PB001: Úvod do informačních technologií

Luděk Matyska

Fakulta informatiky Masarykovy univerzity

podzim 2015

Obsah přednášky

1. Ochrana a bezpečnost
2. Client-server model

Ochrana a bezpečnost

- Obecná ohrožení:
  - Přístup (čtení)
    - Nezanechává přímo stopy
  - Zápis (modifikace)
    - Následné využití útočníkem modifikovaných dat
    - Zahrnuje i smazání/přepsání
  - Znepřístupnění služby (denial of service)
- Možné útoky
  - Přihlášení, impersonifikace, ...
  - Trojský kůň
  - Viry

Více o útocích

- Sociální inženýrství
  - Uhodnutí nebo získání hesla
  - Využívá důvěřivosti a naivity lidí
  - Technologie může pomoci jen do jisté úrovně
    - Nutnost koordinované shody dvou či více lidí – 7 klíčů k korunovačním klenotům
    - Kombinace fyzických nástrojů a tajemství (po krádeži karty je třeba ještě získat pin a naopak, samotný pin bez karty není k ničemu)
- Využití technických nedostatků
  - Bezpečnostní ,,díry“, ,,zadní vrátka“ apod.
  - Je možné minimalizovat korektními programátorskými praktikami
  - a pravidelnou aplikací záplat
  - Automatizované nástroje pro ,,oťukání“ systému
- Botnety
  - Sítě již napadených počítačů
  - Využitelné k dalším útokům

Principy návrhu bezpečných systémů

- Zveřejnění šifrovacích a souvisejících algoritmů
- Standardní nastavení = žádný přístup
  - Správce/uživatel musí aktivně rozhodnout, co komu dovolí
- Minimální oprávnění
- Pravidelné kontroly
  - „Díry“, nadbytečná oprávnění, ...
- Jednoduchý a uniformní mechanismus
  - Složitost vede k nepochopení a to k chybám
- Úrovně oprávnění
  - Delegace oprávnění na konkrétní akci

Ochrana souborů

- Základní operace:
  - čtení, zápis (včetně vytvoření), smazání, prodloužení a spuštění souboru
- Základní ochrana
  - Různá pro různé operace
  - Specifikace, kdo smí co: Ochranné domény:
    - Skupina, která má stejná práva
    - Statické versus dynamické
    - Např.: Já, moji přátelé, ostatní
    - POSIX (UNIX): user:group:other
    - Možná i jiná schemata

Řízení přístupu k souborům

Access Control List, ACL (seznamy přístupových oprávnění)

- ke každému souboru je připojen seznam přístupových oprávnění
- sestává se z uspořádaných dvojic (doména,operace)

Zjednodušená varianta (z UNIXových systémů):

- pouze tři záznamy: *u* uživatel, *g* skupina, *o*: ostatní
- operace:
  - `r`: čtení souboru (čtení obsahu adresáře)
  - `w`: zápis souboru (včetně vytvoření)
  - `x`: spuštění (sestoupení do podadresáře)
- Příklad
  - rw-r-----
  - Uživatel může číst i zapisovat, skupina smí jen číst, ostatní nesmí nic

Řízení přístupu k souborům

Plné ACL:

- libovolný počet záznamů
- více práv: smazání, změna oprávnění...
- negativní záznamy (explicitní odepření operace)
- dynamická dědičnost – propagace změn do podadresářů
- např. AFS, Windows od verze 2000, ext4 s ACL

Řízení přístupu k souborům

Capability List, CL

- Uspořádání podle domén, nikoliv podle souborů
- Schopnost (capability) tj. práva přístupu patří procesu a ten je může:
  - předávat dalším procesům (delegace)
  - modifikovat (degradovat, nemůže rozšířit práva)
  - smazat
- Proces se při přístupu k souboru prokazuje odpovídající schopností
- Možnost transferu schopností mezi procesy: vhodné pro distribuované systémy

Ochrana přístupu uvnitř OS

- Kernel a uživatelský prostor
- Oddělení na hw úrovni
- Každá stránka někomu patří
- Pouze kernel má přístup k hardware
  - Kontroluje práva přístupu
  - Obsluhuje zařízení (pro všechny)
  - Garantuje serializaci přístupu
- Uživatelské procesy používají *volání* kernelu (jádra)
- Korektnost kernelu kritická

Přístup k paměti

- Příslušnost virtuálních stránek k procesu
- Výpadek stránky: nepovolený přístup
- Ochrana
  - Mezi procesem a jádrem
  - Mezi procesy
  - Uvnitř procesu

Client-server model

- Distribuované počítání
  - Využití prvků (počítače) propojených počítačovou sítí
  - Dekompozice úlohy na podúlohy
  - Paralelní vykonávání podúloh
  - Na různých systémech propojených sítí
- Client-server model
  - Speciální případ distribuovaného počítání
  - Více strukturované
  - Asymetrické: *klient* posílá požadavek na zpracování *serveru*
  - Server pro jednoho klienta může být klientem pro jiný server.

Vlastnosti modelu client-server

- Klient a server samostatné procesy
- Na stejném nebo různých počítačích
- Interní informace je „soukromá“ pro každý proces
  - Klient i server se mohou vzájemně prokázat (autentizace)
- Komunikují duplexním protokolem
  - Komunikace může být šifrovaná

Požadované vlastnosti

- Interoperabilita
  - Klient a server mohou běžet na zcela odlišných systémech
- Portabilita
  - Stačí zajistit u klientů
- Integrace
- Transparence
  - Klient vidí jen ,,svůj“ server, nikoliv jeho další komunikaci
- Bezpečnost
  - Autentizace klienta i serveru
  - Šifrovaná komunikace
  - Důvěryhodný server

Příklady

- `telnet`, `ssh`
- X Window systém na Unixu
- Světová pavučina (World Wide Web)
- Distribuované systémy souborů (AFS, NFS, Samba/CIFS)

Třívrstevný model klient-server architektury

- Základní rozčlenění
  - Data
  - Logika
  - Prezentace
- Sousední možno kombinovat/rozdělit (tj. např. Logika může být součástí datové i prezentační vrstvy, a to i současně)

„Tlustý“ a „tenký“

- Platí pro server i klient, podstatné zejména v souvislosti s klienty
- „Tlustý“ (fat) klient:
  - Značná spotřeba lokálních zdrojů (CPU, paměť, disk)
  - Komplexní provedení i instalace
  - Příklad: Mozilla
- „Tenký“ (thin) klient:
  - Jednodušší
  - Snadná správa a přenositelnost
  - Menší škálovatelnost (příliš mnoho práce dělá server)
  - Zpravidla vyšší nároky na propustnost sítě

Middleware

- Software zajišťující funkcionalitu distribuovaných systémů
  - „Zkratka“ v rámci protokolů
  - Stojí ,,nad“ operačním systémem, ale ,,pod“ aplikací
  - Propojuje oddělené komponenty distribuovaného systému
- Dovoluje aplikacím komunikaci přímo na vyšší abstraktní úrovni
- Realizuje jednu (RPC) nebo více (DCE) funkcí

Middleware – příklady

- Primitivní: přenos souborů
- Základní: RPC (Remote Procedure Call)
- Integrované: DCE (Distributed Computing Environment)
- Distribuované objektové služby: CORBA, OGSA (Open Grid Service Architecture), Web Services

CORBA

- Common Object Request Broker Architecture
- Základem ORB: vrstva, která zprostředkovává komunikaci (middleware pro middleware)
- Kompomenty:
  - Rozhraní (řetězce)
    - Umožňují volání procedur mezi klientem a serverem
  - Pojmenování (naming service)
  - „Obchodní“ služba (trader)
    - Vyhledávání vhodného serveru
  - A mnoho dalších

Webové služby

- Využívají standardní protokoly zavedneé v rámci WWW konsorica (W3C)
- Určeny pro interakci mezi počítači přes počítačovou síť
- WSDL (Web Services Description Language)
  - Popisuje rozhraní služby
- SOAP (Simple Object Access Protocol)
  - Protokol pro výměnu zpráv
- XML (eXtensible Markup Language)
  - Značkovací jazyk používaný pro popis objektů a vlastní komunikaci