Masarykova univerzita
Fakulta informatiky

Rozpoznanie užívateľa na základe informácií o HTTP komunikácií

Diplomová práca

Matej Majdiš

Brno, jeseň 2016

Namiesto tejto stránky vložte kópiu oficiálneho podpísaného zadania práce a prehlásenie autora školského diela.

Prehlásenie

Prehlasujem, že táto diplomová práca je mojím pôvodným autorským dielom, ktoré som vypracoval samostatne. Všetky zdroje, pramene a literatúru, ktoré som pri vypracovaní používal alebo z nich čerpal, v práci riadne citujem s uvedením úplného odkazu na príslušný zdroj.

Matej Majdiš

Vedúci práce: doc. RNDr. Vlastislav Dohnal Ph.D. title

Poďakovanie

TODO

Zhrnutie

TODO

Kľúčové slová

keyword1, keyword2, …

Obsah

1 Úvod **1**
1.1 *Aplikácie typu Klient-Server* 2
1.1.1 Klient-Server model 2
1.1.2 Architektúra 3

2 Sieťové vrstvy **5**
2.1 *Aplikačná vrstva* 5
2.2 *Transportná vrstva* 5
2.3 *Sieťová vrstva* 5
2.4 *Vrstva sieťového rozhrania* 5

3 Útoky typu Denial of Service **7**

4 Existujúce prístupy k identifikácií **9**
4.1 *Využitie sieťovej vrstvy* 9
4.1.1 Internet Protocol (IP) 9
4.1.2 Nedostatky 9
4.2 *Aplikačné identifikátory* 9

5 Tvorba unikátneho identifikátoru **11**
5.1 *Možnosti protokolu HTTP* 11
5.2 *Možnosti TCP* 11
5.3 *Popis Algoritmu* 11
5.4 *Záver* 11

Register **13**

A Príloha **13**

Zoznam tabuliek

Zoznam obrázkov

1.1 Vizualizácia pomeru počtu užívateľov webových a desktopových aplikácií v čase, zdroj: vlastné spracovanie *1*
1.2 Schéma znázorňuje základnú architektúru modelu Klient-Server, zdroj: vlastné spracovanie *2*
1.3 Grafické znázornenie a popis priebehu komunikácie 2-tier architektúry, zdroj: vlastné spracovanie *3*
1.4 Grafické znázornenie a popis priebehu komunikácie 3-tier architektúry, zdroj: vlastné spracovanie *4*

1 Úvod

Problematika jednoznačnej identifikácie používateľa je dnes veľmi dôležitou a riešenou témou. Jedným z hlavných dôvodov je fakt, že väčšina dnešných existujúcich, prípadne novo vznikajúcich systémov a aplikácií je nejakým spôsobom zapojená do Internetu. Zároveň zaznamenávame nárast aplikácií, ktoré poskytujú užívateľom webové rozhranie a ústup takzvaných desktopových aplikácií.

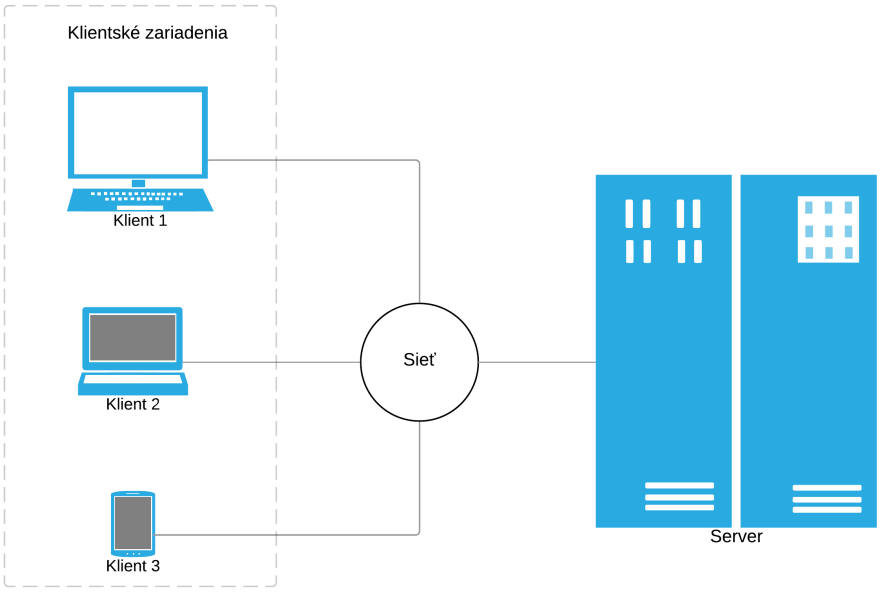
Obr. 1.1: Vizualizácia pomeru počtu užívateľov webových a desktopových aplikácií v čase, zdroj: vlastné spracovanie

Z tohto vyplýva potreba rozoznania a identifikácie používateľov, ktorý s danou aplikáciou interagujú. Existuje niekoľko rôznych prístupov k identifikácií, od mapovania IP adries sieťovej vrstvy až po aplikačnú správu užívateľských účtov. Podrobne sa nimi zaoberá kapitola 4. Cieľom tejto práce je vytvoriť unikátny identifikátor na základe informácií dostupných z *HTTP* protokolu. Pred zostavením samotného algoritmu je preto dôležité popísať niektoré kľúčové oblasti a postupy.

Nasledujúce kapitoly sa preto budú stručne zaoberať fungovaním aplikácií typu klient-server, modelom sieťových vrstiev či útokmi typu *Denial of Service*. Ďalej v práci popíšem spomínané existujúce prístupy a vlastný návrh algoritmu identifikácie užívateľa.

1.1 Aplikácie typu Klient-Server

S pokračujúcim vývojom nových technológií sa Web stáva stále väčšou súčasťou našich životov. Web taktiež už nie je limitovaný prehliadaním na počítačoch. Musí sa prispôsobovať rôznym novým technológiám, ako sú napríklad mobilné, či iné zriadenia. Najčastejšie používaným modelom komunikácie pre architektúru webových aplikácií je tzv. Klient-Server model. Základnou myšlienkou tohto modelu je zaslanie požiadavku (*requestu*) klientom na server, ktorý vystupuje ako poskytovateľ služby.



Obr. 1.2: Schéma znázorňuje základnú architektúru modelu Klient-Server, zdroj: vlastné spracovanie

1.1.1 Klient-Server model

Pretože Klient-Server model je používaný rôznymi typmi aplikácií bolo nutné použiť štandardizované protokoly, na základe ktorý ch bude možné komunikovať. Základné používané protokoly sú: *FTP*

(File Transfer Protocol), *Simple Mail Transfer Protocol (SMTP)* a *Hypertext Transfer Protocol (HTTP)*. Bližšie sieťové vrstvy a jednotlivé protokoly popisuje kapitola 2.

1.1.2 Architektúra

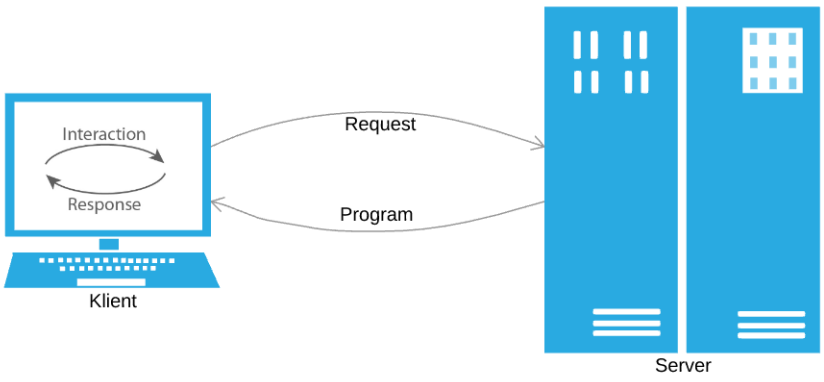
Architektúra modelu Klient-Server sa vo všeobecnosti typicky skladá z troch častí:

- Aplikačný server
- Databázový server
- Zariadenie klienta

Zároveň Existujú dva základné typy architektúr:

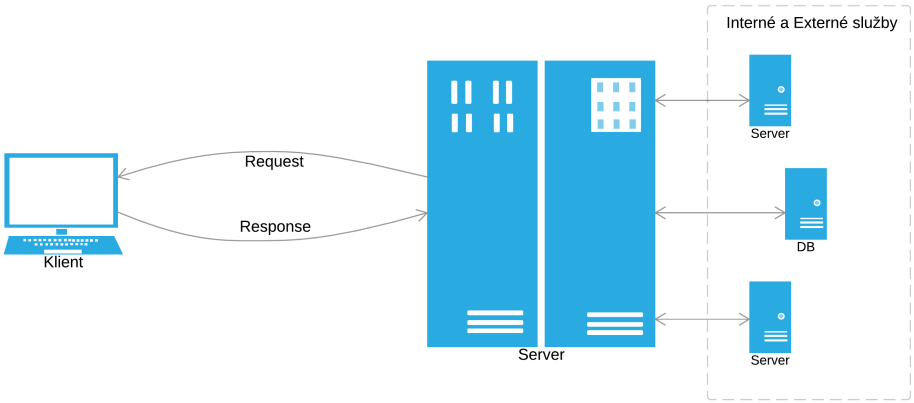
- 2-stupňová *(2-tier)*
- 3-stupňová *(3-tier)*

2-tier architektúra zahrna len zariadenie klienta a databázový server. U tohoto typu architektúry je aplikácia spustená na zariadení klienta, ktoré sa následne pripája priamo na server. Zariadenie tak obsluhuje zároveň *business* logiku aj zobrazovanie aplikácie. Inak tento typ architektúry nazývame aj tučný klient(*thick client*).



Obr. 1.3: Grafické znázornenie a popis priebehu komunikácie 2-tier architektúry, zdroj: vlastné spracovanie

3-tier architektúra, ktorou sa budem zaoberať v tejto práci sa od *2-tier* líši najmätým, že okrem zariadenia klienta a databázového servera zahŕňa aj aplikačný server. Tento je následne používaný na obsluhu *business* logiky aplikácie a komunikáciu s databázou, pričom zariadenie klienta slúž len na zobrazovanie. Iný názov pre takýto typ architektúry je tenký klient (*thincient*).



Obr. 1.4: Grafické znázornenie a popis priebehu komunikácie 3-tier architektúry, zdroj: vlastné spracovanie

Nasledujúce kapitoly tejto práce sa budú zaoberať jedným z najdôležitejších problémov Webových aplikácií, ktorým je identifikácia užívateľa. Najskôr v kapitole 2 popisujem jednotlivé sieťové vrstvy a protokoly, ktorých informácie je možné použiť ka následnej identifikácií. Ďalšou časťou je zhrnutie existujúcich prístupov k rozoznaniu užívateľov v kapitole 4 a popis útokov typu DOS v kapitole 3. Najdôležitejšou časťou je však samozrejme kapitola 5, ktorá popisuje návrh samotného algoritmu identifikátoru.

2 Sieťové vrstvy

//TODO - Úvod

2.1 Aplikačná vrstva

2.2 Transportná vrstva

2.3 Sieťová vrstva

2.4 Vrstva sieťového rozhrania

3 Útoky typu Denial of Service

4 Existujúce prístupy k identifikácií

//TODO - Úvod

4.1 Využitie sieťovej vrstvy

//TODO - Úvod

4.1.1 Internet Protocol (IP)

4.1.2 Nedostatky

4.2 Aplikačné identifikátory

5 Tvorba unikátneho identifikátoru

//TODO - Úvod

5.1 Možnosti protokolu HTTP

5.2 Možnosti TCP

5.3 Popis Algoritmu

5.4 Záver

A Príloha

Here you can insert the appendices of your thesis.