 제한 있음

5. 계층형(Tree, 분산형)

1) 분산처리시스템 구성 방식

6. 망형(Mesh)

1) 모든 지점의 컴퓨터와 단말장치 서로 연결, 노드 연결성 높음

2) 많은 단말장치로부터 많은 양의 통신이 필요할 때 유리

3) 통신 회선 총 경로가 가장 길다

4) 총 회선 = n(n-1)/2, 노드당 n-1개 포트 필요

5) 통신 회선 장애 시 다른 경로를 이용해 전송 가능

7. 네트워크 분류

1) 근거리 통신망(LAN : Local Area Network)

- 회사,학교 비교적 가까운 거리에 있는 컴퓨터, 프린터, 등 자원 연결

- 자원 공유 목적

- 전송 속도 빠르고, 에러 발생률 낮음

- 주로 버스형, 링형 사용

2) 광대역 통신망(WAN : Wide Area Network)

- 멀리 떨어진 사이트들을 연결하여 구성

- 통신 속도가 느리고, 에러 발생률 높음

- 가까운 것을 근거리 통신망으로 연결 후 각 근거리 통신망을 연결하는 방식

3) VLAN(Virtual Local Area Network)

- LAN의 물리적 배치와 상관없이 논리적으로 분리하는 기술, 접속된 장비 성능 및 보안성 향상

8. LAN의 표준안

1) IEEE 802 위원회에서 지정한 LAN 표준 규격

- 802.1 : 전체구성, OSI참조모델과 관계, 통신망 관리 등에 대한 규약

- 802.2 : 논리 링크 제어(LLC)계층에 관한 규약

- 802.3 : CSMA/CD방식의 매체 접근 제어 계층에 관한 규약

- 802.4 : 토큰 버스 방식의 매체 접근 제어 계층에 관한 규약

- 802.5 : 토큰 링 방식의 매체 접근 제어 계층에 관한 규약

- 802.6 : 도시형 통신망(MAN)에 관한 규약

- 802.9 : 종합 음성/데이터 네트워크에 관한 규약

- 802.11 : 무선 LAN에 관한 규약

2) 802.11의 버전

- 802.11 : 초기버전, 2.4GHz 대역 전파와 CSMA/CA기술을 사용해 최고 2Mbps까지 속도 지원

- 802.11a : 5GHz 대역의 전파와 OFDM기술을 사용해 최고 54Mbps까지 속도 지원

- 802.11b : 초기 번전 개선안, 최고 11Mbps 속도로 기존의 5배 이상 빨라짐

- 802.11e : 802.11의 부가 기능 표준,QoS 기능이 지원되도록 매체 접근 제어(MAC)계층에 해당하는 부분 수정

- 802.11g : 2.4GHz 대역의 전파를 사용하지만 5GHz 대역의 전파 사용하는 802.11a와 동일한 최고 54Mbps까지 속도 지원

- 802.11i : 802.11의 보안 기능 표준, 인증방식에 WPA/WPA2 사용(와이파이에서 제정한 무선랜 인증 및 암호화 관련 표준)

- 802.11n : 2.4GHz 대역과 5GHz 대역을 사용하는 규격, 최고 600Mbps까지 속도 지원

9. CSMA/CA

1) 매체가 비어있음을 확인한 뒤 충돌을 피하기 위해 일정 기간 기다린 후 데이터 전송

2) 회선을 사용하지 않을 때도 확인 신호 전송하여 동시 전송에 의한 충돌 방지

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

180주차 - 경로 제어/트래픽 제어

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 경로 제어(Routing)의 개요

1) 송수신 측 간의 전송 경로 중에서 최적 패킷 교환 경로 결정

2) 어느 한 경로에 데이터가 집중되는 것을 피하고 최저 비용으로 최단 시간에 송신할 수 있는 경로

3) 경로 제어표(Routing Table) 를 참조해서 이루어지고 라우터가 수행함

4) 경로 제어 요소 : 성능 기준, 경로의 결정 시간과 장소, 정보 발생지, 경로 정보의 갱신 시간

2. 경로 제어 프로토콜(Routing Protocol)

1) IGP(Interior Gateway Protocol, 내부 게이트웨이 프로토콜)

- 하나의 자율 시스템(AS) 내의 라우팅에 사용

- 자율 시스템 : 하나의 도메인에 속한 라우터 집합

- RIP(Routing Information Protocol)

- 현재 가장 널리 사용되는 프로토콜, 벡터 라우팅 프로토콜이라 불림

- 최단 경로 탐색에 벨만-포드 알고리즘 사용(가중치가 음수여도 사용 가능)

- 소규모 동종 네트워크 내 효율적

- 최대 홉 수를 15로 제한이므로 대규모 네트워크에서는 RIP 사용 불가

- 라우팅 정보를 30초마다 네트워크 내 모든 라우터에 알리고, 180초 이내에 새 라우팅 정보가 수신 않으면 해당경로를 이상상태로 간주

- OSPF(Open Shortest First Protocol)

- RIP단점 해결, 대규모 네트워크에서 사용

- 라우팅 정보에 노드 간 거리, 링크 상태 실시간으로 반영

- 최단 경로 탐색에 다익스트라 알고리즘 사용

- 라우팅 정보 변화 생기면 변화된 정보만 네트워크 내 모든 라우터에 알림

- 하나의 자율 시스템 내에서 동작하면서 내부 라우팅 프로토콜의 그룹에 도달한다

2) EGP(Exterior Gateway Protocol, 외부 게이트웨이 프로토콜)

- 자율 시스템 간의 라우팅, 게이트웨이 간의 라우팅에 사용

3) BGP(Border Gateway Protocol)

- 자율 시스템 간의 라우팅, EGP 단점 보완

- 초기에 BGP 라우터들이 연결될 때 전체 경로 제어표(라우팅 테이블) 교환, 이후에는 변화된 정보만 교환

3. 트래픽 제어(Traffic Control)의 개요

1) 네트워크 보호, 성능 유지, 네트워크 자원의 효율적인 이용

2) 흐름제어,혼합제어,교착상태 방지 기법이 있음

4. 흐름 제어(Flow Control)

1) 송수신 측 사이에 전송되는 패킷의 양이나 속도 규제

2) 송수신 측 처리 속도 차이나 버퍼 크기 차이에 의해 생길 수 있는 버퍼 오버플로 방지 기능

3) 정지-대기(Stop-and-Wait)

- 수신 측 확인 신호(ACK) 받은 후 다음 패킷 전송

- 한번에 하나의 패킷 전송 가능

4) 슬라이딩 윈도우(Sliding Window)

- 확인 신호를 이용해 송신 데이터 양 조절

- 확인 신호 받지 않아도 미리 정해지 패킷 수만큼 전송

- 한번에 여러개 패킷 전송 가능

- ACK없이 보낼 수 있는 패킷의 최대치를 미리 약속 받는데 최대치가 윈도우 크기를 의미함

- 윈도우 크기는 상황에 따라 변함, ACK일 경우 윈도우 크기 증가, NAK일 경우 윈도우 크기 감소

5. 폭주(혼합)제어(Congestion Control)

1) 네트워크 내 패킷 수 조절, 네트워크 오버플로 방지

2) 느린 시작(Slow Start)

- 윈도우 크기를 1,2,4,8 과 같이 2배씩 지속적 증가 초기엔 느리지만 갈수록 빨라짐

- 전송 데이터 크기가 임계 값에 도달하면 혼잡 회피 단계로 넘어감

3) 혼잡 회피(Congestion Avoidance)

- 임계 값에 도달하면 혼잡으로 간주하고 윈도우 크기를 1씩 선형적으로 증가시켜 혼잡 예방

6. 교착상태(Dead Lock) 방지

1) 교환기 내 패킷 축적하는 기억 공간이 꽉 차 있을 때 다음 패킷이 무한정 기다리는 현상

2) 패킷이 같은 목적지를 갖지 않도록 할당하고, 교착상태 시 그 단말장치를 선택해 패킷 버퍼 폐기함

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

181주차 - SW관련 신기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. SW 관련 용어

1) 인공지능

2) 뉴럴링크

- 뇌와 컴퓨터 결합하는 기술, 일론머스크

- 작은 전극을 뇌에 이식해 생각 업로드, 다운로드

3) 딥 러닝

- 두뇌를 모델로 만들어진 인공 신경망(ANN : Artificial Neural Network) 기반 기계학습기술

- 스스로 필요한 데이터 수집분석

4) 전문가 시스템

- 의료 진단 같은 전문가가 수행하는 고도의 업무 지원 위한 컴퓨터 응용 프로그램

- 지식 베이스 기초하여 추론 기구를 활용해 결정, 해결함

5) 가상현실

- 실제와 유사하지만 실제가 아닌 환경

6) 증강현실

- 실제 촬영한 화면에 가상 정보 부가

7) 혼합현실

- 가상현실과 현실세계를 합쳐 현실의 물리적인 객체와 가상 객체가 상호작용

8) 그레이웨어

- 사용자 입장에서는 악의적일수도있고 유용할수도있음

- 애드웨어(소프트웨어 자체에 광고 포함),트랙웨어(동의없이 사용자 정보 수집,스파이웨어),기타 악성 코드 등

9) 매시업

- 웹에서 제공하는 정보 및 서비스를 이용해 새 소프트웨어, 서비스, 디비를 만드는 기술

10) 리치 인터넷 애플리케이션(RIA)

- 플래시 애니메이션 기술과 웹 서버 애플리케이션 기술을 통합

- 기존 HTML보다 역동적인 웹페이지 제공하는 플래시웹페이지 제작 기술

11) 시맨틱 웹

- 컴퓨터가 사람을 대신해 정보 읽고 이해,가공,만들어내는 차세대 지능형 웹

- 웹 자원을 서술하는 기술, 온톨로지를 통한 지식 서술 기술, 통합 운영을 위한 에이전트 기술

12) 증발품(Vaporware)

- 판매 계획 또는 배포 계획은 발표되었으나 실제로 판매나 배포되지 않은 소프트웨어

13) 오픈 그리드 서비스 아키텍처(OGSA : Open Gride Service Architecture)

- 앱 공유를 위한 웹 서비스를 그리드 상에서 제공하기 위한 개방형 표준

- 웹 서비스 표준을 적극적으로 따름, 웹 개발 툴 그대로 사용

14) 서비스 지향 아키텍처(SOA : Service Oriented Architecture)

- 기업의 소프트웨어 인프라인 정보시스템을 공유와 재사용이 가능한 서비스 단위나 컴포넌트 중심으로 구축하는 정보기술 아키텍처

- 비즈니스에 맞춰 유용하게 사용

- SOA 기반 애플리케이션 구성 계층

- 표현, 업무 프로세스, 서비스 중간, 애플리케이션, 데이터 저장 계층

15) 서비스형 소프트웨어(SaaS : Software as a Service)

- 여러 기능 중 필요로 하는 서비스만 이용

- 하나의 플랫폼을 이용해 다수의 고객에게 서비스 제공, 사용자는 이용한 만큼 돈 지불

16) 복잡 이벤트 처리(CEP : Complex Event Processing)

- 실시간 많은 사건 중 의미가 있는 것만 추출하는 데이터 처리 방법

- 금융,통신,전력 등 대용량 데이터 스트림에 대한 요구에 실시간으로 대응

17) 디지털 트윈

- 현실사물을 가상화한 모델, 정확한 정보,최적화, 모든 과정의 효율성 향상

- 모의 실험하는 용도

18) 텐서플로

- 구글 브레인 팀이 만든 다양한 작업에 대해 데이터 흐름 프로그래밍을 위한 오픈소스 소프트웨어 라이브러리

- C++로 제작, 구글 검색,음성 인식, 번역 등 사용

19) 도커

- 컨테이너(패키지) 기술 자동화하여 쉽게 사용하는 오픈소스 프로젝트

- 컨테이너 안에 응요 프로그램을 배치시키는 일을 자동화해주는 역할

20) 스크래피

- 파이썬 기반 웹 크롤링 프레임워크, 코드 재사용성,대규모 크롤링 프로젝트

- 웹 크롤링 : 다양한 정보 자원 수집하여 분류 및 저장

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

182주차 - 보안 관련 신기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 보안 관련 용어

1) 블록체인(Blockchain)

- P2P네트워크를 이용해 온라인 금융거래정보를 온라인 네트워크 참여자의 디지털 장비에 분산 저장

2) 분장 원장 기술(DLT : Distributed Ledger Technology)

- 중앙 관리자나 중앙 데이터 저장소 존재하지 않고 P2P망내 참여자들에게 모두 거래 목록이 분산 저장

- 거래 발생 시 지속적으로 갱신되는 디지털 원장

- 대표적인 사례로 블록체인이 있다.

3) 양자 암호키 분배(QKD : Quantum Key Distribution)

- 양자 통신을 위해 비밀키 분배하여 관리하는 기술

- 두 시스템이 암호 알고리즘 동작을 위한 비밀키를 안전하게 공유하기 위해 양자 암호키 분배 시스템 설치

4) 프라이버시 강화 기술(PET : Privacy Enhancing Technology)

- 개인정보 위험 관리 기술

- 암호화 익명화 등 개인정보 보호 기술에서 사용자가 직접 개인정보 통제하는 기술까지 사용자 프라이버시 보호 기술

5) 공통 평가 기준(CC : Common Criteria)

- ISO 15408 표준, 정보 보호 제품의 평가 기준

- 시스템 평가 원칙과 평가 모델, 시스템 보안 기능 요구사항(11개), 시스템 7등급 평가를 위한 보증 요구사항(8개)

6) 개인정보 영향평가 제도(PIA : Privacy Impact Assessment)

- 새로운 시스템 도입 및 중요한 변경 사 국민의 사생활에 미칠 영향에 대해 미리 조사,분석,평가하는 제도

7) 소프트웨어 영향평가 제도(임치, Software Escrow)

- 개발자의 지식재산권 보호, 안정적으로 사용 및 유지보수할 수 있도록 제 3의 기관에 소스프로그램,기술 등 보관

8) 서비스형 블록체인(BaaS : Blockchain as a Service)

- 블록체인 앱 개발 환경을 클라우드 기반으로 제공

- 블록체인 네트워크에 노드 추가,제거 쉽고, 플랫폼마다 다른 기술을 보다 편리하게 사용

9) OWASP(the Open Web Application Security Project, 오픈 웹 애플리케이션 보안 프로젝트)

- 웹 정보 노출, 악성 코드, 보안이 취약한 부분 연구하는 비영리 단체

- 보안에 미치는 영향이 큰 것을 기준으로 10 대 웹 애플리케이션 취약점을 3~4년마다 발표

10) TCP 래퍼(TCP Wrapper)

- 외부 컴퓨터의 접속 인가 여부를 점검하여 허용 및 거부

- 네트워크에 접속 시 로그인한 다른 컴퓨터 사용자 ID 로그 조회하여 악용 가능한데 이를 방지하기 위한 방화벽 역할 수행

11) 허니팟(Hoenypot)

- 비정상적인 접근을 탐지하기 위한 시스템

- 침입자 속여 공격당하는 것처럼 보여줌으로써 추적 및 공격기법에 대한 정보 수집

12) DPI(Deep Packet Inspection)

- OSI 7 Layer 전 계층의 프로토콜과 패킷 내부의 콘텐츠를 파악하여 침입 시도, 해킹 등을 탐지하고, 트래픽 조정을 위한 패킷분석기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

185주차 - HW 관련 신기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 고가용성 솔루션(HACMP : High Availability Clustering Muti Processing)

1) 긴 시간동안 안정적인 서비스 운영을 위해 장애 발생 시 즉시 다른 시스템으로 대체 가능한 환경을 구축하는 매커니즘

2) 각 시스템 간에 공유 디스크를 중심으로 클러스터링으로 엮어 시스템끼리 동시에 연결

3) 2개 서버 연결하는 이중화를 통해 서버 안정성 높임

2. 3D Printing(Three Dimension Printing)

1) 대상을 손으로 만질 수 있는 실체 물체로 만들어 내는것

2) 아주 얇은 두께로 한층씩 적층

3. 4D Printing(Fourth Dimension Printing)

1) 특정 시간이나 환경 조건이 갖추어지면 스스로 형태를 변화시키거나 제조되는 자가 조립 기술이 적용된 제품을 3D Printing함

2) TED 강연에서 스카일러 티빗츠 교수가 공개

3) 스마트 소재가 필요하며, 이는 형상기억합금이나 나노 기술을 통해 전기회로를 내장하는 방법 등으로 제조

4. RAID(Redundant Array of Inexpensive(Independent) Disk)

1) 여러개의 하드디스크들로 디스크 배열을 구성하여 파일을 구성하고 있는 데이터 블록을 서로 다른 디스크에 분산 저장할 경우

그 블록들을 여러 디스크에서 동시에 읽거나 쓸 수 있으므로 디스크 속도 매우 향상됨

2) 디스크 하나만 손상되도 전체 데이터 파일이 손상됨, 복구할 수 있도록 여분의 디스크 추가하여 방지

3) 오류 검출 및 정정 방법에 따라 RAID1~5, 5개 있음

5. 4K 해상도

1) 차세대 고화질 모니터의 해상도

2) 3840 \* 2160 픽셀, FULL HDTV(1920\*1080)의 총 4배에 해당

3) UHDTV는 차세대 TV 규격, HDTV의 4배에 해당하는 4K, 16배에 해당하는 8K 해상도

6. 앤 스크린(N-Screen)

1) N개의 서로 다른 단말기에서 동일한 콘텐츠를 자유롭게 이요

2) 콘텐츠를 끊김 없이 이용할 수 있어야 한다.

7. 컴패니언 스크린(Companion Screen)

1) 세컨드 스크린이라고도 불림

2) 방송을 보면서 채팅치기, 방송 내용 공유하며 추가적인 기능 등을 수행할 수 있는 스마트폰, PC등을 의미

8. 신 클라이언트 PC(Thin Client PC)

1) 하드디스크나 주변장치 없이 기본적인 메모리만 갖추고 서버와 네트워크로 운용되는 개인용 컴퓨터, 서버 기반 컴퓨팅과 관계 깊음

2) 클라이언트는 필요할때마다 서버에 접속하여 내려받으면 되고, 기억장치가 없으므로 데이터는 서버 측에서 한꺼번에 관리

3) PC 분실해도 정보 유출될 우려 없음

9. 패블릿(Phablet)

1) 폰과 태블릿 합성어, 태블릿 기능을 포함한 5인치 이상 대화면 스마트폰

2) 한 번 사용하면 작은 기기를 사용할 수 없다는 톱니 효과가 적용될 수 있음

10. C형 유에스비(Universal Serial Bus Type-C, USB Type-C, USB-C)

1) USB IF에서 발표, A형에 비해 크기가 작고, 24핀으로 위아래 구분없음

2) 초당 10기가비트, 전력은 최대 100W까지 전송 가능

11. 멤스(MEMS : Micro-Electro Mechanical Systems)

1) 초정밀 반도체 제조 기술 바탕 센서, 액추에이터 등 기계 구조를 미세 가공해 전기기계적 동작을 할 수 있도록 한 초미세 장치

12. 트러스트존 기술(TrustZone Technology)

1) ARM에서 개발, 하나의 프로세서 내에 일반 애플리케이션 처리하는 일반구역,보안이 필요한 애플리케이션 처리하는 보안구역으로 분할

2) 하드웨어 기반 보안 기술

3) 운영체제 수준에서 안전하게 보호

13. 엠디스크(M-DISC, Millennial DISC)

1) 한 번의 기록만으로 자료를 영구 보관할 수 있는 광 저장 장치

2) 디스크 표면의 무기물층에 레이저를 이용해 자료를 조각해 기록

3) 외부 요인에 영향을 받지 않음

4) DVD, 블루레이디스크에 적용

14. 멤리스터(Memristor)

1) 메모리와 레지스터의 합성어, 전류의 방향과 양 등 기존의 경험을 모두 기억하는 특별 소자

2) 네 번째 전자회로 구성 요소

3) 전원 공급이 끊어져도 직전에 통과한 전류의 방향과 양을 기억, 다시 전원 공급되면 기존 상태 그대로 복원

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

186주차 - Secure OS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Secure OS 개요

1) 기존 OS에 내재된 보안 취약점을 해소하기 위해 보안 기능을 갖춘 커널을 이식해 외부로부터 시스템 자원 보호하는 OS

2) 보안 커널은 TCB 기반으로 참조 모니터의 개념을 구현하고 집행

3) 커널의 보호 대상 : 메모리,보조기억장치,데이터,하드웨어장치,자료구조,명령어,보호매커니즘

4) 보호 방법 종류(복잡한 순서로 위부터 아래로)

- 암호적 분리 : 내부 정보 암호화

- 논리적 분리 : 프로세스의 논리적 구역 지정, 구역 벗어나는 행위 제한

- 시간적 분리 : 동일 시간에 하나의 프로세스만 수행되도록 제한

- 물리적 분리 : 사용자별로 특정 장비만 사용하도록 제한

2. 참조 모니터(Reference Monitor)

1) 보호대상의 객체에 대한 접근통제를 수행하는 추상머신, 이걸 실제로 구현한 게 보안 커널임

2) 보안 커널 데이터베이스(SKDB)를 참조해 객체에 대한 접근 허가 여부를 결정함

3) 참조 모니터와 보안 커널 특징

- 격리성 : 부정 조작 불가능

- 검증가능성 : 적절히 구현된걸 확인 가능

- 완전성 : 우회 불가능

3. Secure OS의 보안 기능

1) 식별 및 인증

2) 임의적 접근 통제(DAC)

3) 강제적 접근 통제(MAC)

4) 객체 재사용 보호

5) 완전한 조정

6) 신뢰 경로

7) 감사 및 감사기록 축소

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

187주차 - DB 관련 신기술

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 빅데이터

1) 막대한 양의 정형 또는 비정형 데이터 집합

2. 브로드 데이터

1) 다양한 채널에서 소비자와 상호 작용을 통해 생성된 데이터, 이전에 없던 새로운데이터 또는 기존 데이터에 새로운 가치가 더해진 데이터

2) 다양한 정보를 의미함

3. 메타 데이터

1) 일련의 데이터를 정의하고 설명해주는 데이터

4. 디지털 아카이빙(Digital Archiving)

1) 디지털 정보 자원을 장기적으로 보존하기 위한 작업

2) 아날로그는 디지털로 변환 후 압축, 디지털은 체계적 분류 후 메타 데이터를 만들어 DB화하는 작업

3) 늘어나는 정보 자원을 효율적으로 관리

5. 하둡

1) 오픈 소스 기반 분산 컴퓨팅 플랫폼

2) 일반 PC급 컴퓨터로 가상화된 대형 스토리지 형성하고 그 안에 보관된 거대한 데이터 세트를 병렬로 처리할 수

있도록 개발된 자바 소프트웨어 프레임워크, 구글,야후에서 사용

3) 하둡과 관계형 디비 간 대용량 데이터 전송할 때 스쿱 사용

6. 맵리듀스

1) 대용량 데이터를 분산 처리하기 위한 목적

2) 연관성 있는 데이터 분류로 묶는 Map작업 후 중복 데이터 제거, 원하는 데이터 추출하는 Reduce 작업 수행

3) 구글, 대푠적인 대용량 데이터 처리를 위한 병렬 처리 기법

7. 타조

1) 하둡 기반의 분산 데이터 웨어하우스 프로젝트로 우리나라가 주도 개발

2) 빅데이터를 분석할 때 맵리듀스 사용하지 않고, SQL 사용하여 하둡파일 바로 읽어냄

3) 대규모 데이터처리와 실시간 상호 분석에 모두 사용

8. 데이터 다이어트

1) 데이터를 삭제하는 것이 아닌 압축하고 중복배제하고 새로운 기준에 다라 나누어 저장하는 작업

9. 데이터 마이닝

1) 데이터 웨어하우스에 저장된 데이터 집합에서 사용자의 요구에 따라 유용하고 가능성 있는 정보를 발견하기 위한 기법

2) 대량 데이터 분석, 상호관계 규명하여 패턴화함

10. OLAP(Online Analytical Processing)

1) 다차원으로 이루어진 데이터로부터 통계적 요약 정보를 분석하여 의사결정에 활용하는 방식

2) 데이터 웨어하우스나 데이터 마트와 같은 시스템과 상호 연관되는 정보 시스템

3) OLAP 연산 : Roll-up,Drill-down,Drill-through,Drill-across,Pivoting,Slicing,Dicing

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

188주차 - 회복/병행제어

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 회복(Recovery)

1) 장애의 유형

- 트랜잭션 장애 : 트랜잭션 내 비정상적인 상황으로 실행 중지되는 현상

- 시스템 장애 : 데이터베이스에 손상을 입지 않으나 하드웨어 오동작,교착상태 등 모든 트랜잭션의 연속적인 수행에 장애를 주는 현상

- 미디어 장애 : 저장장치 디스크 블록의 손상, 디스크 헤드 충돌등 일부나 전부가 물리적으로 손상된 상태

2) 회복 관리기

- DBMS 구성 요소

- 모든 변화를 Undo 시키고 수행 이전 상태로 복구

- 메모리 덤프(주기적으로 디비 전체 복사), 로그(갱신 전후 기록,저널) 이용

2. 회복 기법

1) 연기 갱신 기법(Deferred Update)

- 트랜잭션 성공 완료될 때까지 실질적인 갱신 연기

- 갱신 내용 일단 log에 보관

- 부분 완료 시점에 log 내용 실제 디비에 기록

- 부분 완료되기 전에 장애 발생하여 트랜잭션 롤백되면 영향 미치지 않았기 때문에 Undo할 필요 없다.

- Redo(재실행)만 하면 됨

2) 즉각 갱신 기법(Immediate Update)

- 데이터 갱신하면 바로 실제 디비에 반영

- 갱신 내용 일단 log에 보관

- Redo/Undo 둘 다 가능

3) 그림자 페이지 대체 기법(Shadow Paging)

- 갱신 이전 디비를 일정 크기의 페이지 단위로 구성하여 각 페이지마다 복사본인 그림자 페이지로 별도 보관

- 롤백시킬 때 그림자 페이지를 대체하여 회복

- 로그,undo,redo 필요없음

4) 검사점 기법(Check Point)

- 특정 단계에서 재실행할 수 있도록 정보와 함께 검사점을 로그에 보관

- 장애 시 검사점부터 회복 작업

3. 병행 제어(Concurrency Control)

1) 트랜잭션 병행 수행 할 때 일관성 파괴하지 않도록 제어

2) 목적

- 디비 공유 최대화

- 시스템 활용도 최대화

- 디비 일관성 유지

- 사용자에 대한 응답 시간 최소화

4. 병행제어 기법의 종류

1) 로킹

- 주요 데이터의 액세스를 상호 배타적

- 액세스하기 전에 락 요청

2) 타임 스탬프 순서

- 직렬성 처리 순서 미리 선택

- 실행 전 시간표를 부여하여 시간에 따라 작업 수행

- 교착상태 발생하지 않음

3) 최적 병행수행(검증 기법,확인 기법,낙관적 기법)

- 판독전용(Read Only) 일 경우 충돌률이 매우 낮아서 병행제어 기법 사용하지 않고 실행

4) 다중 버전 기법

- 타임 스탬프 개념 이용, 다중 버전 타임 스탬프 기법이라고도 함

- 갱신될 때마다 버전을 부여하여 관리

5. 병행수행의 문제점

1) 갱신 분실(Lost Update)

- 두 개 이상 트랜잭션이 같은 자료 공유할 때 갱신 결과 일부가 없어지는 현상

2) 비완료 의존성(Uncommitted Dependency)

- 하나의 트랜잭션이 실패 후 회복되기 전에 다른 트랜잭션이 참조하는 현상, 임시 갱신이라고도 함

3) 모순성(Inconsistency)

- 두 개의 트랜잭션이 병행수행될 때 원치 않는 자료를 이용할 때, 불일치 분석이라고도 함

4) 연쇄 복귀(Cascading Rollback)

- 병행 수행되던 트랜잭션이 하나가 롤백될 때 다른 트랜잭션도 같이 롤백되는 현상

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

189주차 - 교착상태

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 교착상태의 개요(Dead Lock)

1) 상호 배제에 의해 생긴 문제, 자원을 무한정 기다리는 현상

2. 교착상태 발생의 필요 충분 조건

1) 상호 배제(Mutual Exclusion) : 한번에 한개 프로세스만 공유 자원 이용

2) 점유와 대기(Hold and Wait) : 최소한 하나 자원 점유하면서 사용되고 있는 다른 자원을 추가로 점유하기 위해 대기하는 프로세스가 있어야한다.

3) 비선점(Non-preemption) : 다른 프로세스에 할당된 자원은 사용 끝날 때까지 강제로 빼앗을 수 없다

4) 환형 대기(Circular Wait) : 대기 프로세스가 원형으로 구성되어 할당된 자원을 점유하면서 앞,뒤 프로세스의 자원을 요구해야8 함

3. 교착상태의 해결 방법

1) 예방 기법 : 위 4가지 조건 중 어느 하나를 제거, 자원 낭비 가장 심함

2) 회피 기법 : 교착상태 가능성 배제하지 않고 피함, 은행원 알고리즘 사용, 다익스트라가 제안한 것

3) 발견 기법 : 교착상태 발생했는지 점검하여 프로세스와 자원 발견하는 것, 교착상태 발견 알고리즘과 자원 할당 그래프 이용

4) 회복 기법 : 프로세스 종료하거나 교착상태의 프로세스에 할당된 자원을 선점하는 것

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

191주차 - Secure SDLC

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. Secure SDLC의 개요

1) SDLC(소프트웨어 개발 생명 주기)에 보안 강화를 위한 프로세스를 포함한 것

2) 유지보수 단계에서 보안 이슈를 해결하기 위해 소모되는 많은 비용을 최소화하기 위해 등장

3) 전체 단계에 걸처 수행되어야 할 보안 활동 제시

4) Secure SDLC의 대표적인 방법론

- CLASP

- Secure Software 사에서 개발, SDLC 초기 단계 보안 강화

- 활동 중심,역할 기반의 프로세스로 구성, 현재 운용 중인 시스템에 적용하기 적합

- SDL

- 마이크로소프트 사에서 기존 SDLC 개선한 방법론

- 전통적인 나선형 모델 기반

- Seven Touchpoints

- 소프트웨어 보안의 모범사례를 SDLC에 통합

- 모든 산출물에 위험분석 및 테스트 수행

- SDLC 각 단계에 관련된 7개 보안 강화 활동 수행

2. 요구사항 분석 단계에서의 보안 활동

1) 보안 요소별로 등급을 구분하여 분류

2) 보안 요소

- 기밀성 : 허가된 사용자만 접근 가능

- 무결성 : 허가된 사용자만 수정 가능

- 가용성 : 허가된 사용자는 언제든 사용 가능

- 인증 : 합법적인 사용자인지 확인하는 과정

- 부인 방지 : 데이터 송수신한 자가 이 사실을 부인할 수 없도록 송수신 증거 제공

3. 설계 단계에서의 보안 활동

1) 식별된 보안 요구사항들을 소프트웨어 설계서에 반영, 보안 설계서 작성

- 네트워크 : 네트워크를 분리하거나 방화벽 설치

- 서버 : 보안이 뛰어난 운영체제 사용,보안 업데이트,접근 통제

- 물리적 보안 : 출입통제, 개발 공간 제한, 폐쇄회로 등 감시설비 설치

- 개발 프로그램 : 허가되지 않은 프로그램 통제, 무결성 검사 실시

4. 구현 단계에서의 보안 활동

1) 표준 코딩 정의서 및 소프트웨어 개발 보안 가이드 준수, 설계서에 따라 보안 요구사항 구현

2) 개발 과정 중 지속적인 단위 테스트

3) 시큐어 코딩(Secure Coding)

- 구현단계에서 발생할 수 있는 보안 취약점들을 최소화하기 위해 보안 요소를 고려해 코딩하는 것

- 안정성,신뢰성

- 보안 정책을 바탕으로 시큐어 코딩 가이드 작성, 개발 참여자들에게 시큐어 코딩 교육 실시

5. 테스트 단계에서의 보안 활동

1) 보안 사항들이 동작되는지 점검

2) 동적 분석 도구 또는 모의 침투테스트를 통해 검증

3) 테스트 계획 수립, 모든 결과는 문서화, 개발자에게 피드백

6. 유지보수 단계에서의 보안 활동

1) 보안사고 식별하고 보안 패치를 실시

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

192주차 - 세션 통제

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 세션 통제의 개요

1) 서버와 클라이언트의 연결을 의미함, 그 때의 정보 관리

2) 요구사항 분석 및 설계 단계에서 진단해야 하는 보안 점검 내용

3) 불충분한 세션 관리, 잘못된 세션에 의한 정보 노출이 약점

2. 불충분한 세션 관리

1) 일정한 규칙이 존재하는 세션ID가 발급되거나 타임 아웃이 너무 길게 설정되어 있는 경우 발생

2) 세션 하이재킹과 같은 공격을 통해 획득한 세션ID로 인가되지 않은 시스템 조작 가능

- 세션 하이재킹

- 서버에 접속한 클라이언트의 세션 정보 가로채는 공격 기법

- 정상적인 연결을 RST(Reset)을 통해 종료시킨 후 재연결 시 공격자에게 연결하는 기법

- 공격자는 서버와 상호 간의 동기화된 시퀀스 번호를 이용해 인가되지 않은 시스템에 접근할 수 있다.

- 탐지 방법 : 비동기화 상태 탐지,ACK Strom(패킷량 비정상적 증가) 탐지,패킷의 유실 탐지, 예상치 못한 접속의 리셋 탐지

3. 잘못된 세션에 의한 정보 노출

1) 다중 스레드 환경에서 멤버 변수에 정보를 저장할 때 발생하는 보안 약점

2) 싱글톤 패턴에서 발생하는 레이스컨디션으로 인해 동기화 오류가 발생하거나 멤버 변수의 정보가 노출될 수 있다.

- 싱글톤 : 하나의 객체를 생성하면 어디서든 참조할 수 있지만, 여러 프로세스가 동시에 참조할 수 없다.

- 레이스컨디션 : 두 개 이상의 프로세스가 공용 자원을 획득하려고 경쟁하는 상태

3) 멤버변수보다 지역 변수를 활용하여 변수 범위를 제한함으로써 방지

4. 세션 설계시 고려사항

1) 모든 페이지에서 로그아웃 가능하도록 UI 구성

2) 로그아웃 요청 시 할당된 세션이 완전히 제거

3) 세션 타임아웃은 중요도가 높으면 2~5분 낮으면 15~30분으로 설정

4) 이전 세션이 종료되지 않으면 새 세션이 생성되지 못하도록 설계

5) 중복 로그인 허용 않은 경우 클라이언트의 중복 접근에 대한 세션 관리 정책 수립

6) 패스워드 변경 시 활성화된 세션 삭제 후 재할당

5. 세션ID 관리 방법

1) 안전한 서버에서 최소 128비트 길이로 생성

2) 예측 불가능하도록 안전한 난수 알고리즘

3) 세션ID노출되지 않도록 URL Rewrite(쿠키 사용 못할 때 URL에 세션 ID포함시키는 것) 기능을 사용하지 않는 방향으로 설계

4) 로그인 시 로그인 전 세션ID 삭제, 재할당

5) 장기간 접속 시 주기적 재할당

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

193주차 - 입력 데이터 검증 및 표현

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 입력 데이터 검증 및 표현의 개요

1) 입력 데이터로 인해 발생한 문제를 예방하기 위해 구현 단계에서 검증 해야하는 보안 점검 항목

2) 구현 단계에서 유효성 검증 체계 갖추고, 검증되지 않은 데이터가 입력되는 것을 처리하도록 구현

3) 일관된 언어셋 사용하도록 코딩

2. 입력 데이터 검증 및 표현의 보안 약점

1) SQL 삽입, 자원 삽입, 크로스사이트 스크립팅(XSS), 운영체제 명령어 삽입 등 공격에 취약

2) 보안 약점 종류

- SQL 삽입

- 웹 응용 프로그램에 SQL 삽입하여 내부 디비 서버의 데이터 유출 및 변조, 관리자 인증 우회

- 동적 쿼리에 사용되는 입력 데이터에 예약어 및 특수문자가 입력되지 않게 필터링 되도록 설정하여 방지

- 자원 삽입

- 데이터 입출력 경로를 조작하여 서버 자원 수정삭제

- 사용자 입력값을 식별자로 사용하는 경우, 경로 순회 공격을 막는 필터를 사용하여 방지

- 크로스사이트 스크립팅(XSS)

- 웹페이지에 악의적으로 스크립트 삽입하여 정보 탈취하거나 비정상적 기능 수행 유발

- HTML 태그 사ㅛㅇㅇ 제한하거나 스크립트에 삽입되지 않도록 "/><\"등의 문자를 다른 문자로 치환하여 방지

- 운영체제 명령어 삽입

- 외부 입력값을 통해 시스템 명령어 실행 유도하여 권한 탈취 및 시스템 장애 유발

- 웹 인터페이스를 통해 시스템 명령어가 전달되지 않도록 하고 외부 입력값을 검증하여 방지

- 위험한 형식 파일 업로드

- 악의적 명ㄹ령어가 포함된 스크립트 파일 업로드함으로써 시스템 손상 줌

- 업로드 파일 확장자 제한, 파일명 암호화 통해 방지

- 신뢰되지 않는 URL주소로 자동접속 연결

- 입력 값으로 사이트 주소를 받는 경우 이를 조작하여 방문자를 피싱 사이트로 유도하는 보안 약점

- 연결되는 외부 사이트의 주소를 화이트 리스트로 관리하여 방지

- 메모리 버퍼 오버플로

- 연속된 메모리 공간을 사용하는 프로그램에서 할당된 메모리 범위를 넘어선 위치에서 자료를 읽거나 쓰려고 할때 발생

- 프로그램 오동작 유발, 악의적 코드 실행시켜 공격자가 프로그램 통제할 권한 획득

- 메모리 버퍼를 사용할 경우 적절한 버퍼 크기 설정, 올바르게 쓰도록 함으로써 방지

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

194주차 - 보안 기능

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 보안 기능의 개요

1) 구현 단계에서 코딩하는 기능인 인증, 접근제어, 기밀성, 암호화 등을 올바르게 구현하기 위한 보안점검항목

2. 보안 기능의 보안 약점

1) 적절한 인증 없이 중요기능 허용

- 보안 검사 우회를 조심하고, 재인증 기능 수행하도록 하여 방지

2) 부적절한 인가

- 모든 실행 경로에 대해 접근제어 검사 수행

3) 중요한 자원에 대한 잘못된 권한 설정

4) 취약한 암호화 알고리즘 사용

5) 중요정보 편문 저장 및 전송

6) 하드코드(직접 입력한)된 비밀번호

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

197주차 - 코드 오류

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 코드 오류의 개요

1) 구현 단계 코딩 중 실수하기 쉬운 형변환,자원 반환 등 오류 예방

2. 널 포인터 역참조

1) 아무것도 가리키지 않는 포인터를 참조한는 것

2) 스택 가드

- 주소가 저장되는 스택에서 발생하는 보안 약점 막는 기술

- 메모리상에서 복귀 주소와 변수 사이에 특정 값을 저장 후 값이 변경될 경우 오버플로우 상태로 판단해 실행 중단하여

잘못된 복귀 주소 호출 막는 기술

3. 부적절한 자원 해제

1) 반환 코드 누락이나 오류로 인한 반환 못하는 상태

2) 힙 메모리, 소켓 등 유한한 시스템 자원이 계속 점유하고 있으면 자원 부족으로 새로운 입력 못할 수 있다.

4. 해제된 자원 사용

1) 이미 사용이 종료되어 반환된 메모리 참조하는 경우

2) 쓰레기 값이 들어갈 수 있어 포인터 초기화해서 방지

5. 초기화되지 않은 변수 사용

1) 초기화하지 않아서 이전에 정보가 남아있을 수 있음

2) 초기화하면 됨

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

200주차 - 암호 알고리즘

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 암호 알고리즘 개요

1) 해시를 사용하는 단방향 암호화 방식과, 개인키 및 공개키로 분류되는 양방햐 암호화 방식이 있다.

2. 개인키 암호화 기법

1) 대칭 암호 기법, 비밀키 암호화 기법

2) 암호화 대상이 n개일 떄 키 개수는 n(n-1)/2이다.

3) 블록 암호화 방식

- 한번에 하나의 데이터 블록 암호화

- DES,SEED,AES,ARIA,IDEA

4) 스트림 암호화 방식

- 평문과 동일한 길이의 스트림을 생성해 비트/바이트/워드 단위로 암호화

- LFSR, RC4

5) 장점

- 암호화/복호화 속도 빠름, 단순, 파일 크기 작음

6) 단점

- 사용자 증가에 따라 관리해야할 키 수가 많아짐

3. 공개키 암호화 기법

1) 데이터베이스 사용자가 공개키로 암호, 관리자가 비밀리에 관리하는 비밀키로 복호

2) 암호화 대상이 n개일 때 사용되는 키 개수는 2n개이다.

3) 비대칭 암호 기법,RSA

4) 장점

- 키 분배 쉬움, 키 개수 적음

5) 단점

- 암호화/복호화 속도 느림, 복잡, 파일 크기 큼

6) 공개키 기반 구조(PKI : Public Key Infrastructure)

- X.509 방식 : 인증기관에서 발생하는 인증서 기반 상호 인증

- 비 X.509 방식 : 국가별,지역별로 맞게 보완 및 개발

4. 양방향 알고리즘 종류

1) SEED

- 한국인터넷진흥원(KISA)에서 개발한 블록 암호화 알고리즘

- 블록 크기 128비트, 키 길이에 따라 128,256으로 분류됨

2) ARIA

- 국가정보원과 산학연합회에서 개발한 블록 암호화 알고리즘

- 학계,연구기관,정부 앞글자로 구성

- 블록 크기 128비트, 키 길이에 따라 128,192,256으로 분류됨

3) DES

- 미국 NBS에서 발표한 개인키 암호화 알고리즘

- DES 3번 적용한 3DES도 있음

- 블록 크기 64비트, 키 길이 56비트

4) AES

- 미국 표준 기술 연구소(NIST)에서 발표한 개인키 암호화 알고리즘

- DES한계를 느낀 NIST에서 발표

- 블록 크기 128비트, 키 길이에 따라 128,192,256으로 분류됨

5) RSA

- 라이베스트,샤미르,애들먼이 제안한 공개키 암호화 알고리즘

- 큰 숫자는 소인수분해 하기 어렵다는 것에 기반

- 키 = 상수를 의미함

5. 해시

1) 임의의 길이의 입력 데이터나 메시지를 고정된 길이의 값이나 키로 변환

2) 해시함수,해시키

3) 무결성 검증, 정보보호

4) 복호화가 거의 불가능한 일방향 함수

5) 종류

- SHA시리즈

- 미국 국가안보국(NSA)가 설계, NIST 발표

- SHA-0,SHA-1,SHA-2,SHA-224,SHA-256,SHA-384,SHA-512

- MD5

- 라이베스트가 MD4 대체하기 위한 함수

- 블록 크기 512비트, 키 길이 128비트

- N-NASH

- 일본 전신전화주식회사(NTT) 발표

- 블록 크기 128비트, 키 길이 128비트

- SNEFRU

- 머클이 발표

- 32비트 프로세스 구현 용이

- 블록 크기 512비트. 키 길이에 따라 128비트,256비트로 분류

6. 솔트(Salt)

1) 암호화를 수행하기 전에 원문에 무작위 값을 덧붙이는 과정

2) 덧붙이는 무작위 값을 솔트라고 함

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

201주차 - 서비스 공격 유형

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 서비스 거부(DoS : Denial of Service) 공격의 개념

1) 서버의 자원을 고갈시킬 목적으로 다수의 공격자 혹은 시스템에서 대량의 데이터를 한 곳의 서버에 집중적으로 전송

2) 공격 유형 :Ping of Death, SMURFING, SYN Flooding, TearDrop, Land, DDOS

2. Ping of Death(죽음의 핑)

1) Ping 명령을 전송할 때 패킷의 크기를 인터넷 프로토콜 허용 범위(65,536바이트) 이상으로 전송하여 네트워크 마비시킴

2) 큰 패킷은 수백 개의 패킷으로 분리되어 전송되는데 전송된 패킷을 재조립하는 부담과

각각의 ICMP Ping 메시지에 대한 응답 처리하느라 시스템 다운됨

3) jolt, sPING, ICMP bbug, IceNewk 등 변종 공격에 대해 ICMP Ping 메시지가 전송되지 못하도록 방화벽에서 차단

3. SMURFING

1) IP나 ICMP 특성을 악용해 엄청난 양의 데이터를 한 사이트에 집중시켜 네트워크 불능상태로 만듦

2) 공격자는 송신 주소를 공격 대상지의 IP로 위장하고 해당 네트워크 라우터의 브로드캐스트 주소를 수신지로 하여 패킷을 전송

라우터의 브로드캐스트 주소로 수신된 패킷은 해당 네트워크 내의 모든 컴퓨터로 전송됨

3) 응답 메시지를 공격 대상지로 집중적으로 전송, 네트워크 과부하가 걸림

4) 각 네트워크 라우터에서 브로드캐스트 주소를 사용할 수 없게 미리 설정해놓아 방지

4. SYN Flooding

1) TCP는 3-way-handshake를 거친 후 데이터 전송하는데 가상의 클라이언트로 위장해 3-way-handshake를 의도적으로 중단시켜

대상 서버가 대기 상태에 놓여 서비스 수행하지 못하는 것

2) 공격자는 사용할 수 없는 IP주소를 이용해 대상서버에 SYN신호르 보냄

3) 대상 서버는 응답을 보낸뒤 접속을 받아들이기 위해 메모리 일정 공간 확보

4) 해당 IP는 가상이므로 응답전송도 안되고,응답이 오지도 않아서 메모리 공간 확보한 채로 대기

5) 위 과정을 반복

6) 'SYN' 수신 대기 시간을 줄이거나 침입 차단 시스템 활용해 방지

5. TearDrop

1) 데이터 송수신 과정에서 패킷의 크기가 커 여러개로 분할되어 전송될 때 분할 순서를 알 수 있도록 Fragment Offset 값을 함께 전송함

2) 이 값을 변경시켜 다시 조립할 때 오류로 인한 과부하를 발생시킴

3) Fragment Offset이 잘못된 경우 해당 패킷을 폐기함으로써 방지

6. Land

1) 송수신 IP주소를 모두 공격 대상 IP주소로 함

2) 이 패킷을 받은 공격 대상은 송신 IP주소가 자신이므로 자신에게 응답 수행하여 반복적으로 행할 시 무한히 응답함

3) IP 주소 적정성을 검사함으로써 방지

7. DDoS(Distributed Denial of Service, 분산 서비스 거부) 공격

1) 여러 곳에 분산된 공격 지점에서 한 곳의 서버에 공격 수행

2) 네트워크 취약점이 있는 호스트 탐색 후 호스트에 분산 서비스 공격용 툴을 설치 후 에이전트로 만든 후 DDoS에 이용

3) 공격 범위를 확대하기 위해 일부 호스트에 다수 에이전트를 관리할 수 있는 핸들러 프로그램을 설치해 마스터로 지정

8. 네트워크 침해 공격 관련 용어

1) 스미싱(Smishing)

- 문자 메시지를 이용해 사용자 개인 신용 정보 빼옴

- 초기에는 문자를 통해 소액 결제 유도하는 형태

- 현재는 행사 안내,경품 안내 등에 링크를 걸어 apk 파일 설치 유도하여 정보 빼옴

2) 스피어 피싱(Spear Phishing)

- 사회 공학의 한 기법, 특정 대상 선정 후 메일을 지속적으로 보내 클릭 유도해 개인정보 탈취

3) APT(Advanced Persistent Threats, 지능형 지속 위협)

- 다양한 IT 기술을 통해 특정 기업이나 조직 네트워크에 침투해 활동 거점 마련 후 보안 무력화시킬 방법 찾은 후 외부로 빼돌림

- 공격 방법

- 악성코드 이메일 오랜 기간 꾸준히 보내 한번이라도 클릭되길 기다리는 형태

- 스턱스넷과 같이 악성코드가 담긴 이동식 디스크(USB)로 전파하는 형태

- 악성코드에 감염된 P2P사이트에 접속하면 악성코드에 감염되는 형태

4) 무작위 대입 공격(Brute Force Attack)

- 암호화된 문서의 암호키를 찾기 위해 모든 값 대입하여 공격

5) 큐싱(Qshing)

- QR코드로 악성 앱 다운로드 유도

6) Phishing

- 개인 정보와 낚시의 합성어, 이메일이나,메신저 통해 사칭하여 개인정보 빼냄

7) Ping Flood

- 특정 사이트에 매우 많은 ICMP메시지를 보다 응답으로 시스템 자원을 모두 써서 정상적으로 동작하지 못하도록 공격

8) Evil Twin Attack

- 실제 존재하는 동일한 이름의 WiFi 신호를 송출해 로그온한 사람들의 계정정보 등을 빼냄

9) 스위치 재밍(Switch Jamming)

- 위조된 매체 접근 제어(MAC) 주소를 지속적으로 네트워크로 흘려보내 스위치 MAC주소 테이블의 저장 기능을 혼란시켜 더미허브처럼

작동하게 하는 것

9. 블루투스 관련 공격

1) 블루버그 : 블루투스 장비 사이의 취약한 연결 관리를 악용한 공격, 폰을 원격 조정하거나 통화 감청 가능

2) 블루스나프 : 블루투스 취약점을 통해 장비의 파일에 접근하는 공격, 인증 없이 정보 교환할 수 있는 OOP를 사용하여 정보 열람

3) 블루프린팅 : 공격 대상이 될 블루투스 장비 검색

4) 블루재킹 : 블루투스를 이용해 스팸처럼 메시지 익명으로 전달

10. 정보 보안 침해 관련 용어

1) 좀비 PC

- 악성코드에 감염되어 다른 프로그램이나 컴퓨터를 조종하는 컴퓨터, C&C서버의 제어를 받아 DDoS공격에 이용

2) C&C서버

- 해커가 원격지에서 감염된 좀비PC에 명령 내리고 악성코드를 제어하기 위한 용도로 사용되는 서버

3) 봇넷

- 악성 프로그램에 감염되어 악의적인 의도로 사용될 수 있는 다수의 컴퓨터들이 네트워크로 연결된 형태

4) 웜

- 네트워크를 통해 연속적으로 자신을 복제하여 시스템 부하 높여 다운시키는 바이러스로, DDoS,버퍼오버플로,슬래머 등

5) 제로 데이 공격

- 보안 취약점이 발견되었을 때 알려지기 전에 빠르게 공격하는 것

6) 키로거 공격

- 키보드 움직임 탐지해 개인 정보 빼감

7) 랜섬웨어

- 사용자의 컴퓨터에 잠입해 문서를 암호화하는 것, 해독을 빌미로 돈을 요구함

8) 백도어

- 개발자가 액세스 편의를 위해 시스템 보안을 제거하여 만들어놓은 비밀 통로

- 탐지 방법 : 무결성 검사,열린포트확인,로그분석,SetUID 파일 검사 등

9) 트로이목마

- 정상기능 프로그램인 척 숨어 있다가 프로그램이 동작할 때 활성화되어 부작용 일으킴, 자가복제능력은 없음

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

202주차 - 서버 인증

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 보안 서버의 개념

1) 인터넷을 통해 개인정보를 암호화하여 송수신할 수 있는 기능을 갖춘 서버

2) 서버에 SSL(Secure Socket Layer) 인증서를 설치하여 전송 정보를 암호화하여 송수신하는 기능

3) 서버에 암호화 응용 프로그램을 설치

4) 스니핑을 이용한 정보 유출, 피싱을 이용한 위조 사이트 등에 대비하기 위해 보안 서버 구축 필요

2. 인증의 개념

1) 지식 기반 인증(Something You Know)

- 사용자가 기억하고 있는 정보 기반 인증

- 비밀번호, 패스 프레이즈, 아이핀 등

- 패스 프레이즈 : 길이가 길고 기억하기 쉬운 문장으로 비밀번호 구성

- 아이핀 : 사이버 주민등록번호, 온라인으로 발행

2) 소유 기반 인증(Something You Have)

- 사용자가 소유하고 있는 것을 기반으로 인증

- 신분증, 메모리 카드(토큰), 스마트 카드, OTP(One Time Password)

3) 생체 기반 인증(Something You Are)

- 사용자의 고유한 생체 정보 기반으로 인증

- 지문, 홍채/망막, 얼굴, 정맥

4) 위치 기반 인증(Somewhere You Are)

- 인증을 시도하는 위치의 적절성 확인

- 콜백(전화가 왔을 때 끊고 다시 걸어 유효한 번호인지 확인하는 방법), GPS나 IP주소를 이용한 위치 기반 인증

5) 행위 기반 인증(Something You Do)

- 사용자의 행동 정보를 이용해 인증 수행

- 서명, 동작, 음성

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

203주차 - 보안 아키텍처/보안 프레임워크

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 보안 아키텍처(Security Architecture)

1) 정보 시스템의 무결성,가용성,기밀성을 확보하기 위해 보안 요소 및 체계를 식별하고 이들 관계를 정의한 구조

2) 관리적 물리적 기술적 보안 개념 수립, 보안 관리 능력 향상, 일관된 보안 수준의 유지를 기대할 수 있다.

- 관리적 보안 : 정보보호 정책, 정보보호 조직 등

- 물리적 보안 : 건물 및 사무실 출입 통제, 전산실 관리 지침 등

- 기술적 보안 : 사용자 인증, 접근 제어, PC, 서버 등 보안 지침 정의

3) 보안 아키텍처 모델의 대표적인 표준 : ITU-T X.805

2. 보안 프레임워크

1) 안전한 정보 시스템 환경 유지, 보안 수준 향상시키는 체계

2) ISO 27001

- 정보보안 관리를 위한 국제 표준, 보안 인증이자 가장 대표적인 보안 프레임워크

- 영구 BSI가 제정한 BS 7799를 기반으로 만들어짐

- 조직에 대한 정보보안 관리 규격이 정의되어 있어 실제 심사/인증용으로 사용됨

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

205주차 - 보안 솔루션

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. 보안 솔루션의 개념

1) 접근 통제, 침입, 침입 차단 및 탐지 등 수행 등 수행하여 외부로부터 불법적인 침입을 막는 기술 및 시스템

2. 방화벽(Firewall)

1) 네트워크나 인터넷 간에 전송되는 정보를 선별하여 수용,거부,수정하는 기능을 가진 침입 차단 시스템

2) 내부에서 외부로 나가는 패킷은 통과시키고, 외부에서 내부로 들어노는 패킷을 엄밀히 체크, 인증된 패킷만 통과

3) 해킹 등에 의한 외부로의 정보 유출 막기위해 사용

3. 침입 탐지 시스템(IDS : Intrusion Detection System)

1) 시스템의 비정상적인 사용 등을 실시간으로 탐지한느 시스템

2) 방화벽과 같은 침입 차단만으로는 내부 사용자의 불법적인 행동과 외부 해킹 100퍼센트 완벽하게 대처할 수 없음

3) 오용 탐지 : 미리 입력해 둔 공격패턴이 감지

4) 이상 탐지 : 평균적인 시스템 상태 기준 비정상적인 행위나 자원 사용 감지

5) 침입 탐지 시스템 종류

- HIDS(Host-Based Intrusion Detection System)

- 시스템 내부 감시,분석에 중점을 둔 시스템

- 변화를 실시간으로 감시, 기록 추적

- 종류 : OSSEC, md5deep, AIDE, Samhain

- NIDS(Network-Based Intrusion Detection System)

- 외부로부터 침입 감시,분석에 중점을 둔 시스템

- 네트워크 트래픽 감시, 서비스 거부 공격, 포트 스캔 등 악의적인 시도 탐지

- 종류 : Snort, Zeek

6) 침입 탐지 시스템의 위치

- 패킷이 라우터로 들어오긴 전 : 네트워크에 시도되는 모든 공격 탐지

- 라우터 뒤 : 라우터에 의해 패킷 필터링 통과한 공격 탐지

- 방화벽 뒤 : 내부에서 외부로 향하는 공격 탐지

- 내부 네트워크 : 내부에서 내부 네트워크의 해킹 공격 탐지

- DMZ : 외부인터넷에 서비스 제공 서버가 위치한 네트워크, 강력한 내외부 공격으로부터 서비스 중단 방지

4. 침입 방지 시스템(IPS : Intrusion Prevention System)

1) 방화벽과 침입 탐지 시스템 결합한 것

2) 비정상적 트래픽을 능동적으로 차단 후 격리 등의 방어 조치

3) 침입 탐지 기능으로 패킷 하나씩 검사 후 비정상적 패킷이 탐지되면 방화벽 기능으로 해당 패킷 차단

5. 데이터 유출 방지(DLP : Data Leakage/Loss Prevention)

1) 내부 정보의 외부 유출을 방지

2) 사내 직원의 컴퓨터와 네트워크상 정보 모두 검사

6. 웹 방화벽(Web Firewall)

1) 일반 방화벽이 탐지 못하는 SQL삽입공격,Croos-Site Scripting(XSS) 등의 웹 기반 공격을 방어

2) 웹 관련 공격 감시, 웹 서버에 도달 전에 차단

7. VPN(Virtual Private Network, 가상 사설 통신망)

1) 인터넷 등 통신 사업자의 공중 네트워크와 암호화 기술을 이용해 사용자가 마치 자신의 전용 회선을 사용하는 것처럼 해줌

2) 비용 부담 줄고, 지역적인 제한 없이 업무 수행 가능

8. NAC(Network Access Control)

1) 네트워크에 접속하는 MAC주소를 IP관리 시스템에 등록 후 일관된 보안 관리 기능 제공

2) 불법적인 소프트웨어 설치 방지

3) 백신,보안 패치 등의 설치 및 업그레이드 수행

9. SSH(Secure SHell, 시큐어 셸)

1) 다른 컴퓨터에 로그인, 원격 명령 실행, 파일 복사 등 수행할 수 있도록 다양한 기능을 지원하는 프로토콜

2) 데이터 암호화와 강력한 인증 방법으로 보안성 낮은 네트워크에서도 안전하게 통신

3) 키를 통한 인증 방법 사용하려면 공개키를 서버에 등록해야 됨

4) 기본으로 22번 포트 사용