UNIVERZITA PARDUBICE
ÚSTAV ELEKTROTECHNIKY
A INFORMATIKY

DOHLEDOVÝ SYSTÉM NAD POČÍTAČI V UČEBNĚ

BAKALÁŘSKÁ PRÁCE

Autor práce: Nagyová Kateřina
Vedoucí práce: Mgr. Tomáš Hudec

UNIVERSITY OF PARDUBICE
INSTITUTE OF ELECTRICAL ENGINEERING AND INFORMATICS

COMPUTER SUPERVISION SYSTEM IN CLASSROOM

BACHELOR WORK

AUTHOR: Nagyová Kateřina
SUPERVISOR: Mgr. Tomáš Hudec

Vysokoškolský ústav: Ústav elektrotechniky a informatiky
Katedra/Ústav: Ústav elektrotechniky a informatiky
Akademický rok: 2006/2007

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Pro: Nagyová Kateřina

Studijní program: Informační technologie

Studijní obor: Informační technologie

Název tématu: Dohledový systém nad počítači v učebně

Zásady pro zpracování:

1. Vytvořte přehled technologií multiplatformní vzdálené (sdílené) pracovní plochy (jako je VNC).
2. Vyberte otevřené řešení (open source) a doplňte je o možnosti sledování více počítačů současně s možností zoomu. Sledovací systém má též umožňovat sledování nedetekovatelné na sledovaném počítači a také volitelné sledování bez možnosti zásahu a s možností zásahu do plochy.

Seznam odborné literatury:

- VNC [online]. URL: http://www.realvnc.com-documentation.html
- Tristan Richardson: The RFB Protocol [online], RealVNC Ltd., říjen 2006, [cit. 2006-10-31]. URL: http://www.realvnc.com/docs/rfbproto.pdf
- Neil Matthew, Richard Stones: Linux – začínáme programovat, Computer Press 2000
- Neil Matthew, Richard Stones a kol.: Linux Programujeme profesionálně, Computer Press 2001
- Eric S.Raymond: Umění programování v Unixu, Computer Press 2004

Rozsah: Přibližně 30 stran

Vedoucí práce: Mgr. Hudec Tomáš

Vedoucí katedry (ústavu): prof. Ing. Pavel Bezoušek, CSc.

Datum zadání práce: 31. 10. 2006

Termín odevzdání práce: 18. 5. 2007

Prohlašuji:

Tuto práci jsem vypracovala samostatně. Veškeré literární prameny a informace, které jsem v práci využila, jsou uvedeny v seznamu použité literatury.

Byla jsem seznámena s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Univerzity Pardubice.

V Pardubicích dne 18. 5. 2007

Nagyová Kateřina

Poděkování

Děkuji tímto vedoucímu své bakalářské práce, panu Mgr. Tomáši Hudcovi, za vedení, cenné rady a podnětné připomínky při tvorbě této práce.

Abstrakt

Bakalářská práce nabízí ucelený přehled technologií, které se používají ke vzdálené správě počítače. Ke každé z nich je doplněn popis jak fungují, jejich klady a zápory a přehled aplikací. Po přečtení by měl čtenář mít dostatečný přehled, aby si mohl zvolit, která z technologií mu nejvíce vyhovuje. Cílem práce je i řešit situaci, kdy učitel potřebuje mít dohled nad počítači studentů. Touto částí se zabývá též programová část mé práce, a to upravením již existujícího programu, který však ve své základní verzi nenabízí dostatečné možnosti.

Obsah

Seznam obrázků

1. Úvod

Technologie vzdálené pracovní plochy se používají při správě počítače, u kterého nejsme zrovna fyzicky přítomni, ale se kterým jsme spojeni buď v rámci místní sítě nebo přes internet protokolem TCP/IP.

Umožňují připojit se z lokálního počítače k jinému počítači a pracovat s ním. Na monitoru lokálního počítače je vidět totéž, jako na ploše toho, ke kterému jsme připojeni. Můžeme tedy vzdálený počítač plně ovládat.

Před vysvětlením základních funkcí je třeba definovat pojmy. Serverový počítač je počítač, který chceme spravovat. Běží na něm server, tj. program, který nám umožňuje se k němu přihlásit. Klientský počítač je počítač u kterého jsme právě přítomni a pomocí klienta se můžeme k dostat k serveru.

Server a klient spolu komunikují přes protokoly – VNC, RDP, NX a X protokol. Server zpřístupňuje plochu serverového počítače, případně generuje vlastní, kterou dává klientovi k dispozici. Klient plochu zobrazuje na klientském počítači. V případě, že se plocha změní, dává to pomocí protokolu vědět serveru.

Jak je z popisu patrné, jedná se o vhodné řešení, pokud si potřebujete z domova na pracovním počítači přečíst poštu nebo když od vás někdo potřebuje pomoci s nastavením počítače. Oblíbené využití je například přístup k datům na jiném počítači či používání programů, které nejsou nainstalovány na klientském počítači. Vhodné zejména pro správce sítě, kteří tak mohou ze svého počítače spravovat kompletně celou síť. U všech těchto případů stačí mít na počítači, ze kterého se připojujete spuštěného klienta (v některých případech dokonce stačí jen webový prohlížeč podporující Javu) a na vzdáleném server.

Rozsáhlé možnosti využití se objevují ve školních počítačových učebnách. Aplikace umožňují studentům pohled na počítač učitele. Další z možností je dohled učitele nad studentskými počítači. Touto situací se blíže zabývám v programové části bakalářské práce.

2. Technologie VNC / Protokol RFB

2.1. Charakteristika

VNC je název technologie jako takové, protokol, který využívá, se v některých zdrojích uvádí pod názvem VNC a v některých pod názvem RFB. Jedná se stále o jeden a tentýž protokol.

VNC (Virtual Network Computing) je systém, který umožňuje zobrazit si pracovní plochu vzdáleného počítače, popř. sdílet tuto pracovní plochu s více uživateli, a to nezávisle na operačním systému a architektuře, ať už vzdáleného nebo lokálního počítače.

Skládá se ze dvou komponent – serveru, který generuje obraz, a klienta (prohlížeče), který obraz vykresluje na naši obrazovku. Server může být spuštěn na zcela jiné architektuře než prohlížeč. Protokol, který spojuje server a klienta, je jednoduchý a nezávislý na platformě. V prohlížeči není ukládán žádný stav, přerušení spojení nemá tedy za následek žádnou ztrátu dat a může být kdykoliv znovu navázáno.

Většina implementací VNC serverů na UNIXu vytváří svoji vlastní grafickou plochu, která není spojena s fyzickým displejem. To znamená, že uživatelů přihlašujících se na počítač se serverem může být více a přitom se navzájem neovlivňují a neovlivňují ani činnost prováděnou uživatelem na fyzickém displeji.

Existují klienti a servery pro většinu grafických operačních systémů, na server se lze dokonce připojit i jakýmkoliv webovým prohlížečem schopným spouštět Java applety.

Původní návrh pochází od Olivetti Research Labs v Cambridge, kde bylo VNC používáno v přenosném systému, který se při pohybu v budově vždy

přihlásil k nejbližšímu vhodnému terminálu, ze kterého poté bylo možné spravovat plochu svého počítače. Poté společnost AT&T odkoupila vybavení a VNC technici spadají nyní pod společnost RealVNC, která má na starosti komerční podporu pro produkty VNC.

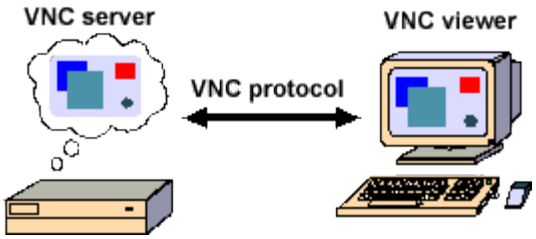
VNC je volně šiřitelný software pod licencí GNU GPL. To například znamená, že všechny modifikace musí navazovat na původní zdrojový kód.

2.2. Komunikace

VNC server nahrazuje některý grafický server (obvykle v unixové architektuře) nebo je připojen k fyzickému displeji. Klient vytváří okno, přenáší do něj grafiku ze serveru a stará se o přijímání signálů od uživatele a jejich odesílání na server.

Na části komunikace od uživatele k serveru již není nic složitého. Jedná se o překlad signálů z grafického prostředí na signály protokolu RFB a překlad z RFB na signály obecně odlišného grafického prostředí.

Komunikace opačným směrem je o něco zajímavější. Po navázání spojení se o update obrazovky stará klient. Na jeho žádost jsou ze serveru odeslány pouze části obrazovky, ve kterých došlo ke změně. Tyto části mají tvar obdélníku a přichází s nimi i jejich souřadnice v rámci obrazovky. Pro větší rychlost a menší zátěž sítě mohou být data navíc zaslána v některém z komprimovaných tvarů. Seznam podporovaných kompresí posílá klient serveru při inicializaci a server má možnost vybrat si z tohoto seznamu nejvhodnější kompresi. Z toho plyne, že implementace VNC je celkem snadná, protože klient si může určit v jakém formátu je data ochoten přijmout, zatímco server nepodporované formáty prostě negeneruje. Nejčastějším formátem komprese je pouze informace, kde v již nakresleném obrázku, je umístěn obdélník shodný s právě kresleným. Dále je podporována komprese jpeg.



Ilustrace 1: VNC komunikace

Zdroj [14]

VNC standardně používá porty 5900 až 5906 (záleží na počtu ploch) a porty 5801 a výš (taktéž).

Většina počítačů s nainstalovaným systémem Windows může mít spuštěnu jen jednu plochu obvykle se standardním číslem 0 a standardním portem 5900.

2.3. Zabezpečení

Autentizace je u protokolu VNC vcelku bezpečná, pro ověření znalosti hesla se používá systém náhodného požadavku a odpovědi, takže heslo nikdy neputuje po síti v nezašifrované podobě. Pokud jsme již jednou připojeni, síťový provoz mezi klientem a serverem putuje nezašifrovaně a může být sledován. Pokud je pro nás bezpečnost na prvním místě, doporučuje se protokol VNC tunelovat přes bezpečnější protokol, jakým je třeba SSH.

SSH (Secure Shell) slouží k připojení na vzdálený počítač, který danou službu podporuje. Veškerý provoz mezi takto připojenými počítači je šifrován pomocí techniky veřejného klíče, takže i když někdo tuto komunikaci zachytí, může ji jen těžko zpět dešifrovat. Kromě bezpečnosti nabízí SSH navíc takzvané forwardování portů, což je prostředek, kterým lze vytvořit zabezpečený tunel.

SSH klienti jsou dostupní pro většinu platforem.

Některé pozdější projekty (třeba UltraVNC) v sobě mají zabudovaný plugin pro podporu šifrovaného spojení, případně nabízejí dobrou ochranu při zakoupení komerčního balíčku (RealVNC).

Výhody SSH tunelů:

- Zabezpečená komunikace (end-to-end).
- Hesla neputují přes síť.
- Možnost zpřístupnění dat za firewallem či za NATem.
- Možnost autentizace jedním heslem na více strojů bez prozrazení opravdového hesla.
- Tunelování více portů z různých destinací.
- Automatický X forwarding.
- Spolupráce s jinými SSH servery a klienty.
- Umí přenášet soubory (sftp, scp).

Zdroj [18]

Jednou z možností jak zajistit bezpečný přístup do sítě je vytvořit si vlastní soukromou síť, a to pomocí VPN (Virtual Private Networking). Nejčastěji bývá založena na protokolu IPSec. Virtuální znamená, že se vzdálená strana, ať už jednotlivé počítače nebo celá síť, chová jako by byla součástí lokální sítě. Privátní pak znamená, že i když se jedná o připojení přes libovolnou IP síť/internet, díky šifrování jsou veškerá data zabezpečená.

Výhodou tohoto řešení je, že je univerzální. Jedná se o přístup do celé sítě. Na druhou stranu takový počítač může představovat potenciální bezpečnostní riziko. Pokud je na cestách nakažen virem, pak během VPN připojení do mateřské sítě může tuto nákazu rozšířit. Nevýhodou je pak nutnost instalace VPN klienta na každém přistupujícím počítači, což přináší problémy se správou a údržbou. IPSec připojení rovněž nemusí fungovat za všech okolností. IPSec „nesnáší“ překlad IP adres a při použití v cizí síti se může stát, že bude blokován na firewallu.

Dobrou alternativou je SSL VPN, které využívá pro bezpečný přístup protokol SSL/TLS. Ten je podporován v každém webovém prohlížeči. Na straně klienta není tedy nutno nic instalovat. Protokol HTTPS nemá problém s překladem IP adres a jeho provoz je povolen prakticky na všech firewallech. Na straně centrály, ke které se připojujeme, je nutné použít SSL VPN bránu.

2.4. Rozdíl mezi X serverem a VNC serverem

X server a VNC server mají plno společných prvků. Oba jsou postaveny na architektuře klient/server. Pro oba platí, že server je tam, kde je plocha se všemi aplikacemi. V případě X serveru to je však na počítači před námi, kde řídí grafickou kartu. X klient je aplikace, která potřebuje zobrazit okno na serveru. V případě VNC je server na vzdáleném počítači a VNC klient je aplikace, která zobrazuje zpřístupněnou plochu uživateli, jinak práce X serveru. Přitom oba dva mají na starost správu ploch (jinými slovy framebuffer) a zobrazování obrazu uživateli.

2.6. Programy používající protokol RFB

Programy používající protokol RFB se skládají z aplikace a prohlížeče. Po nainstalování a spuštění aplikace VNC na určitém počítači je možné se k němu připojit pomocí prohlížeče a pracovat na něm tak, jako by to byl počítač u kterého právě sedíte. Můžete ovládat vzdáleně myš, tiskárnu, spouštět aplikace, psát, tisknout atd.

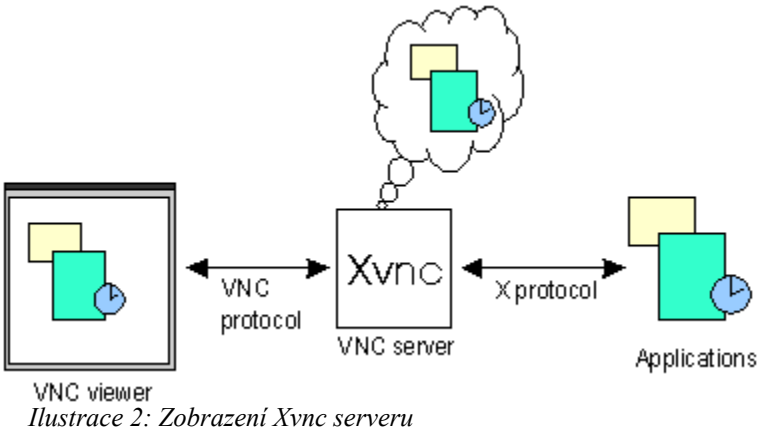
2.6.1. Xvnc

Xvnc, které je součástí balíku xorg-x11-vnc (součást X serveru), vytváří virtuální obrazovky, ke kterým se lze přihlásit ze vzdáleného počítače a pracovat jako by tento počítač měl více obrazovek, přičemž každá by mohla být pro jiného uživatele. Typickým příkladem využití tohoto rozložení

je síť, kde jeden centrální počítač slouží jiným, mnohem slabším, jako nástroj pro provádění například výpočtů, které by na slabších klientech trvaly příliš dlouhou dobu. Vlastnosti takového virtuálního desktopu lze přizpůsobit klientovi – rozlišením, barevnou hloubkou a řadou dalších nastavení.

Nejprve je třeba pomocí programu vncpasswd pod rootem vytvořit heslo pro uživatele, který se připojuje. Je třeba otevřít ve firewallu porty 5801 a výš (dle počtu virtuálních obrazovek) a 5901 a výš (taktéž).

Xvnc je unixový VNC server, který je založen na standardním X serveru. Ve skutečnosti jsou to dva servery v jednom. Vůči aplikacím je to X server a vůči vzdálenému VNC uživateli je to VNC server.



Ilustrace 2: Zobrazení Xvnc serveru

Jelikož číslo pracovní plochy VNC serveru a X serveru je totéž, můžeme se na ni v obou případech odkazovat jméno_stroje:číslo_pracovní_plochy.

XVNC se obvykle startuje pomocí scriptu vncserver, který je navržen pro zjednodušení procesu a který je napsán v Perlu. Xvnc má v podstatě stejné parametry jako standardní X server s několika málo rozšířeními. Spuštěním Xvnc -h získáte jejich seznam.

VNC server lze spouštět bez jakýchkoliv parametrů. V tomto případě se vybere první volná pracovní plocha, spustí se Xvnc jako tato pracovní plocha a několik základních aplikací. Číslo pracovní plochy můžete také specifikovat sami. Jestliže je volná, použije se, jinak se program ukončí.

VNC serverů může být na jednom počítači spuštěno víc, ale každý používá jinou pracovní plochu. Uživatel může mít na konzoli spuštěná X (normálně z konzole startx) a z druhé konzole vncserver.

2.6.2. RealVNC

RealVNC je aplikace pro vzdálenou administraci a správu počítače. Je to jediná profesionálně podporovaná verze VNC. Vytváří ji společnost, která získala práva od té původní. Chybí jí několik vylepšení, které obsahují verze níže uvedené, přesto patří mezi nejstahovanější produkty.

Rychlost reakce a kvalita zobrazení je závislá na internetové konektivitě. K dispozici je i omezení na konkrétní IP adresy, v kombinaci s heslem se stává počítač zabezpečený proti náhodným pokusům zvědavých uživatelů. K serverovému počítači se lze připojit pomocí jakéhokoliv počítače, který disponuje javou, internetovým prohlížečem a připojením k internetu.

2.6.3. TightVNC

TightVNC je aplikace, která obsahuje řadu vylepšení a optimalizací oproti původnímu VNC. Přitom je tato verze stále zdarma, nezávislá na platformě a plně kompatibilní se standardním protokolem RFB, což znamená, že lze použít TightVNC klienta se standardním VNC serverem naopak. Stále se vyvíjí, je tedy možné očekávat v budoucnu další změny.

Tato aplikace je optimalizovaná pro rychlost, takže i na pomalých linkách, jako je modemové spojení, pracuje relativně svižně. Aplikace podporuje SSH tunneling.

TightVNC je projekt spravovaný Constantinem Kaplinskym. Mnoho dalších jednotlivců a společností se zapojuje do vývoje, testování a podpory.

Server může běžet pod Linuxem i Windows, prohlížeč je pro Linux, Windows a Javu. Je dokonce možné spojit mezi sebou TightVNC a VNC, nedojde pak ovšem ke zrychlení komunikace, které je v TightVNC rozšířením původního protokolu.

. V následujících několika bodech jsou nastíněny hlavní změny. Jen je třeba podotknout, že tato vylepšení fungují pouze když jsou podporována oboustranně (jak na straně serveru, tak i na straně klienta).

- Přenos souborů ve verzi pro Windows.

- Podpora pro video mirror driver – driver připojený do systému jako další grafický adaptér – rychlejší než klasické VNC, které snímá obrazovku.

- Zoomování vzdálené plochy (prohlížeč pro Windows a Java prohlížeč). Je možné zmenšit vzdálenou plochu tak, aby byla vidět celá. Případně je možné si ji zvětšit a vidět i detaily.

- Efektivní kódování s optimální JPEG kompresí. Nové Tight kódování je optimalizováno pro pomalá a středně rychlá spojení. Generuje mnohem méně zpráv než tradiční kódování pro VNC. Na rozdíl od ostatních kódovacích metod, Tight kódování je konfigurovatelné prostřednitvím kompresních vrstev a nastavením kvality JPEG obrázků.

- Vylepšený přístup přes Web. TightVNC obsahuje vylepšený Java prohlížeč s plnou podporou pro Tight kódování, 24bitové barevné rozlišení atd.

- Podpora dvou hesel pro přístup k serveru. V závislosti na zadaném hesle server vybere, zda klient může převzít cele řízení plochy, či zda ji dostane přidělenou jen k prohlížení.

- Automatické SSH „tunelování“ na Unixu. Unixová verze TightVNC prohlížeče může tunelovat spojení přes SSH, automaticky používá lokální SSH/OpenSSH klientskou instalaci.

Zdroj [10]

2.6.3. x11vnc

Práce s již existujícím X sezením není podporována normálními VNC balíčky. Řešením je nainstalovat si x11vnc, nebo nakonfigurovat X server na podporu VNC rozšíření.

Většina implementací VNC serverů vytváří své vlastní grafické plochy, které nejsou nijak propojeny s tou skutečnou. Využívá se hlavně v případě, že je potřeba pracovat s počítačem nezávisle na tom, jestli u něj někdo sedí a nějak ho využívá. Někdy je ovšem třeba použít plochu skutečnou. Bývá to i rychlejším řešením v případě pomalého počítače.

Nejpoužívanějším klientem pro připojování se k již existujícímu sezení je x11vnc. Spojení může být zabezpečeno pomocí SSH nebo SSL.

X11vnc je součást LibVNCServer projektu. Je to free software podléhající GNU General Public License. Bylo napsáno Karlem Rungem, jako reakce na x0rfbserver a podobně funkční aplikace, ale s nižší přenosností.

Existuje plno dalších programů s podobnou funkčností, občas bývají kolektivně nazývány VNC :0 viewers. Jsou to například x4vnc modul od XFree86 verze 4, x0vncserver od RealVNC, gemsvnc nebo vzdálena plocha v KDE. Nejsou však tak efektivní.

Pomocí x11vnc se lze připojit ke svému PC z jiného a kontrolovat jej, případně si z něj odeslat nějaká data. Jiným případem využití může být školství – je možné snadno kontrolovat děje na obrazovkách žáků z učitelského PC a dohlížet nad jejich aktivitami.

Zprovoznění x11vnc je shodné s xvnc – pouze s tím rozdílem, že otevírané porty ve firewallu jsou 5800 a 5900. Oba programy je tedy možné provozovat paralelně.

Připojení k oběma programům probíhá stejně – buďto pomocí vncviewer <server>:<cislo_obrazovky>, nebo pomocí libovolného prohlížeče, který podporuje Javu. Do řádku pro adresu pak stačí zadat http://<server>:<port>, s porty 5800 a výš. Port 5800 pro obrazovku :0 a vyšší pro další.

Aby se x11vnc spustilo s podporou http a běželo i po odpojení uživatele, je třeba jej spouštět následovně:

x11vnc -rfbauth /root/.vnc/passwd -shared -http -forever -display :0

2.6.4. WinVNC

WinVNC je VNC server který umožňuje zobrazit pracovní plochu systému Windows v jakémkoliv VNC prohlížeči. WinVNC tedy zpřístupňuje existující plochu vzdáleně, nevytváří se žadná nová pracovní plocha jako v Unixu. V podstatě je to pro Windows stejný server jako x11vnc pro Unix/X11. Jeden stroj se systémem Windows může být tedy v jednom okamžiku přístupný více uživatelům, ale všichni vidí tu samou pracovní plochu.

2.6.5. UltraVNC

Klient i server jsou dostupné pouze pro Windows, ale klienta lze spustit i v Linuxu pod Wine a to i jako uživatel, bez rootovských oprávnění. Případně lze použít Java prohlížeč, který povoluje připojení (i přenos souborů) z webových prohlížečů jakéhokoliv operačního systému podporujícího Javu.

Výhody Ultra VNC:

- Možnost šifrované komunikace (podpora SSH).
- Možnost „windows-based“ loginu.
- Dynamicky mění barevnou hloubku, kterou používá v závislosti na propustnosti linky.
- Mirror driver – driver připojený do systému jako další grafický adaptér – rychlejší než klasické VNC, které snímá obrazovku.
- Chat mezi serverem a klientem.
- Přenos souborů (při přenosu složky automaticky zkomprimuje a zase dekomprimuje).

- Možnost přenášet ve VNC jen konkrétní okno (snížení datového toku).
- Dávková bezobslužná instalace.
- Existence „one click server“ verze.

Zdroj [15]

2.6.6. Krfb/Vino

Krfb je VNC kompatibilní server pro KDE desktop. Na rozdíl od většiny VNC serverů se nenapojuje přímo na X, takže je docela pomalý. Lepší alternativou je spíše nainstalované Xvnc, případně x11vnc, záleží na použití.

Vino je defaultně například v Ubuntu jako Remote Desktop Server. Jednoduché na používání.

2.6.7. Krdc (KDE Remote Desktop Connection)

Krdc je pouze nadstavbou pro KDE, který umí využít dvou přístupových protokolů:

1. RDP (Remote Desktop Protocol) od MS (rozepsaný dále).
2. RFB (Remote Framebuffer Protocol – opensource), který pochází z projektu VNC (Virtual network computing)

Díky integraci s KDE tedy Krdc umí uložit parametry (heslo, host adresu atd.) do Úschovny (Kwallet) a nabízí jednoduché menu po jehož naklikání se zavolá buď vncviewer (po zadání vnc:/host nebo jen host) nebo rdesktop s příslušnými parametry (pokud jako adresu dáme rdp:/host).

3. X window system/X protokol

3.1. Charakteristika

Tento systém je součástí moderních unixových systémů a obsahují ho téměř všechny distribuce v oblasti UNIXových systémů.

Pouštět programy v grafickém režimu na vzdálených počítačích a ovládat je nebyl v Unixu nikdy problém. Nejsou potřeba ani žádné speciální aplikace, vše je umožněno díky síťové architektuře X Window systému. Spouštěný program v podstatě ani nepozná, že je zobrazen a ovládán odjinud než z lokálního počítače.

Ke komunikaci mezi klientskou a serverovou částí je určen protokol X. Důvodem pro jeho vznik byla potřeba grafického uživatelského rozhraní, hlavně pro operační systém UNIX. Může být provozován buď na jednom počítači, nebo v počítačové síti. Je nezávislý na operačním systému a tím pádem použitelný na mnoha různých platformách. Pro síťovou komunikaci zajišťuje spojení typu klient-server.

Systém X Window je distribuovaný, uživatel může mít na jedné obrazovce okna programů běžících na různých počítačích a naopak aplikace z jednoho počítače mohou komunikovat s uživateli prostřednictvím více X serverů.

Další důležitou vlastností je otevřenost. X se skládá z mnoha částí komunikujících mezi sebou přes standardizovaná rozhraní. Proto mohou bez problémů spolupracovat implementace X od různých výrobců.

X-Window není úplným GUI, jedná se pouze o systém pro správu oken, jenž poskytuje funkce pro práci s okny a ovládání oken.

Zabezpečení komunikace je opět možno realizovat SSH tunelováním.

Prvotní myšlenka se objevila u Jima Gettyse a Boba Scheiflera (1984) z MIT. X prošla více vývojovými stádii (X9, X10) až X11, které se objevilo v září roku 1987. V roce 1993 se X Consortium stalo následníkem MIT X Consortium a došlo k uvolnění. Roku 1997 přešlo X Consortium pod správu Open Group, které vydávalo verze až po X11R6.4 patch 3. V roce 1999 však začal vývoj lehce stagnovat. XFree86 obsahovalo mnoho technických inovací vzatých od X Consortia, zatímco X.Org mělo širokou podporu hardware a využití v Linuxu. Z těchto důvodů došlo ke spojení a XFree86 se stalo čestným členem X.Org. Nastal rok 2003. Popularita Linuxu stoupala spolu s popularitou X.Org, ke kterému začali přecházet i stálí vývojáři Xfree86, v tu chvíli nastával rozpor, který bylo potřeba řešit diskuzí nad reorganizaci a kontrolou otevřeného vývoje. X jsou šířena jako free software pod licencí MIT a podobnými.

3.2. Architektura X

Architektura X Window System je založena na modelu klient-server.

Systém X Window tvoří čtyři základní komponenty:

3.2.1. X server

X server je zařízení (místní počítač) s grafickým displejem, klávesnicí a myší, na kterém je spuštěna serverová část X Window Systemu. Je zodpovědný za řízení grafického displeje a stará se o vstup a výstup. X Server reaguje na požadavky aplikací X klient, které chtějí něco vykreslit na obrazovce anebo číst vstup z klávesnice nebo myši. Předává klientským programům vstup a údaje například o pohybu myši a stisku tlačítek. Nejběžnější implementací systému X pro Linux a další systémy založené na počítačích PC je systém Xfree86.

3.3.2. X klient

X klient je aplikace (uživatelský program), která využívá služeb X serveru. Od X klientů vychází požadavky na X server na provedení určité akce (např. na vykreslení čáry, vykreslení okna). Od X serveru dostávají X klienti různé informace (např. události jako stisk klávesy, pohyb myši), na které následně reagují svými novými požadavky atd. Klientské programy komunikují se serverem prostřednictvím zpráv X protokolu, které jsou posílány a přijímány pomocí funkcí knihovny Xlib.

3.2.3. X protokol

X protokol je síťový protokol určený pro komunikaci mezi X serverem a X klientem. Veškerá komunikace mezi X klienty a X serverem probíhá prostřednictvím zpráv. Typ a použití zpráv tvoří X protokol. Zprávy zasílané X protokolem se dělí na:

- Požadavky (requests) – X klienti žádají X server o provedení určité akce. Kvůli zvýšení výkonu X klient obvykle neočekává odpověď na svoji žádost.

- Odpovědi (replies) – X server odpovídá na některé požadavky X klientů (ty, které to vyžadují).

- Události (events) – X server odešle X klientovi událost (vstup z klávesnice, myši), na kterou čeká.

- Chyby (errors) – X server podá zprávu o chybě X klientovi. Chyby jsou speciálním typem události.

Komunikace pomocí X protokolu může být lokální či vzdálená. Tím je umožněno spouštění vzdálených aplikací komunikujících s lokálním X serverem. Komunikace probíhá přes síťové (typicky TCP) sockety.

Jestliže klient i server běží na stejném počítači, používají se lokální sockety (AF_UNIX) a pro přenosy objemnějších dat i sdílená paměť.

3.2.4. X knihovny

X protokol je zajímavý hlavně pro programátory, kteří implementují X servery. Většina aplikací pro systém X používá jako programovací rozhraní knihovny funkcí jazyka C. Knihovna, která poskytuje API pro komunikaci prostřednictvím X protokolu, se jmenuje Xlib. Umí jen kreslit na obrazovku a reagovat na pohyb myši. Pokud požadujeme nabídky, tlačítka, posuvné lišty atd., musíme si je napsat sami nebo využít některé pokročilé sady implementačních nástrojů (Xt, OpenLook, Motif, Qt, GTK+).

Zdroj [23]

3.3. Používané programy

Základním předpokladem pro spojení je nějaký běžící manažer (správce) na vzdáleném počítači. Zjednodušeně řečeno: správce, který je vidět při přihlašování do systému v grafickém režimu. Ten provede autentizaci uživatele a po přihlášení spustí jeho pracovní prostředí.

Správců displeje máme v Linuxu několik. Základní správce dodávaný s XFree86 XDM (X Display Manager) není dnes příliš používán. Více se používají KDM a GDM (jeden je součástí KDE, druhý GNOME). Správce displeje umí poskytovat své služby vzdáleně protokolem XDMCP (XDM Control Protocol). XDMCP bývá ve většině instalací zakázáno a je třeba jej povolit. Pro GDM se používá program gdmconfig, KDM lze nastavit v Správce přihlášení.

Pro spojení je používán port číslo 6000. Ke spuštění vzdáleného desktopu se používá program Xnest.

3.3.1. XDM

Základní metoda postavená na X protokolu. Tato metoda má vyšší nároky na přenosy po síti. Snížení přenosů je možno zajistit pomocí dxpc, což je metoda komprese X protokolu tak, aby byl použitelný ze vzdálených míst s nízkou šířkou pásma (modem). Je to však stabilní a již ozkoušené řešení.

XDMCP

Je komunikační protokol pro X Window System. Umožňuje spustit X terminálový server, ve kterém je prezentován seznam připojitelných serverů.

Seznam může být předdefinovaný, případné ho může XDMCP server získat síťovým broadcastem.

Během vybírání serveru ze seznamu se X server běžící na lokálním počítači připojí na X display manager a otevře okno (X terminal) na spuštění window managera nebo plochy na vzdálením počítači, zatímco lokální počítač slouží jen k zobrazování výstupu.

Display manager naslouchá na UDP portu 177 a na požadavky QUERY a BROADCAST_QUERY odpovídá posíláním WILLING paketu.

X terminály jako hardware se dnes už ani nevyrábějí. Moderní terminál bývá spíše tvořen počítačem bez harddisku, který bootuje na síti a na kterém běží grafický server.

3.3.2. Synergy

Synergy je software, který umožňuje ovládat několik počítačů a fyzických obrazovek pomocí jediné sady vstupních zařízení, tj. jedné klávesnice a jedné myši. Software je spouštěn ve všech propojených

počítačích jako nenápadná služba. Pokud přejedete kurzorem myši přes okraj obrazovky počítače, ke kterému jsou klávesnice a myš fyzicky připojeny, Synergy na základě nakonfigurovaného rozložení obrazovek určí, který počítač budete pomocí vstupních zařízení ovládat. Do tohoto počítače pak bude odesílat informace o stiscích kláves a pohybu myši.

Rozložení obrazovek je možné konfigurovat do té míry, že se obrazovky mohou překrývat jen částečně: Pokud máte fyzická zobrazovací zařízení sražená k sobě, můžete v konfiguraci Synergy jemně nastavit jejich výškový nebo stranový překryv tak, že se kurzor myši při pohybu přes okraj nijak neposune ve směru kolmém na směr pohybu.

Spuštěná služba synchronizuje schránky jednotlivých počítačů, můžete tedy snadno přes schránku kopírovat text. Synergy i překládá kódování a konce řádků podle konvence systému, do kterého nebo z kterého text kopírujete. Synergy umí i synchronně spouštět šetřiče obrazovky, ukončovat je a v případě šetřičů se zamknutím počítače také zamčené počítače synchronně odemyká.

Spadá pod licenci GPL.

Zdroj [24]

4. Protokol RDP

4.1. Charakteristika

V prostředí Windows zabezpečuje přístup ke vzdálené ploše počítače v síti serverová aplikace „vzdálená plocha“ a klientský program „připojení ke vzdálené ploše“. Tyto produkty společnosti Microsoft realizují protokol RDP (Remote Desktop Protocol). Tento protokol umožňuje uživatelům připojit se k počítači, kde běží Microsoft Terminal Services.

Protokol RDP nabízí rychlejší a bezpečnější spojení než je VNC. Chyby, které se v minulosti vyskytovaly, jsou již odstraněny. Klienti existují pro většinu operačních systémů.

Server naslouchá na defaultním TCP portu 3389.

Protokol RDP je založen na protokolu T.120 a je definován pouze nad TCP/IP. Obsahuje 64000 oddělených kanálů, které mohou přenášet různé typy dat (obraz, klávesnice, myš, zvuk, tisk, schránka apod.) . Protokol RDP podporuje i základní šifrování přenášených dat – 56bit a 128bit RC4.

4.2. Používání terminálových služeb

Terminálové služby Microsoft používají protokol RDP

Od Windows 2003 Serveru SP1 lze pro zabezpečení RDP protokolu použít i TLS. Terminálové služby standardně naslouchají na TCP portu 3389.

Terminálové služby vznikají za spolupráce společností Citrix a Microsoft na jádře Windows NT 3.51 – produkt WinFrame. Jako prostředek vzdálené správy se objevují poprvé ve Windows XP jako Vzdálená plocha.

Existují dva typy provozu:

- Vzdálená správa (vzdálená plocha) – určeno ke správě serveru, umožňuje současně připojit maximálně dva uživatele, pro provoz není nutná terminálová licence.

- Aplikační režim – vzdálené provozování aplikací na serveru běžným uživatelem, počet současných připojení je omezen pouze kapacitou serveru, pro provoz je nutná licence TSCAL.

Vzdálená správa (vzdálená plocha) se ve Windows 2003 aktivuje stejně jako u Windows XP ve vlastnostech ikony Tento počítač.

4.3. Licence

Licence TSCAL je nutná pouze pro provoz v režimu aplikačním.

Správu TSCAL licencí má na starost terminálový licenční server, který licence vydává a eviduje. Licenční server je stejně jako terminálové služby volitelnou komponentou instalace Windows serveru. Licenční server je nutné aktivovat a poté do něj pomocí průvodce vložit licence TSCAL.

Provoz terminálových služeb bez licenčního serveru je omezen na 90 dní pro Windows 2000 Server a 120 dní pro Windows 2003 Server. Po této době budou terminálové služby nová spojení odmítat.

Typy TSCAL licencí:

- Na zařízení (perDevice) – nejčastěji používaná licence. Opravňuje jedno zařízení, které je používáno libovolným počtem uživatelů, využívat terminálové služby.
- Na uživatele (perUser) – opravňuje jednoho uživatele využívat terminálové služby z libovolného počtu zařízení.
- Zabudovaná (built-in) – je obsažena v klientských operačních systémech Windows 2000 Workstation a Windows XP Professional

a opravňuje tyto operační systémy využívat terminálové služby provozované na Windows 2000 serveru.

V jedné organizaci lze využívat najednou oba typy TSCAL – perDevice i perUser, ale terminálové služby jednoho serveru akceptují vždy jen jeden typ licencí. Který typ bude daný server akceptovat, lze nastavit ve vlastnostech RDP protokolu.

Zdroj [13]

4.4. Programy

4.4.1. Vzdálená plocha

Vzdálená plocha se poprvé objevila v systému Windows XP. Slouží jako server pro vzdálené připojování se k počítači. Je dodávána již se systémem. Jediné co potřebuje ke svému provozu je zapnout v systému povolení přihlásit se ke vzdálené ploše. Jako klient se používá připojení ke vzdálené ploše, které je též v systému osazeno již při koupi.

4.4.2. Xrdp

Xrdp je open source terminálový server RDP. Používá protokol RDP k zobrazení grafického uživatelského rozhraní uživateli.

Produkt slouží jako server/daemon, který běží na počítači, k němuž se uživatel připojuje Umí zobrazit nejen celou pracovní plochu, ale i jednotlivou aplikaci. Má velmi dobře řešený bezpečnostní model. Bohužel produkt je zatím poměrně mladý, a tak stále zbývá několik chyb k vyřešení.

Program poskytuje plně funkční terminálový server pro Linux, který akceptuje spojeni nejen z rDesktopu, ale též z klienta Microsoft terminal sevices připojení ke vzdálené ploše.

Na rozdíl od serverů na Windows NT/2000/2003, nezobrazí uživateli plochu Windows, ale plochu X Window.

Tento projekt spadá pod GPL.

4.4.3. RDesktop

RDesktop je RDP klientem pro většinu unixových operačních systémů Je zdarma, patří do kategorie open source software a je pod GNU General Public License.

RDesktop komunikuje s Microsoft Terminal Services.

4.4.4.WiSSH

Tento program spojuje RDP protokol s SSH protokolem a poskytuje tím vysoce spolehlivé řešení vzdáleného přístupu. Je jednoduchý k používání.

5. Protokol NX

5.1. Charakteristika

Protokol je oblíben hlavně pro svoji rychlost komunikace. Správa vzdáleného počítače se jeví skoro stejně rychlá jako sprava počítače lokálního. Rychlost spojení je příznivá především díky dobré kompresi.

Protokol NX je využíván v komerčních i nekomerčních variantách.

V několika bodech shrnutí, co vše NX nabízí:

- Rozvíjí RDP, VNC a X11 plochy bezpečně přes internet.
- Hladce integruje X11 aplikace s nativní plochou.
- Nechává puštěné aplikace i během odpojení a opětovním připojení.
- Dovoluje aplikacím přistupovat k souborovému systému na klientovi jako by byly na serveru.
- Uživatel může kopírovat mezí lokálními a vzdálenými aplikacemi.
- Je možné za serveru tisknout klientskou tiskárnu.
- Přehrává hudbu a multimedia ze vzdálených aplikací.
- Podporuje šifrování SSL a TLS (Transport Layer Security).
- Připojuje se k serverům XDMCP i Windows Terminal Services.
- Nabízí jednoduchou administraci přes webové rozhraní.

Komerční varianty nabízejí ještě další možnosti:

- Integrují autentifikační insfrastrukturu LDAP a MS Active Directory.
- Automaticky nastavují uživateli „guest“ běh v omezeném módu.
- Podporují profily uživatelů, u kterých lze nastavovat jednotlivé přístupy k aplikacím.

5.2. Programy

5.2.1. NX server / NX klient

NoMachine NX server je terminálový server pro řešení vzdáleného přístupu založený na open-source technologiích. Díky technologii X-Window vyvinuté společností NoMachine umožňuje NX spouštět aplikace na libovolném operačním systému v rámci sítě. Kromě X-Window protokolu je NX schopné přeložit a přesměrovat jiné běžně používané protokoly (RDP, VNC) do X-Window. Samozřejmostí zůstává bezpečnost podnikové sítě a rychlost přístupu k ní.

NX server běží na Linuxu i Solarisu, umožňuje přístup k aplikacím nainstalovaným na lokálním počítači, nebo může fungovat jako brána k Windows Terminal Serveru či VNC serveru. NX klient je dostupný pro mnoho platforem a operačních systémů.

Všichni NX klienti jsou volně dostupní, NX server je licencován na počet serverů, nikoli na počet přístupů k nim, což značně snižuje náklady na pořízení v poměru k podobným konkurenčním produktům.

NXClient je dodáván jako freeware pro Windows, Linux a Solaris, PlayStation2, iPAQ a Sharp Zaurus. Protože ale freeware není Free Software (neobsahuje svobodné zdrojové kódy), vznikly projekty pro dodání

svobodných alternativ. Linuxová svobodná alternativa se jmenuje knx. Všichni NX klienti jsou volně dostupní,

5.2.2. FreeNX

FreeNX je GPL alternativa ke komerčnímu NoMachine NX Serveru.

5.2.3. NX Enterprise server

NX Enterprise server nabízí vše potřebné pro podporu mnoha běžících sessions v multi-user a multi-OS prostředí. Jeden NX server může bez obtíží podporovat až 100 uživatelů, a to při zachování nízkých nákladů.

Počet user sessions pro tuto licenci/produkt není omezen.

6. Zajímavé komerční programy bez zveřejněného protokolu

6.1. LogMeIn

LogMeIn je vícesložkovým systémem. Na jedné straně stojí počítač, ke kterému je třeba zajistit přístup, a na straně druhé pak počítač, ze kterého má být přístup uskutečněn. Propojení daných stran se realizuje přes server výrobce (je použito HTTPS spojení, čímž lze do značné míry obejít problémy vznikající obvykle u řešení daného typu s firewally či proxy servery), přičemž na kontrolovaném počítači je zapotřebí zprovoznit odpovídající klientský software, zatímco na straně kontrolního počítače stačí pouze webový prohlížeč.

Poté, co se uživatel přihlásí do prostředí služby, nabídne se mu seznam počítačů (včetně indikace toho, zda je daný stroj online), na které může přistupovat. Jeden z nich zvolí a zadá odpovídající přístupové heslo. Nyní mu prohlížeč nabídne různé typy služeb, závislé na tom, jakou edici služby uživatel využívá.

Celá komunikace mezi klientem a serverem probíhá šifrovaně s využitím SSL protokolu (https spojení, port 443) a SSL/TLS certifikátů k ověření přístupu. Šifrování zajišťuje, že komunikace nebude odposlechnuta či pozměněna ani serverem LogMeIn, který funguje jako prostředník. K zabránění neautorizovaného přístupu slouží separátní hesla k LogMeIn účtu a k přihlášení do Windows vzdáleného počítače. LogMeIn používá 128/256bitové šifrování, bezpečnostní rysy placených verzí pak rozšiřuje např. možnost IP filtrování a RSA SecurID ověřování.

Celý systém je jištěn na několika úrovních. Přihlašovacím heslem k uživatelskému účtu LogMeIn, 256bitovým SSL šifrováním komunikace

a v neposlední řadě vlastním přihlašovacím jménem a heslem k systému Windows v ovládaném počítači. Podporovány jsou Windows 98, 2000, XP a Server 2003.

Základní výhodou LogMeIn je ve srovnání s mnoha konkurenčními řešeními jeho práce prostřednictvím serverů společnosti LogMeIn. To kromě již zmíněného přístupu přes webový prohlížeč umožňuje rovněž ignorovat překladače adres (NAT) a v mnoha případech i obcházet firewall. Řešení LogMeIn je tak ideální pro uživatele, kteří nemají veřejnou IP.

Aplikace nám nabízí plno možností k použití. K dispozici je takto především Remote Control, tedy možnost zobrazit si vzdálenou plochu a tu ovládat. Příslušné rozhraní přitom nabízí vše, na co jsme zvykli u řešení daného typu: kontrolovat lze rozlišení obrazovky i bitovou hloubku, možné je zobrazení v okně či na celou obrazovku s funkcí zoom, podporováno je využití více monitorů a k dispozici jsou i různá nastavení kvality zobrazení. Odpovídající nabídka dovoluje simulovat různé specializované klávesy (např. PrintScreen, Alt-Tab, funkční klávesy aj.). Produkt také dovoluje vyměňovat mezi propojenými počítači obsah schránky a možný je i vzdálený tisk. Nabízí i chatovací funkci, která dovoluje s uživatelem na kontrolovaném počítači komunikovat pomocí instant messagingu.

LogMeIn File Manager je určen pro přenos souborů. Přenos probíhá zabezpečeně a data jsou pro něj zkomprimována, což šetří kapacitu přenosových linek.

Jinou zásadní otázkou je cena služby. V podobě Free lze LogMeIn využívat zdarma, takto se ovšem nabízí pouze vzdálený přístup. Pro všechny ostatní funkce je třeba investovat 70 USD ročně (nebo se nabízí i méně výhodné měsíční platby) za LogMeIn Pro, které si lze nicméně po prvotní registraci vyzkoušet bezplatně (po uplynutí 30 dnů či 120 minutách užívání přejde služba do módu Free). K dispozici jsou pak i další možnosti licencí, které jsou určeny zejména pro vzdálený servis počítačových pracovišť.

Zdroj [20]

6.2. Radmin

Radmin disponuje všemi klíčovými funkcemi jako jsou vzdálené ovládání, transfer souborů, NT zabezpečení s podporou NTLMv2, Telnet, vícejazyčná podpora. Komunikace využívá TCP/IP protokolu a 256bitového šifrování všech datových toků.

I prostřednictvím modemu lze dosáhnout slušného obnovování obrazovky (okolo 5–10 obrazovek za sekundu). Při připojení přes lokální síť je možné dosáhnout 100–500 obnovení! Frekvence obnovování může být pro úsporu sítě nastavena (i snížena).

Hlavní rysy:

- Možnost spuštění Radmin serveru jako služby na systémech WinNT a Win9x, což umožňuje vzdálené přihlášení a odhlášení uživatelů.
- Podpora současného připojení k více vzdáleným počítačům.
- Podpora navázání na přerušené spojení.
- Transfer souborů v prostředí podobném Windows Exploreru.
- Možnost vzdáleného vypnutí počítače bez nutnosti otevření obrazovky vzdáleného počítače.
- Připojení službou Telnet k počítači s Windows NT.
- Podpora Windows NT/2000 zabezpečení a specifikace úrovně přístupových práv jednotlivým uživatelům nebo skupinám uživatelů.

- Je-li uživatel Radmin přihlášen v doméně WinNT, může být pro připojení ke vzdálenému počítači použito jeho uživatelské jméno a heslo v doméně.

- Podpora IP filtrace k omezení vzdáleného přístupu jen ze specifických IP adres a subnetů.

- Všechny události jsou zaznamenávány do log souboru.

- Možnost aktivovat autorizaci přístupu a upozornění při připojení k Radmin serveru na straně vzdáleného počítače.

- Plná kompatibilita s Windows Vista.

- Technologie DirectScreenTransfer poskytuje bezkonkurenční rychlost při minimální zátěži CPU.

- Systém pro bezpečnou textovou a hlasovou komunikaci, rozšiřující možnosti použití pro prezentační a demonstrační účely.

- Textový režim umožňuje vedení skupinové nebo privátní komunikace, hlasový režim nabízí konferenční nebo veřejné komunikační kanály.

- Podpora Windows Security, 256bitové AES šifrování všech přenesených dat, podpora NTLM/Kerberos a Active Directory, IP filtrace

- Podpora více monitorů a současného vícenásobného připojení ke stejnému počítači.

Zdroj [22]

7. Dokumentace k programu

Jako vhodné řešení jsem zvolila úpravu kódu KRDC.

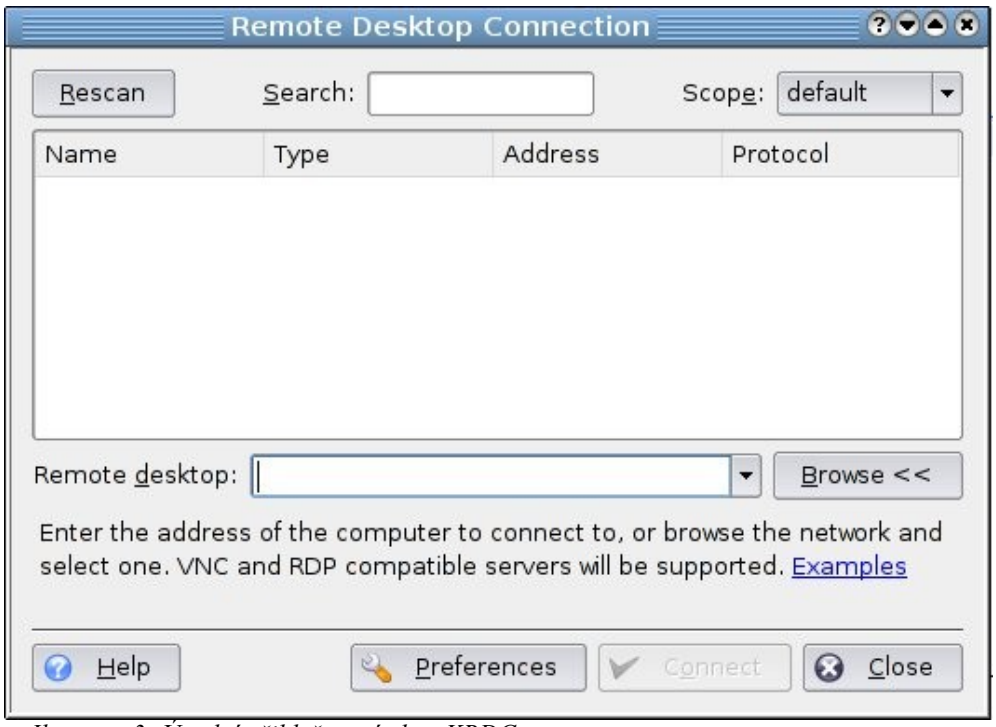
KRDC je nadstavba pro KDE, která zvládá využít RDP i RFB protokol. Produkt je lehce dostupný a je např. přítomen již v základní instalaci linuxového systému Ubuntu. Zároveň spadá pod licenci GPL, to mj. znamená, že zdrojové kódy jsou volně přístupné a je možné si je přizpůsobovat k obrazu svému.

Program je naprogramován v jazyce C++ s využitím knihoven KDE/Qt. K programování jsem použila KDevelop a QT 3 Designer. Zkompilovatelný je ve všech verzích KDE od verze 3.4.3.

7.1. Uživatelský pohled

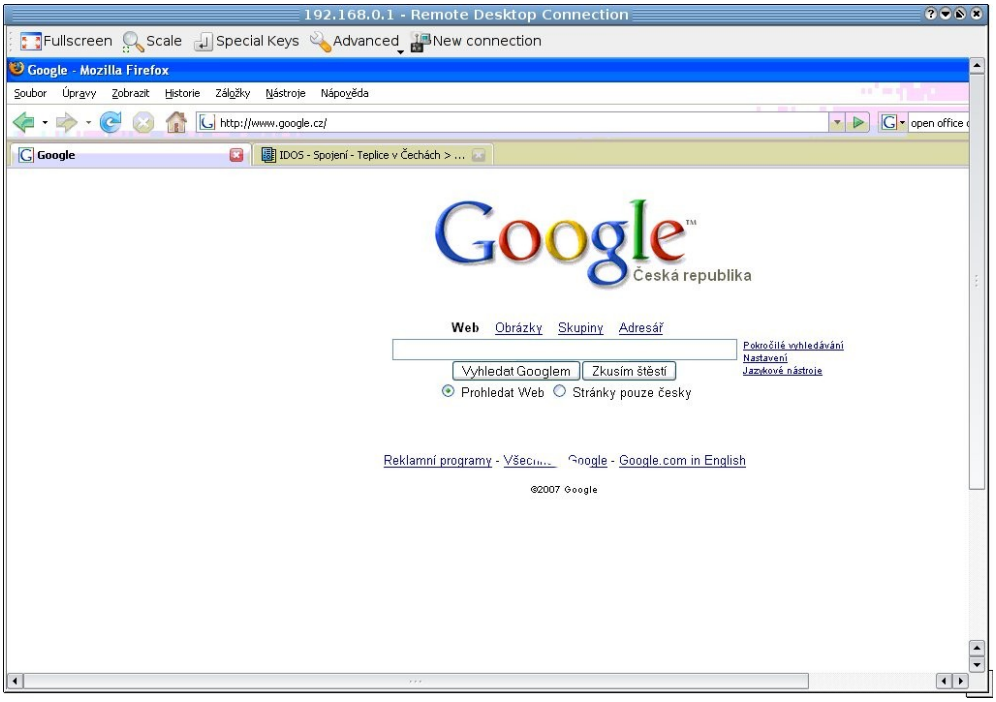
Program se spustí po kliknutí v K Menu (kategorie Internet) na položku Remote Desktop Connection (Krdc), nebo jednodušeji napsaním do příkazového řádku názvem programu – krdc. Naběhne úvodní okénko programu (viz níže). Do řádku Remote desktop je třeba zadat typ připojení a IP adresu vzdáleného počítače spolu s číslem plochy. Po kliknutí na šipku hned vedle se rozjede nabídka IP adres zadaných při dřívějším přihlášení. V případě, že uživatel neví, jak svoji volbu zapsat, volba Examples nabízí přehlednou ukázku.

V položce Preferences je možné nastavit kvalitu spojení (čím vyšší kvalita, tím je vyžadováno rychlejší spojení). Nejpomalejší vyzkoušené spojení z mé strany bylo 3 Mb, v této rychlosti však již kvalita spojení nehrála roli.



Ilustrace 3: Úvodní přihlašovací okno KRDC

Takto vypadá okénko po správném připojení.



Ilustrace 4: Vzdálená plocha

První položka – Fullscreen nabízí jak už sám název napovídá přechod z režimu Fullscreen (tedy celá obrazovka) a zpět.

Scale slouží k zoomování obrázku, přizpůsobí velikost zobrazovaného serveru, aby byla vidět celá plocha.

Special Keys se využívá, pokud je potřeba zadat na serveru klávesové zkratky (např. Ctrl+Alt+Del).

V nabídce Advanced lze nalézt zaškrtávací políčko View Only, které se stará o nastavení možnosti zasahovat do serverového počítače, nebo pouze sledovat, co se odehrává na ploše.

New connection připojuje další počítač, aby bylo možné jich sledovat více najednou.

7.2. Realizace sledování více počítačů

K rozšíření programu jsem zvolila začlenění Kparts. Kparts je vlastní technologie prostředí KDE, která umožňuje tzv. komponentový model GUI aplikací.

Základní rozdíl mezi komponentou (KPart) a widgetem je dynamičnost – určení komponenty se děje až za běhu aplikace. Komponenta není úplně autonomní. To znamená, že systém KParts umožňuje komponentě definovat jak má oknu, ve kterém je zobrazována, měnit strukturu menu nebo taskbar.

KParts je struktura pro KDE komponenty založená na standardních KDE/Qt objektech jako QWidget a KTMainWindow.

Výběrem řešení jsem se nechala inspirovat v dokumentaci programu, kde bylo napsáno, že do budoucna se plánuje KParts do programu začlenit.

K programu jsem tedy přidala dva nové soubory – krdcpart.h a krdcpart.cpp. Jsou vloženy jako součást přílohy i s dodanými komentáři.

Krdcpart obsahuje dvě třídy – KrdcPart a Part Widget. Třída KrdcPart je zděděná od Kparts::ReadOnlyPart. V podstatě vytváří novou Kpart, nastavuje velikost a ostatní vlastnosti , provádí opětovnou kontrolu jména a hesla, zadaného protokolu a spouští widget. K vytváření volá v konstruktoru třídu PartWidget.

K vytváření nové instance je využita knihovna generic factory.

V souboru Makefile.am bylo třeba nastavit sdílené knihovny, respektive vytvořit dynamickou knihovnu. Dynamické knihovny jsou vytvořeny jiným způsobem než statické. Linker objektové kódy předzpracuje tak, aby mohly být připojeny ke kódu programu až za chodu. Při spouštění programu je knihovna umístěna do sdílené paměti, kde se o ni podělí všechny programy, které ji potřebují.

Dále bylo potřeba upravit krdc. desktop, kde se v podstatě dává systému vědět o komponentě.

Spravování komponent je řešeno přes PartManager.

7.2. Řešení nedetekovatelného sledování

Řešení ve Windows:

VNC servery ve Windows se obvykle prezentují v pravé spodní části hlavní lišty, kde se objevují spuštěné služby. Při připojení klienta změní ikonka barvu.

Některé programy mají již ve svých vlastnostech možnost skrývat tuto ikonku (např. TightVNC, Radmin).

Zobrazování programů, které tuto možnost nemají, lze nastavit ve službách. Lze se k nim dostat přes nabídku Start -> Ovládací panely -> Nástroje pro správu -> Služby. Mezi službami lze nalézt VNC server. Ve vlastnostech na záložce přihlášení je třeba odškrtnout položku „povolit službě používání plochy“. Po restartu systému již ikonka nenaběhne, VNC server však automaticky běží.

Řešení v Linuxu:

Konkrétně je to vyzkoušeno s x11vnc. Stačí program spustit poklikáním přímo na položku v /usr/bin/x11vnc, ne z příkazového řádku.

7.3. Zásah do plochy

Okno se zobrazeným serverem jsem v programu nastavila, aby nabíhalo v módu, kdy kurzor myšky je sice vidět, ale plochu vzdáleného počítače nijak neovlivňuje. V horní části lišty je dále dále umístěna ikonka Advanced,

kde lze odškrtnout režim View Only a počítač je opět možné spravovat. Režimy je možné střídat po celou dobu spojení.

View Only je vlastnost v konstruktoru okna. Okno tedy již nabíhá s nastaveným parametrem. Metoda viewOnly kontroluje, zda je program nastaven v prohlížecím módu, v případě, že ano, vrací true, jinak false.

Možnost prohlížení okna bez zásahu byla v programu již vyřešena, já jsem přidala možnost nastavit tuto vlastnost již během spouštění a nastavila jsem automatické spouštění na hodnotu true.

8. Závěr

Tuto bakalářskou práci jsem si vybrala, protože mě již delší dobu fascinují různé způsoby jak spravovat počítač, aniž bych u něj byla přítomna. Navíc tyto služby hojně využívám, když potřebuji přistupovat k počítači u nás doma, ať už k datům, která tam mám uložená, nebo když rodiče potřebují něco na počítači nastavit a já nejsem doma několik týdnů. Vzala jsem to jako výzvu dozvědět se o této problematice co nejvíce informací.

V práci je uveden poměrně podrobný přehled technologií vzdálené plochy, který v takto ucelené podobě na českém (i anglickém) internetu stále chybí. Většinu podkladů jsem překládala z anglických stránek, případně vypozorovala z diskuzí na serveru www.abclinuxu.cz.

Ve chvíli, kdy jsem otevřela KDevelop se rozplynuly mé lehce naivní představy o kompilaci KDE projektů během jednoho odpoledne. Následující týdny jsem tedy trávila studováním článků o programování v Linuxu a zkoušením, který ze zdrojových kódu uveřejněných na internetu má ten správný Makefile, či která verze libtoolu a automaku je schopna komunikovat s mým KDevelopem. Problém jsem tedy zdárně vyřešila a mohla se vrhnout na programování. Směle jsem uposlechla autorova doporučení a začala začleňovat KParts do programu. Poměrně dlouhé zaseknutí nastalo u nastavování sdílených knihoven, které bylo potřeba vyřešit přepsáním souboru Makefile.am. V tuto chvíli jsem se obrátila na několik zkušenějších programátorů, nikdo z nich se však nevěnoval programování pro KDE. Tento problém jsem nakonec také zdárně vyřešila. Vytváření nové komponenty funguje.

Program tedy zvládá nastavování ovlivňovat možnost zasahování či nezasahování do plochy vzdáleného počítače. Má v sobě začleněn Kparts, zvládá tedy vytvořit novou komponentu (nové okno s následujícím

spojením). Práce s part managerem však plně nefunguje. Vzhledem ke složitosti problému a absenci použitelných zdrojových kódů, kde by byla tato technologie využita, nemám v tuto chvíli dostatečné znalosti, abych mohla práci dořešit. K vyřešení problému by bylo třeba lépe pochopit, jak jsou knihovny napsané a jakým způsobem přesně pracují. Mohla by to být součást mé budoucí diplomové práce. Režim View Only je plně funkční a nastavení nedetekovatelného sledování je též vyřešeno.

Práce poskytuje informace, které mohou správce sítě dovést k nejefektivnějšímu způsobu dohlížení nad počítači v celé spravované síti i učitele k dohledu nad počítači studentů.

Použité zdroje:

1. MATTHEW N., STONES R. *Linux – začínáme programovat*. Computer Press, 2000. 897 stran, ISBN 80-7226-307-2

2. Wikipedia. *Remote Desktop Protocol* [online]. [cit. 2007-15-03], dostupné z http://en.wikipedia.org/wiki/Remote_Desktop_Protocol

3. Microsoft Corporation. *Understanding the Remote Desktop Protocol* [online]. [cit. 2007-27-03], dostupné z http://support.microsoft.com/kb/186607

4. Karl J. Runge. *x11vnc: a VNC server for real X displays* [online]. [cit. 2007-12-03], dostupné z http://www.karlrunge.com/x11vnc

5. Wikipedia. *VNC* [online], [cit. 2007-22-02], dostupné z http://gentoo-wiki.com/VNC

6. Faure David. *Creating and Using Components (Kparts)* [online]. [cit. 2007-21-02], Chapter 13, dostupé z http://developer.kde.org/documentation/tutorials/kparts/index.html

7. Malenko, K. *Desktop Environment – architektura* [online], [cit. 2007-21-02], dostupné z http://www.ms.mff.cuni.cz/~beran/vyuka/X/ref_malenko/KDE.html .cs

8. Wikipedia. *VNC* [online]. [cit. 2007-22-02], dostupné z http://en.wikipedia.org/wiki/VNC

9. Waugh Tim. *VNC – Where it came from, where it's going* [online]. [cit. 2007-25-02], dostupné z http://cyberelk.net/tim/articles/VNC/

10. TightVNC Software. *Introduction to TightVNC* [online].
[cit. 2007-20-04], dostupné z http://www.tightvnc.com/intro.html

11. Beran Martin. *Programování pro X Window systém* [online]. 2004
[cit. 2007-15-01], dostupné
z http://www.root.cz/clanky/programovani-pro-x-window-system/

12. Hruška Karel. *xVNC vs. x11vnc* [online], [cit. 2007-20-04], dostupné
z http://www.abclinuxu.cz/blog/FluxBlog/2007/3/6/171845

13. Microsoft Corporation. *Terminálové služby* [online].
[cit. 2007-10-03], dostupné
z http://www.microsoft.com/cze/technet/clanky/terminal_services.mspx

14. RealVNC Ltd. *VNC – How it works* [online], [cit. 2007-25-04],
dostupné z http://www.realvnc.com/howitworks.html

15. UltraVNC. *Remote Control Software for all* [online].
[cit. 2007-25-04], dostupné z http://www.uvnc.com/index.html

16. AT&T. *X-based VNC server* [online]. [cit. 2007-17-03], dostupné
z http://www.cl.cam.ac.uk/research/dtg/attarchive/vnc/xvnc.html

17. Bíbr Ivan. *Vzdálená administrace III* [online]. 2004,
[cit. 2007-17-03], dostupné z http://casopis.systemonline.cz/195-jak-na-to-v-linuxu.htm

18. Koloros Petr. *Šifrovaný tunel skrz nezabezpečenou síť* [online].
[cit. 2007-17-03], dostupné
z http://www.cryptofest.cz/2003/slajdy/sshtunel/ssh.html

19. Čadík Martin, Šaroun Martin. *Systém X-Window* [online].
[cit. 2007-10-03], dostupné
z http://www.cgg.cvut.cz/~cadikm/school/vps/

20. Krejčí Richard. *LogMeIn: řešení pro vzdálený přístup na publikačním pracovišti* [online]. 2005, [cit. 2007-10-03], dostupné z http://www.grafika.cz/art/sw/logmein.html

21. Microsoft Corporation. *Understanding the Remote Desktop Protocol (RDP)* [online]. 2007, [cit. 2007-12-04], dostupné z http://support.microsoft.com/kb/186607

22. Mokry systems. Radmin Remote Control [online]. [cit. 2007-15-04], dostupné z http://www.mokry.cz/radmin/radmin-remote-control.php

23. Martin Pokorný. *Systém X Window* [online]. [cit. 2007-15-04], dostupné z http://tamnekde.unas.cz/data/prg/prg_g1/prg_g1_10.php

24. Jakub Hegenbart. *Synergy* [online]. [cit. 2007-15-04], dostupné z http://www.abclinuxu.cz/software/pracovni-prostredi/doplnky/synergy

Příloha 1

Zdrojový kód je lehce zkrácen (v obou přílohách), kompletní je na přiloženém CD

KrdcPart.h

Widget, který obsahuje scrollview a kpart

```
class PartWidget:  public QWidget{
public:
  PartWidget(QWidget *parent, const char* name) : QWidget(parent,
name) { scroll = NULL; }
   void resizeEvent( QResizeEvent *event){
    if (scroll != NULL)
      scroll->resize(event->size().width(), event->size().height());
   }
void setScrollView(QScrollView2 *newScroll) { scroll = newScroll; }
  QScrollView2* getScrollView() { return scroll; }
private:
  QScrollView2 *scroll;
};
```

Aktuální kpart

```
class KrdcPart: public KParts::ReadOnlyPart
{
  Q_OBJECT
public:
  KrdcPart(QWidget *parentWidget, const char *widgetName, QObject
*parent, const char *name, const QStringList& args);
   virtual ~KrdcPart();
   static KAboutData *createAboutData(); ;
   virtual bool openFile();
   virtual bool openURL(const KURL &url);
private:
  PartWidget *frame;
  KRemoteView *remote;
  QScrollView2 *scroll;
};
```

Příloha 2

KrdcPart.cpp

Factory objekt

```
typedef KParts::GenericFactory<KrdcPart> KrdcPartFactory;
K_EXPORT_COMPONENT_FACTORY( libkrdc, KrdcPartFactory )
```

Konstruktor

```
KrdcPart::KrdcPart(QWidget *parentWidget, const char *widgetName,
QObject *parent, const char *name, const QStringList& args)
    : KParts::ReadOnlyPart(parent, name){
  remote = NULL;
```

Vytvoření nového widgetu a scrollwidgetu

```
  frame = new PartWidget(parentWidget,widgetName);
  scroll = new QScrollView2(frame, "scrollview");
  frame->setScrollView(scroll);
  frame->show();
  frame->resize(parentWidget->width(),parentWidget->height());
```

Nastavení parametru scrollwidgetu

```
  scroll->setFrameStyle(QFrame::NoFrame);
  scroll->setSizePolicy(QSizePolicy(QSizePolicy::Expanding,
                                    QSizePolicy::Expanding));
  scroll->setResizePolicy(QScrollView::AutoOne);
  scroll->show();
  scroll->resize(parentWidget->width(),parentWidget->height());
```

Nastavení rámu hlavního widgetu

```
  setWidget(frame);
}

bool KrdcPart::openURL(const KURL &url)
{
  int port = url.port();
  QString str, host, user, pass;
```

Odstranění případné staré session

```
  if (remote != NULL)delete remote;
  host = url.host();
```

Kontrola hesla

```
if (url.hasPass())pass = url.pass();
if (url.hasUser())user = url.user();
```

Výběr správného protokolu

```
if (url.protocol() == "vnc"){
  if (port < 100)
    port += 5900;
  remote = new KVncView(frame, 0, host, port, pass);
 }
 else if (url.protocol() == "rdp" || url.protocol() == "smb")
  remote = new KRdpView(frame, 0, host, port, user, pass);
 else
  return false;
```

Přidání vzdáleného widgetu do scrollwidgetu a následný start

```
 scroll->addChild(remote);
 remote->start();
 remote->show();
 return true;
}
```

ÚDAJE PRO KNIHOVNICKOU DATABÁZI

Název práce	Dohledový systém nad počítači v učebně
Autor práce	Nagyová Kateřina
Obor	Informační technologie
Rok obhajoby	2007
Vedoucí práce	Mgr. Tomáš Hudec
Anotace	Bakalářská práce nabízí ucelený přehled technologií, které se používají ke vzdálené správě počítače. Ke každé z nich je doplněn popis jak fungují, jejich klady a zápory a přehled aplikací. Po přečtení by měl čtenář mít dostatečný přehled, aby si mohl zvolit, která z technologií mu nejvíce vyhovuje. Cílem práce je i řešit situaci, kdy učitel potřebuje mít dohled nad počítači studentů.
Klíčová slova	VNC, RDP, RFB, NX, X server, vzdálená správa