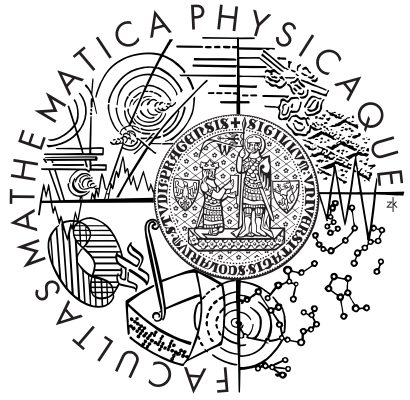
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Michal Vachna

Přehled metod komunikace s externími zařízeními v laboratoři

Ústav částicové a jaderné fyziky

Vedoucí bakalářské práce: RNDr. Peter Kodyš, CSc.

Studijní program: Informatika, Správa počítačových systémů

2009

Chcel by som poďakovať vedúcemu mojej práce, RNDr. Petrovi Kodyšovi, CSc., ktorý mi dovolil pracovať na tejto téme, usmerňoval ma a poskytoval rady počas písania práce.

Prehlasujem, že som svoju bakalársku prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím so zapožičiavaním práce a jej zverejňovaním.

V Prahe dňa 27. 7. 2009 Michal Vachna

Obsah

Názov práce: Přehled metod komunikace s externími zařízeními v laboratoři
Autor: Michal Vachna
Katedra (ústav): Ústav částicové a jaderné fyziky
Vedúci bakalárskej práce: RNDr. Peter Kodyš, CSc.
e-mail vedúceho: Peter.Kodys@mff.cuni.cz

Abstrakt: V predloženej práci je cieľom zmapovať súčasnú situáciu počítačovej infraštruktúry vo fyzikálnom laboratóriu, preskúmať požiadavky a problémy pri práci v laboratóriu v prípadoch ako zálohovanie dát, vzdialený prístup, zdieľanie diskového priestoru, organizácia prístupu procesov k externým zariadeniam, správa verzií vyvíjaných meracích aplikácii, spúšťanie procesov na vzdialených počítačoch. Posúdiť existujúce riešenia, zvážiť možnosti vylepšení a navrhnúť nové riešenia vhodné pre potreby laboratória. Pokiaľ to bude možné, vzhľadom na súčasnú prevádzku laboratória, realizovať riešenia pre zvolené problémy a požiadavky. Práca má byť metodickým materiálom a vodítkom pre ďalší rozvoj laboratória v nasledujúcich rokoch.

Klúčové slová: zálohovanie dát, vzdialený prístup, zamykanie súborov, systém na správu verzií, zdieľanie súborov

Title: Communication with external devices in laboratory
Author: Michal Vachna
Department: Institute of Particle and Nuclear Physics
Supervisor: Peter Kodyš, Ph.D.
Supervisor's e-mail address: Peter.Kodys@mff.cuni.cz

Abstract: In the present work the goal is to map present circumstances of computer infrastructure in the physical laboratory, examine the demands and problems at work in the laboratory such as data backup, remote control, disk space sharing, access organization of processes to external devices, version control of measuring application development, running processes on remote hosts. Assess existing solutions, consider the possibility of improvements and propose new solutions appropriate to the needs of the laboratory. As far as possible, in regard to the current operations of the laboratory, implement solutions to selected problems and requirements. The work has to be methodical material and guide for further development of the laboratory in the coming years.

Keywords: data backup, remote control, files locking, version control system, file sharing

Kapitola 1

Úvod

Laboratórium v Těžkých laboratořích Ústavu částicové a jaderné fyziky sa zaoberá testovaním a simuláciami kremíkových polovodičových detektorov nabitých častíc v časticovej fyzike, laserovými testami, vývojom programov na zber dát a analýzou dát získaných pri meraniach. Pri tejto práci sa využívajú meracie počítače (DAQ PCs) a k ním pripojené externé zariadenia. Počas testov, meraní a samotného chodu laboratória sa objavujú prevádzkové problémy a vznikajú požiadavky pracovníkov na funkčnosť laboratória spojené s nasledujúcimi oblasťami. Komunikácia medzi meracími počítačmi s rozdielnym operačným systémom umožní pracovať s viacerými počítačmi v rámci jedného merania. Organizácia práce s extérnymi zariadeniami urýchli a zefektívni merania. Možnosť pre pracovníkov laboratória vykonávať testy a merania nielen osobne v laboratóriu ale aj vzdialene z kancelárie alebo domova, im ušetrí čas a poskytne pohodlie. Výsledkom práce v laboratóriu sú dôležité dáta, ktoré je potrebné chrániť pred stratou a poškodením. Cieľom práce by malo byť zmapovania súčasného stavu výpočtovej infraštruktúry v laboratóriu, zosumarizovať implementované postupy, problémy a nové požiadavky. Zvážiť priority problémov, ktoré najviac zlepšia pracovné podmienky a pre tie najdôležitejšie navrhnúť riešenia a vykonať ich realizáciu.

Kapitola 2

Stav v laboratóriu

Laboratórium predstavuje takzvaná šedá miestnosť, prechodová miestnosť a čistá miestnosť upravená k udržiavaniu zvýšenej čistoty ovzdušia v triede čistoty ISO 7 podľa štandardu ISO 14644-1. V čistej miestnosti sa nachádzajú dva počítače a sada externých zariadení. Počítače slúžia na vykonávanie pokusov a meraní pomocou externých zariadení. Dáta z nich získané sa následne ukladajú a ďalej spracovávajú. Komunikácia s externými zariadeniami prebieha cez rôzne typy rozhraní. Šedá miestnosť pozostáva z počítača, ktorý slúži na kontrolu čistej miestnosti. Monitoruje stav čistej miestnosti a zaznamenáva dáta zo špeciálnych detektorov[1]. Celé laboratórium je pre prípad výpadku elektrickej siete vybavené záložnými zdrojmi napájania UPS[2].

UPS je zariadenie alebo systém, ktorý zabezpečuje plynulú dodávku elektriny pre zariadenia, ktoré nesmú byť neočakávane vypnuté. UPS je obvykle zapojený medzi primárny zdroj napätia (elektrického prúdu) a zálohovaný vstup napájania chráneného zariadenia. UPS funguje na princípe akumulátora. Pokiaľ nie je dodávka elektrického prúdu z primárneho zdroje prerušená, je UPS udržiavaný v nabitom stave. Zároveň slúži ako ochrana proti podpätiu alebo prepätiu v sieti. V okamihu prerušenia dodávky elektriny zabezpečuje napájanie zariadenia až do svojho vybitia alebo obnovenia dodávky. Obdobie, počas ktorého UPS udrží zariadenie v behu, je dané kapacitou akumulátorov a odoberaným výkonom, pohybuje se od niekoľko minút po niekoľko hodín.[11]

2.1 Meracie počítače

Ako bolo spomenuté v úvode, úlohou meracích počítačov **DAQ PCs**[3] je vykonávanie pokusov a meraní pomocou externých zariadení. Dáta z nich získané následne ukladať a ďalej spracovávať. Kedže sa jedná o dôležité dáta, ktoré môžu byť výsledkom niekoľko hodinových meraní, je nutné, aby boli zaloho-

[1] napríklad detektory teploty, tlaku, vlhkosti ovzdušia

[2] skratka anglického termínu Uninterruptible Power Supply - neprerušiteľný zdroj energie

[3] anglický termín **D**ata **Aq**uisition **P**ersonal **C**omputer**s**

vané. Časť dát sa po spracovaní odosiela na server mimo laboratória, kde sa dáta použijú ako obsah internetových stránok. Pokusy a merania je možné vykonávať osobne v laboratóriu alebo vzdialeným prístupom po sieti.

V čistej miestnosti sa nachádzajú dva meracie počítače s rozdielnym operačným systémom, konkrétne Windows XP a Debian Linux. Oba počítače a tretí počítač v šedej miestnosti sú pripojené do spoločného HUBu, ktorý umožňuje prístup do internetu a je tiež pripojený na záložný zdroj napájania UPS. Meracie počítače zdieľajú monitor, myš a klávesnicu pripojené na hardwarový prepínač umožňujúci prepínať ovládanie medzi počítačmi. Toto riešenie bolo navrhnuté a realizované kvôli úspore miesta.

Väčšina externých zariadení je pripojená k počítaču s operačným systémom Windows z dôvodu, že v súčasnosti chýbajú alebo nie sú vôbec dostupné vhodné ovladače rozhraní pre operačný systém Linux. Naopak niektoré zariadenia musia byť pripojené k počítaču s operačným systémom Linux, pretože v laboratóriu sa používajú aj špeciálne meracie programy, ktoré existujú len vo verzii pre operačný systém Linux a nemajú ekvivalentnú náhradu pre operačný systém Windows. Kvôli tomu, že viac počítačov pracuje s externými zariadeniami, je nutné riešiť okrem organizácie prístupu procesov podieľajúcich sa na meraní k zariadeniam v rámci jedného počítača aj organizáciu prístupu medzi viacerými počítačmi a ich vzájomnú komunikáciu.

2.2 Monitorovací počítač

Monitorovací počítač sa nachádza v šedej miestnosti. Monitoruje podmienky v čistej miestnosi ako teplotu a stav ovzdušia, stav externých zariadení a stav napájania miestnosti. V prípade problému informuje osoby zodpovedné za chod laboratória prostredníctom posielania mailov a krátkych textových správ SMS.

2.3 Externé zariadenia

V čistej miestnosti sú zariadenia uvedené v nasledujúcej tabuľke. V tabuľke sú taktiež uvedené rozhrania, prostredníctvom ktorých sú externé zariadenia pripájané.

externé zariadenie	rozhranie OS Windows	rozhranie OS Linux
DAQ - SCT	VXI-MXI-2	-
DAQ - DEPFET (2ks)	USB	USB, 100BaseTX
polohovací stolček (5ks)	RS232	-
generátor signálov (2ks)	IEEE-488	-
optický atenuátor	RS232	-
zdroj nízkeho napätia (2ks)	RS232	-
zdroj vysokého napätia (2ks)	IEEE-488	-
osciloskop	IEEE-488	-
optický power meter*	RS232	-
voltmeter*	RS232	-
osciloskop*	RS232, USB	-

Externé zariadenia v tabuľke označené * nie sú momentálne používané pri meraniach, ale uvažuje sa o ich zaradení do testov.

Popis rozhraní:

USB(Universal Serial Bus) je štandard sériovej zbernice určenej na pripojenie zariadení k počítaču. USB bolo navrhnuté pre možnosť pripojenia viacerých periférií použitím štandardizovaného rozhrania a zlepšiť schopnosti plug and play pridaním hot swappingu[4]. Ďalšie výhodné vlastnosti zahrňujú poskytovanie napájania zariadeniam s nízkou spotrebou, čo eliminuje potrebu externého zdroja napájania a možnosť použitia väčšiny zariadení bez potreby inštalovania špeciálnych ovládačov od výrobcu zariadení. USB podporuje nasledujúce komunikačné rýchlosti[12]:

- **Low-Speed** - rýchlosť 1.5 Mbit/s, definovaná v USB 1.0. Využíva sa najviac na takzvané Human Interface Devices ako sú klávesnice, myši a joysticky.
- **Full-Speed** - rýchlosť 12 Mbit/s, základná komunikačná rýchlosť definovaná v USB 1.1 Všetky USB rozbočovače podporuju Full Speed.
- **High-Speed** - rýchlosť 480 Mbit/s bola uvedená v roku 2001 a definovaná v USB 2.0. Všetky High-Speed zariadenia sú v prípade potreby schopné pracovať ako Full-Speed.
- **SuperSpeed** - rýchlosť 5.0 Gbit/s, definovaná v USB 3.0. Špecifikácia USB 3.0 bola vydaná firmou Intel a jej partnermi v auguste 2008.

100BaseTX je štandart 100 Mbps Fast Ethernet technológie. 100BaseTX je sieť postavená na dvoch pároch krútenej dvojlinky kategórie 5 a vyššie so segmentom do 100 metrov. Sieťový adaptér je pripojený cez konektor RJ-45. Prenosová rýchlosť je 100 Mbit/s.[2]

[4]dovoľuje pripojiť a odpojiť zariadenia bez reštartu počítača alebo vypnutia zariadenia

RS232[5] alebo **sériový port** je rozhranie pre prenos informácií vytvorené pôvodne pre komunikáciu dvoch zariadení do vzdialenosti 20 metrov (na väčšie vzdialenosti sa používajú rozhrania RS422 a RS485). Pre väčšiu odolnosť proti rušeniu je informácia po prepojovacích vodičoch prenášaná väčším napätím, než je štandartných 5 voltov. Prenos informácií prebieha asynchrónne, pomocou pevne nastavenej prenosovej rýchlosti a synchronizácie zostupnou hranou štartovacieho impulzu. Prenosové rýchlosti závisia od dĺžky a kapacity použitého prepájacieho káblu: 20 kbit/s, 115200 bit/s, maximálne 1.5 Mbit/s.[3]

IEEE-488[6] je štandardné rozhranie pre prepojenie meracích prístrojov s počítačom pre účely automatizácie merania. Maximálne 14 meracích zariadení a systémový kontrolér môže byť paralelne prepojených na zbernici. Maximálna dĺžka zbernice je 20 metrov a maximálna vzdialenosť medzi zariadeniami je 2 metre. Prenosová rýchlosť je 1 MB/s podľa pôvodného štandardu, s neskoršími rozšíreniami až 8 MB/s.[10]

VXI-MXI-2 rozhranie je rozšírenie triedy veľkosti C a B pre mainframe VXI zbernice (VMEbus Extensions for Instrumentation). VXI-MXI-2 modul rozširuje architektúru VXI zbernice mimo mainframe VXI zbernice cez MXI-2, druhú generáciu MXI zberníc (Multisystem Extension Interface bus). Mainframe VXI zbernice vybavený rozhraním VXI-MXI-2 môže byť prepojený s iným zariadením so zbernicou MXI ako mainframe VXI zbernice, zariadenie so zbernicou MXI alebo počítačom so zbernicou MXI. Prenosová rýchlosť je maximálne 33 MB/s.[5]

[5]Recommended Standard 232, jeho poslednou variantou je RS-232C

[6]známe je pod názvom HP-IB (Hewlett-Packard Interface Bus), GPIB (General Purpose Interface Bus) a IMS-2

Kapitola 3

Požiadavky a problémy

3.1 Komunikácia po sieti smerom z laboratória

Skúmané oblasti:

- Synchronizácia meracích počítačov - synchronizovanie času na meracích počítačoch pri vykonávaní aktualizácie internetových stránok a zálohovaní dát na úložisko mimo laboratória

Súčasný mechanizmus nedostatočne synchronizuje meracie počítače a dochádza k situáciam, kedy sa rozdiel časov v priebehu týždňa dostane na hodnoty rádovo minúty. To sťažuje vykonávanie naplánovaných operácií, pretože je vyžadované, aby prebiehali súčasne. Porovnávanie presných časov slúži tiež na kontrolu funkčnosti počítačov a prekročením stanovených limitov sa spúšťajú automatické poplašné opatrenia.

3.2 Komunikácia po sieti smerom do laboratória

Skúmané oblasti:

- Vzdialený prístup - pripojenie k meracím počítačom z miesta mimo laboratória, plánovanie, kontrola a ovládanie meraní.

Súčasné riešenie pracuje nezabezpečene po otvorenej linke. Navrhované riešenie musí prebiehať zabezpečenou linkou.

3.3 Komunikácia v rámci čistej miestnosti

Skúmané oblasti:

- Komunikácia medzi procesmi jedného meracieho počítača - organizácia prístupu procesov k externým zariadeniam v rámci jedného meracieho počítača.
- Komunikácia medzi procesmi viacerých meracích počítačov - organizácia práce procesov s externými zariadeniami v rámci viacerých meracích počítačov s rôznymi operačnými systémami (Linux, Windows).
- Možnosti spúšťania procesov na vzdialenom meracom počítači s odlišným operačným systémom - uvažovaná funkcionalita, ktorá by v prípade realizovateľnosti prispela k novým možnostiam pri meraniach.

Organizácia prístupu procesov meraní k externým zariadeniam je dôležitá, pretože s konkrétnym zariadením môže pracovať vždy len jeden proces. Počas doby práce so zariadením musí byť exkluzívne používané práve jedným procesom, nesmie dochádzať k tomu, že ho počas tejto doby začne používať aj iný proces. To platí aj pri začiatku a ukončení práce so zariadením, proces si musí vytvoriť pomyselný zámok na zariadenie a po skončení práce ho korektne zrušiť. Nesmie nastať, aby si zámok na konkrétne zariadenie vytvorilo viac procesov súčasne. Funkčnou organizáciou prístupu sa zabezpečí efektívne využívanie externých zariadení.

3.4 Zálohovanie, zdieľanie a správa verzií

Skúmané oblasti:

- Zálohovanie - potreba zálohovania výsledkov meraní.
- Zdieľanie a správa verzií - potrebné zdieľanie a správa verzií ROOT kódov, C++ zdrojových kódov, binárnych súborov, dynamických knižníc.

Zálohovanie by mohlo prebiehať na lokálny externý disk pripájaný cez rozhranie USB, prípadne na vzdialené úložisko. Ďalšou možnosťou je zapojenie diskov ako RAID 1, ide o jednoduché zrkadlenie dát, kde na polovici diskov sa nachádzajú dáta, s ktorými sa pracuje a na druhú sa dáta len zrkadlia (zálohujú).

K zdieľaniu a správe verzií bude vhodné použiť rozumný software k tomu určený, pričom dáta bude potrebné umiestniť na dobre prístupné miesto, teda server.

Kapitola 4

Analýza a návrh riešení

4.1 Komunikácia po sieti smerom z laboratória

Súčasťou komunikácie je posielanie varovných mailov a správ SMS, aktualizácia internetových stránok a zálohovanie dát. Varovné maily a správy SMS sa posielajú v prípade problému prostredníctvom monitorovacieho počítača. Odosielania sú veľmi nepravidelné (niekoľkokrát za deň) a jedná sa o dáta do 1KB. Pri aktualizácií internetových stránok sa odosielajú texty a obrázky. Odosielania je pravidelné (1 až 4 krát za hodinu) a jedná sa o dáta rádovo v desiatkach až stovkách KB. O zálohovaní dát na úložisko mimo laboratória sa v súčasnosti len uvažuje. Prebiehať by malo pravidelne a za podmienky, že od poslednej zálohy prebehli merania a jedná sa o dáta rádovo až v GB.

Komunikácia po sieti má ako proces vysokú prioritu a ostatné bežiace procesy sú kvôli tomu obmedzované alebo pozastavené. To vadí hlavne v prípade, že prebieha meranie. Preto bude potrebné správne naplánovanie hlavne aktualizácií internetových stránok a zálohovania dát. Posielanie SMS je v kritickom prípade, keď nastal závažný problém, čo má väčšiu prioritu ako prebiehajúce merania. Aktualizácia by preto mala prebiehať v presne stanovený čas, len v tom prípade, ak práve neprebieha žiadne meranie. Presné časy aktualizácie by mali byť stanovené podľa potreby ako často je potrebné ju vykonať. Zálohovanie by malo prebiehať v čase, keď sa zvyčajne laboratórium nepoužíva, teda neprebieha meranie a ešte určitú dobu prebiehať nebude, obvykle v skorých ranných hodinách (0:00 - 4:00).

Používaním súčasného mechanizmu dochádza k rozsynchronizovaniu časov rádovo v minútach. Pre zabezpečenie synchronizácie času na meracích počítačoch je vhodné použiť v systéme Windows v nastaveniach dátumu a času funkciu **Internet time** a v systéme Linux program **ntpd**[1]. Nastavením syn-

[1]NTP(Network Time Protocol) daemon - je démon, ktorý synchronizuje systémový čas s NTP serverom

chronizácie oboch systémov na rovnaký NTP server sa docieli ich samotná časová synchronizácia. Pre zachovanie synchronizácie je ešte potrebné nastaviť interval pre opätovné synchronizovanie s NTP serverom. V systéme Windows je to možné nastaviť v registroch. Taktiež je v registroch možné doplniť ďalšie adresy NTP serverov. V Linuxe sa celá konfigurácia programu ntpd vykonáva prostredníctvom konfiguračného súboru.

4.2 Komunikácia po sieti smerom do laboratória

Na vzdialený prístup na meracie počítače v laboratóriu sa používa voľne šíriteľná aplikácia **VNC**[2].

VNC je grafická aplikácia používajúca protokol RFB[3] na prístup k vzdialenému počítaču. Prenáša signál s klávesnice a myši, odosiela a zobrazuje grafickú obrazovku na vzdialenom počítači. To všetko prostredníctvom siete. VNC je platformovo nezávislé, VNC prehliadač (klient) na jednom operačnom systéme sa môže pripojiť k VNC serveru na inom operačnom systéme.[13] Takto je možné vzdialene spúšťať merania v laboratóriu bez nutnosti, aby v laboratóriu bola osoba fyzicky prítomná.

Problémom je bezpečnosť aplikácie, konkrétne komunikácie po sieti. Základná verzia VNC nepoužíva bezpečný protokol. Heslá síce nie sú posielané ako obyčajný text, ale už jednoduchými metódami na lámanie šifier je možné zistiť šifrovací kľuč a teda samotné heslo z odchytených prenesených dát po sieti. Preto sa odporúča použiť heslo dĺžky aspoň osem znakov, niektoré verzie VNC to nepovoľujú.

Jedným z riešení zabezpečenia VNC je, že VNC môže byť tunelované cez SSH alebo VPN[4] pripojenie, čo pridá bezpečnostnú vrstvu so silnejším šifrovaním. SSH klienti a SSH tunely sú dostupné v systéme Windows aj Linux. Taktiež existujú voľne šíriteľné aplikácie, ktoré dokážu vytvoriť priamy VPN tunel medzi počítačmi.

Ďalšou možnosťou je verzia VNC nazvaná **UltraVNC**, ktorá podporuje použitie šifrovacieho modulu, ktorý šifruje celé spojenie vrátane autentifikácie heslom a prenosu dát. Použitie tohto modulu však robí túto verziu nekompatibilnú s ostatnými verziami a naviac UltraVNC je vytvorené len pre platformu Windows a súčasťou laboratória je aj počítač so systémom Linux.

[2]Virtual Network Computing

[3]Remote FrameBuffer

[4]Virtual Private Network - virtuálna privátna sieť je spôsob akým simulovať privátnu sieť naprieč verejnou sieťou ako internet. Prívlastok virtuálna je kvôli používaniu virtuálnych spojení t.j. dočasných spojení, ktoré fyzicky neexistujú, ale pozostávajú z paketov smerovaných cez verejnú sieť. Zabezpečené virtuálne spojenie je vytvárané medzi dvoma počítačmi, počítačom a sieťou alebo dvoma sieťami[7]

Verzia VNC nazvaná **RealVNC** ponúka silné šifrovanie v rámci komerčného balíčka a je určená pre platformu Windows aj Linux. Cena licencie sa pohybuje okolo 50 amerických dolárov. Viac informácii o RealVNC v [6].

4.3 Komunikácia v rámci čistej miestnosti

V súčasnosti je implementovaná len komunikácia medzi procesmi jedného meracieho počítača a je len čiastočne funkčná. Funguje nasledovne: ak sa pri meraní má pracovať s externým zariadením, v určenom súbore sa overí, či je zariadenie používané (súbor obsahuje 1), v tomto prípade sa vyčká určitý čas a overí sa hodnota v súbore opäť, alebo ho práve iný proces nepoužíva (súbor obsahuje 0) a v tomto prípade sa pri inicializácii procesu vytvorí zámok daného zariadenia tým, že sa 0 prepíše na 1. Medzi overením, či je zariadenie nepoužívané (načítanie 0 zo súboru) a vytvorením zámku (zápis 1 do súboru) prebehne určitý čas, počas ktorého môže dôjsť k situácii, že v rámci iného merania sa zistí, že zariadenie je nepoužívané, pretože zámok ešte nebol vytvorený. Tým získajú prístup k jednému zariadeniu aspoň dva procesy, čo nastať nesmie. Všetky procesy meraní sú súčasťou aplikácii napísaných v jazyku C++. Preto by navrhované riešenie malo byť rozumne implementovateľné v tomto programovacom jazyku.

Pri komunikácii medzi procesmi viacerých meracích počítačov je hlavným problémom prístup k diskovému priestoru z oboch systémov tak, aby bolo možné priamo pracovať s aktuálnymi dátami. Procesy meraní by potrebné informácie pre komunikáciu zapisovali do príslušných súborov, kde by boli prístupné pre ostané procesy a aplikácie.

Preskúmanie možností spúšťania procesov na vzdialenom meracom počítači v prípade nájdenia realizovateľného riešenia prinesie nové možnosti meraní. Bude možné použiť viac meracích počítačov so zariadeniami k ním pripojenými na komplexnejšie merania.

4.3.1 Komunikácia medzi procesmi jedného meracieho počítača

Na existujúcej komunikácii stačí vyriešiť spomenutý problém zamykania extérneho zariadenia a komunikácia bude plne funkčná.

Riešenie by mohlo fungovať podľa nasledujúceho princípu. Pre prácu so súborom, ktorý slúži ako zámok na externé zariadenie, sa bude používať zámok na čítanie a zámok na zápis. Zámok na čítanie sa použije pri čítaní zo súboru, zámok na zápis pri zápise. Ak bude proces chcieť zapisovať, bude musieť vymeniť zámok na čítanie za zámok na zápis. Súbor nemôže byť zamknutý na zápis, ak ho má aspoň jeden iný proces zamknutý na čítanie alebo zápis. Preto proces pred vymenením zámku musí čakať, kým sa uvoľnia všetky ostatné zámky

príslušného súboru. Takáto verzia riešenia ale nie je veľmi dobrá, pretože môže vzniknúť deadlock[5], ak dva procesy budú vlastniť zámok na čítanie toho istého súboru a obidva ho budú chcieť vymeniť za zámok na zápis. Vhodnejšou verziou riešenia by bolo, aby proces musel najprv uvoľniť zámok na čítanie a potom požiadať o zámok na zápis (a čakať, kým sa všetky zámky na čítanie aj zápis uvoľnia).

Ako jednoduchšie riešenie sa naskytá upravenie úrovne zamykania (zámok na čítanie a zámok na zápis), a to nahradením dvoch zámkov jedným. Jednalo by sa o zamykanie prístupu k súboru, ktorý slúži ako zámok na externé zariadenie. Mechanizmus zamykania zaručí, že k súboru nezíska prístup viac procesov simultánne. Väčšina operačných systémov ponúka platformovo-špecifické knižnice implementujúce zamykanie. Vhodnejšou cestou však bude jednoduchá a prenositeľná technika zamykania bez využitia platformovo-závislých služieb.

Ako navrhnúť zásady zamykania? Na meracom počítači môže prebiehať viac meraní paralelne. Pre voltmeter ako externé zariadenie existuje súbor voltmeter.ez, ktorý slúži ako zámok na dané externé zariadenie. Procesy spustené v meraniach v prípade, že s voltmetrom chcú pracovať, čítajú z tohto súboru a môžu do neho zapisovať.

Pre zaistenie toho, aby len jediný proces modifikoval súbor, vytvorí sa zámok na daný súbor. Zámok bude v skutočnosti obyčajným súborom na disku rovnakého mena ako súbor, na ktorý je vytváraný a s koncovkou .lck. Ak teda bude existovať voltmeter.lck, súbor voltmeter.ez bude považovaný za uzamknutý a iné procesy budú mať odopreté upravovať ho. Ak zámok voltmeter.lck neexistuje a proces chce modifikovať súbor voltmeter.ez, musí najprv zámok voltmeter.lck vytvoriť. Len potom môže do súboru voltmeter.ez zapisovať. Ak sa súbežne pokusí viac procesov vytvoriť zámok voltmeter.lck, kvôli atomickosti tejto operácie len jeden z nich dostane návratovou hodnotou informáciu o úspechu. Ostatné procesy dostanú návratovou hodnotou informáciu o neúspechu a môžu sa neskôr ďalej pokúšať o vytvorenie zámku voltmeter.lck a teda sprístupniť si voltmeter pre seba. Ak proces skončí prácu s externým zariadením, po druhýkrát modifikuje súbor voltmeter.ez a následne zmaže zámok voltmeter.lck. Uvoľnením zámku sprístupní zariadenie ostatným čakajúcim procesom.

Navrhovaný mechanizmus zamykania súboru vlastne pozostáva len z vytvorenia ďalšieho súboru s koncovkou .lck v tom istom adresári, kde je umiestnený súbor, ktorý sa zamyká. Treba pamätať, že tento uzamykací mechanizmus je založený na dobrovoľnej spolupráci všetkých zúčastnených procesov: všetky procesy, ktoré budú pracovať s extérnymi zariadeniami a tým používať súbory

[5]uviaznutie - situácia, keď aspoň dva procesy čakajú natrvalo na udalosť, ktorú môže vyvolať len niektorý z čakajúcich procesov[8]

s koncovkou .ez, musia mať daný mechanizmus implementovaný. V prípade, ak by proces nevykonal kontrolu, či je súbor uzamknutý, pred tým než sa pokusí do neho zapisovať, môže nastať situácia, že viac procesov bude pracovať so súborom a nastane kolízia, ktorá je nežiadúca. Kolízii je možné predísť použitím a nastavením schém zoznamov prístupových práv pre dané súbory, takzvané Access Control Lists. Tým bude prístup k súborom závislý na identite užívateľa. Vytvorením špeciálneho užívateľa, ktorý ako jediný bude mať povolený prístup a iba procesy spúšťané pod jeho identitou budú môcť modifikovať zamykané súbory, sa zamedzí náhodným procesom v prístupe k týmto súborom. Neochráni to súbory pred procesmi spúšťanými ako procesy špeciálneho užívateľa, na rešpektovanie mechanizmu musia dohliadať osoby, ktoré programujú aplikácie vykonávajúce merania, preto sa jedná o dobrovoľnú spoluprácu procesov.

Iný problém sa môže vyskytnúť, ak proces, ktorý drží zámok, ukončí svoj beh pred uvoľnením zámku. Dôvodom ukončenia behu môže byť neošetrená výnimka alebo signál kill poslaný iným procesom. V tomto prípade ostane súbor voltmeter.ez uzamknutý zámkom voltmeter.lck. Problému je možné vyhnúť sa tým, že sa zámku nastaví dátum a čas. Táto informácia sa stane automatickou súčasťou súboru voltmeter.lck. Po expirácií tejto doby môže iný proces tento súbor zmazať a znova skúšať požiadavku na obsadenie uvoľneného externého zariadania štandartným postupom.

Hlavnou výhodou tohto mechanizmu uzamykania je jednoduchosť a platformová nezávislosť. Viac sofistikované mechanizmy ako **mutexy**[6] a **spinlocky**[7] sú bezpečnejšie a ponúkajú väčšiu funkcionalitu prostredníctvom vysokej granularity prístupových politík. Napríklad dovoľujú vytvoriť pravidlo, ktoré povolí prístup k súboru maximálne štyrom procesom. Rozhodnutie, ktorý mechanizmus použiť, by malo byť založené na tom, aký stupeň bezpečnosti je požadovaný a aké špecializované synchronizačné knižnice sú dostupné. Spomenuté mechanizmy sú použiteľné medziplatformovo a však pre daný problém komunikácie medzi procesmi sú príliš robustné. Navrhovaný jednoduchý mechanizmus by mal požiadavkám na funkcionalitu dostatočne vyhovovať. Navyše sa dá rozširovať jeho funkcionalita. Napríklad je možné zapísať PID[8] vlastníka zámku do zámku, aby ostatné procesy vedeli, kto zámok práve drží. Taktiež vytvárať takzvaný log súbor, v ktorom by sa zaznamenávalo kedy, kto a prípadne akú akciu vykonal so zámkom respektíve externým zariadením.

[6]binárny semafór zabezpečujúci vzájomné vylúčenie (**mut**ual **ex**clusion). Semafór je synchronizačný nástroj pre zaistenie výhradného prístupu do kritických častí kódu. Častí, v ktorých sa pracuje so zdieľanými zdrojmi a v danom momente k ním môže pristúpiť len jeden proces, nikdy nie viac súčasne[8]

[7]typ semafóru, pri ktorom proces v prípade, že nemôže vstúpiť do kritickej časti kódu a vytvoriť si zámok, čaká v cykle, v ktorom sa tak povediac točí (z anglického spin). Jedná sa o techniku aktívneho čakania (anglický termín busy waiting)[8]

[8]process identifier

4.3.2 Komunikácia medzi procesmi viacerých meracích počítačov

Ako bolo naznačené v úvode časti 3.3, daná komunikácia nie je vôbec implementovaná. Je potrebné zaistiť, aby procesy na dvoch rôznych počítačoch mali prístup k tým istým súborom na prácu s externými zariadeniami a ďalším súborom potrebným pre komunikáciu. Rozšírenie by malo spočívať v doplnení mechanizmu, ktorý zabezpečí, že meracie počítače a ich procesy budú mať prístup k adresáru na disku, ktorý bude obsahovať dané súbory a budú môcť vykonávať operácie: čítanie zo súboru, zápis do súboru, vytvorenie súboru a zmazanie súboru. Týmto sa daný problém odstráni a komunikácia procesov viacerých meracích počítačov bude vyriešená.

Je nutné navrhnúť riešenie zdieľania diskového priestoru, ktorý bude prístupný z operačného systému Windows a tiež z operačného systému Linux. Vzhľadom na to, že v laboratóriu sú iba dva počítače, bude postačujúce, ak diskový priestor bude fyzicky umiestnený na jednom z meracích počítačov. V prípade, že by išlo o zdieľanie diskového priestoru medzi viacerými počítačmi, vhodnejším riešením by bolo umiestnenie diskového priestoru na serveri prístupnom pre všetky počítače.

Sú možné dve verzie riešenia vzhľadom na umiestnenie zdieľaného diskového priestoru:

- Verzia Windows - diskový priestor bude zdieľaný meracím počítačom s operačným systémom Windows.
- Verzia Linux - diskový priestor bude zdieľaný meracím počítačom s operačným systémom Linux.

Verzia Windows

Zdieľané dáta budú umiestnené na disku meracieho počítača s operačným systémom Windows. Jednou z možností ako dáta sprístupniť meraciemu počítaču s operačným systémom Linux je použiť komerčný software **DiskShare** od firmy Javvin Technologies. DiskShare dovoľuje počítačom s operačným systémom Windows NT/2000/2003/XP prevádzkovať NFS[9] server. Tým sú prístupné dáta počítačom s operačným systémom Linux a ďalším operačným systémom, ktoré podporujú NFS. Výrobca sľubuje jednoduchú správu, bezpečnosť a výkon. Kedže sa jedná o produkt pre operačný systém Windows, užívateľské rozhranie je jednoducho ovládateľné a obsahuje prvky ako ikony a objektovo-orientované menu. DiskShare je však komerčný produkt a cena licencie sa pohybuje okolo 400 amerických dolárov. Za ďalší príplatok, čo činí 100 amerických dolárov, si môže zakúpiť zákazník ročnú technickú podporu. Viac informácií o aplikácii DiskShare na internetových stránkach výrobcu [4].

[9]Network File System

Ďalšou možnosťou je nastavenie meracieho počítača s operačným systémom Linux tak, aby dokázal pracovať so zdieľaným adresárom alebo diskom meracieho počítača s operačným systémom Windows. K tomu bude nutné nakonfigurovať **mountovanie**[10] daného zdieľaného diskového priestoru na meracom počítači s operačným systémom Linux. Toto riešenie je jednoduché a využíva funkcie dostupné v operačnom systéme. V systéme Windows je to zdieľanie súborov a v systéme Linux použitie dostupných balíčkov **smbclient** a **smbfs**, príkazu **mount** a úpravy konfiguračných súborov mountovania.

Verzia Linux

Zdieľané dáta budú umiestnené na disku meracieho počítača s operačným systémom Linux. Mountovanie bude prebiehať na meracom počítači s operačným systémom Windows pomocou **VBScript**[11] skriptu alebo použitím nástroja **Map Network Drive** v Explorer Windows. V systéme Linux bude nutné nainštalovať balíček **Samba** zabezpečujúci zdieľanie diskového priestoru, ktorý systém Windows dokáže rozpoznať a namountovať. Navrhované riešenie si nevyžaduje inštaláciu ďalších aplikácii.

4.3.3 Možnosti spúšťania procesov na vzdialenom meracom počítači

Základom pre možnosť spúšťania procesov na vzdialenom meracom počítači je, že operačný systém obsahuje služby zabezpečujúce vzdialený prístup pre prácu s procesmi. Systém Linux zahrňuje takéto služby. Najčastejšie sú používané **RSH**[12] a **SSH**[13]. Systém Windows dané služby podporuje menej. Existujú aplikácie pre systém Linux, ktorými je možné spúšťať procesy na vzdialenom systéme Windows, no opačným smerom nefungujú. Vhodnejšie by bolo, aby komunikácia oboma smermi prebiehala rovnakým spôsobom. Teda možnosť nainštalovať službu SSH do systému Windows by zaistila rovnaké podmienky na oboch systémoch. Riešením je **Cygwin** s nainštalovaným balíčkom **OpenSSH** pre Windows. Cygwin v systéme Windows vytvára prostredie Linuxu pomocou dynamickej knižnice cygwin1.dll a tým dovoľuje získať podstatnú funkcionalitu systému Linux. Balíček OpenSSH ponúka podporu SSH, čím sa dá SSH používať na systéme Windows rovnako ako na systéme Linux.

[10] je to proces prípravy súborového systému k použitiu operačným systémom, typicky načítaním istej indexovanej datovej štruktúry z úložiska do pamäte v predstihu

[11] Visual Basic Script

[12] Remote Shell vykonáva príkazy na vzdialenom počítači nezabezpečeným spôsobom[1]

[13] Secure Shell je populárna, výkonná, softwarovo založená koncepcia bezpečnosti siete. Dáta posielané po sieti sú vždy automaticky šifrovane. Na strane príjemcu dát sú následne automaticky dešifrované. Výsledkom je transparentné šifrovanie z pohľadu užívateľa. SSH používa moderné a bezpečné šifrovacie algoritmy a je dostatočne efektívny pre použitie s aplikáciami, ktoré si vyžadujú vysokú bezpečnosť.[1]

Aplikácie vykonávajúce merania sú napísané v jazyku C++ a sú kompilované a spúšťané v prostredí ROOT. Aby dané aplikácie mohli spúšťať procesy na vzdialenom počítači, musí byť v nich implementovaná komunikácia prostredníctvom SSH. K tomu bude možné použiť knižnicu implementujúcu klienta SSH, pomocou ktorého sa dá komunikovať so serverom SSH. NetSieben SSH Library od výrobcu NetSieben Technologies, ktorá je dostupná aj zdarma, ponúka túto možnosť. Funkcie knižnice sú jednoducho použiteľné pri vytváraní vlastných aplikácií, v ktorých je potrebné používať SSH. Knižnica je platformovo nezávislá, teda funguje ako v systéme Windows tak aj v systéme Linux.

Ďalšou možnosťou je použiť metódu `gSystem->Exec()` prostredia ROOT, ktorá dovoľuje volať príkazy prístupné v príkazovom riadku. Inštalácia aplikácie Cygwin pridáva s balíčkom OpenSSH možnosť používať príkaz SSH. Tým je možné v zdrojovom kóde použiť túto metódu a volaním príkazu SSH spustiť príkaz na vzdialenom meracom počítači.

4.4 Zálohovanie, zdieľanie a správa verzií

4.4.1 Zálohovanie

Predmetom zálohovania sú dáta, ktoré sú výsledkom meraní. Tieto výsledky sa istý čas archivujú a preto je ich zálohovanie žiadané. Jedná sa o dáta o veľkosti rádovo GB. Zálohovanie by sa malo vykonávať pravidelne a za podmienky, že prebehli merania a sú k dispozícii nové dáta k zálohovaniu. Je dôležité vykonávať zálohovanie v čase, keď v laboratóriu neprebieha žiadne meranie alebo ho vykonávať tak, aby procesor nebol preťažovaný na úkor meraní.

Ako prvá možnosť zálohovania dát bol uvažovaný externý disk pripájaný cez rozhranie USB. Výhodou tohto riešenia je hlavne rýchlosť (rádovo desiatky MB za sekundu), nižšia záťaž pre procesor a prípadna mobilita, disk je možné preniesť a pripojiť k ľubovoľnému počítaču. Nevýhodou je však nutnosť dáta zálohovať ručne alebo použiť zálohovací software. Ručné zálohovanie si vyžaduje osobu, ktorá ju bude vykonávať, čo je ťažko realizovateľné. Zálohovací software splňajúci požadované vlastnosti nemusí byť voľne šíriteľný, čo si vyžaduje ďalšiu peňažnú investíciu do zakúpenia softwaru. Za nevýhodu by sa tiež dala považovať veľkosť disku, ktorá je 500GB.

Ďalšou možnosťou je zálohovanie na vzdialené úložisko ako server, diskové pole. Výhodou je hlavne veľký úložný priestor, rádovo to môžu byť jednotky, desiatky az stovky TB. Nevýhodou je rýchlosť prenosu, závislosť na stabilnosti a spoľahlivosti siete.

Realizovateľnou alternatívou by mohlo byť zapojenie diskov do poľa **RAID 1**.

RAID je skratka pre Redundant Array of Independent Disks. Použitím diskového radiča sa špeciálnymi funkciami a viacerými pevnými diskami zároveň dajú získať určité špeciálne vlastnosti. Sú to predovšetkým rýchlosť, spoľahlivosť a ich kombinácia.

RAID 1 (mirroring / zrkadlenie) - Na dva samostatné disky sú ukladané úplne rovnaké dáta, druhý disk je tak vernou kópiou prvého disku. Kompletná redundancia dát je pre prípad poruchy jedného z diskov, užívateľ neprichádza o dáta. Účelom RAID 1 je spoľahlivosť. Celková kapacita poľa RAID 1 je rovná kapacite jedného pevného disku, teda polovici súčtu kapacity dvoch pevných diskov (za predpokladu rovnakých veľkostí). Tým, že sú dáta v RAID 1 zapisované viackrát, môže to spôsobiť pokles rýchlosti, ale čítanie dát sa môže vykonávať paralelne, čím sa zvýši rýchlosť čítania.[9]

RAID 1 sa hodí pre všetky prípady, kedy sú na disk ukladané dôležité dáta. Administrácia RAIDu 1 je jednoduchá, kedykoľvek môže byť jeden s diskov odobraný, kedykoľvek môže byť synchronizovaný stav. Dopady na výkon sú závislé na spôsobe implementácie v radiči. Teoreticky je rýchlosť zápisu pomalšia, pretože je nutné rovnaké dáta zapísať na dva disky. Radič ale môže využívať oneskorenie zápisu a na druhý (mirror - zrkadlový disk) zapisovať až v okamihu menšieho vyťaženia. Pokiaľ sa druhý disk využívá len k zápisu, je rýchlosť čítania rovnaká ako u jedného pevného disku, opäť tu ale pripadajú v úvahu dve modifikácie. Prvá spočívá v čítaní z oboch diskov zároveň a porovnávania čítaných dát (v prípade rozdielov je nahlásená chyba čítania). V takomto prípade je rýchlosť čítania tou pomalšou z dvoch diskov. Druhá naopak číta z oboch diskov iné dáta, takže pole sa pri čítaní chová ako RAID 0, to má pozitívny vplyv na sekvenčné čítanie, kedy sa jeho rýchlosť výrazne zvyšuje.

Pre zaobstaranie si RAID poľa sú potrebné dve veci: RAID radič a párny počet pevných diskov. Dva pevné disky sú otázkou ceny. Je nutné poznamenať, že pri výbere značky je vhodné preštudovať porovnávacie testy pevných diskov zapojených do RAIDu. Pokiaľ ide o RAID radič, existujú dva varianty. Prídavný radič do PCI slotu, na PCI napojený radič integrovaný na základnej doske, alebo v čipovej sade integrovaný radič. Výhodou prídavných radičov často bývajú výrazne lepšie možnosti konfigurovania a to, že do systému pridajú ďalšie IDE/Serial ATA porty pre pevné disky a nezaberú tak pozície z čipovej sady. Nevýhodou naopak je, že PCI zbernica je pomalá a výkon takého radiča často nie je optimálny, obzvlášť pri napojení dalších zariadení na PCI zbernici ako napríklad zvukových kariet, TV tunerov nebo grafických kariet. Radič v čipsete je na tom presne naopak. Jeho možnosti niekedy nebývajú tak veľké, ale má veľmi rychlé nezdieľané spojenie s ostatnými časťami čipovej sady (t.j. i rychlé spojenie s pamäťou). Nevýhodou všetkých lacných RAID radičov je, že niektoré ich funkcie sú závislé na výkone procesora a že nemajú žiadnu vyrovnávaciu paměť RAM pre urýchlenie práce s diskami. Profesionálne riešenie pre

zbernicu PCI-X, PCI-Express či 64bitovú PCI je samozrejme omnoho lepšie, tiež ale stojí rádovo tisíce až desaťtisíce korún. U lacných radičov do PCI a v čipovej sade integrovaných radičov nízkej výkonnosti odpovedá nízka predajná cena.

Pre stavbu RAID poľa je ideálne rozhranie Serial ATA. To predovšetkým z toho dôvodu, že káble Serial ATA lepšie odolávajú rušeniu (sú tienené) a disk je na samostatnom kanále, teda poskytuje optimálny výkon. Použitie staršieho rozhrania IDE je síce tiež možné, ale tam už je nutné dávať pozor na zapojenie diskov na jednotlivé kanály. Optimálne je mať každý disk nastavený ako Master, teda na samostatnom kanále.

Prípad poruchy

Pokiaľ sa jeden z pevných diskov pokazí, je užívateľ pri štarte počítača ihneď informovaný, že pole nie je funkčné. Zobrazí sa ponuka a očakáva sa, že užívateľ rozhodne, čo ďalej. Možnosti sú:

- Power off and check the failed drive - Vypnúť počítač a skontrolovať chýbajúci pevný disk.
- Destroy the Mirroring Relationship - Odstraní záznam o tom, že disk je viazaný v RAID 1 poli. Dáta na disku zostanú zachované, pri ďalšom štarte sa už bude predpokladať, že disk je samostatný a žiadnu druhú kópiu k sebe nemá. Pokiaľ po odstranení záznamu o poli bude vložený do systému druhý disk poľa, opäť vyskočí táto ponuka. Inými slovami na každý disk z poľa je viazaná informácia, že má k sebe kolegu.
- Choose replacement drive and rebuild - Umožní zvoliť náhradný disk za pokazený a vytvoriť RAID 1 pole na ňom.
- Continue to boot - Nič nezmení a umožní bootovanie do operačného systému. Pretože RAID 1 je zrkadlenie, je možné nabootovať s jedným diskom. Je nutné si však uvedomiť, že nabootovaním z jedného disku (a nie z poľa ako celku) dôjde k rozsynchronizovaniu údajov na diskoch. Operačný systém v priebehu bootovania ukladá dáta na disk, napríklad protokoly o spustení systémových služieb. Ak bude vložený druhý disk, je potrebné opäť spustiť synchronizáciu, pretože v opačnom prípade sa dáta na diskoch nebudú zhodovať a funkcia zrkadlenia bude narušená.

RAID 1 pole neochráni pred chybami spôsobenými softwarom, teda predovšetkým pred vírmi a červami, chybami programov a operačného systému, kolapsami súborového systému. Pokiaľ je však k dispozícii stabilný hardware, správne ovladače a dodržujú sa bezpečnostné zásady (inštalácia záplat, firewall), je riziko straty dát výrazne redukované.

riešenie	výhody	nevýhody
extérny disk	rýchlosť, cena, mobilita	diskový priestor, funkcia zálohovania
vzdialené úložisko	diskový priestor, zálohovanie	cena, rýchlosť
RAID 1	rýchlosť čítania, jednoduchá administrácia, spoľahlivosť	možné spomalenie zápisu

4.4.2 Zdieľanie a správa verzií

V súčasností je dostupných mnoho nástrojov na správu verzií. Jedným zo základných je **RCS**[14], ktorý je často súčasťou distribúcií Linuxu. Taktiež je vo verzii pre systém Windows. RCS slúži na zaznamenávanie zmien v súboroch. Je možné zistiť ako vyzeral súbor v istom čase, napríklad pred tým, než niečo prestalo fungovať, kto túto zmenu previedol a mnoho ďalšieho. Základné ovládanie pozostáva z dvoch príkazov. Jedným príkazom sa súbor a zmeny v ňom ukládajú spolu s pripojeným komentárom zmien, zároveň sa súbor uzamkne, až pokiaľ ho druhým príkazom užívateľ neodomkne k tomu, aby ho mohol pozmeniť.

Ďalším nástrojom je **CVS**[15]. Funguje podobne ako RCS ale má viac vlastností. Napríklad jeden súbor môže upravovať naraz viacero užívateľov. Pracuje ako klient/server aplikácia, čím dovoľuje mať uloženú históriu zmien centrálne na serveri a klientské stanice si udržujú kópiu, ktorú môžu lokálne upravovať. Následne prebieha aktualizácia smerom na server.

Na základe CVS vznikol nástroj **Subversion**. Subversion funguje na báze klient-server. Serverová časť je vlastne súborový server, ktorý si pamätá všetky zmeny súborov a štruktúr adresárov. Umožnuje ukladať akékoľvek typy súborov, či už binárne alebo textové. Zjednodušuje spoluprácu viacerých ľudí na spoločných súboroch. Uchováva si všetky verzie súborov, takže je jednoduché kedykoľvek sa vrátiť k nejakej staršej verzii. Serverová časť vybavuje požiadavky klientov. Klientska časť poskytuje nástroje pre prácu s verziami priamo v pracovnom adresári a komunikáciu so serverovou časťou. Existuje niekoľko klientskych nástrojov, od príkazového riadku, cez webové rozhranie až po nástroje integrované do GUI operačného systému. Záleží na tom, čo ktorému užívateľovi vyhovuje najviac. Subversion je distribuovaný pod licenciou, ktorá umožnuje bezplatné komerčné použitie. Existujú verzie pre všetky dôležité platformy.

[14]Revision Control System
[15]Concurrent Versions System

Git je distribuovaný systém na správu verzii zameraný na rýchlosť, efektívnosť a použitie pre veľké projekty. Git uchováva všetky verzované súbory a históriu zmien lokálne pre daného užívateľa, nemá centrálny server. Zmeny smerom k ostatným užívateľom sú publikované cez HTTP, FTP, rsync, alebo Git protokol. Je vo verzii pre systém Windows aj Linux.

Mercurial je tak ako Git distribuovaným systémom na správu verzií. Primárne je ovládaný z príkazového riadku. Mercurial bol navrhnutý na vysoký výkon, škálovateľnosť, decentralizáciu, plne distribuovanú spoluprácu užívateľov, robustnú správu textových a binárnych súborov, pričom stále ostáva konceptuálne jednoduchý. Obsahuje integrované webové rozhranie. Je vo verzii pre systém Windows aj Linux.

Bazaar je ďalším distribuovaným systémom na správu verzii. Podporuje spôsob práce od samostatného cez centralizovaný až po decentralizovaný s veľkou variáciou medzistupňov. Má jednoduché užívateľské rozhranie a je platformovo nezávislý.

Vzľadom na to, že na vývoji aplikácii v laboratóriu sa podieľa malá skupina ľudí, bude vhodné použiť jednoduchší a bežne používaný systém na správu verzii napríklad Bazaar alebo Subversion. Kedže všetky spomenuté systémy sú voľne šíriteľné a medzi niektorými je možná migrácia, dovoľuje to vyskúšať viacero systémov.

Kapitola 5

Realizácia navrhnutých riešení

5.1 Mechanizmus zamykania súboru

Vytvorenie zámku musí byť atomické. Na zaistenie atomickosti tejto operácie bude vhodné použiť tradičné Unixové rozhranie pre zápis/čítanie so súboru deklarované v hlavičkovom súbore `fcntl.h` (pre operačné systémy Unix a Linux) alebo `io.h` (pre operačný systém Windows).

Pred tým ako bude môcť proces zapisovať do súboru voltmeter.ez, musí obdržať zámok nasledovne:

```
#include <fcntl.h>  // pre možnosť použitia open()
#include <cerrno>   // pre možnosť použitia errno
#include <cstdio>   // pre možnosť použitia perror()
...
int fd;
...

int getlock()
{
...
   fd=open("voltmeter.lck", O_WRONLY | O_CREAT | O_EXCL);
...
}
```

V prípade kompilovania kódu v systéme Windows je potrebné nahradiť v `#include <fcntl.h>` hlavičkový súbor `fcntl.h` súborom `io.h`. Funkciu `open()` a príznaky `O_WRONLY`, `O_CREAT` a `O_EXCL` je potrebné prefixovať znakom _ .

Ak volanie funkcie `open()` uspeje, návratovou hodnotou je desktriptor, ktorý je malá kladná hodnota identifikujúca súbor. V prípade neúspechu funkcia vráti hodnotu -1 a priradí odpovedajúci kód chyby do globálnej premennej

`errno`. Príznak `O_CREAT` indikuje v prípade, že súbor voltmeter.lck neexistuje, to aby bol funkciou `open()` vytvorený. Príznak `O_EXCL` zaisťuje, že volanie funkcie je atomické. Ak súbor voltmeter.lck už existuje, funkcia `open()` zlyhá a nastaví premennú `errno` na hodnotu `EEXIST`. Tento spôsob garantuje, že len jeden proces v danom čase môže držať zámok.

Overenie návratového kódu funkcie `open()`:

```
int getlock() // pri úspechu vráti deskriptor zámku
{
...
   if (fd<0 && errno==EEXIST)
   {
      // súbor už existuje, iný proces drží zámok
      cout<<"the file is currently locked; try again later";
      return -1;
   }
   else if (fd < 0)
   {
      // perror() pripojí slovný popis súčasnej hodnoty errno
      perror("locking failed for the following reason");
      return -1;
   }
   // na tomto mieste kódu proces už vlastní zámok
   return fd;
}
```

Keď už proces vlastní zámok, môže bezpečne modifikovať súbor voltmeter.ez. Po ukončení práce s externým zariadením a poslednom zápise do súboru voltmeter.ez, proces musí uvoľniť zámok:

```
close(fd);
remove("voltmeter.lck");
```

V tejto chvíli je súbor voltmeter.ez odomknutý a externé zariadenie voľné pre použitie iným procesom.

Mechanizmus zamykania je funkčný a otestovaný. Ostáva ho implementovať do jestvujúcich aplikácii a začať používať pri návrhu nových.

5.2 Zdieľanie diskového priestoru

Navrhované riešenie v časti 3.3.2 pôvodne uvažovalo použitie iba jednej verzie zdieľania diskového priestoru. Pri realizácii riešenia došlo k rozšíreniu požiadavky, aby procesy na danom meracom počítači mali vždy prístup do adresára so súbormi pre prácu s externými zariadeniami, ktoré sú pripojené k danému meraciemu počítaču. Týmto spôsobom sa predíde problému, že vzdialený diskový priestor nie je možné namountovať napríklad z dôvodu nefunkčnej siete, chyby alebo technickej poruchy na vzdialenom meracom počítači.

Obe verzie riešenia boli v laboratóriu realizované podľa postupu popísanom nižšie, odskúšané a sú funkčné. Vo verzii Windows bol použitý spôsob mountovania pomocou CIFS. Vo verzii Linux bol v systéme Windows na mountovanie použitý Map Network Drive, v ktorom je možné nastaviť mountovanie po reštarte systému, takže použitie skriptu nie je potrebné.

Postup konfigurácie verzie riešenia Windows je nasledovný. Najprv je nutné v systéme Windows povoliť zdieľanie adresára alebo celého disku. To je, čo sa týka konfigurácie v systéme Windows všetko. V systéme Linux je nutné najprv overiť, či jadro podporuje CIFS a SMBFS. V prípade, že nie, je nutné nakonfigurovať a prekompilovať jadro. Ak je jadro v poriadku, je potrebné doinštalovať balíčky smbfs a smbclient, ktoré sú potrebné pre mountovanie. Sú dva spôsoby ako mountovať disk, tradičný **SMBFS** a nový **CIFS**. SMBFS[1] je súborový systém mountovateľný do systému Linux. Začína byť zastaralý, môže teda nastať, že v novších systémoch nebude fungovať. Príkaz na mountovanie vyzerá nasledovne:

```
mount -t smbfs -o username=cr2,password=cr2_pass
   //ipnp03/devices /mnt/
```

Kde `cr2` je meno užívateľa na systéme Windows, `cr2_pass` je heslo užívateľa `cr2`, `ipnp03` je NetBIOS meno počítača so systémom Windows, `devices` je zdieľaný adresár obsahujúci súbory s koncovkou .ez a `mnt` je cieľový adresár pre mountovanie na počítači so systémom Linux.

CIFS[2] je protokol pre zdieľanie súborov založený na SMBFS a postupne ho nahrádza, príkaz na mountovanie vyzerá nasledovne:

```
mount -t cifs //ipnp/devices /mnt/
   -o username=IPNP_VDG/cr2,password=cr2_pass
```

Kde `IPNP_VDG` je doménové meno alebo názov pracovnej skupiny lokálnej siete, ostatné parametre sú rovnaké ako v predchádzajúcom prípade.

[1] Server Message Block File System
[2] Common Internet File System

Týmito príkazmi sa však mountuje len jednorázovo, po reštarte systému nebude v adresári `/mnt/` nič namountované. Pre zaistenie automatického mountovania, ktoré bude fungovať aj po reštarte systému, sú dve varianty riešenia.

Prvou je úprava súboru `/etc/fstab`, v ktorom sa nachádzajú statické informácie o súborových systémoch, podľa ktorých sa pri každom štarte systému automaticky namountujú. Riadok, ktorý je nutné doplniť do súboru `/etc/fstab` a odpovedá príkazom uvedeným vyššie, vyzerá nasledovne:

```
//ipnp03/devices /mnt/ smbfs fmask=0664,dmask=0775,
   username=cr2,password=cr2_pass 0 0
```

alebo

```
//ipnp03/devices /mnt/ cifs fmask=0664,dmask=0775,
   username=cr2,password=cr2_pass 0 0
```

Tento riadok však obsahuje heslo, ktoré si môže ľubovoľný užívateľ prečítať. Pre zvýšenie bezpečnosti sa užívateľské meno a heslo môžu umiestniť do samostatného súboru napríklad `/etc/smbpass`, ktorému sa ako vlastník nastaví užívateľ root a prístupové práva súboru sa upravia tak, aby len vlastník mohol pracovať so súborom. V riadku sa následne nahradí nasledujúci reťazec:

```
username=cr2,password=cr2_pass
```

reťazcom

```
credentials=/etc/smbpass
```

Druhou variantou je použitie automountera **AutoFS**. Jedná sa o modul jadra systému Linux, ktorý automaticky mountuje súborové systémy ak je potrebné s nimi pracovať a automaticky ich odpája, ak sa s nimi nepracuje stanovený čas. Po nainštalovaní AutoFS (v prípade, že nainštalovaný nebol) je nutné upraviť dva konfiguračné súbory `/etc/auto.master` a `/etc/auto.mnt`. Súbor `/etc/auto.master` je hlavným konfiguračným súborom AutoFS. V ňom je nutné doplniť nasledujúci riadok:

```
/mnt /etc/auto.mnt --timeout=60
```

Ten informuje AutoFS, že má podľa konfiguračného súboru `/etc/auto.mnt` mountovať zdieľaný adresár systému Windows do `/mnt`. Súbor `/etc/auto.mnt` môže obsahovať jeden s nasledujúcich dvoch riadkov:

```
mnt -fstype=smbfs,credentials=/etc/smbpass //ipnp03/devices
```

alebo

```
mnt -fstype=cifs,file_mode=0664,dir_mode=0775,
   credentials=/etc/smbpass //ipnp03/devices
```

Postup konfigurácie verzie riešenia Linux je nasledovný. Po naištalovaní balíčka Samba je potrebné v konfiguračnom súbore `/etc/samba/smb.conf` nastaviť meno pracovnej skupiny lokálnej siete:

```
workgroup = IPNP_VDG
```

Cestu k adresáru a povolenie čítania a zápisu do adresára, ktorý má byť zdieľaný:

```
[share disk]
comment = Devices Folder
path = /devices
public = yes
read only = no
create mask = 0777
directory mask = 0777
```

Ďalej je potrebné nastaviť a povoliť pre Sambu užívateľa systému Linux, pod akým sa zo systému Windows bude k zdieľanému adresáru pripájať, pomocou príkazu smbpasswd:

```
smbpasswd -L -a depfet
smbpasswd -L -e depfet
```

Tým je Samba nakonfigurovaná. V systéme Windows je možne namountovať adresár pomocou nástroja Map Network Drive v Explorer Windows (Obrázok 1), kde je možné zvoliť, aby sa mountovanie vykonalo aj po reštarte systému. Alebo pomocou VBScript skriptu, ktorý môže vyzerať nasledovne:

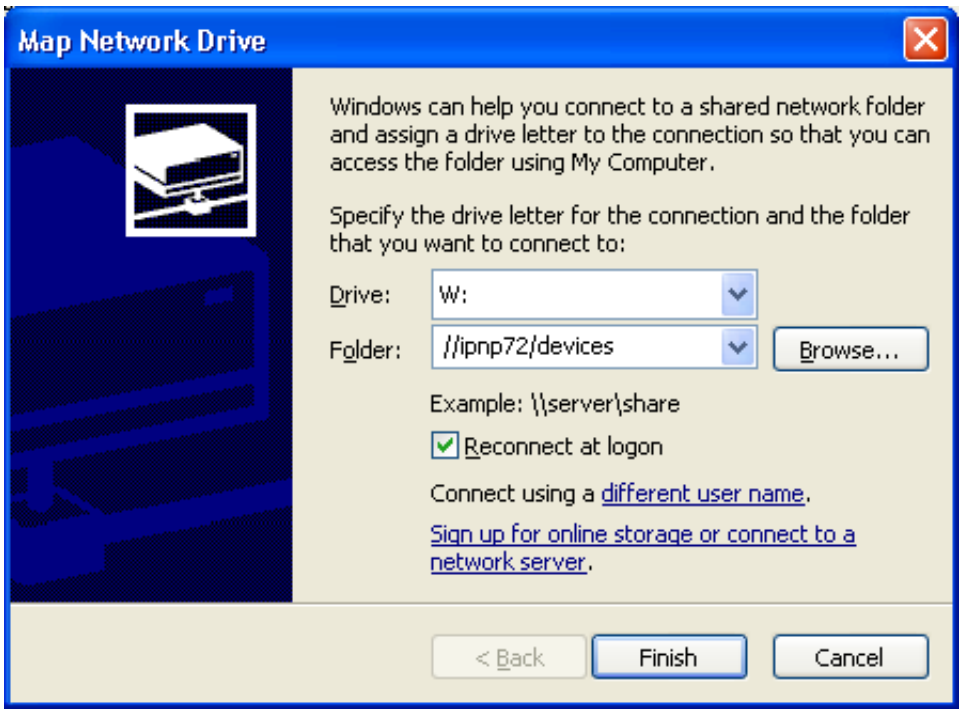
```
Dim objNetwork
Set objNetwork = WScript.CreateObject("WScript.Network")
strLocalDrive = "Z:"
strRemoteShare = "\\ipnp72\devices"
strPer = "FALSE"
strUsr = "username"
strPas = "password"
objNetwork.MapNetworkDrive strLocalDrive, strRemoteShare,
strPer, strUsr, strPas
```

Kde `strUsr` je meno užívateľa na systéme Windows, `strPas` je heslo užívateľa `user`, `ipnp72` je meno počítača so systémom Linux, `devices` je zdieľaný

adresár obsahujúci súbory s koncovkou .ez, `strLocalDrive` je cieľový adresár pre mountovanie na počítači so systémom Windows a `strRemoteShare` je mountovaný adresár na počítači so systémom Linux.

Skript obsahuje heslo, ktoré je tak voľne prístupné. Pre zvýšenie bezpečnosti je možné užívateľské meno a heslo zadávať do dialogového okna, ktoré bude vyvolané funkciou `InputBox()`.

```
strUsr = InputBox("Username for Z: drive:")
strPas = InputBox("Username for Z: drive:")
```

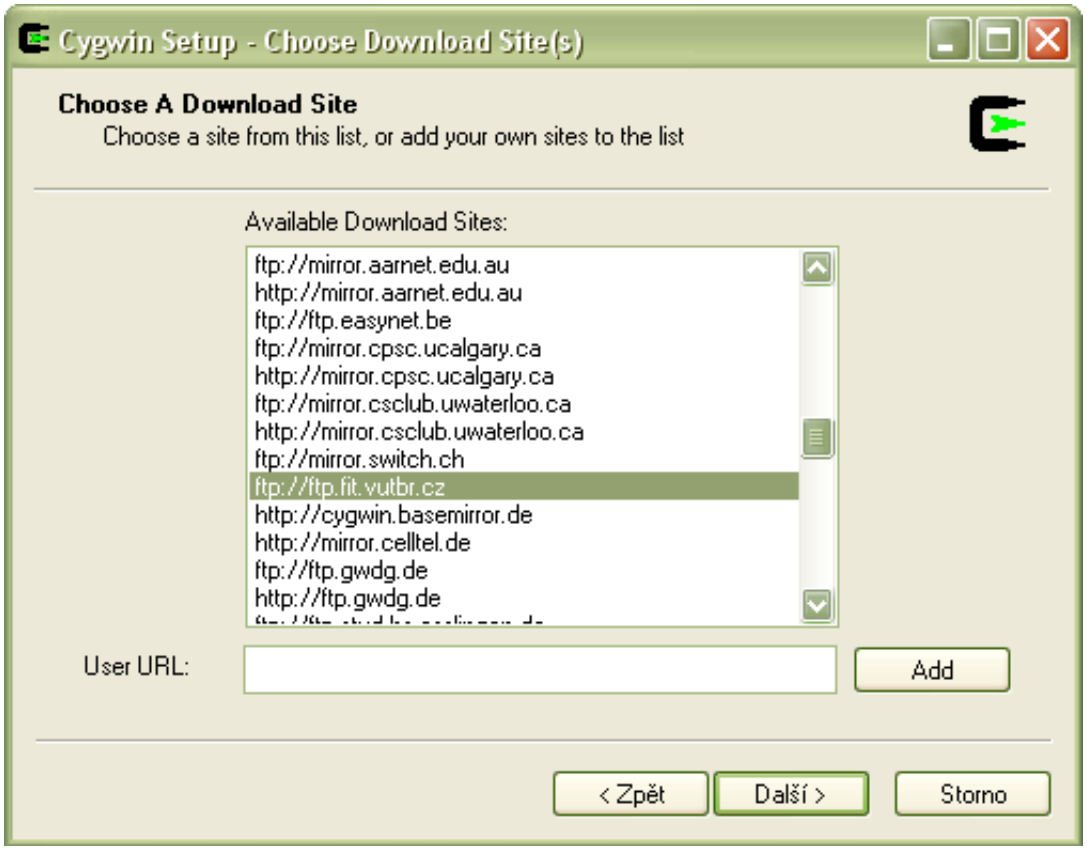


Obrázok 1

5.3 Spúšťanie procesov na vzdialenom meracom počítači

Operačný systém Linux priamo podporuje používanie služby SSH klienta a SSH servera. SSH klient je väčšinou súčasťou inštalácie, SSH server je v prípade niektorých distribúcií potrebné doinštalovať. Pre operačný systém Windows existujú aplikácie poskytujúce priamo len SSH klienta, napríklad PuTTY. Sprístupnenie služby SSH servera a SSH klienta zároveň v systéme Windows

je možné nainštalovaním a nakonfigurovaním aplikácie Cygwin s balíčkom OpenSSH nasledujúcim postupom. Je nutné stiahnuť inštalačný súbor setup.exe a uložiť ho do adresára `C:\cygwin`. Následne spustiť setup.exe. V sprievodcovi inštaláciou nie je nutné meniť žiadne prednastavené vlastnosti inštalácie až do chvíle, kedy je potrebné vybrať názov servera (prípadne zadať názov servera, ktorý nie je v zozname), z ktorého majú byť stiahnuté potrebné balíčky (Obrázok 2).



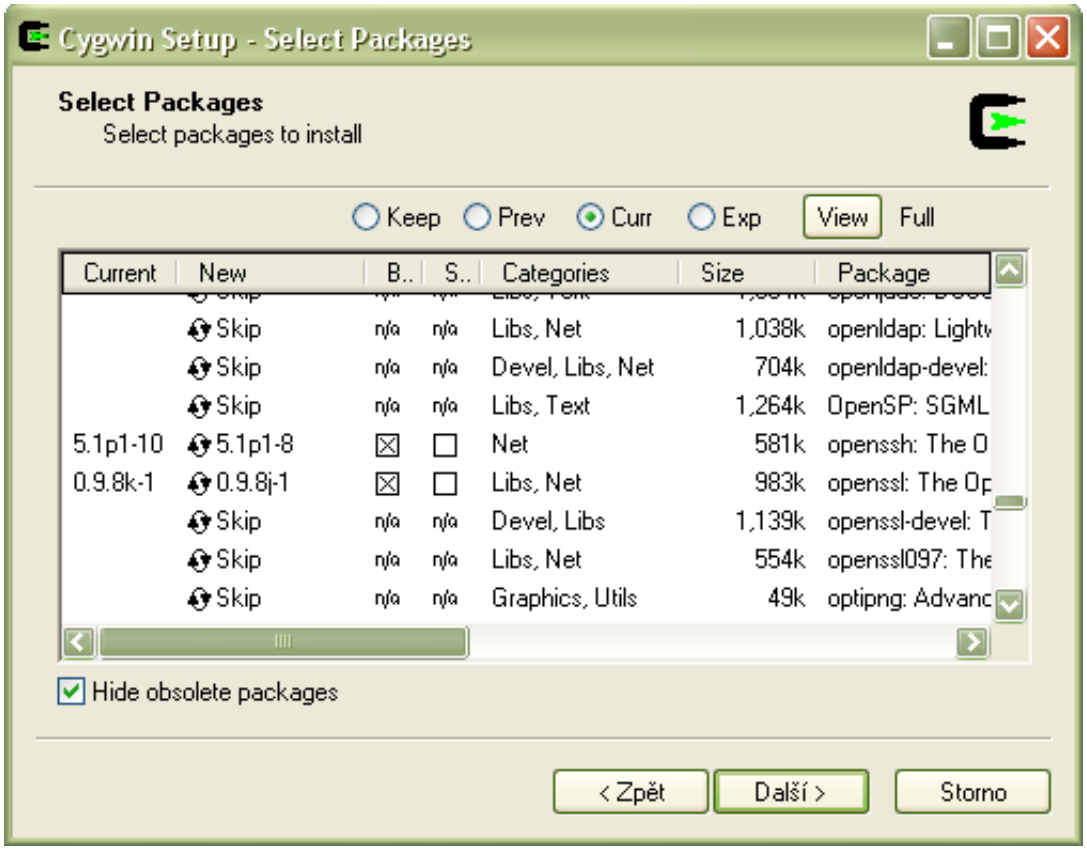
Obrázok 2

Vyberie sa server, ktorý sa nachádza v blízkej lokalite, očakáva sa tým dobrá konektivita. Po načítani ponuky balíčkov zo servera je nutné pridať balíčky OpenSSH a tcp_wrappers kliknutím na položku s hodnotou skip v riadku, ktorý reprezentuje daný balíček (Obrázok 3). Ak sú všetky potrebné balíčky vybraté, pokračovaním v inštalácii dôjde k ich stiahnutiu a nainštalovaniu, čím končí celková inštalácia a ostáva nakonfigurovanie aplikácie Cygwin, SSH démona a operačného systému Windows.

V systéme Windows je potrebné nastaviť systémovú premennú PATH a vytvoriť novú premennú CYGWIN (Obrázok 4). Premennú PATH je potrebné upraviť tak, že na koniec jej hodnoty sa pridá nasledujúci reťazec:

```
;C:\cygwin\bin
```

Nová premenná sa nazve CYGWIN a ako hodnota sa jej nastaví `ntsec tty`. Posledné, čo je potrebné nastaviť, je povolenie portu 22 vo firewalle.



Obrázok 3

V aplikácii Cygwin je nutné spustiť príkaz `ssh-host-config -y` pre nainštalovanie SSH démona. Aby daný príkaz úspešne nakonfiguroval SSH démona, musia byť súborom `/etc/passwd` a `/etc/group` zmenené prístupové práva príkazom `chmod` z parametrom `+r` a adresár `/var` musí mať nastavené prístupove práva príkazom `chmod` s parametrom `755`. Príkazom `net start sshd` sa spustí SSH démon a bude sa spúšťať aj po reštarte systému. Príkazmi `mkpasswd -cl > /etc/passwd` a `mkgroup --local > /etc/group` sa harmonizuje Cygwin s užívateľmi a skupinami systému Windows. Týmto je konfigurácia ukončená a je možné vzdialene sa pripojiť pomocou SSH.

Obrázok 4

Pri testovaní metódy `gSystem->Exec()` prostredia ROOT nastal problém, pretože volanie príkazu SSH si vyžaduje zadanie hesla, ktoré príkazu nie je možné zadať ako parameter. Je to nevhodné, pretože počas merania môže dané volanie nastať mnohokrát a meranie často prebieha aj hodiny. Odhliadnuc od faktu, že by na priebeh merania musela dohliadať osoba, je nereálne, aby opakovane zadávala heslo.

Metóda, ktorou je možné tento problém riešiť, sa nazýva autentifikácia verejným kľúčom. Je to metóda používajúca šifrovanie verejným kľúčom na overenie identity klienta. Na prístup k účtu na počítači s SSH serverom musí klient dokázať, že vlastní tajomstvo t.j. privátny protipól autorizovaného verejného kľúča.[1]

Aby sa táto metóda mohla používať a nahradila by zadávanie hesla, je nutné vygenerovať pár kľučov, privátny a verejný. K tomu slúži program `ssh-keygen`. Spustením príkazu `ssh-keygen` sa vygeneruje privátny a verejný kľúč, ktorým je možné nastaviť cestu, kde majú byť uložené. Doporučuje sa zvoliť prednastavenú hodnotu cesty, prípadne zmeniť len názov súboru v nej. Ďalším krokom je umiestnenie kľúčov do správnych adresárov. Privátny kľúč je potrebné umiestniť do adresára `/home/userL/.ssh/`, ktorý patrí užívateľovi `userL` a je na lokálnom meracom počítači, z ktorého sa bude SSH klientom pripájať na SSH server vzdialeného meracieho počítača. Obsah súboru, ktorý predstavuje verejný kľúč, je potrebné pripojiť do súboru `/home/userR/.ssh/authorized_keys` na vzdialenom meracom počítači, kde bude spustený SSH server a kam sa bude pripájať užívateľ `userL` ako užívateľ `userR`.

Po tejto konfigurácii je možné spúšťať príkazy na vzdialenom meracom počítači pomocou SSH bez zadávania hesla.

V prípade potreby zvýšenia bezpečnosti je možné pár kľúčov chrániť použitím passphrase. Tým vznikne potreba zadávať passphrase pri autentifikácii verejným kľúčom. Táto situácia sa dá vyriešiť použitím programu `ssh-agent`. Ten si udržuje privátne kľúče a poskytuje autentifikačné služby SSH klientom. Po prihlásení do operačného systému stačí agentom načítať privátny kľúč a zadať passphrase a SSH klienti už nebudú žiadať zadávanie passphrase.

Spúšťanie príkazov a aplikácii na pozadí na vzdialenom meracom počítači bolo úspešne otestované na oboch systémoch. To dáva dobrý predpoklad pre ďalšie použitie tejto funkcionality a nové možnosti pri meraniach.

5.4 Zálohovanie dát

Realizácia poľa RAID 1 bola zvolená ako najvhodnejšie riešenie, ktoré ponúka spoľahlivosť, rýchlosť čítania, jednoduchú administráciu a prijateľnú obstarávacu cenu.

Nasledovným postupom je možné vytvoriť pole RAID 1 použitím dvoch diskov. Za predpokladu, že už sú dva pevné disky pripojené na rozhranie Serial ATA, je možné zapnúť počítač. Pri bootovaní sa každý radič prezentuje vlastnou obrazovkou (obvykle tesne po obrazovke BIOSu), kde umožňuje nakonfigurovanie poľa. Za normálnych okolností sa dva Serial ATA disky hlásia BIOSu ako samostatné zariadenia s možnosťou bootovania z ľubovoľného z nich. Stlačením tabulátora se prejde do konfiguračnej utility RAID BIOSu. Pokiaľ by neboli pripojené dva disky, zostanú všetky ponuky neaktívne. RAID pole je možné vytvárať z dvoch diskov, v prípade že doska využíva čip PHY[3], až zo štyroch diskov. Je nutné uvedomiť si, ktorý disk je ktorý. Je možné skontrolovať sériové čísla. Dôležité je to preto, že pri synchronizácii je nutné

[3] **phy**sical layer chip - čip, ktorý vytvára rozhranie medzi čipovou sadou dosky a konektormi, pomocou kódovania dát

vedieť, z ktorého disku sa májú kam kopírovať dáta. Pole sa vytvára kliknutím na Create Array. Otvorí sa ďalšia ponuka v nej pri druhej položke zvolíme RAID 1 for data protection - následne sa v menu objaví Array Mode RAID 1 (Mirroring). Pokračuje sa voľbou Select Disk Drives, kde sa zvolia pevné disky určené k vytvoreniu RAID poľa. Tu je dôležité kliknúť najprv na disk, ktorý bude zdrojom dát - Source. Druhý disk potom bude zrkadliť stav zdroja, nazvaný je Mirror. V praxi je vcelku jedno, ktorý z diskov bude Source a ktorý Mirror, dôležité je to pre synchronizáciu, pretože dáta sa kopírujú zo Source na Mirror, nie naopak. Pokiaľ už na jednom disku sú dáta (napríklad nainštalovaný operačný systém), bude vhodné preniesť dáta z tohoto disku na druhý, prázdny disk. Preto sa volí disk s dátami ako Source a prázdny disk ako Mirror. Keby to bolo vykonané naopak, obsah disku s operačným systémom sa premaže dátami z prázdneho disku (teda nulami). Ak už je všetko nastavené, je potrebné spustiť synchronizáciu kliknutím na Start Create Process. Tým sa dáta z disku Source prekopírujú na disk Mirror (premažú pôvodný obsah na Mirror). Pokiaľ sa v budúcnosti niekedy pole rozhodí (obsah na Source nebude odpovedať obsahu na Mirror), je možné takto znovu vytvoriť presnú kópiu. Samotné kopírovanie diskov prebieha bit po bite sekvenčným čítaním z jednoho disku a sekvenčným zápisom na druhý disk. Po dokončení synchronizácie je ešte potrebné v hlavnom menu nastaviť pole ako Boot. Po reštarte systému detekuje RAID BIOS oba disky a vypisuje, že se jedná o Array 0 typu RAID 1. BIOS základnej dosky teraz neponúka bootovanie z jednotlivých pevných diskov, ale z poľa ako celku. Dva disky zlúčené do poľa sú potom i pre operačný systém viditeľné len ako pole, nie ako jednotlivé disky.

Kapitola 6

Záver

V práci bol preskúmaný stav vypočtovej infraštruktúry laboratória, popísané požiadavky a problémy v súčasnej prevádzke laboratória. Po prevedení analýzy boli navrhnuté riešenia, a nasledujúce z nich boli realizované:

- Pre možnosť vzdialeného prístupu do laboratória bola na počítače nainštalovaná aplikácia RealVNC, ku ktorej bol zakúpený potrebný počet licencií. Výber riešenia RealVNC bol urobený vzhľadom na skúsenosti pracovníkov laboratória s aplikáciu VNC a potrebou zabezpečenia spojenia.

- Na zálohovanie dát bolo zvolené pole RAID 1 na základe výhod popísaných v časti 3.4.1. Pracovníci laboratória pole RAID 1 do počítačov úspešne nainštalovali a otestovali.

- Obe navrhované verzie zdieľania diskového priestoru boli realizované a odskúšané v laboratóriu. Na každom systéme je možné pristupovať k zdieľanému adresáru druhého systému ako k lokálnym adresárom s možnosťou vykonávania operácii zápis, čítanie, vytvorenie a zmazanie súboru, čo splňuje požiadavky na funkcionalitu riešenia.

- Spúšťanie procesov na vzdialenom počítači sa ukázalo ako realizovateľné a bude postačovať na použitie príkazov a aplikácii, ktoré majú vykonať sadu operácii a výstup uložiť na disk alebo vypísať na štandartný výstup. Táto funkcionalita ponúka možnosť rozšíriť merania a pracovať s viacerími meracími počítačmi ako s celkom t.j. ovládať a vykonávať meranie z jedného počítača.

Nasledujúce riešenia neboli realizované alebo boli realizované len čiastočne, no v budúcnosti sa o nich uvažuje:

- Nedospelo sa k záveru, ktorý konkrétny systém na správu verzií bude nainštalovaný a bude sa v laboratóriu používať. Ale ako bolo naznačené

v časti 3.4.2, je dostupné veľké množstvo bezplatných sytémov s možnosťou migrácie medzi nimi, čo dovoľuje otestovať viac alternatív a vybrať si tú, ktorá bude najviac vyhovovať užívateľom. Vzhľadom na súčasné požiadavky by v úvahu pripadali systémy Bazaar alebo Subversion.

- Mechanizmus zamykania bol otestovaný na oboch operačných systémoch. Je implementovateľný v jazyku C++ a vhodný na použitie v kóde aplikácií. To odstraňuje problém organizácie prístupu procesov k externým zariadeniam. Účelom bolo navrhnúť a vyriešiť daný problém a keďže sa aplikácie vykonávajúce merania často menia a vznikajú nové, bolo by zbytočné do všetkých jestvujúcich aplikácii mechanizmus implementovať. Vhodnejšie bude, ak si osoby podieľajúce sa na vývoji aplikácií osvoja mechanizmus zamykania a pri písaní aplikácií ho v kóde použijú.

- Synchronizácia času na meracích počítačoch nebola otestovaná a ani realizovaná. Navrhované riešenie sa bude brať v úvahu pri budúcej realizácií.

Laboratórium sa rozvíja a potreby prevádzky majú svoje priority a dynamicky sa menia. Preto sa naskytá možnosť v tejto práci pokračovať aj v nadväzujúcom magisterskom štúdiu.

Literatúra

[1] Barrett D. J. , Silverman R.: *SSH, The Secure Shell: The Definitive Guide*, O'Reilly, 2001.

[2] Gook M.: *PC Hardware Interfaces: A Developer's Reference*, A-LIST Publishing, 2004.

[3] HW server: *Sériová linka RS-232*, 2009(online).

`http://hw.cz/rs-232`

[4] Javvin Technologies: *DiskShare 6.0*, 2009(online).

`http://www.javvin.com/diskshare.html`

[5] National Instruments Corporation: *VXI-MXI-2 User Manual*, August 1996 Edition, 2009(online).

`http://www.ni.com/pdf/manuals/371692a.pdf`

[6] RealVNC: *RealVNC remote control software*, 2009(online).

`http://www.realvnc.com`

[7] Scott C., Wolfe P., Erwin M.: *Virtual Private Networks, Second Edition*, O'Reilly, 1999.

[8] Silberschatz A., Galvin P. B., Gagne G.: *Operating System Concepts, 7th Edition* , John Wiley & Sons, 2005.

[9] Vadala D.: *Managing RAID on Linux* , O'Reilly, 2003.

[10] Wikipédia, Slobodná encyklopédia: *IEEE-488*, 2009(online).

`http://en.wikipedia.org/wiki/IEEE-488`

[11] Wikipédia, Slobodná encyklopédia: *Neprerušiteľný zdroj energie*, 2009(online).

`http://sk.wikipedia.org/wiki/UPS`

[12] Wikipédia, Slobodná encyklopédia: *Universal Serial Bus*, 2009(online).

`http://en.wikipedia.org/wiki/Universal_Serial_Bus`

[13] Wikipédia, Slobodná encyklopédia: *Virtual Network Computing*, 2009(online).

`http://en.wikipedia.org/wiki/Virtual_Network_Computing`