Ďalšie služby počítačových sietí

Mgr. Dan Keder keder@fi.muni.cz

Centrum výpočetní techniky
Fakulta informatiky
Masarykova univerzita

30. september 2009

Obsah

Elektronická pošta

Elektronická pošta

- Jedna z najstarších a najpoužívanejších služieb
- Prenos textových správ (emailov) po sieti
- Formát adresy: `xnovak@fi.muni.cz`, `xnovak@aisa.fi.muni.cz`
- Protokoly
  - SMTP
  - POP3, IMAP
  - i proprietárne protokoly (MS Exchange, IBM Lotus Notes)

Elektronická pošta

- Používa sa tzv. store-and-forward model
  - Správa je postupne preposielaná medzi viacerými servermi, až kým sa nedostane do cieľa
- Kdo sa zúčastňuje na prenose správ:
  - **MUA** (Mail User Agent) – užívateľská aplikácia (Thunderbird, Outlook, . . .)
  - **MSA** (Mail Submission Agent, často integrovaný s MUA) – posiela emailové správy MTA
  - **MTA** (Mail Transfer Agent) – prijíma emailové správy od MSA alebo iného MTA, preposiela ich ďalej
  - **MDA** (Mail Delivery Agent) – pripíma emailové správy od MTA a lokálne ich doručuje do mailboxu adresáta

Formát správy
Elektronická pošta

- Definovaný v RFC 5322
- Správa sa skladá z hlavičiek a tela správy, oddelených prázdnym riadkom
- Správa môže obsahovať len 7bitové ASCII znaky
- MIME (Multipurpose Internet Mail Extensions)
  - pridáva možnosť používať v správe iné znakové sady než ASCII – znaky sa kódujú do ASCII (quoted-printable, BASE64)
  - Telo správy môže mať stromovú štruktúru (prílohy)

Formát správy – Hlavičky

Elektronická pošta

- Majú formát `Názov: Hodnota`, na veľkosti písmen nezáleží
- Na každom riadku jedna, riadok musí začínať názvom (tzn. pred názvom nesmú byť medzery, tab a pod.)
- Ak riadok začína bielym znakom, jedná sa o pokračovanie predchádzajúceho riadku
- Hlavičky sa ukončia prázdnym riadkom
- Typy hlavičiek:
  - štrukturované – jednotlivé údaje sa nejak interpretujú
  - neštrukturované – údaj je určený pre človeka

Formát správy – Hlavičky II.

Elektronická pošta

- Príklady hlavičiek:
  - `From:` a `To:` – odosielateľ a adresát správy
  - `Subject:` – subjekt správy, o čom správa je
  - `Cc:` a `Bcc:` – komu sa má poslať kópia správy
  - `Date:` – dátum odoslania správy
  - `Received:` – táto hlavička sa pridá pri každom spracovaní počas transportu správy (relay), dá sa z nej určiť, kadiaľ správa putovala
  - `Reply-To:` – Adresa pre zaslanie odpovede
  - `Sender:` – Identifikácia skutočného odosielateľa (tvorcu) správy
- Chyby pri transporte sa zasielajú na `Sender:` (ak je uvedené)
- Odpoveď sa zasiela na `Reply-To:`. Ak nie je uvedené, tak na `From:`
- Hlavičky `From:`, `To:` sa **neinterpretujú**, skutočný adresát a odosielateľ sa neurčuje podľa nich

Formát správy – Telo správy

Elektronická pošta

- Môže obsahovať len 7bitové ASCII znaky
- Nesmie obsahovať jeden znak '.' na začíatku riadku (ak riadok začína na '.', do správy sa dá '..')
- Na logickej úrovni môže mať stromovú štruktúru (MIME)
- Binárne dáta sa kódujú pomocou BASE64

Konvencie pri písaní emailovej správy

Elektronická pošta

- Uvádzať zmysluplný `Subject`
- Neposielať veľké množstvo dát v prílohách
- Nepoužívať HTML formátovanie v mailoch, príp. zároveň uvádzať i textovú verziu správy
- Dodržovať max. dĺžku riadku 78 znakov
- Pri odpovede na mail nevytvárať nový mail, ale použíť funkciu Odpovedať (Reply), príp. Odpovedať všetkým (Reply to All)
  - Subject by mal obsahovať na začiatku text `Re:`
- Text pôvodnej správy uvodzovať znakom >, napr.:

```
> Povodny text
>
Odpoved
```

SMTP
Elektronická pošta

- Protokol, ktorým medzi sebou kominukujú MTA (prípadne i MSA a MTA)
- Obálka správy vs. text správy
  - Hlavičky v správe nemajú nič spoločného s obálkovými hlavičkami
  - Skutočný adresát správy sa určuje podľa obálkovej adresy
- Príkazy: `HELO` (príp. `EHLO`), `MAIL FROM:`, `RCPT TO:`, `QUIT`

SMTP – Príklad

Elektronická pošta

- v Linuxe: `telnet anxur.fi.muni.cz smtp`

```
220 anxur.fi.muni.cz ESMTP NO UCE NO SPAM - CVT Vas lubi.
HELO nemesis
250 anxur.fi.muni.cz
MAIL FROM: keder@fi.muni.cz
250 2.1.0 Ok
RCPT TO: keder@fi.muni.cz
250 2.1.5 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
From: keder@fi.muni.cz
To: keder@fi.muni.cz
Subject: test

test
.
250 2.0.0 Ok: queued as 454664937C1
QUIT
221 2.0.0 Bye
```

POP3, IMAP

Elektronická pošta

- Protokoly na klientský prístup k pošte
- POP3 (RFC 1939) je jednoduchší, umožňuje len stiahnuť obsah vzdialeného mailboxu do lokálneho mailboxu
  - Port 110 (príp. 995 v prípade POP3 nad SSL)
  - Problém pri prístupe do mailboxu z viacerých strojov
- IMAP (RFC 3501) je zložitejší, ale poskytuje širšie možnosti použitia
  - Port 143 (príp. 993 v prípade IMAPu nad SSL)
  - Podpora simultánneho prístupu viacerých klientov k jednému mailboxu
  - Umožňuje sťahovať správy po častiach, príp. len hlavičky
  - Vytváranie, mazanie, premenovávanie mailboxov na serveri
  - Vyhľadávanie mailov podľa rôznych kritérii na strane serveru
  - Udržuje informácie o stave správy (zmazaná, prečítaná, . . .)

Adresárové a autentizačné služby

Adresárové a autentizačné služby

- **Autentizácia** – overenie totožnosti užívateľa
- **Adresárové služby (Directory services)**
  - poskytujú prístup k rôznym informáciám (napr. údaje o užívateľoch)
  - optimalizované pre read-only prístup
- **Autentizačné služby**
  - umožňujú overiť totožnosť užívateľa
  - Dva prístupy: lokálne na každom stroji alebo centralizovane pre mnoho počítačov
- Single Sign-on

Flat files

- Lokálne uložené informácie o užívateľoch a skupinách užívateľov
- Formát DSV (delimiter separated values), ľahko prístupné a strojovo spracovateľné
- `/etc/passwd`
  - `login:passwd:uid:gid:gecos:home_dir:shell`
- `/etc/shadow`
  - `login:enc_passwd:a:b:c:d:e:f:reserved`
  - `a-f` – password aging information
  - `enc_passwd` – šifrované heslo (`crypt(5)`, MD5)
- `/etc/group`
  - `name:passwd:gid:members`

LDAP
Lightweight Directory Access Protocol

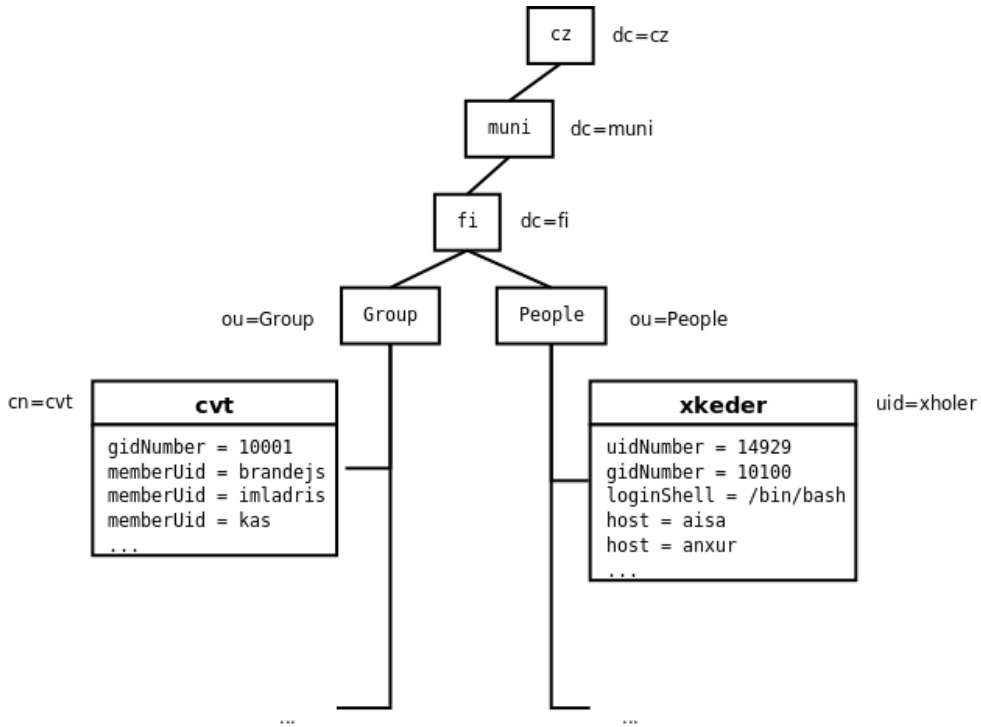
- Centralizovaná adresárová služba
- Vychádza z ISO štandardov X.500, v praxi sa moc nepoužíva, príliš zložité
- LDAP je "odľahčená" verzia protokolu X.500 Directory Access Protocol
- Pôvodne ako brána k X.500, neskôr plnohodnotná služba s mnohými rozšíreniami
  - štandardné API, datové formáty, ...
- Sieťová komunikácia nad TCP/IP
- Výhody: internetový štandard, ľahká integrácia do aplikácií
- Konkrétne implementácie: OpenLDAP, Active Directory (Microsoft), Open Directory (Apple)

- **Directory Information Tree (DIT)** – konkrétny návrh štruktúry stromu
- **Distinguished Name (DN)** – jednoznačne identifikuje položku v globálnom mennom priestore adresárového stromu
- **Relative DN** – jednoznačne identifikuje položku v rámci jednej vetvy stromu

- Schéma má hierarchickú stromovú štruktúru (používa sa DNS)
- Uzly i listy stromu uchovávajú záznamy
  - záznamy sú zložené z atribútov
  - každý záznam má určitý typ (*objectClass*)
  - povinné a voliteľné atribúty
  - záznam môže mať súčasne niekoľko typov
  - príklad atribútov
    - *dc* – domain component
    - *c* – country
    - *o* – organization
    - *ou* – organization unit name
    - *cn* – common name

LDAP
LDAP strom na FI

- `ldap.fi.muni.cz`
- Udržuje informácie o unixových užívateľoch a skupinách (ekvivalent `/etc/passwd` a `/etc/group`)



- v Linuxe:

```
$ ldapsearch -x -H ldap://ldap.fi.muni.cz \
-b ou=People,dc=fi,dc=muni,dc=cz uid=xkeder

$ ldapsearch -x -H ldap://ldap.fi.muni.cz \
-b ou=Group,dc=fi,dc=muni,dc=cz cn=cvt
```

- Výstup vo formáte LDIF (LDAP Data Interchange Format):

```
dn: uid=xkeder,ou=People,dc=fi,dc=muni,dc=cz
uid: xkeder
cn: xkeder
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
userPassword:: e2NyeXB0fXg=
loginShell: /bin/bash
uidNumber: 14929
gidNumber: 10100
homeDirectory: /home/xkeder
gecos: Daniel Keder
(...)
```

Kerberos

Kerberos

- Centralizovaná autentizačná služba
- RFC 1510, 1964
- Overenie identity bez posielania tajných dát sieťou
- Dve strany (A, B) a dôveryhodný server (KDC, Key Distribution Center)
- A i B zdieľajú s KDC nejaké tajomstvo – heslo
- KDC generuje lístok, zašifruje heslom A a pošle klientovi
- Klient A dešifruje svojím heslom a lístok použije k preukázaniu svojej identity
- Pekný popis algoritmu na `http://en.wikipedia.org/wiki/Kerberos_(protocol)`

- **principal** – užívateľ, stroj alebo služba
- `primary/instance@realm`
  - jedinečná identifikácia užívateľa, služby alebo stroja v databáze Kerbera
  - `primary` – meno užívateľa alebo služby, 'host'
  - `instance` – voliteľná časť v závislosti na `primary` (napr. hostname)
  - `realm` – logická sieť reprezentovaná kerberovskou databázou a KDC (väčšinou zhodná s DNS doménou)
- Napríklad
  - `xkeder@FI.MUNI.CZ`
  - `host/aisa.fi.muni.cz@FI.MUNI.CZ`
  - `ftp/aisa.fi.muni.cz@FI.MUNI.CZ`

Kerberos
Praktický príklad

- Vyskúšajte si na strojoch nymfe:
  - `kinit` – získanie lístku od KDC
  - `klist` – výpis získaných lístkov
  - `kdestroy` – zmazanie získaných lístkov

PAM
Pluggable Authentication Modules

- PAM je sada knižníc integrujúca rôzne autentizačné mechanizmy do jedného API.
- Oddelenie detailov autentizácie od aplikácie, konfigurovateľnosť autentizačného procesu
- Dostupný na väčšine unixových systémov
- Modulárny princíp, je jednoduché zmeniť proces či pridať nový spôsob autentizácie (napr. odtlačky prstov)
- Existuje množstvo modulov, i pre Kerberos, LDAP

- Konfigurácia uložená súboroch v adresári `/etc/pam.d/`
- Každá aplikácia tu má svoj vlastný konfiguračný súbor
- Štyri časti autentizačného procesu:
  - `account` – ako overiť existenciu a platnosť užívateľského účtu
  - `auth` – ako overiť totožnosť užívateľa
  - `password` – ako urobiť zmenu hesla
  - `session` – správa sedenia (napr. auditing)
- Pre každú časť by mal byť nastavený aspoň jeden modul
- Každý modul má nastavenú "dôležitosť" (required, sufficient, optional, requisite)

PAM
Príklad konfigurácie

```
auth            required        pam_env.so
auth            sufficient      pam_thinkfinger.so
auth            sufficient      pam_unix.so \
try_first_pass likeauth nullok
auth            required        pam_deny.so
account         required        pam_unix.so
password        sufficient      pam_unix.so \
try_first_pass use_authtok nullok md5 shadow
password        required        pam_deny.so
session         required        pam_limits.so
session         required        pam_unix.so
```

Zdieľanie súborov v sieti

FTP
File Transfer Protocol

- Protokol pre prenos súborov nad TCP
- Riadiaci kanál (port 21), dátový kanál (port 20)
- Dva režimy prenosu dát:
  - aktívny – *klient* serveru oznámi neprivilegovaný port, na ktorom počúva, server otvorí dátové spojenie
  - pasívny – *server* klientovi oznámi neprivilegovaný port, na ktorom počúva, klient sám otvorí dátové spojenie
- Problémy
  - chýbajúce šifrovanie (hlavne hesiel a dát)
  - samostatné dátové spojenie pre každý prenos
  - aktívny režim nefunguje za NAT (Network Address Translation)

SCP
Secure Copy Protocol

- Prenos súboru medzi dvoma stanicami cez SSH
- Klient komunikuje so vzdialene spusteným `scp`
- Obmedzenia
  - max. 4GB súbory
  - nepodporuje obnovenie prenosu
  - nedá sa vypísať obsah adresára
  - podpora kompresie dát (vhodné pre pomalé linky)
- Triviálny príklad:

```
$ scp test.txt user@server.example.com
```

SFTP
SSH File Transfer Protocol

- Podobné vlastnosti ako SCP, bez vypísaných obmedzení
- Nemá nič spoločného s protokolom FTP
- Od základu novo navrhnutý, používaný v SSHv2 (ako jeho subsystém)
- Užitočný program pre MS Windows: WinSCP

NFS
Network File System

- Protokol sprístupňujúci po sieti vzdialené zväzky
- Niekoľko generácií:
  - v1: experimentálny
  - v2: pôvodne bezstavový protokol nad UDP s podporou zámkov; neskôr dopracovaná podpora TCP
  - v3: vyšší výkon, súbory $> 4GB$; málo bezpečné, ale **rýchle**
  - v4: vyšší výkon, väčsia bezpečnosť (šifrovanie, autentizácia), internetový štandard

CIFS
Common Internet File System

- Pôvodne protokol SMB (Server Message Block) od IBM, značne prepracovaný Microsoftom.
- Nielen zdieľanie súborov, ale i zdrojov (tlačiarne)
- Pôvod vo Windows, existujú i otvorené implementácie pre iné operačné systémy
- Horšia implementácia vo Windows, často zneužívané chyby
- Výkonovo horšie než NFS

Synchronizácia času

Synchronizácia času

- Prečo synchronizovať čas
  - potreba presného času nie je až tak dôležitá
  - dôležitá je predstava o súvislosti dejov
  - je dôležité mať všade **rovnaký** čas
- Možné riešenia:
  - vzájomná dohoda uzlov na čase
  - časový synchronizačný server so (sprostredkovaným) presným časom
- Protokoly na synchronizáciu času: Time, Daytime, NTP

NTP
Network Time Protocol

- RFC 778, 891, 956, 958, 1305
- Protokol prenosu aktuálneho času cez médium s premenlivou dobou doručenia informácie
- UDP port 123
- Časové servery hierarchicky radené
  - stratum 0 – atómové hodiny a pod., veľká presnosť
  - stratum 1 – synchronizované so stratum 0
  - stratum 2 a 3 – synchronizácia koncových počítačov
  - (. . .) – teoreticky až 256 úrovní

NTP
Network Time Protocol

- Čas uplynutý od 1.1.1900 00:00 uložený v 64bitovej hodnote
  - 32b počet sekúnd
  - 32b desatinná časť
- Pre korekciu času používa Marzullov algoritmus
- Podpora vo Windows i unixových systémoch
- Synchronizujte si svoj čas podľa `time.fi.muni.cz`

Otázky?

Koniec
Kde hľadať podrobnejšie informácie

- Repozitár RFC dokumentov
  - `http://tools.ietf.org/html/`
- Manuálové stránky a oficiálna dokumentácia k programom
- Google, Wikipedia
- Predmety na FI
  - PV004 UNIX
  - PV065 UNIX – programování a správa systému I
  - PV077 UNIX – programování a správa systému II
  - PV090 UNIX – seminář ze správy systému