

Implementácia firemnej siete

Semestrálna práca

Andrej Hucík, Miroslav Kozák, Andrej Šišila

Katedra informačných sietí
Žilinská Univerzita v Žiline - Fakulta riadenia a informatiky
Žilina
2017

Obsah

1	Úvod	4
2	Ciele práce	5
3	Topológia siete	6
4	Linux	7
4.1	Inštalácia operačného systému	7
4.2	Základná konfigurácia	8
4.3	VLAN	8
4.4	DNS	9
4.5	DHCP	13
4.6	NTP	13
4.7	Web server	15
4.7.1	Joomla	16
4.7.2	Mediawiki	16
4.8	Poštový server	17
4.9	Firewall	17
4.9.1	Spúšťanie iptables skriptu po štarte	21
5	Windows	23
5.1	Inštalácia operačného systému	23
5.2	Základná konfigurácia	26
5.3	DNS	26
5.4	DHCP	27
5.5	NTP	28
5.6	Web server	28
5.7	Poštový server	30
5.8	NAT	31
6	Záver	32

Zoznam obrázkov

1	Logická topológia siete	6
2	Verejné DNS záznamy	10
3	Vnútorne DNS záznamy	11
4	Konfigurácia DNS pohľadov	11
5	Konfigurácia DNS forwardera	12
6	Konfiguračný súbor pre DHCP	13
7	Konfigurácia NTP servera	14
8	Konfigurácia NTP na klientovi	14
9	Kontrola externých NTP serverov	15
10	Stav synchronizácie s NTP serverom	15
11	Konfiguračný súbor webstránky	17
12	Prehľad konfiguračných súborov webstránok	17
13	Server Manager	24
14	Inštalácia DHCP, DNS a IIS	24
15	Spôsob inštalácie novej role pre Windows Server	25
16	Nainštalované role pre systém Windows Server	25
17	DNS záznamy	27
18	Konfigurácia DHCP	28
19	Inštalácia webového servera IIS	29
20	Zdrojový kód stránky "web2"	30
21	Ukážka webových stránok "web1" (vpravo) a "web2" (vľavo)	30
22	Inštalácia poštového servera MainEnable	31
23	Povolenie vzdialeného prístupu	31

Kapitola 1

Úvod

Základom každej firmy, či už malej, strednej alebo veľkej, je stabilná a bezpečná sieťová infraštruktúra.

V našej práci sa budeme zaoberať implementáciou a konfiguráciou menšej firemnej siete, ktorá bude pozostávať z jedného smerovača, jedného prepínača, dvoch serverových systémov a dvoch pracovných staníc. Potom na potrebných uzloch nastavíme potrebné služby, ktoré umožnia firme poskytovať služby pre vnútornú, ale aj vonkajšiu sieť.

Kapitola 2

Ciele práce

Hlavným cieľom našej práce je vytvorenie jednoduchej sieťovej infraštruktúry, ktorú bude možné nasadiť do podnikového prostredia. Vytvorenie a konfigurácia takejto siete si vyžadovalo splnenie nasledovných čiastkových cieľov:

- Inštalácia operačného systému na serverové systémy, pracovné stanice a smerovač, a ich následná základná konfigurácia
- DNS: Master/Slave riešenie s replikáciou a overením
- Podpora viacerých virtuálnych web serverov a s inštaláciou niektorej z web aplikácii typu CMS or Wiki
- Firewall na smerovači cez netfilter s NAT
- Firewall na server systémoch
- NTP čas pre firmu
- SSH prístup
- DHCP
- Email

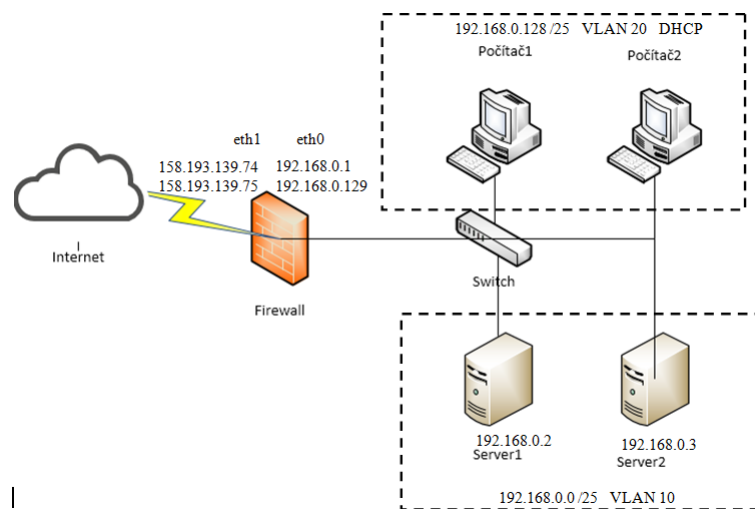
Služby bolo potrebné sprevádzkovať najprv na linuxovej distribúcii Debian 8.6.0 x64 Stable a potom aj na Windows Server 2016, preto je aj táto práca rozdelená na dve hlavné časti: konfiguráciu v prostredí Windows a Linux.

Kapitola 3

Topológia siete

Naša topológia siete pozostáva z firewallu ku ktorému je na jednom rozhraní pripojený prepínač ku ktorému su napojené dva servery a dva počítače. Na druhom rozhraní je Internet. Na linuxoch aj na windowsoch bola rovnaká topológia s výnimkou, že pri pracovaní s linuxom sa nachádzali počítače a servery v rozdielnych VLAN: Servery vo VLAN-e 10 a desktopy vo VLAN-e 20.

Keďže si Windows Server nerozumel s VLAN-ami, topológia siete sa líši v tom, že všetky koncové uzly sú v defaultnej VLAN-e (VLAN 1).



Obr. 1: Logická topológia siete

Kapitola 4

Linux

4.1 Inštalácia operačného systému

Ako serverový aj klientský operačný systém sme použili linuxovú distribúciu “Debian 8.6.0 x64 - kódové meno ‘Jessie’ ” Najprv sme si stiahli inštalačný iso súbor s “Debian 8.6.0 stable 64 bit” vo verzii “netinst” t.j. operačný systém si sťahuje aktuálne balíčky z internetu.

V rámci nastavení počítačov (serverov a desktopov) vo VirtualBox-e sme zmenili tieto nastavenia:

- System -> Motherboard -> Base memory = 1024MB
- System -> Processor -> Processor(s) = 1 CPU
- System -> Processor -> Enable PAE/NX = zaškrtnúť
- System -> Acceleration -> Hardware Virtualization -> Enable VT-x/AMD-V = zaškrtnúť
- Storage -> Controller: IDE -> klikneme na Empty disk -> v pravom paneli klikneme na ikonku CD a vyberieme možnosť Choose Virtual Optical Disk File. Otvorí sa dialógové okno, v ktorom vyhľadáme stiahnutý iso súbor.
- Pre servery a desktopy: Network -> Adapter 1 -> Enable Network Adapter a nastavíme Attached to -> Bridged Adapter -> eth1. Sieťové nastavenia pre firewall sú rovnaké, ako pre desktopy a servery, s tým, že Adapter 1 sme nastavili na Bridged Adapter pripojený na eth1 (lokálne rozhranie) a navyše sme zapli aj Adapter 2 ako Bridged Adapter pripojený na eth0 (verejné rozhranie).

Po nastavení virtuálky sme ju spustili. Ďalej sme postupovali v štandardnej inštalácii Debian-u. Ako doménu sme nastavili takú, ktorá nám bola pridelená: “sos3.local”. Ako hostname sme nastavili doménové meno, ktoré je odvodené od jej funkcie napr. všetky servery majú označenie “SX” a desktopy “DX”, kde X je poradové číslo servera/desktopu. Z doplnkových balíkov sme pre servery neinštalovali nič, jedine pre desktopy sme nainštalovali grafické rozhranie XFCE.

4.2 Základná konfigurácia

Každý inštalácia Debian-u obsahovala navyše tieto balíčky (inštalovali sme ich príkazom “sudo apt-get install”):

```
tcpdump vim openssh-client openssh-server
```

4.3 VLAN

Servery sú vo VLAN 10, desktopy vo VLAN 20. Smerovanie medzi VLANami je vykonávané na FW. Preto sme na FW museli nainštalovať balíčky “isc-dhcp-server” a “vlan” t.j. “apt-get install isc-dhcp-server vlan”. Potom sme editovali súbor “/etc/network/interfaces” tak, že sme odstránili adresné informácie z vnútorného interfacu eth1, ale nechali sme “auto eth1”, aby sa port zapol (UP). Následne sme pridal subinterface eth1.10 pre VLAN 10 (servery) a eth1.20 pre VLAN 20 (desktopy). Adresný rozsah pre jednotlivé VLAN bola sieť 192.168.0.0/24 rozdelená na dve /25 siete: 192.168.0.0 - 192.168.0.127 pre VLAN 10 a 192.168.0.128 - 192.168.0.255 pre VLAN 20

FW je DHCP Relay server. Všetky DHCP požiadavky prepošle FW DHCP serveru, ktorý prideli klientovi IP adresu a ďalšie nakonfigurované informácie. Týmto spôsobom je FW zodpovedný iba za filtrovanie premávky a server za služby poskytované na sieti. IP adresa DHCP servera sa do konfiguračného súboru “/etc/default/isc-dhcp-relay” DHCP relay agenta musí zadať BEZ úvodzoviek a musíme počúvať na obidvoch podrozhraniach t.j. eth1.10 aj eth1.20. Konfigurácia DHCP Relay servera je uvedená nižšie.

```
# Defaults for isc-dhcp-relay initscript
# sourced by /etc/init.d/isc-dhcp-relay
# installed at /etc/default/isc-dhcp-relay by the maintainer scripts

#
# This is a POSIX shell fragment
#

# What servers should the DHCP relay forward requests to?
# Forwarduj DHCP požiadavky na S2
SERVERS=192.168.0.3

# On what interfaces should the DHCP relay (dhrelay) serve DHCP requests?
#Pocuvame DHCP REQUESTY na vlane 20 pre desktopy, ale aj na vlane 10, aby sme mohli REQU
INTERFACES="eth1.10 eth1.20"

# Additional options that are passed to the DHCP relay daemon?
OPTIONS=""
```

4.4 DNS

Systém názvov domén alebo systém mien domén, alebo systém doménových mien (Domain Name System), skr. DNS, je systém, ktorý ukladá prístup k informácii o názve stroja (hostname) a názve domény v istej distribuovanej databáze v počítačových sieťach ako internet. Najdôležitejšie je, že poskytuje mechanizmus získania IP adresy pre každé meno stroja (lookup) a naopak (reverse), a uvádza poštové servery (MX záznam) akceptujúce poštu pre danú doménu.

DNS poskytuje na internete všeobecne dôležitú službu, pretože kým počítače a sieťový hardvér pracujú s IP adresami, ľudia si vo všeobecnosti ľahšie pamätajú mená strojov a domén pri použití napr. v URL a e-mailovej adrese (obzvlášť nepríjemné by to bolo pri IPv6 adrese). DNS tak tvorí prostredníka medzi potrebami človeka a softvéru.

V rámci našej doménovej zóny “sos3.local” sme si museli nastaviť dva DNS servery: Master (S1) a Slave (S2). Slave zrkadlí hlavný DNS server a v prípade poruchy ho zastúpi. Keďže si Slave DNS server všetko stiahne z Master DNS servera, netreba ho primárne konfigurovať, ale stačí mu nastaviť “allow-transfer” na privátnu IP Master DNS.

Na obidva servery sme nainštalovali DNS server a nástroje na overenie jeho funkčnosti príkazom “apt-get install bind9 bind9utils dnsutils”.

Master DNS serveru sme upravovali súbory “/etc/resolv.conf” (konfigurácia adries DNS serverov), “/etc/bind/master/db.sos3.local” (view-lokálny), “/etc/bind/master/db.sos3.external” (view-verejný), “/etc/bind/named.conf.local” (definovanie lokálnych a verejných DNS View). Obsah súboru “/etc/resolv.conf” je uvedený nižšie.

```
domain sos3.local
nameserver 192.168.0.2
nameserver 192.168.0.3
```

V adresári “/etc/bind” na S1 sme vytvorili adresár “master”, do ktorého sme ukladali zónové súbory pre DNS.

Views sme nastavovali na Master DNS serveri súbormi “/etc/bind/named.conf.local”, “/etc/bind/master/db.sos3.local”, “/etc/bind/master/db.sos3.external”. Pri dotazovaní na doménové meno nášho DNS zvnútra sa použijú privátne adersy DNS serverov zo súboru “/etc/bind/master/db.sos3.local”. Pri dotazovaní na doménové meno nášho DNS zvonku sa použijú verejné adersy DNS serverov zo súboru “/etc/bind/master/db.sos3.external”. O tom, aký súbor sa použije, rozhoduje súbor “/etc/bind/named.conf.local”

Počítače v lokálnej sieti sa dokážu navzájom pingovať pomocou svojich hostname. Preklad hostname názvov na IP adresy je definovaný v súbore “/etc/bind/master/db.sos3.local”.

Firewall bol nakonfigurovaný tak, aby prepúšťal DNS požiadavky na lokálnej sieti, a tiež aby prepúšťal požiadavky z internetu na obidva DNS servery t.j. aby boli obidva DNS servery viditeľné zvonku (PREROUTING). Záznamy pre DNS sú pre obidve verejné IP adresy pre udp aj tcp port 53 (zdrojový aj cieľový).

V prípade, že sa vyskytli problémy, skúšali sme vypnúť firewall, kontrolovali sme konfiguračné súbory Master DNS servera príkazmi “named-checkconf” a “named-checkzone” a príkazom “tcpdump” sme monitorovali prenášané správy. Pri každej zmene konfiguračných súborov bolo treba reštartovať bind9 / isc-dhcp-server / interface.

Zdroje:

https://www.howtoforge.com/two_in_one_dns_bind9_views

```
GNU nano 2.2.6          File: db.sos3.external          Modified
;
$TTL      1H
@         IN      SOA      ns1.sos3.local. hostmaster.sos3.local. (
                        2016112202      ; Serial
                        604800          ; Refresh
                        86400           ; Retry
                        2419200         ; Expire
                        604800 )        ; Negative Cache TTL
;
@         IN      NS       ns1.sos3.local.
@         IN      NS       ns2.sos3.local.
ns1       IN      A        158.193.139.74
ns2       IN      A        158.193.139.75
;web1     IN      A        158.193.139.74
web1      IN      CNAME    ns1
web2      IN      CNAME    ns1
@         IN      MX       10      mail.sos3.local.
mail      IN      A        158.193.139.74

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 2: Verejné DNS záznamy

```
GNU nano 2.2.6 File: db.sos3.local Modified
$TTL      1H
@          IN      SOA      ns1.sos3.local. hostmaster.sos3.local. (
                                2016112202      ; Serial
                                604800           ; Refresh
                                86400            ; Retry
                                2419200          ; Expire
                                604800 )         ; Negative Cache TTL
;
@          IN      NS       ns1.sos3.local.
@          IN      NS       ns2.sos3.local.
ns1        IN      A        192.168.0.2
;ns1       IN      CNAME     server1
server1    IN      A        192.168.0.2
ns2        IN      A        192.168.0.3
;firewall  IN      A        192.168.0.1
server2    IN      A        192.168.0.3
web1       IN      CNAME     ns1
web2       IN      CNAME     ns1
@          IN      MX       10      mail.sos3.local.
mail       IN      A        192.168.0.3

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

Obr. 3: Vnútročné DNS záznamy

```
secret "cmFuZG9t";
};

view "local" {
    match-clients { !key fero; 192.168.0.0/24; 127.0.0.1; };
    zone "sos3.local" {
        type master;
        file "/etc/bind/master/db.sos3.local";
        allow-query { any; };
        allow-update { 192.168.0.0/24; };
        allow-transfer { 192.168.0.3; };
    };
};

view "external" {
    match-clients { key fero; 192.168.0.3; !192.168.0.0/24; any; };
    server 192.168.0.3 { keys fero; };
    zone "sos3.local" {
        type master;
        allow-query { any; };
        file "/etc/bind/master/db.sos3.external";
        allow-transfer { 192.168.0.3; };
    };
};
};

26,2 Bot
```

Obr. 4: Konfigurácia DNS pohľadov

```
GNU nano 2.2.6      File: named.conf.options

    forwarders {
        158.193.152.2;
    };
allow-recursion {any;};
//=====
// If BIND logs error messages about the root key being expired,
// you will need to update your keys.  See https://www.isc.org/bind-keys
//=====
dnssec-validation no;

auth-nxdomain no;      # conform to RFC1035
listen-on-v6 { any; };
};

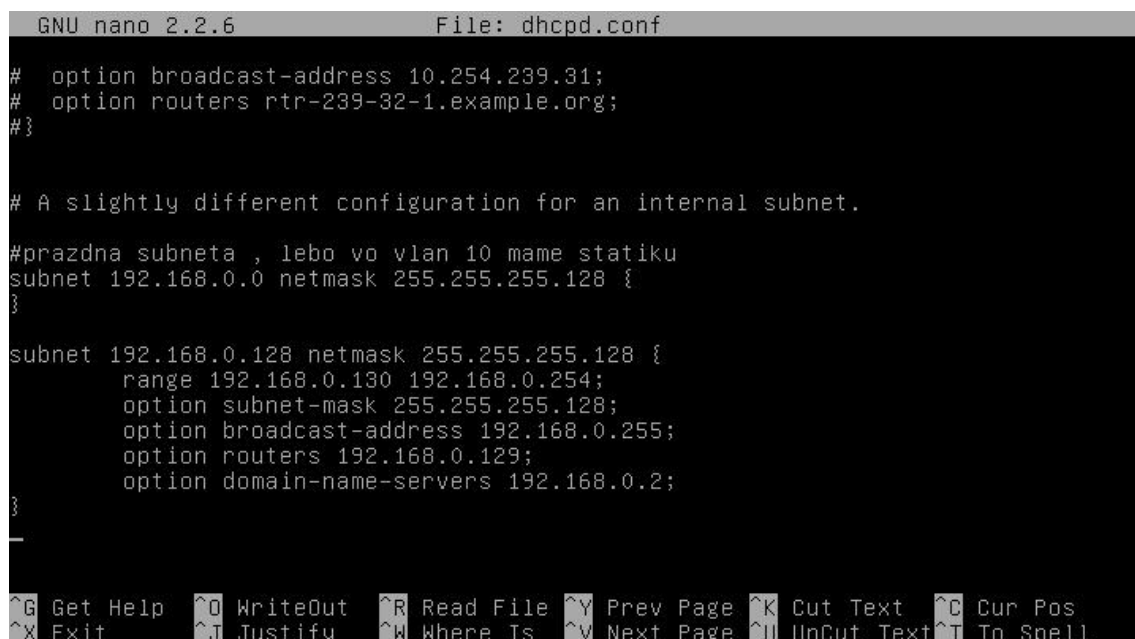
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 5: Konfigurácia DNS forwardera

4.5 DHCP

DHCP server (S2) sme museli upraviť tak, aby prideloval aj DNS adresy serverov. Súbor “/etc/dhcp/dhcpd.conf” na S1 sme upravili tak, že sme doň pridali privátne IP adresy DNS serverov (option domain-name-servers). Do časti pre podsieť sme definovali názvy týchto serverov. Voľbu “option host-name” sme zmenili z pôvodného “example.org” na “sos3.local”. Tým, že sme nastavili DNS server, nemusíme meniť na jednotlivých hostoch súbor “/etc/resolv.conf”.

Dynamic Host Configuration Protocol (DHCP) je súbor zásad, ktoré využívajú komunikačné zariadenia (počítač, router alebo sieťový adaptér), umožňujúci zariadeniu vyžiadať si a získať IP adresu od servera, ktorý má zoznam adres voľných na použitie. DHCP Server (Dynamic Host Configuration Protocol) vykonáva automatické pridelenie IP adres svojim klientom. Môžu to byť akékoľvek systémy, podporujúce DHCP. DHCP je štandardný protokol, môžu ho využívať aj systémy mimo Microsoft. Z Microsoft operačných systémov podporujú funkciu DHCP klienta všetky až na veľmi exotický LAN Manager pre OS / 2. V rámci siete potom máme DHCP Server - pridelujúca adresy a počítače - ktoré je od neho preberajú (DHCP Clients). V sieti môžu byť aj počítače, ktoré majú tieto adresy nastavené manuálne.



```
GNU nano 2.2.6      File: dhcpd.conf

# option broadcast-address 10.254.239.31;
# option routers rtr-239-32-1.example.org;
#}

# A slightly different configuration for an internal subnet.
#prazdna subneta , lebo vo vlan 10 mame statiku
subnet 192.168.0.0 netmask 255.255.255.128 {
}

subnet 192.168.0.128 netmask 255.255.255.128 {
    range 192.168.0.130 192.168.0.254;
    option subnet-mask 255.255.255.128;
    option broadcast-address 192.168.0.255;
    option routers 192.168.0.129;
    option domain-name-servers 192.168.0.2;
}

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 6: Konfiguračný súbor pre DHCP

4.6 NTP

NTP(Network Time Protocol) je protokol na synchronizáciu všetkých počítačov pripojených do vnútornej siete. Tento protokol zaisťuje, aby všetky počítače v sieti mali rovnaký a presný čas. Bol navrhnutý aby odolával následkom premenlivého zdržania pri doručovaní paketov. NTP používa UDP na porte 123. NTP server sme zvolili server2, ktorý má ip adresu 192.168.0.3. Nainštalovali sme NTP príkazom apt-get install ntp. Na klientoch sme nainštalovali NTP pomocou príkazu apt-get install

ntp ntpdate. V súbore na serveri s2 /etc/ntp.conf sme pridali slovenské servery zo stránky www.pool.ntp.org/zone/sk. a ostatné servery sme zakomentovali. Klienti si z master serveru aktualizujú čas.

```
GNU nano 2.2.6      File: /etc/ntp.conf

filegen clockstats file clockstats type day enable

# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example

# pool.ntp.org maps to about 1000 low-stratum NTP servers.  Your server will
# pick a different set every time it starts up.  Please consider joining the
# pool: <http://www.pool.ntp.org/join.html>
#server 0.debian.pool.ntp.org iburst
#server 1.debian.pool.ntp.org iburst
#server 2.debian.pool.ntp.org iburst
#server 3.debian.pool.ntp.org iburst
server 192.168.0.3
restrict 192.168.0.3 mask 255.255.255.0

#ignore all
restrict default ignore

# Access control configuration; see /usr/share/doc/ntp-doc/html/acconf.html for
# details.

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 7: Konfigurácia NTP servera

```
GNU nano 2.2.6      File: ntp.conf

filegen clockstats file clockstats type day enable

# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example

# pool.ntp.org maps to about 1000 low-stratum NTP servers.  Your server will
# pick a different set every time it starts up.  Please consider joining the
# pool: <http://www.pool.ntp.org/join.html>
server 2.sk.pool.ntp.org
server 0.europe.pool.ntp.org
server 2.europe.pool.ntp.org
#server 0.debian.pool.ntp.org iburst
#server 1.debian.pool.ntp.org iburst
#server 2.debian.pool.ntp.org iburst
#server 3.debian.pool.ntp.org iburst

# Access control configuration; see /usr/share/doc/ntp-doc/html/acconf.html for
# details.  The web page <http://support.ntp.org/bin/view/Support/AccessRestrictions>
# details.

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 8: Konfigurácia NTP na klientovi

```

64 bytes from 91-235-53-86.s.azet.sk (91.235.53.86): icmp_seq=2 ttl=247 time=14.
3 ms
^C
--- pokec.sk ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 14.329/14.701/15.074/0.391 ms
root@server2:/home/s2# ntpq -p
      remote           refid      st t when poll reach   delay   offset   jitter
=====
ntp2.kajot.cz      5.1.56.123        3 u   2   64    7   15.556  -85.810   59.793
ns3.0x00.lv       194.100.2.194     2 u   1   64    7   57.556  -86.395   54.944
stratum2-4.NTP. 129.70.130.70     2 u   -   64    3   38.957  -88.568   39.425
root@server2:/home/s2# ntpstat
unsynchronised
time server re-starting
polling server every 8 s
root@server2:/home/s2# ntpstat
synchronised to NTP server (85.254.216.1) at stratum 3
time correct to within 1165 ms
polling server every 64 s
root@server2:/home/s2# ntpstat
synchronised to NTP server (85.254.216.1) at stratum 3
time correct to within 455 ms
polling server every 64 s
root@server2:/home/s2# _

```

Obr. 9: Kontrola externých NTP serverov

```

s1@server1:/etc/bind/master$ timedatectl
Local time: Tue 2016-10-25 15:28:54 CEST
Universal time: Tue 2016-10-25 13:28:54 UTC
RTC time: Tue 2016-10-25 13:29:14
Time zone: Europe/Prague (CEST, +0200)
NTP enabled: no
NTP synchronized: yes
RTC in local TZ: no
DST active: yes
Last DST change: DST began at
Sun 2016-03-27 01:59:59 CET
Sun 2016-03-27 03:00:00 CEST
Next DST change: DST ends (the clock jumps one hour backwards) at
Sun 2016-10-30 02:59:59 CEST
Sun 2016-10-30 02:00:00 CET
s1@server1:/etc/bind/master$ _

```

Obr. 10: Stav synchronizácie s NTP serverom

4.7 Web server

Apache HTTP Server je softwarový webový server s Opensource licenciou pre Linux, BSD, Microsoft Windows a iné platformy.

PHP (PHP: Hypertext Preprocessor) je populárny opensource skriptovací jazyk, ktorý sa používa najmä na programovanie klient-server aplikácií (na strane servera) a pre vývoj dynamických webových stránok.

MySQL je slobodný a otvorený viacvláknový, viacúčítateľský SQL relačný databázový server. MySQL je podporovaný na viacerých platformách (ako Linux, Windows či Solaris) a je implementovaný vo viacerých programovacích jazykoch ako PHP, C++ či Perl. Databázový systém je relačný, typu DBMS (database management system). Každá databáza je v MySQL tvorená z jednej alebo z viacerých tabuliek, ktoré majú riadky a stĺpce. V riadkoch sa rozoznávajú jednotlivé záznamy, stĺpce udávajú dátový typ jednotlivých záznamov, pracuje sa s nimi ako s poľami. Práca s MySQL databázou je vykonávaná pomocou takzvaných dotazov, ktoré vychádzajú z programovacieho jazyka SQL (StructuredQueryLanguage).

Na webový server sme použili apache. Apache HTTP Server je softwarový webový server s Opensource licenciou pre Linux, BSD, Microsoft Windows a iné platformy. V dnešnej dobe je najrozšírenejším na celom svete. Pre plnú funkcionality webového servera sme museli nainštalovať apache, mysql, php príkazom “apt-get install apache2 mysql php5” .

V adresári /var/www/ sme vytvorili priečinky s názvami web1 a web2. Kde web1 a web2 predstavovali dva virtuálne webové servery. Následne sme v etc/apache2/sites-available 003-wiki.sos3.local.conf sme pridali cestu ku web stránke /var/www/web1 a ServerName web2.sos3.local. Pre joomlu v súbore 002-joomla.sos3.local.conf sme pridali cestu k adresaru keď už DocumentRoot /var/www/web1 a ServerName web1.sos3.local

Následne do DNS záznamov sme museli pridať:

```
db.sos1.local
web1 IN A 192.168.0.4
web2 IN A 192.168.0.4

db.sos1.public
web1 IN A 158.193.139.74
web2 IN A 158.193.139.74
```

4.7.1 Joomla

V priečinku /var/www/web2 sme stiahli joomlu verziu 3.6 pomocou príkazu “wget https://github.com/.../Joomla_3.6.0-Stable-Full_Package.zip” . V ďalšom kroku sme odzipovali tento súbor príkazom “unzip Joomla_3.6.0-Stable-Full_Package.zip” . Následne sme v prehliadači otvorili web1.sos3.local a podľa príslušných krokov sme nainštalovali joomlu.

4.7.2 Mediawiki

V priečinku var/www/web1 sme stiahli Wikimedia pomocou príkazu “wget https://www.mediawiki.org/wiki/Download/1.2.8.zip” . Následne sme odzipovali tento súbor príkazom “unzip mediawiki-1.2.8.zip” . A v poslednom kroku sme v prehliadači web2.sos3.local nainštalovali mediawiki.


```
GNU nano 2.2.6      File: 003-wiki.sos3.local.conf

<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    ServerName web2.sos3.local
    #Redirect / https://web2.sos3.local

    ServerAdmin webmaster@sos3.local
    DocumentRoot /var/www/web1

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn
    ErrorLog ${APACHE_LOG_DIR}/web1-error.log

^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell
```

Obr. 11: Konfiguračný súbor webstránky

```
root@server1:/etc/apache2# cd sites-available/
root@server1:/etc/apache2/sites-available# ls
000-default.conf          003-wiki.sos3.local.conf
002-web1.sos3.local.conf  003-wiki.sos3.local-ssl.conf
002-web1.sos3.local-ssl.conf default-ssl.conf
root@server1:/etc/apache2/sites-available# _
```

Obr. 12: Prehľad konfiguračných súborov webstránok

4.8 Poštový server

Na poštový server sme použili postfix. Postfix je počítačový program pre unixové systémy pro prepravu elektronickej pošty (MTA).

Najprv bolo potrebné nainštalovať postfix príkazom `apt-get install postfix` prešli sme inštaláciou kde sme nastavili hostname `sos3.local`. Následne sme museli reštartovať postfix `service postfix restart`. V súbore `/etc/postfix/main.cf` je potrebné upraviť `myhostname = sos3.local`, odkomentovať

pridať konfigurak a obrázky

4.9 Firewall

Firewall sme konfigurovali priebežne s úlohami počas semestra. Nižšie je uvedený konfiguračný skript na pridávanie záznamov do “iptables”.

```
#!/bin/bash
```

I=/sbin/iptables

#vycisti povodne pravidla

\$I -F -t filter

\$I -F -t nat

#zakaz vsetko, co nie je vyslovene povolene

#\$I -P INPUT DROP

#\$I -P FORWARD DROP

\$I -P INPUT ACCEPT

\$I -P FORWARD ACCEPT

#povol loopback

\$I -A INPUT -i lo -j ACCEPT

STAVOVY FIREWALL

#FW CONNTRACK - nastav sledovanie aktivnych pripojeni inicializovanych z FW

#Co odide z firewallu, sa vrati na firewall

\$I -A INPUT -i eth0 -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

#VLAN CONNTRACK - aby sa nam vsetko z netu vratilo na VLANy

#Co odide z VLANiek, sa vrati do VLANiek

\$I -A FORWARD -i eth0 -o eth1.10 -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

\$I -A FORWARD -i eth0 -o eth1.20 -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT

SNAT

\$I -t nat -A POSTROUTING -o eth0 -j SNAT --to-source 158.193.139.74

#a co s druhou ipckou? treba na nu tiez prekladat? mozme pridat aj nat na druhu ip

VLAN ROUTING

#Povol routing medzi vlanami

\$I -A FORWARD -i eth1.10 -o eth1.20 -j ACCEPT

\$I -A FORWARD -i eth1.20 -o eth1.10 -j ACCEPT

ICMP

#Povol pingy na fw

\$I -A INPUT -p icmp -j ACCEPT

#Povol pingy z VLANiek von

\$I -A FORWARD -i eth1.10 -o eth0 -p icmp -j ACCEPT

\$I -A FORWARD -i eth1.20 -o eth0 -p icmp -j ACCEPT

SSH

#Povol SSH na firewall zvonku aj zvnutra

\$I -A INPUT -p tcp --dport 22 -j ACCEPT

SSH

#Povol SSH na firewall zvonku aj zvnutra

```

$I -A INPUT -p tcp --dport 22 -j ACCEPT

#Navrat SSH na FW
$I -A INPUT -i eth1.10 -p tcp --sport 22 -j ACCEPT
$I -A INPUT -i eth1.20 -p tcp --sport 22 -j ACCEPT

#Povol spatne SSH z vnutornej siete na FW
$I -A INPUT -i eth1.10 -p tcp --sport 22 -j ACCEPT
$I -A INPUT -i eth1.20 -p tcp --sport 22 -j ACCEPT

#S1 SSH zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p tcp --dport 3002 -j DNAT
--to-destination 192.168.0.2:22
$I -A FORWARD -p tcp -d 192.168.0.2 --dport 22 -j ACCEPT
$I -A FORWARD -p tcp -s 192.168.0.2 --sport 22 -j ACCEPT

#S2 SSH zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p tcp --dport 3003 -j DNAT
--to-destination 192.168.0.3:22
$I -A FORWARD -p tcp -d 192.168.0.3 --dport 22 -j ACCEPT
$I -A FORWARD -p tcp -s 192.168.0.3 --sport 22 -j ACCEPT

#S1 zvnutra von
$I -A FORWARD -d 192.168.0.2 -p tcp --dport 22 -j ACCEPT
$I -A FORWARD -s 192.168.0.2 -p tcp --sport 22 -j ACCEPT

#S2 zvnutra von
$I -A FORWARD -d 192.168.0.3 -p tcp --dport 22 -j ACCEPT
$I -A FORWARD -s 192.168.0.3 -p tcp --sport 22 -j ACCEPT

##### DNS #####
#Povol DNS na FW
$I -A INPUT -p udp --sport 53 -j ACCEPT
$I -A INPUT -p udp --dport 53 -j ACCEPT

#Povol DNS zvnutra von
$I -A FORWARD -i eth1.10 -o eth0 -p tcp --dport 53 -j ACCEPT
$I -A FORWARD -i eth1.10 -o eth0 -p udp --dport 53 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p tcp --dport 53 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p udp --dport 53 -j ACCEPT

#MASTER DNS (S1) lokalne
$I -A FORWARD -d 192.168.0.2 -p tcp --dport 53 -j ACCEPT
$I -A FORWARD -s 192.168.0.2 -p tcp --sport 53 -j ACCEPT
$I -A FORWARD -d 192.168.0.2 -p udp --dport 53 -j ACCEPT
$I -A FORWARD -s 192.168.0.2 -p udp --sport 53 -j ACCEPT
#SLAVE DNS (S2) lokalne
$I -A FORWARD -d 192.168.0.3 -p tcp --dport 53 -j ACCEPT
$I -A FORWARD -s 192.168.0.3 -p tcp --sport 53 -j ACCEPT
$I -A FORWARD -d 192.168.0.3 -p udp --dport 53 -j ACCEPT

```

```
#S1 DNS zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p tcp --dport 53 -j DNAT
--to-destination 192.168.0.2
$I -t nat -A PREROUTING -d 158.193.139.74 -p udp --dport 53 -j DNAT
--to-destination 192.168.0.2

#S2 DNS zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.75 -p tcp --dport 53 -j DNAT
--to-destination 192.168.0.3
$I -t nat -A PREROUTING -d 158.193.139.75 -p udp --dport 53 -j DNAT
--to-destination 192.168.0.3

#Povovl DNS zvoknu dnu
$I -A FORWARD -i eth0 -o eth1.10 -d 192.168.0.2 -p udp --dport 53 -j ACCEPT
$I -A FORWARD -i eth0 -o eth1.10 -d 192.168.0.3 -p udp --dport 53 -j ACCEPT

##### NTP #####
#Povol NTP na FW
$I -A INPUT -p udp --sport 123 -j ACCEPT

#Povol pripojenia na NTP server
$I -A FORWARD -d 192.168.0.3 -i eth1.10 -p udp --sport 123 -j ACCEPT

#Povol NTP vnutri
$I -A FORWARD -d 192.168.0.3 -p udp --dport 123 -j ACCEPT
$I -A FORWARD -s 192.168.0.3 -p udp --sport 123 -j ACCEPT

#NTP DNAT - Povol NTP zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p udp --dport 123 -j DNAT
--to-destination 192.168.0.3:123
$I -t nat -A PREROUTING -d 158.193.139.75 -p udp --dport 123 -j DNAT
--to-destination 192.168.0.3:123

##### HTTP #####
#Povol HTTP von
$I -A FORWARD -i eth1.10 -o eth0 -p tcp --dport 80 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p tcp --dport 80 -j ACCEPT

#HTTP lokalne
$I -A FORWARD -d 192.168.0.2 -p tcp --dport 80 -j ACCEPT
$I -A FORWARD -s 192.168.0.2 -p tcp --sport 80 -j ACCEPT

#HTTP zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p tcp --dport 80 -j DNAT
--to-destination 192.168.0.2

##### HTTPS #####
#Povol HTTPS von
$I -A FORWARD -i eth1.10 -o eth0 -p tcp --dport 443 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p tcp --dport 443 -j ACCEPT
```

```

#HTTPS lokalne
$I -A FORWARD -d 192.168.0.2 -p tcp --dport 443 -j ACCEPT
$I -A FORWARD -s 192.168.0.2 -p tcp --sport 443 -j ACCEPT

#HTTP zvonku dnu
$I -t nat -A PREROUTING -d 158.193.139.74 -p tcp --dport 443 -j DNAT
--to-destination 192.168.0.2

##### FTP #####
#Povol FTP von
$I -A FORWARD -i eth1.10 -o eth0 -p tcp --dport 20 -j ACCEPT
$I -A FORWARD -i eth1.10 -o eth0 -p tcp --dport 21 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p tcp --dport 20 -j ACCEPT
$I -A FORWARD -i eth1.20 -o eth0 -p tcp --dport 21 -j ACCEPT

##### DHCP #####
#Povol DHCP
#$I -I FORWARD -i eth1.10 -p udp --dport 67:68 --sport 67:68 -j ACCEPT
#$I -I FORWARD -i eth1.20 -p udp --dport 67:68 --sport 67:68 -j ACCEPT

##### SMTP #####
$I -t nat -A PREROUTING -p tcp --dport 25 -j DNAT --to-destination 192.168.0.3:25
$I -A FORWARD -p tcp -d 192.168.0.3 --dport 25 -j ACCEPT
$I -A FORWARD -p tcp -s 192.168.0.3 --sport 25 -j ACCEPT

##### IMAP #####
$I -A FORWARD -p tcp -d 192.168.0.3 --dport 143 -j ACCEPT
$I -A FORWARD -p tcp -s 192.168.0.3 --sport 143 -j ACCEPT

# Pridaj logovanie do /var/messages/kern.log kvoli debugingu
$I -A INPUT -j LOG
$I -A FORWARD -j LOG

# Zapni Forwarding pre iptables (aby fungovali "FORWARD" prikazy)
echo 1 > /proc/sys/net/ipv4/ip_forward

```

4.9.1 Spúšťanie iptables skriptu po štarte

Skript na pridávanie záznamov “iptables” spúšťame príkazom “up” vrámci konfigurácií sieťového rozhrania “eth0” v súbore “/etc/network/interfaces”, ktorý je uvedený nižšie.

```

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
allow-hotplug eth0

```

```
iface eth0 inet static
    address 158.193.139.74
    netmask 255.255.255.0
    gateway 158.193.139.1
up /usr/local/etc/firewall.sh

auto eth0:1
allow-hotplug eth0:1
iface eth0:1 inet static
    address 158.193.139.75
    netmask 255.255.255.0
    gateway 158.193.139.1

# Vnutorna (firemna) sietovka
auto eth1

#VLAN 10 subinterface
auto eth1.10
iface eth1.10 inet static
    address 192.168.0.1
    netmask 255.255.255.128
    dns-nameservers 192.168.0.2 192.168.0.3

#VLAN 20 subinterface
auto eth1.20
iface eth1.20 inet static
    address 192.168.0.129
    netmask 255.255.255.128
    dns-nameservers 192.168.0.2 192.168.0.3

# This is an autoconfigured IPv6 interface
iface eth0 inet6 auto
```

Kapitola 5

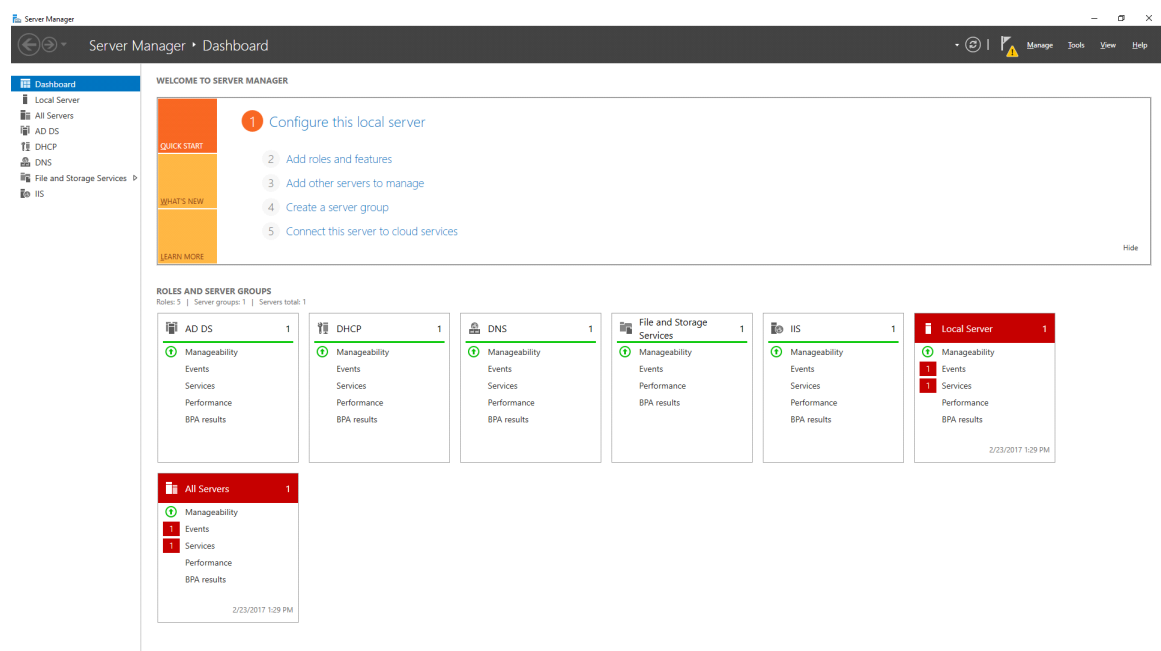
Windows

5.1 Inštalácia operačného systému

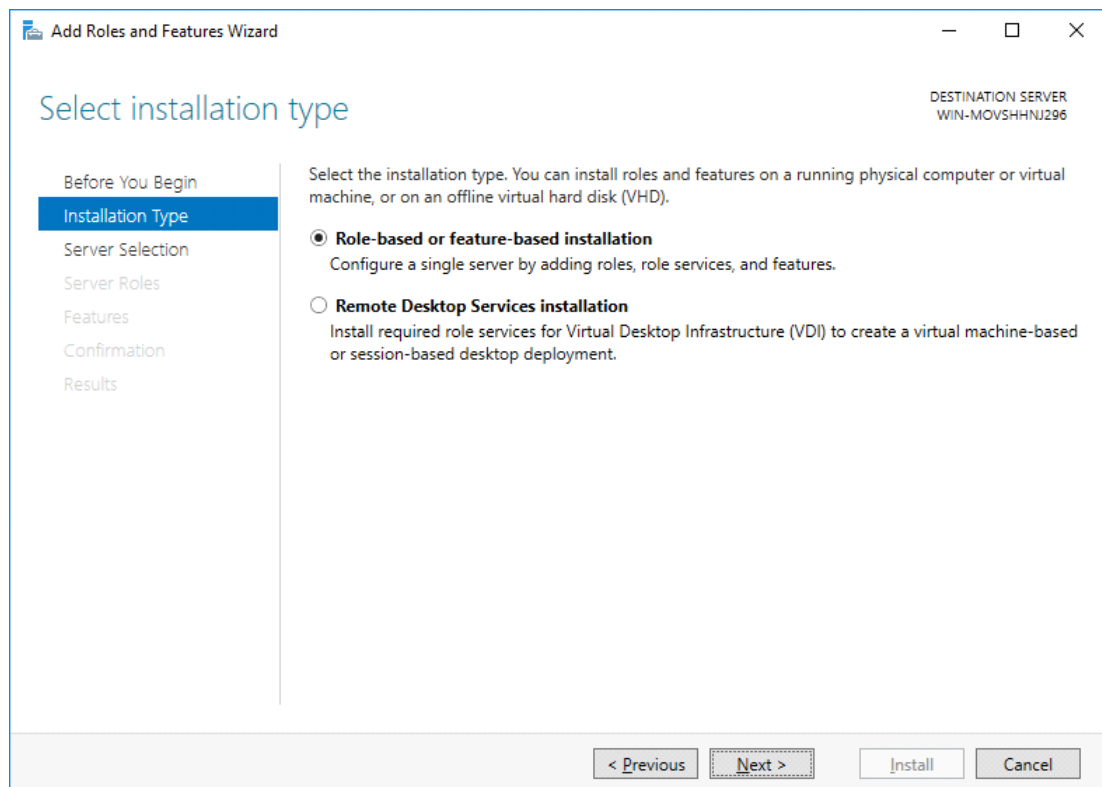
Ako operačný systém pre servery sme nainštalovali Windows Server 2016 x64. Pre desktopy sme použili už predinštalovaný virtuálny stroj s Windows 7 x64.

Windows Server sme v rámci VirtualBox-u nastavili rovnako ako pri Debian-e s tým rozdielom, že sme namiesto "Debian x64" použili šablónu "Windows 10 x64". Servery mali jeden sieťový adaptér a bol nastavený rovnako ako pri Debian serveroch. Dva sieťové adaptéry boli nastavené na Windows Server firewall-e, rovnako ako aj Debian na firewall-e, s rovnakou konfiguráciou. Osobitne sme však museli zmeniť nastavenia v časti Display -> Video Memory na maximálnu možnú hodnotu (128MB s 2D akceleráciou), aby bol zážitok z používania čo najplynulejší. Tiež sme zdvihli množstvo dostupnej operačnej pamäte na 2048MB a počet procesorov na 2. Nakoniec sme v časti Storage pridali inštaláčné médium pre Windows Server.

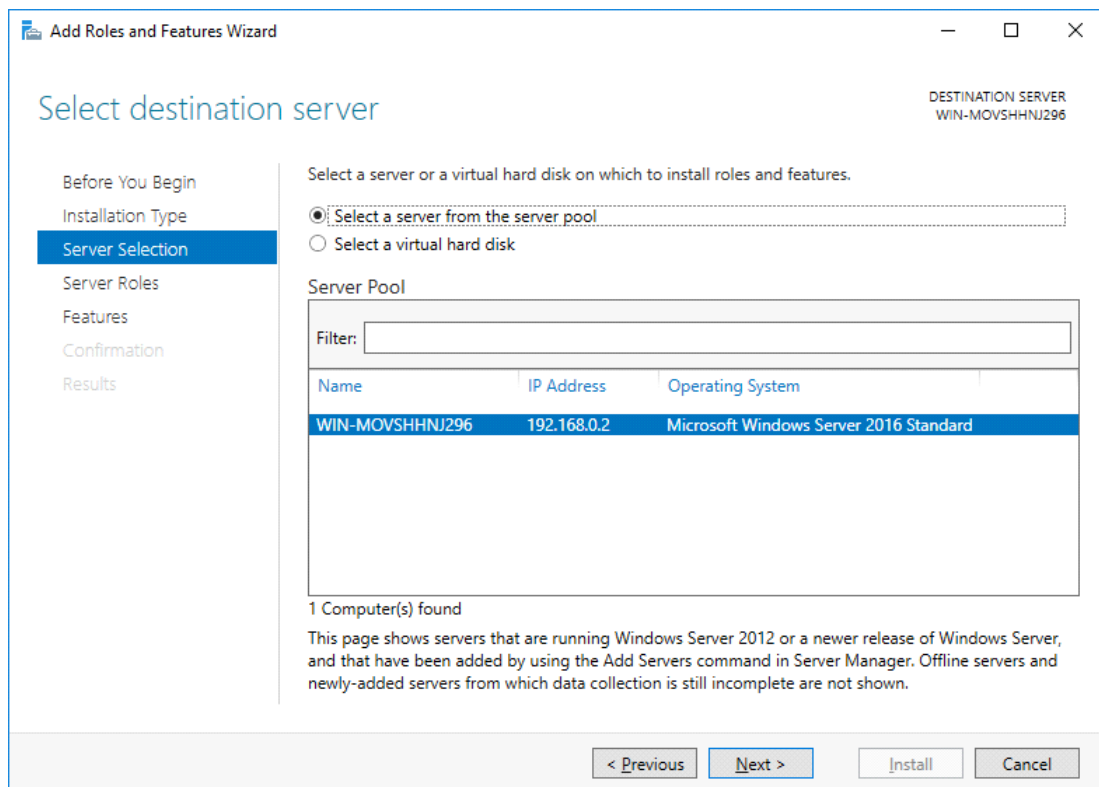
Po vykonaní potrebných nastavení vo VirtualBox-e sme spustili Windows Server virtuálku. Inštalácia bola veľmi jednoduchá - stačilo stále klikať na tlačidlo "Next". Pri voľbe typu inštalácie sme zvolili "Custom installation", vymazali sme všetky existujúce partície, vytvorili jedinou partíciu maximálnej veľkosti. Nastavenia sme potvrdili a Windows sa po chvíli nainštaloval. Po prihlásení sme po chvíli videli aplikáciu "Server Manager", pomocou ktorej budeme spravovať jednotlivé súčasti systému Windows Server.



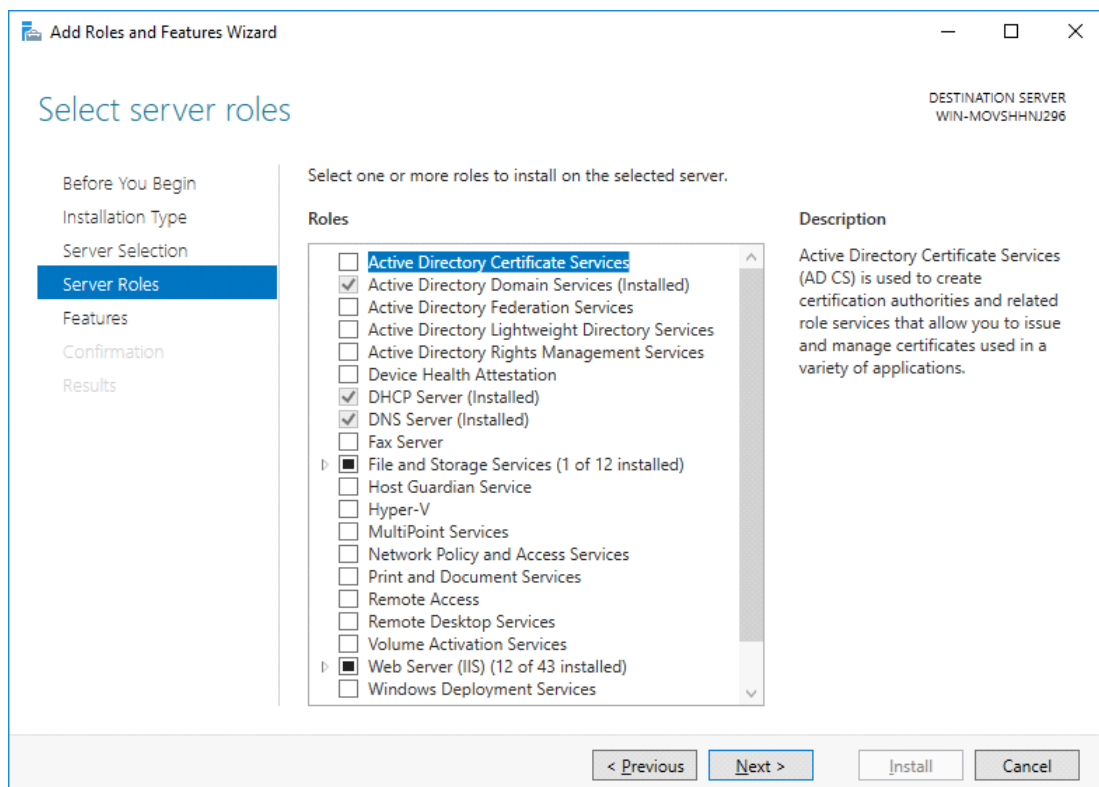
Obr. 13: Server Manager



Obr. 14: Inštalácia DHCP, DNS a IIS



Obr. 15: Spôsob inštalácie novej role pre Windows Server



Obr. 16: Nainštalované role pre systém Windows Server

5.2 Základná konfigurácia

Do Windows Server-u 2016 sme nainštalovali webový prehliadač Mozilla Firefox alebo Google Chrome (podľa preferencie - Internet Explorer bol prakticky nepoužiteľný). Podobne ako Windows 10, aj Windows Server 2016 inštaluje aktualizácie automaticky, pričom sa aktualizácie inštalujú do systému samé od seba, bez vedomia administrátora. Preto sme sa rozhodli automatické aktualizácie vypnúť priamo v nastavení služieb systému Windows. Konkrétne sme vypli službu "Windows Update" tým, že sme jej stav nastavili na "Disabled". Po nastavení sme reštartovali servery.

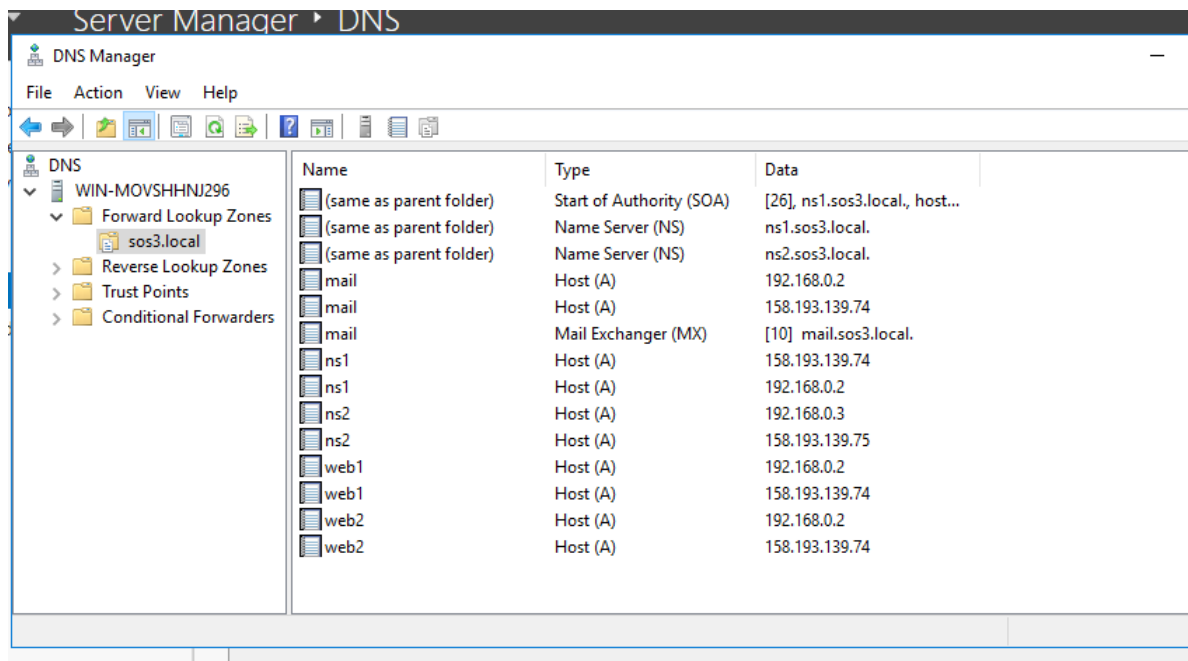
5.3 DNS

V prvom rade sme si zvolili Master a Slave. Master je server1 (192.168.0.2) a Slave server2 (192.168.0.3)

DNS master nainštalujeme pomocou Windows Server Manager. Klikneme na Manage , vyberieme možnosť Add roles and features ďalej Role-based or feature-based installation, zobrazí sa zoznam serverov, my vyberieme náš server a zvolíme zo zoznamu roles DNS Server a dokončíme inštaláciu.

Po inštalácii DNS balíka sme sa dostali cez Tools -> DNS -> Configure a DNS server -> Create a forward lookup zone k vytvoreniu primárnej forward lookup zóny sos1.local , nastavili sme aj nech záznamy preposiela na Slave 192.168.0.3.

Prešli sme k inštalácii DNS Slave. Postup ako pri DNS Master avšak DNS server bolo potrebné nastaviť na slavemode. Vybrali sme Tools -> Forward lookup zones -> New zone. Hneď v prvom kroku sme vybrali možnosť nie Primary zone ale Secondary zone a taktiež meno zóny .V ďalšom kroku určíme DNS Masterserver, v našom riešení ma IP 192.168.0.2 .Dokončíme vytváranie zóny pomocou Next a Finish. O chvíľu si Slave stiahne záznamy z DNS Master servera.



Obr. 17: DNS záznamy

5.4 DHCP

Inštaláciu sme vykonali vo Windows service manager - Add Roles and Features, vybrali si možnosť DHCP server.

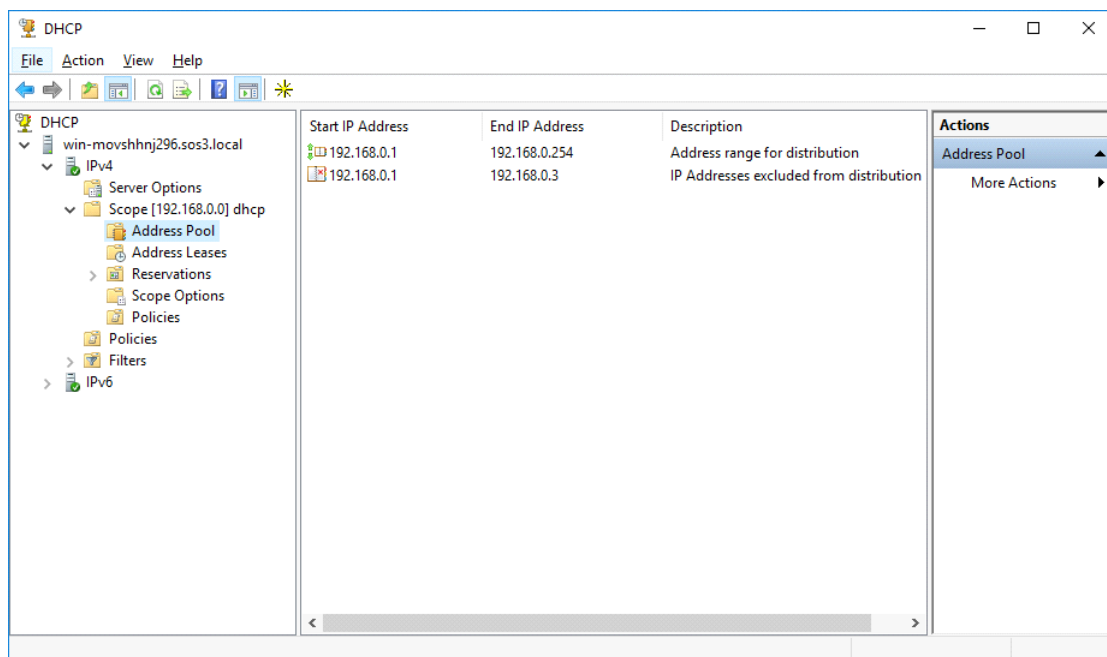
Pre konfiguráciu sme klikli na TOOLS a následne DHCP. Zobrazilo sa nám okno s ponukou, my sme vybrali náš server, IPv4 a možnosť new scope. Spustil sa New Scope Wizard. V prvom kroku sme napísali názov pravidla na pridelenie IP adries. Ďalej sme zvolili rozsah IP adries a masku.

Rozsah IP adries od 192.168.0.1 po 192.168.0.254
Maska 255.255.255.0

Následne sme využili možnosti pridať výnimku z predtým zadaného rozsahu, teda adresy ktoré sa nebudu pridelené napriek tomu, že sú zo nami zadaného rozsahu v predchádzajúcom kroku. Ide o adresy serverov 192.168.0.2 a 192.168.0.3.

Potom sme zvolili aký dlhý čas si server bude pamätať IP adresy ktoré niekomu pridelil. Stačilo nám 5 hodín (dĺžka cvičenia aj s rezervou).

Nakoniec sme nastavili bránu na „192.168.1.1“, pridali sme IP adresy našich DNS serverov, teda „192.168.1.2“ a „192.168.1.3“. , a dokončili inštaláciu kliknutím na Finish.



Obr. 18: Konfigurácia DHCP

5.5 NTP

Na spustenie NTP na Windows servery sme museli vykonať zmeny v registroch. Spustíme okno RUN (WIN+R) , kde napíšeme regedit. Následne sa dostaneme cestou HKEY LOCAL MACHINE — SYSTEM — CurrentControlSet — Services — W32Time — TimeProviders — NtpServer až k hodnote Enabled , ktorá bola nastavená na 0 , a my ju zmeníme na 1. Využijeme opäť win+R , zadáme w32tm /config /update, čím vlastne spustíme NTP server na danom zariadení.

Na aplikáciu zmien sme reštartovali Windows Timeservice príkazom zadaným do commandline:

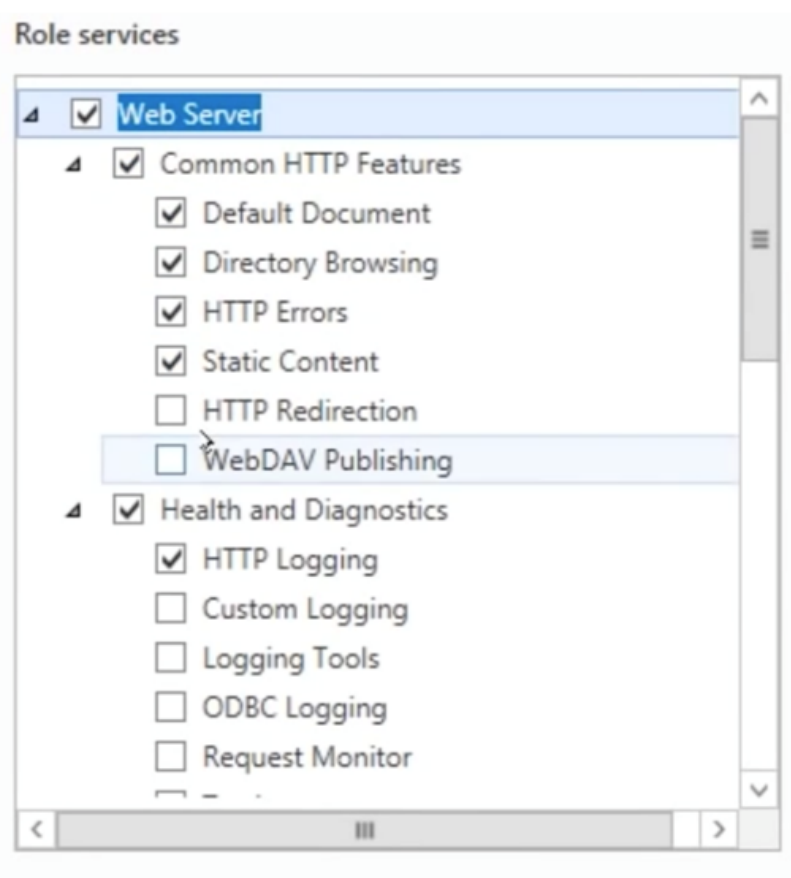
```
net stop w32time && net start w32tim.
```

5.6 Web server

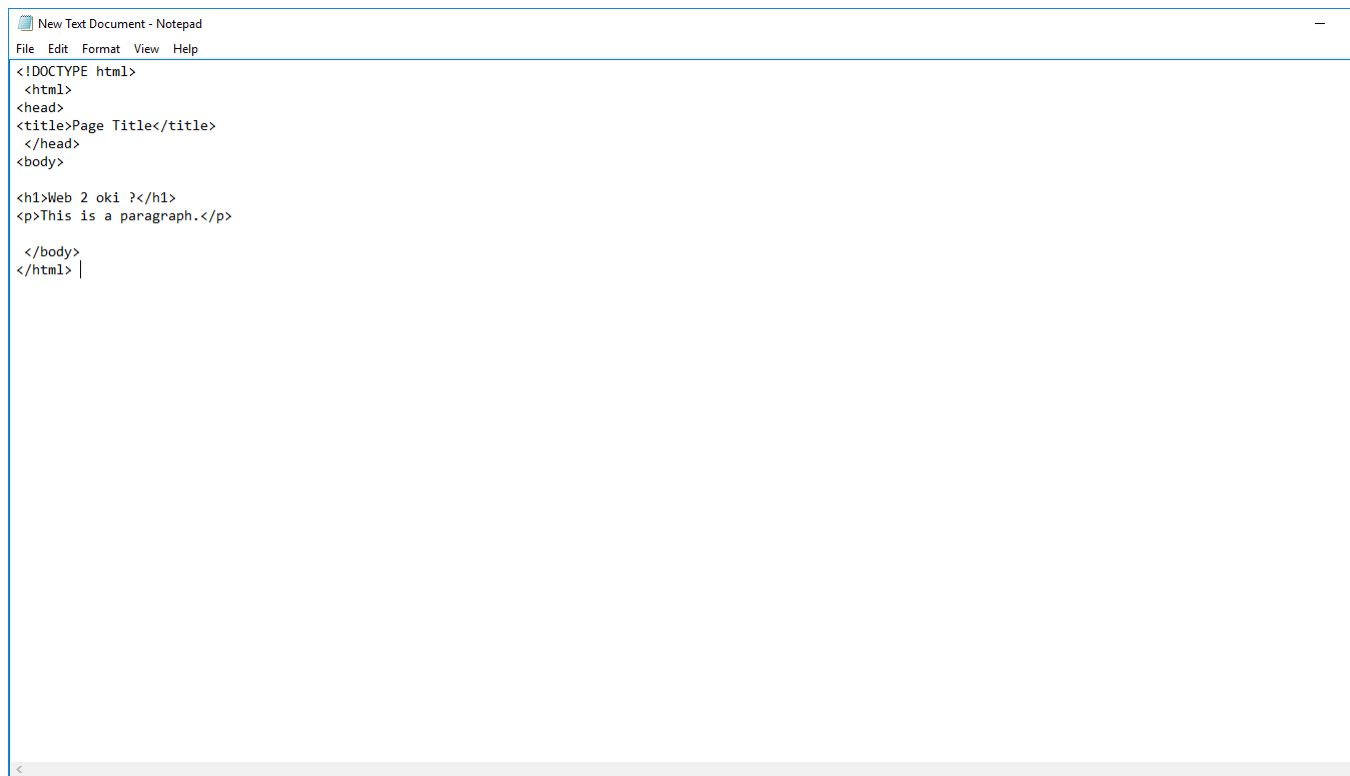
Webserver IIS (Internet Information Server) sme pridali cez windows server manager tlačidlom Add roles and features, kde sme vyhľadali Web Server ISS a pokračujeme ďalej. Pri ponuke Role Services

Následne nainštalujeme služby na server. Po úspešnej inštalácii sa IIS objaví na ľavom paneli v server manager-i. Klikneme na ikonu IIS a v zozname dostupných serverov sa zjaví jeden - ten, na ktorom uskutočňujeme konfiguráciu. Klikneme naň pravým tlačidlom myši a z ponuky zvolíme možnosť Internet Information Services (IIS) Manager. Otvorí sa nové okno, v ktorého ľavom paneli sa nachádza náš server. Rozbalíme jeho ponuku a klikneme na Sites. Pravý klik na Default Web Site nám ponúkne viacero možností vrátane nastavenia webstránky a pridania novej.

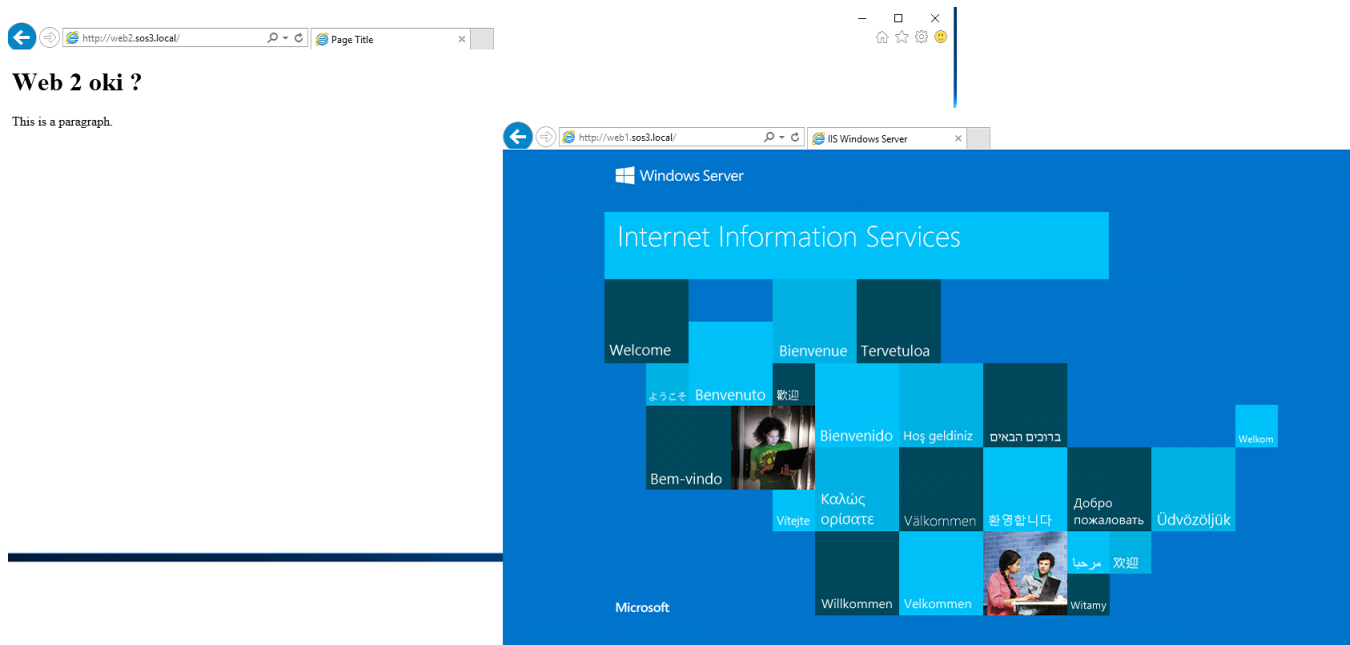
Po inštalácii sa nachádza ISS v ľavom paneli vo Windows Server Manager-i. Po kliknutí na tools v pravom hornom rohu klikneme Internet Information Services (ISS) manager. A po rozkliknutí na ľavom rohu je už vytvorená default sites. Otvoriť ju je možné zadaním do browseru “localhost”.



Obr. 19: Inštalácia webového servera IIS



Obr. 20: Zdrojový kód stránky "web2"



Obr. 21: Ukážka webových stránok "web1" (vpravo) a "web2" (vľavo)

5.7 Poštový server

V server manageri klikneme na tools a v záložke DNS, nasmerujeme sa ku DNS serveru a vytvoríme nové záznamy pre mail server. Cname záznam mail 158.193.139.74, dva MX(Mail exchanger) záznamy 0 mail sos3.local a 10 mail sos3.local.

Zo stránky mailenable.com stiahneme standart edition. Začneme inštaláciou stiahnutého balíčka, zaklikneme web mail service(server). V nasledujúcich krokoch napíšeme do Domain Name: sos3.local a DNS host: 192.168.0.2 a smtp port: 25. Počas inštalácie nám vybehne tabuľka, kde odklikneme aby sa mailserver inštaloval ako webserver ISS. V server manageri po kliknutí servers -> localhost -> system -> diagnose si skontrolujeme či všetky políčka sú pass, čo nám značí že mail enable funguje. V ďalšom kroku servers -> localhost -> services and connectors a na SMTP klikneme pravým tlačidlom a klikneme na properties. V záložke general nastavíme default mail domain name čo je v našom prípade mail.sos3.local. Ďalej v záložke smart host nastavíme IP/DOMAIN 158.193.139.74. Po reštarte serveru vidíme že, všetky service sú running.

Service	Status
✓ MailEnable IMAP Service	Running
✓ MailEnable List Connector	Running
✓ MailEnable Mail Transfer Agent	Running
✓ MailEnable POP Service	Running
✓ MailEnable Postoffice Connector	Running
✓ MailEnable SMTP Connector	Running

Obr. 22: Inštalácia poštového servera MailEnable

5.8 NAT

Inštaláciu sme vykonali vo Windows Server Manageri, kde sme cez “Add Roles and Features” pridali službu “Routing and Remote Access” a následne sme ju nainštalovali. Pri inštalácii zvolíme sieťový adaptér eth0, ktorý je pripojený k internetu. Po inštalácii je NAT plne funkčné, ale je potrebné pridať NAT záznamy na porte 53 pre tcp aj udp. Ďalej sme potrebovali nakonfigurovať NAT v Control Panel -> Administrative tools -> Routing and Remote Access. Po kliknutí na NAT, vyberieme záložku s adaptérom, ktorý je pripojený k internet. V Address Pool je potrebné nastaviť položku “from”, čo znamená náš rozsah verejných adries 158.193.139.74 a to, čo je naša koncová adresa 158.193.139.75 a maska 255.255.255.252. V záložke services and ports je potrebné pridať 4 nové záznamy NAT pre DNS(Master-Slave, TCP-UDP).



Obr. 23: Povolenie vzdialeného prístupu

Kapitola 6

Záver

Vytvorili sme dve verzie funkčnej základnej firemnej siete – linuxovú a windowsovú. Linuxová verzia bola postavená na operačnom systéme Debian 8.6.0 x64 Stable, windowsová na Windows Server 2016.

V linuxovej verzií sme sprevádzkovali firewall, VLAN smerovanie, DNS, DHCP, NTP, web server, poštový server.

Vo windowsovej verzií sme sprevádzkovali NAT, DHCP, DNS, NTP, web server IIS.