



Středoškolská technika 2013

Setkání a prezentace prací středoškolských studentů na ČVUT

LSD v informatice

Michal Gruber, Ondřej Linger, Amin Shakery, Denis Sianov

SPŠE Ječná
Ječná 30, Praha 2

Osnova:

Úvod	1
Odložené projekty	2
Projekty, které nyní rozšiřujeme	3
Začínající projekty	5
Závěr	6
Zdroje	6

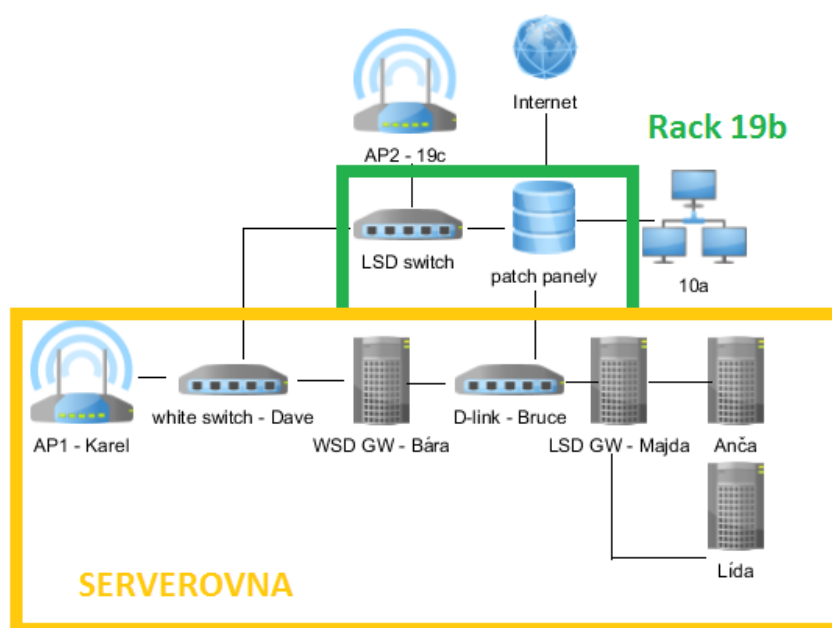
Úvod

Již třetím rokem běží na SPŠE Ječné studentský projekt pojmenovaný LSD, který shromažďuje žáky z 3. a 4. ročníků a umožňuje jim naučit se něco nového, pracovat v týmu díky světově uznávaným metodikám řízení týmu a na konci 4. ročníku obhájit svou dlouhodobou maturitní práci. Naše projekty se z větší části zaměřují na podporu frontální výuky na SPŠE Ječné, ale najdou se u nás také projekty, které vznikly na základě zájmu členů týmu. Naše služby a projekty se neustále snažíme vylepšovat a stabilizovat.

Odložené projekty

Captive portal

Captive portal byl zřízen za účelem kontrolovaného připojení žáků naší školy k LSD wifi (viz schéma LSD sítě Obr.1.1). Captive portal je soupis firewallových pravidel, které ověřují, zda je daný uživatel již přihlášen (jeho MAC adresa je v tabulce aktivních přihlášení) a jestliže není, je přesměrován na přihlašovací stránku. Bohužel z důvodu jeho velké nestability byl celý projekt ukončen.



Obr. 1.1

Bugzilla

Bugzilla je aplikace pracující ve webovém rozhraní pro sledování chyb takzvaný „Bug tracking“. Lze jí snadno stáhnout zdarma, protože jde o opensource. Bugzilla umožňuje uživateli přímo zadávat a odesílat hlášení o chybě. Takové hlášení může odeslat kdokoli a je možné k němu přidat video dané chyby, komentáře, závažnost atd.... Je to velice užitečná pomůcka pro hledání a opravování veškerých chyb a zlepšování kvality softwaru. My jsme Bugzillu nahradili agilní metodikou řízení týmu Scrum.

Projekty, které nyní rozšiřujeme

Webhosting

Na serveru běží FTP daemon, na který se připojují uživatelé. Při připojení klienta se FTP daemon připojí přes view do centrální databáze a ověří, zda má uživatel přístup. Pokud ano, pustí ho do složky, která je stejná jako jeho uživatelské jméno. Tato složka je pak přístupná i přes web. U toho systému není potřeba synchronizace pro samotné uživatelské účty na FTP, ale je nutný skript, který vytváří a maže složky daných uživatelů.

Moodle

Moodle je softwarový balíček pro tvorbu výukových systémů a elektronických kurzů na internetu. Jedná se o neustále se vyvíjející projekt, navržený na základě sociálně konstruktivistického přístupu ke vzdělávání. Moodle je poskytován zdarma jako Open Source software spadající pod obecnou veřejnou licenci GNU. Moodle již ve své základní instalaci má možnost napojení na MySQL, proto bylo potřeba v centrální databázi pouze vytvořit view, který měl příslušné sloupce potřebné pro Moodle (uživatelské jméno, heslo, email) a poté Moodle správně nastavit (přihlašovací údaje do MySQL a synchronizace údajů)

Vylepšení Moodlu do budoucna

Určitě by bylo vhodné vytvořit do Moodlu šablonu, která bude kopírovat jednotný vzhled všech systémů LSD. Bohužel taková šablona zatím neexistuje a příští rok by se této práci mohl nějaký nový grafik ujmout.

FTPS

Stávající verzi FTP jsme vylepšili na šifrovanou verzi FTP, tzv. FTPS (FTP s přidaným šifrováním SSL). Jak to funguje: Klient před odesláním citlivých dat žádá o šifrovanou komunikaci a z toho důvodu bylo nutné přes openssl vygenerovat certifikát (self signed) a privátní klíč pro šifrování komunikace. Pak už jenom stačilo nastavit FTP daemona, aby SSL komunikaci akceptoval. Tady se žádný další port nevyužívá. FTPS by nemělo být zaměňováno s SFTP, jedná se o odlišný systém, i když se i on v principu využívá na přenos souborů.

HTTPS

Z běžného HTTP protokolu, jsme udělali tzv. HTTPS, což je podobné jako s FTPS. Všechna data jsou mezi serverem a klientem šifrovaná pomocí SSL/TLS. I když narozdíl od FTPS u HTTPS jsou všechna data šifrována rovnou, mezi serverem a klientem vůbec nešifrovaná komunikace neprobíhá. Opět byl použit certifikát a privátní klíč jako u FTPS. Pak se nastavil apache (webový server). HTTPS naslouchá na portu 443, oproti tomu HTTP na 80, takže bylo zapotřebí ještě nastavit na gatewayi port forwarding.

MySQL – synchronizační skript

V rámci zrychlení a stabilizace MySQL bylo nutné napsat nový synchronizační skript. Synchronizace s MySQL probíhá následovně: skript se připojí do obou databází (centrální a MySQL) a načte seznamy uživatelů (u Centrální databáze samozřejmě jenom ty, kteří mají přístup ke službě MySQL). Poté projede uživatele v centrální databázi (dále jen CDB) a pokud:

a) uživatel má přístup podle CDB, ale v MySQL neexistuje, je mu vytvořen uživatel a databáze se stejným jménem (a je mu nastaveno oprávnění pouze pro tuto databázi)

b) uživatel má přístup podle CDB a v MySQL už existuje. V tomto případě je ověřeno, že pro něj skutečně existuje databáze, a případně je vytvořena. Dále je mu podle CDB přepsáno heslo do MySQL.

Poté skript projde uživatele v MySQL (kromě systémových) a zkontroluje, zda mají přístup podle CDB. Pokud ano, nic se neděje. Pokud ne, smaže uživatele a jeho databázi i se všemi jeho daty.

MSSQL

U MSSQL nastal obdobný problém, který jsme řešili stejným způsobem (viz Obr. 2.1).

```
mssqlUsersSql = "SELECT name, loginname, password FROM syslogins WHERE sysadmin = 0 AND loginname NOT LIKE '##%'";
mssql_cur.execute(mssqlUsersSql)
mssqlUsers = mssql_cur.fetchall()

#nyni projedeme CDB a natahame uzivatele do MSSQL
for a in mysqlUsers:
    jeTam = False
    for b in mssqlUsers:
        if(a[1] == b[0] and b[0] == b[1]):
            jeTam = True

            #natvrdo prepiseme heslo podle CDB (bez kontroly)
            mssqlChangePassword = "ALTER LOGIN " + a[1] + " WITH PASSWORD = '" + a[2] + "';"
            mssql_cur.execute(mssqlChangePassword)

            #zkontrolujeme, zda pro uzivatele existuje db a pripadne ji vytvorime
            mssqlCheckDb_sql = "SELECT * FROM sysdatabases WHERE name = '" + a[1] + "';"
            mssql_cur.execute(mssqlCheckDb_sql)
            mssqlCheckDb = mssql_cur.fetchall()
            if(mssql_cur.rowcount < 1):
                mssqlCreateDatabase = "CREATE DATABASE " + a[1] + " ";
                mssql_cur.execute(mssqlCreateDatabase)
                mssqlChangeOwnerOfDatabase = "ALTER AUTHORIZATION ON DATABASE::" + a[1] + " TO " + a[1] + " ";
                mssql_cur.execute(mssqlChangeOwnerOfDatabase)
            #pokud uzivatel neexistuje, vytvorime ho a vytvorime mu i db
            if(not jeTam):
                mssqlAddLogin = "CREATE LOGIN " + a[1] + " WITH PASSWORD = '" + a[2] + "', CHECK_POLICY = OFF;"
                mssql_cur.execute(mssqlAddLogin)

                mssqlCreateDatabase = "CREATE DATABASE " + a[1] + " ";
                mssql_cur.execute(mssqlCreateDatabase)

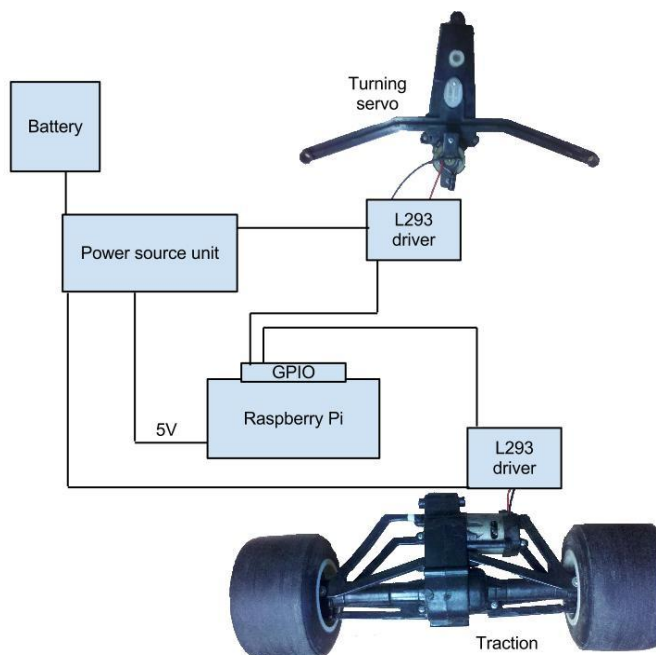
            mssqlSetDefaultDb = "ALTER LOGIN " + a[1] + " WITH DEFAULT DATABASE = " + a[1] + " "
```

Obr. 2.1 – část synchronizačního skriptu mezi MSSQL a Centrální databází

Začínající projekty

Auto dálkově řízené pomocí wi-fi

Jedná se o prototyp rádiem řízeného vozu, ze kterého jsme odstranili ovládací součásti a nahradili je mikropočítačem Rapsberry Pi s USB Wi-fi donglem a dvojicí driverů L239 (viz Obr. 3.1), které budeme ovládat pomocí skriptů napsaných v Pythonu. Na svrchní část krytu jsme připevnili web kameru (viz Obr. 3.2 a Obr. 3.3), která umožňuje přenos obrazu z autíčka na FTP server. Tudiž bude možné řídit autíčko pomocí internetu odkudkoliv ze světa.



Obr. 3.1



Obr. 3.2

Obr. 3.3

Závěr

Většina našich současných projektů byla započata předchozí generací LSD týmu a my věříme, že dokážeme naplno uplatnit jejich potenciál. Co se týče námi započatých projektů, tak ty se snažíme dělat nápadité a užité. Vše velmi pečlivě dokumentujeme a doufáme, že i navazující generace tohoto projektu budou mít tak kvalitní zázemí, jako ho nyní máme my.

Zdroje

Maturitní práce:

- Martin Matuna, Webhostingové a e-learningové služby LSD
- Michal Šanda, Vytvoření stabilní sítě a správa gateway v rámci projektu LSD

Internetové zdroje:

- lsd.spsejecna.net/wiki/