

System DNS

[Povinné]

DNS (*Domain Name System*) je systém pro překlad doménových jmen (DN, *Domain name*) na IP adresy a opačně. Umožňuje jednoznačně identifikovat počítač v interní síti nebo na internetu pomocí textového názvu (jména) namísto IP adresy, jenž je pro normálního uživatele těžko zapamatovatelná. Často se používají i různá rozšíření doménových jmen (adres), např. *e-mailová* adresa není nic jiného než doménové jméno obohacené o identifikaci osoby.

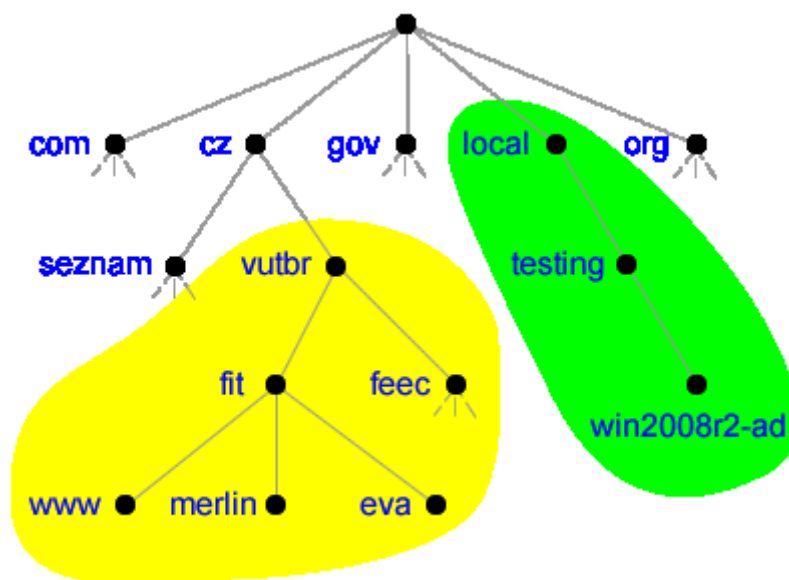
Systém **DNS** lze ovšem výhodně využít i k dalším účelům. Umožňuje flexibilní práci s IP adresami, kdy změna IP adresy počítače znamená pouze opravu mapování doménového jména na odpovídající novou IP adresu, z pohledu klienta nejsou potřeba žádné úpravy. Lze realizovat tzv. *load balancing*, kdy jedno doménové jméno identifikuje více počítačů (nejčastěji serverů), jenž poskytují specifickou službu. Klientovi je pak poskytnuta IP adresa pouze jednoho z těchto počítačů, ideálně takového, který je nejméně vytížený. Doménové jména se také používají k rozlišení služeb, např. doménové jména začínající prefixem *www* často označují *webové* servery.

Systém **DNS** patří mezi decentralizované klient-server systémy, záznamy jsou tedy rozprostřeny po více serverech a je potřeba lokalizovat ty servery, které obsahují potřebné informace. Výhodou decentralizovaného přístupu je jeho robustnost, pokud dojde k výpadku několika serveru, pořád by mělo existovat dost dalších, které jsou schopny poskytnout požadované informace. Robustnost je pro systém jako je **DNS** klíčová vlastnost.

Hierarchie DNS

[Povinné]

Systém **DNS** je hierarchický, prostor doménových jmen (*Domain Name Space*) tvoří obecný strom. Příklad části takového stromu lze vidět na obrázku 1. Kořenem stromu (označovaný jako *the root*) je prázdný (nepojmenovaný) uzel, ostatní uzly stromu jsou pojmenovány textovými řetězci o délce maximálně 63 znaků, název ovšem nesmí obsahovat tečky. Tečky slouží jako oddělovače jednotlivých úrovní stromu. Podle specifikace může mít strom až 127 úrovní.



Obrázek 1. Část stromu prostoru doménových jmen

Plně kvalifikované doménové jméno (**FQDN**, *Fully Qualified Domain Name*) je posloupnost názvů uzlů na cestě ke kořeni oddělená tečkami. Například pro uzel *www* ve stromu na obrázku 1 bude jeho plně kvalifikované doménové jméno **www.fit.vutbr.cz.** (s tečkou na konci). Ukončující tečka vyplývá z existence prázdného kořenového uzlu, i ten je totiž oddělen tečkou od názvů ostatních uzlů. Často se

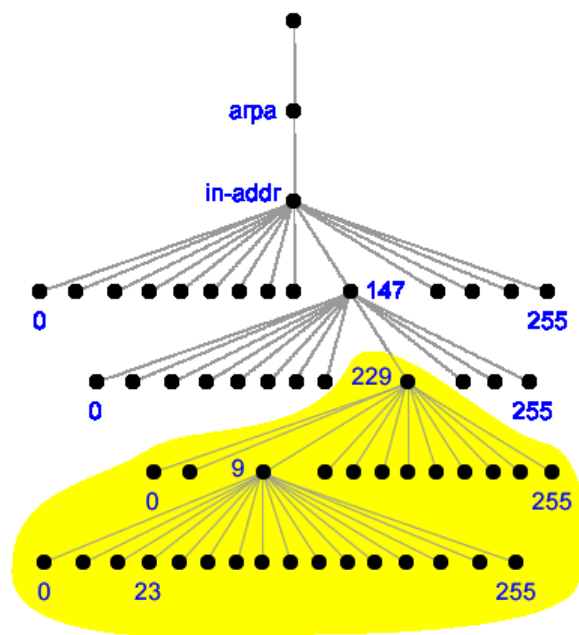
ovšem tato poslední tečka vynechává, jelikož musí být vždy přítomná a může být tedy doplněna automaticky při tvorbě požadavku pro překlad zadaného doménového jména. Maximální délka plně kvalifikovaného doménového jména je 255 znaků.

Relativní doménové jméno je jakékoliv doménové jméno bez ukončující tečky, je interpretováno k relativní doméně, ve které se nachází. Například když na FIT zadáme relativní doménové jméno **www**, bude primárně interpretováno jako **www.fit.vutbr.cz.**, jelikož jsme v doméně **fit.vutbr.cz.**, na relativní doménové jméno lze tedy nahlížet jako na prefix plně kvalifikovaného doménového jména.

Doména je podstrom ve stromu doménových jmen. Název domény (*Domain Name*) je cesta mezi uzlem, jenž tvoří vrchol domény, a kořenem celého stromu. Příkladem mohou být domény **vutbr.cz.** (vyznačena žlutě) a **local.** (vyznačena zeleně) ve stromu na obrázku 1. Domény prvního úrovně, jenž mají vzdálenost 1 od kořene stromu, se často označují TLD (*Top Level Domain*) a jsou buď tematické (např. *org* pro organizace, *gov* pro vládní a *edu* pro vzdělávací instituce) nebo státní (např. *cz* pro Českou republiku, *sk* pro Slovenskou republiku). Ostatní domény se označují podle vzdálenosti od kořene stromu jako domény druhého řádu, třetího řádu atd. nebo jako subdomény domén nižšího řádu. Často se jako domény chybně označují samotné uzly doménového stromu, ty tvoří ovšem jen prefixy konkrétních domén. Na rozdíl od doménových jmen u domén neexistuje žádný relativní zápis, každá doména musí být plně určena.

Výhodou stromové hierarchie **DNS** je možnost administrativního rozdělení jednotlivých domén, kdy konkrétní domény jsou spravovány samostatnými subjekty (buď organizacemi, nebo soukromými osobami).

Jak bylo již řečeno dříve, systém **DNS** neslouží jen k překladu doménových jmen na IP adresy, ale také naopak. Tento překlad se označuje jako *reverzní mapování* a často se využívá k ověření překladu, kdy se kontroluje, zda IP adresa počítače má v DNS odpovídající doménovou adresu. Pokud nemá, lze tuto adresu vyhodnotit jako podvrženou nebo neplatnou. Toto ověření se často provádí u poštovních serverů. Informace pro tento druh překladu jsou uloženy ve speciální doméně jménem **in-addr.arpa.**, kde v případě IPv4 tvoří jednotlivé uzly stromu osmibitová čísla v rozsahu 0 až 255, jenž reprezentují vždy osm bitů IPv4 adresy. Strom pro reverzní mapování IPv4 adres na odpovídající doménové jména je zachycen na obrázku 2. Obdobně pro IPv6 adresy existuje speciální doména **ip6.arpa.**, kde jednotlivé uzly tvoří čtyřbitová čísla.



Obrázek 2. Strom pro reverzní překlad IPv4 adres

Například IPv4 adrese **147.229.9.23** bude ve stromu pro reverzní mapování IPv4 adres odpovídat záznam **23.9.229.147.in-addr.arpa.**, kde **in-addr.arpa.** je speciální doména druhého řádu.

Služba DNS

[Povinné]

Službu **DNS** lze rozdělit na dvě části. První část tvoří **DNS server**, jenž obsahuje záznamy potřebné pro překlad doménového jména na IP adresu a zpět. Druhou částí je **DNS klient**, často označovaný jako tzv. *resolver*, který zprostředkovává překlad doménových jmen aplikacím.

Pro komunikaci se využívá **DNS** protokol, jenž běží nad protokoly **TCP** i **UDP** na portu **53**. Většinou se využívá pro přenos na transportní vrstvě protokol **UDP**, jelikož poskytuje vyšší rychlost přenosu, což je v případě **DNS** klíčové. Protokol **TCP** se používá hlavně pro tzv. *zone transfer*, o kterém bude řeč později.

DNS server

[Povinné]

DNS server zajišťuje samotný překlad doménových jmen na IP adresy a opačně. Každý **DNS server** se stará pouze o část prostoru doménových jmen, tedy jen o určitou doménu, může ovšem delegovat správu jednotlivých subdomén na jiné **DNS servery**. Z hlediska odpovědi lze rozdělit **DNS servery** na dva typy:

- **Autoritativní**, které poskytují odpovědi, jenž obsahují vždy aktuální informace.
- **Neautoritativní**, které poskytují odpovědi, jenž mohou obsahovat již neplatné informace.

DNS servery ovšem nepracují s doménami, nýbrž s tzv. zónami. Přesná definice zóny neexistuje, většinou je zóna ekvivalentním vyjádřením domény. Někdy se jako zóna označuje vše, co je obsahem zónového souboru, což může být více než jen jedna doména. Zóny se dělí na základě obsažených informací a možností manipulace s těmito informacemi na tři typy:

- **Primární** zóna obsahuje veškeré záznamy pro danou doménu a jako jediná umožňuje tyto záznamy přímo modifikovat. Primární zóna je vždy autoritativní. Windows Server ještě navíc rozlišuje dva speciální typy primárních zón podle způsobu jejich uložení:
 - **Standardní primární zóna**, jenž ukládá záznamy do zónového souboru. Standardně jsou zónové soubory uloženy v adresáři `<system>\System32\dns`, kde `<system>` je kořenový adresář systému Windows, a pojmenovány názvem domény, kterou zóna obsahuje, s přidanou příponou **.dns**. Např. zónový soubor pro doménu **fit.vutbr.cz** bude standardně uložen v zónovém souboru **fit.vutbr.cz.dns**.
 - **Zóna integrovaná v Active Directory**, jenž ukládá záznamy do Active Directory. Zóny jsou pak uloženy v oddílu *domény* nebo *aplikace* (podle nastavení rozsahu replikace) v objektu kontejneru **dnsZone** určeného názvem domény, jenž zóna obsahuje. Tento způsob uložení přináší řadu výhod, jak bude popsáno dále.
- **Sekundární** zóna, stejně jako primární, obsahuje veškeré záznamy pro danou doménu. Tyto záznamy jsou ovšem určeny pouze pro čtení, nelze je přímo modifikovat, pouze skrz přenos zón (*Zone Transfer*). Tato zóna je tedy plnou kopií jiné primární či sekundární zóny a je vždy autoritativní.
- **Stub** zóna obsahuje pouze informace potřebné ke kontaktování nějakého autoritativního serveru. Přesněji obsahuje jen SOA a NS záznamy společně s A (případně AAAA) záznamy pro překlad doménových jmen těchto serverů na odpovídající IP adresy. Tato zóna jako jediná není autoritativní.

Zóny se často používají ke kategorizaci **DNS serverů**, kdy na základě typu zóny, kterou daný server obsahuje, se rozlišují celkem tři typy **DNS serverů** (někdy se místo typu mluví o roli **DNS serveru**):

- **Primární** server je server obsahující *primární* zónu a je vždy autoritativní.
- **Sekundární** server je server obsahující *sekundární* zónu a také je vždy autoritativní.
- **Záložní (Caching-Only)** server je server obsahující *stub* zónu. Tento server plní zároveň úlohu klienta i serveru. Uchovává (*kešuje*) informace o překladu doménových jmen na IP adresy a naopak ve vyrovnávací paměti. Pokud přijde požadavek na překlad, podívá se do této paměti,

zda jsou informace potřebné pro provedení překladu k dispozici. V případě, že ano, použije tyto informace k vytvoření odpovědi. Jinak, na základě informací obsažených ve *stub* zóně, kontaktuje nějaký **DNS server** a zašle mu přijatý požadavek na překlad. Informace z obdržené odpovědi pak uloží do vyrovnávací paměti a zároveň je přepošle jako odpověď na původní příchozí požadavek na překlad. Pokud později dojde stejný požadavek, budou již potřebné informace přítomny ve vyrovnávací paměti. Hlavním problémem tohoto typu **DNS serveru** je možná nekonzistence informací ve vyrovnávací paměti. Záznamy se samozřejmě v čase mění a server o těchto změnách neví, může tedy uchovávat zastaralé informace a používat je pro odpovědi na požadavky klientů. Tento typ serveru tedy není autoritativní. Často lze ovšem s výhodou využít pro urychlení překladu doménových jmen v rámci míst (*sites*). Místa totiž bývají navzájem spojená linkami s horší odezvou a propustností a nasazením *záložního* DNS serveru může výrazně snížit potřebný provoz z hlediska požadavků na překlad. Klienti se ptají *záložního* serveru, který první požadavek přepošle dále jinému serveru a po přijetí odpovědi již bude sám schopen na tento požadavek reagovat.

Jelikož může jeden server obsahovat více zón, často se mluví o typu **DNS serveru** vzhledem k nějaké doméně. Jeden server pak může být třeba primárním serverem pro doménu **fit.vutbr.cz.** a zároveň sekundárním serverem pro doménu **feec.vutbr.cz.**, což je celkem častá situace.

DNS záznamy

[Povinné]

DNS server ukládá informace potřebné pro překlad doménových jmen a IP adres ve formě **DNS záznamů**. Existují různé typy **DNS záznamů**, každý typ poskytuje specifické informace. Mezi nejčastěji používané (a nejdůležitější) typy patří:

- **A** (Address) záznam obsahuje mapování doménového jména na odpovídající IPv4 adresu ve formátu **<doménové jméno> IN A <IPv4 adresa>**.
- **AAAA** (IPv6 Address) záznam je obdobou **A** záznamu, jenž mapuje doménové jméno, místo na IPv4 adresu, na IPv6 adresu. Formát je **<doménové jméno> IN AAAA <IPv6 adresa>**.
- **CNAME** (Canonical Name) záznam, označovaný také často jako tzv. *alias*, obsahuje mapování jednoho doménového jména na jiné. Umožňuje flexibilně pojmenovávat jeden server více doménovými jmény. Stejného efektu lze docílit i použitím více **A** záznamů, ovšem v případě změny IP adresy cíle by bylo potřeba změnit veškeré tyto **A** záznamy. Při použití **CNAME** je potřeba změnit pouze jediný **A** záznam a tím dojde ke změně mapování pro všechny **CNAME** záznamy, jenž se odkazovaly na doménové jméno ve změněném **A** záznamu. Formát tohoto záznamu je **<zdrojové doménové jméno> IN CNAME <cílové doménové jméno>**.
- **MX** (Mail Exchange) záznam obsahuje adresu a prioritu serveru pro příjem elektronické pošty pro danou doménu (zónu) ve formátu **<doméma> IN MX <priorita> <doménové jméno>**, kde **<doméma>** identifikuje cílovou doménu, **<doménové jméno>** cílový server pro příjem pošty a **<priorita>** je nezáporné číslo určující prioritu daného serveru. Čím menší číslo, tím má server vyšší prioritu.
- **NS** (Name Server) záznam obsahuje doménové jméno autoritativního **DNS** serveru pro danou doménu ve formátu **<doméma> IN NS <doménové jméno>**. Autoritativních **DNS** serverů pro jednu doménu může být samozřejmě více.
- **PTR** (Pointer) záznam mapuje IP adresu počítače na odpovídající doménové jméno, je tedy obdobou **A** záznamu pro reverzní zóny. Formát je **<arpa doménové jméno> IN PTR <doménové jméno>**, kde **<arpa doménové jméno>** je IP adresa zapsaná ve formě doménového jména z domény **in-addr.arpa.** nebo **ip6.arpa.**, tedy z domény pro reverzní mapování.
- **SOA** (Start of Authority) záznam obsahuje základní informace pro danou zónu, přesněji jméno primárního **DNS** serveru, elektronickou adresu správce zóny (zavináč se nahrazuje tečkou) a několik dalších údajů:

- **Serial** obsahuje informace o sériovém čísle dané zóny. Při každé změně obsahu zóny se navýší toto číslo o jedna.
- **Refresh** je číslo určující v jakém intervalu se bude *sekundární* server dotazovat na změny zóny *primárního* serveru.
- **Retry** je číslo určující za jak dlouho se má *sekundární* server dotazovat na změny zóny *primárního* serveru v případě, že se nepodařilo *primární* server kontaktovat.
- **Expire** je číslo určující za jak dlouho *sekundární* server označí své záznamy za neaktuální a přestane vyřizovat příchozí požadavky na překlad.
- **TTL (Time To Live)** je číslo určující implicitní dobu platnosti jednotlivých DNS záznamů.

Formát je **<doména> IN SOA <primární DNS server> <email> (**
<serial>
<refresh>
<retry>
<expire>
<ttd>)

Tabulka 1 níže shrnuje nejpoužívanější typy **DNS záznamů** spolu s informací o překladu, jenž daný záznam provádí a formátu zápisu tohoto záznamu v zónovém souboru.

Typ záznamu	Překlad	Formát
A	Doménové jméno (DN) → IPv4 adresa (IPv4)	<DN> IN A <IPv4>
AAAA	Doménové jméno (DN) → IPv6 adresa (IPv6)	<DN> IN AAAA <IPv6>
CNAME	Doménové jméno (DN) → Doménové jméno (DN)	<DN> IN CNAME <DN>
MX	Doména (D) → Doménové jméno serveru (DN)	<D> IN MX <priorita> <DN>
PTR	IP adresa (IP) → Doménové jméno (DN)	<IP> IN PTR <DN>
NS	Doména (D) → Doménové jméno serveru (DN)	<D> IN NS <DN>
SOA	Doménové jméno serveru (DN) Doména (D) → Elektronická adresa správce (MAIL) Informace o zóně (viz výše)	<D> IN SOA <DN> <MAIL> (<serial> <refresh> <retry> <expire> <ttd>)

Tabulka 1. Nejpoužívanější typy DNS záznamů

Přenos zón

[Povinné]

Jedna zóna může být obecně na více než jednom **DNS** serveru. Umístění jedné zóny na více **DNS** serverů přináší větší spolehlivost, pokud jeden **DNS** server selže, klient může pořád kontaktovat další. Samozřejmě je potřeba, aby se změny provedené v dané zóně na jednom **DNS** serveru projevily i na ostatních, je tedy nutné provádět jejich vzájemnou synchronizaci. Synchronizace se v tomto případě provádí pomocí tzv. **přenosu zón** (*Zone Transfer*).

Přenos zón by se dal označit jako aktualizace jednoho originálu, kdy se vyžaduje, aby pouze jediná zóna byla primární, tedy explicitně modifikovatelná. Všechny ostatní zóny musí být sekundární a tedy modifikovatelné pouze pomocí **přenosu zón**. V případě existence více primárních zón nebude možné **přenos zón** použít (obecně je povoleno vytvářet neomezeně primárních zón pro danou zónu).

Než proběhne **přenos zón** (synchronizace) mezi dvěma servery (označované jako *master* a *slave*), je ovšem potřeba nejprve zjistit, zda je vůbec přenos potřeba provádět. Existují celkem dva způsoby, jak *slave* server zjistit, zda došlo ke změně dané zóny na *master* serveru:

- *Slave* server v pravidelných intervalech vyžaduje od *master* serveru zaslání SOA záznamu pro danou zónu. Jakmile *master* server zašle tyto informace, *slave* server ověří sériové číslo SOA záznamu a porovná ho se svým. Pokud je sériové číslo SOA záznamu *master* serveru vyšší než

sériové číslo SOA záznamu *slave* serveru, došlo ke změnám zóny a *slave* server si vyžádá od *master* serveru potřebná data (provede tedy **přenos zóny**). Informace o intervalu, ve kterém se *slave* server dotazuje *master* serveru na změny jsou uloženy v SOA záznamu *slave* serveru.

- *Master* server při změně zóny zašle všem *slave* serverům oznámení, že došlo ke změně zóny. *Slave* servery pak postupují stejně jako v předchozím bodě, ještě před přenosem zóny tedy provedou ověření, zda opravdu ke změně došlo.

Samotný přenos zón pak může probíhat dvěma způsoby:

- **Úplným přenosem zóny (AXFR)**, kdy dojde k přenesení celého zónového souboru.
- **Inkrementálním přenosem zóny (IXFR)**, kdy se přenesou pouze změny oproti některé dřívější verzi zónového souboru.

Výhody integrace DNS serveru v Active Directory

[Povinné]

Jak již bylo zmíněno dříve, primární zóny mohou být integrovány v Active Directory. Tato integrace s sebou přináší několik výhod.

První výhodou je možnost využití replikace Active Directory pro synchronizaci (přenos) zón. Tato forma synchronizace je označována jako aktualizace více originálů a umožňuje tedy existenci více primárních serverů pro danou zónu. Synchronizaci jednotlivých zón lze také ovlivňovat nastavením rozsahu replikace. Zóna může být replikována na všechny řadiče domény i přesto, že neplní roli **DNS** serveru, pouze na řadiče domény s rolí **DNS** serveru nebo pouze na specifické řadiče domény, které jsou součástí nějakého oddílu aplikace (*Application Partition*).

Další výhodou je vyšší míra zabezpečení. Zóna je zde objektem Active Directory a tedy, stejně jako u jakýchkoliv dalších objektů v Active Directory, lze využít seznamy pro kontrolu přístupu (ACL, *Access Control List*). Pomocí ACL lze ovlivňovat možnosti přístupu jak k celé zóně, tak pouze ke konkrétním záznamům.

Z dalších výhod lze ještě zmínit efektivnější způsob přenosu zón, který probíhá vždy inkrementálně a s kompresí přenášených dat, nebo možnost použití zabezpečené aktualizace **DNS** záznamů (*Secure dynamic updates*), kdy klienti z Active Directory mohou automaticky ukládat informace o překladu svých doménových jmen na odpovídající IP adresy a naopak do databáze **DNS** serveru.

DNS klient

[Povinné]

DNS klient (tzv. *resolver*) zprostředkovává překlad doménových jmen a IP adres běžícím aplikacím. V případě systému Windows je **klient DNS** implementován ve formě systémové knihovny, jejíž funkce realizují komunikaci se servery DNS. Systém Windows také podporuje ukládání provedených překladů do lokální mezipaměti, což zajišťuje služba **Klient DNS**. Celý proces překladu zadaného doménového jména na jemu odpovídající IP adresu je ovšem složitější než pouze zaslání dotazu a přijetí odpovědi. Postup překladu může být obecně různý, záleží vždy na konkrétní implementaci v daném operačním systému. Často také nedochází přímo k překladu doménového jména, většinou se požaduje překlad tzv. hostitelského jména (*hostname*), což je z pohledu systému **DNS** název listového uzlu ve stromu doménových jmen, tedy název počítače, jehož IP adresu potřebujeme zjistit. V systému Windows probíhá překlad hostitelského jména na IP adresu následovně:

1. Systém ověří, zda se zadané hostitelské jméno shoduje s lokálním hostitelským jménem počítače. V případě, že ano, vrátí jednu z IP adres přiřazenou tomuto počítači.
2. Systém prohledá vyrovnávací paměť **klienta DNS** (*DNS client resolver cache*). Pokud nalezne informace potřebné pro překlad, vrátí IP adresu uloženou v této paměti.
 - **Klient DNS** pro zvýšení efektivity překladu dočasně ukládá informace o dříve vykonaných překladech. Neboli ukládá každou obdržanou odpověď od **DNS serveru** do vyrovnávací paměti. Kromě toho tato paměť obsahuje také informace obsažené v souboru *hosts*, jenž je umístěn v adresáři `<system>\System32\drivers\etc`, kde `<system>` je kořenová složka

systému Windows. Soubor *hosts* je standardní textový soubor obsahující mapování mezi IP adresami a hostitelskými jmény ve formátu *<IP adresa> <hostitelské jméno>*. Jelikož se v těchto záznamech vyhledává dříve, než dojde ke kontaktování **DNS serveru**, lze takto přepsat cílové IP adresy pro jednotlivá hostitelská jména. Také je možné tímto způsobem zajistit překlad specifických hostitelských jmen i v případě, že nelze kontaktovat žádný **DNS server**. Obsah souboru *hosts* je načten do vyrovnávací paměti **klienta DNS** při startu systému a obsažené záznamy mají trvalou platnost (do doby než jsou znova načteny).

3. Systém převede hostitelské jméno na doménové jméno připojením názvu domény, ve které se počítač nachází (nebo jiné, pokud je to explicitně nastaveno) a zašle vytvořený požadavek pro překlad **DNS serveru** a čeká na odpověď. V případě, že obdrží odpověď s požadovanými informacemi, vrátí IP adresu obsaženou v této odpovědi a zároveň tyto informace uloží do vyrovnávací paměti **klienta DNS**. Pokud odpověď nedorazila v daném časovém limitu nebo **DNS server** neobsahuje záznam s informacemi potřebnými pro překlad, pokusí se systém vytvořit nové doménové jméno připojením názvu jiné domény. Systém Windows umožňuje specifikovat celý seznam názvu domén, jenž mohou být použity pro vytváření doménových jmen z hostitelských jmen, tyto domény jsou seřazeny na základě priority.
4. Systém prohledá vyrovnávací paměť **LLMNR** (*LLMNR resolver cache*). V případě, že nalezne informace potřebné pro překlad, vrátí IP adresu uloženou v této paměti.
5. Systém použije **LLMNR** (*Link Local Multicast Name Resolution*), pokud je k dispozici. Pokud se podaří lokalizovat počítač se zadaným hostitelským jménem, vrátí jeho IP adresu a následně uloží informace do vyrovnávací paměti **LLMNR**.
 - **LLMNR** je způsob překladu založený na využití (linkových) skupinových IP adres. Počítač zašle zprávu na IPv6 adresu **FF02::1:3** (všechny uzly na dané lince) nebo na IPv4 adresu **224.0.0.252** s požadavkem pro překlad zadaného hostitelského jména. Pokud některý počítač zjistí, že disponuje hledaným hostitelským jménem, vytvoří odpověď, kterou zašle zpět, tentokrát již na individuální (*unicast*) IP adresu odesílatele. Odpověď vždy obsahuje IP adresu toho rozhraní, přes které je daný počítač připojen na odpovídající linku, tedy odesílatel ho musí být schopen skrz tuto IP adresu kontaktovat. Aby bylo **LLMNR** možné použít, musí být povolena služba zjišťování sítě (*Network Discovery*).
6. Systém prohledá vyrovnávací paměť **NetBIOS** (*local NetBIOS name cache*) a pokud nalezne informace potřebné pro překlad, vrátí IPv4 adresu uloženou v této paměti.
 - **NetBIOS**, neboli NetBIOS přes TCP/IP (*NBT, NetBT, NetBIOS-over-TCP/IP*), je protokol a jmenný systém používaný ve starších verzích systému Windows, jenž je stále zachováván z důvodu zpětné kompatibility s těmito verzemi. **NetBIOS** je tzv. *flat* jmenný systém, což znamená, že se nerozlišují žádné jmenné prostory jako např. domény u **DNS**. Veškerá jména jsou součástí jednoho globálního jmenného prostoru. **NetBIOS** jména musí být maximálně 15 bytů (znaků) dlouhá, obecně je sice pro jméno vyhrazeno 16 bytů, ovšem poslední bajt je rezervován pro specifikaci typu uzlu (stanice, server apod.).
 - **NetBIOS** se použije pouze tehdy, pokud je možné zadané hostitelské jméno převést na **NetBIOS** jméno, což lze pouze v případě, že hostitelské jméno není delší než 15 znaků a není ani doménovým jménem (neobsahuje tečky).
7. Systém kontaktuje **WINS** (*Windows Internet Naming Service*) server, zašle mu požadavek pro překlad **NetBIOS** jména na IPv4 adresu a čeká na odpověď. Pokud obdrží odpověď obsahující požadované informace, vrátí IPv4 adresu, jenž je obsažena v této odpovědi a zároveň tyto informace uloží do vyrovnávací paměti **NetBIOS**.
 - **WINS** server je adresářová služba obsahující mapování mezi **NetBIOS** jmény a jejich IPv4 adresami. Pokud má nějaké rozhraní nastavenou IP adresu **WINS** serveru, automaticky dojde k registraci jeho IPv4 adresy u **WINS** serveru. Výhodou **WINS** serveru je možnost překladu **NetBIOS** jmen i mimo lokální (pod)síť.

8. Systém použije všesměrové vysílání **NetBIOS** (*NetBIOS broadcast*). V případě, že se podaří lokalizovat počítač se zadaným **NetBIOS** jménem, vrátí jeho IPv4 adresu a následně uloží informace do vyrovnávací paměti **NetBIOS**.
 - Všesměrové vysílání **NetBIOS** pracuje na podobném principu jako **LLMNR** (jenž prakticky tento způsob překladu nahradilo), pracuje ovšem pouze s IPv4. Počítač rozešle zprávy na adresu pro všesměrové vysílání, a pokud některý z kontaktovaných počítačů obsahuje hledané **NetBIOS** jméno, zašle odpověď, obsahující jeho IPv4 adresu, zpět odesilateli. Nevýhodou tohoto přístupu je omezení pouze na lokální (pod)síť, jelikož dále se nemůže všesměrové vysílání (standardně) šířit.
9. Systém prohledá soubor *lmhosts*. Pokud nalezne hledané **NetBIOS** jméno, vrátí odpovídající IPv4 adresu.
 - Soubor *lmhosts* je standardní textový soubor, jenž obsahuje mapování **NetBIOS** jmen na odpovídající IPv4 adresy ve formátu *<IPv4 adresa> <NetBIOS jméno>*. Tento soubor je umístěn v adresáři *<system>\System32\drivers\etc*, kde *<system>* je kořenová složka systému Windows.

Posloupnost výše zmíněných kroků nemusí vždy probíhat v uvedeném pořadí, záleží na konkrétní situaci. Obecně ale systém vždy preferuje použití **DNS**, po něm **LLMNR** a až nakonec **NetBIOS**. Také je důležité si uvědomit, že všechny tři způsoby překladu se nejprve dívají do svých vyrovnávacích pamětí (*cache*), které neposkytují autoritativní odpovědi. Od Windows 7 a Windows Server 2012 probíhá obvykle překlad **LLMNR** a **NetBIOS** paralelně.

Zóna globálních jmen

[Povinné]

Zóna globálních jmen (GNZ, *Global Names Zones*) je částečná náhrada **WINS** serverů představená ve Windows Server 2008. Tato zóna musí být integrována v Active Directory a pojmenována **GlobalNames**. Nesmí mít povoleny dynamické aktualizace záznamů, jelikož musí obsahovat pouze záznamy typu CNAME, jejichž názvy mohou být maximálně 15 znaků dlouhé. Replikace této zóny musí být nastavena na celý les (*forest*) Active Directory (seskupení doménových stromů) a zóna samotná musí být povolena na každém řadiči domény v celém lese.

DNS dotazy

[Povinné]

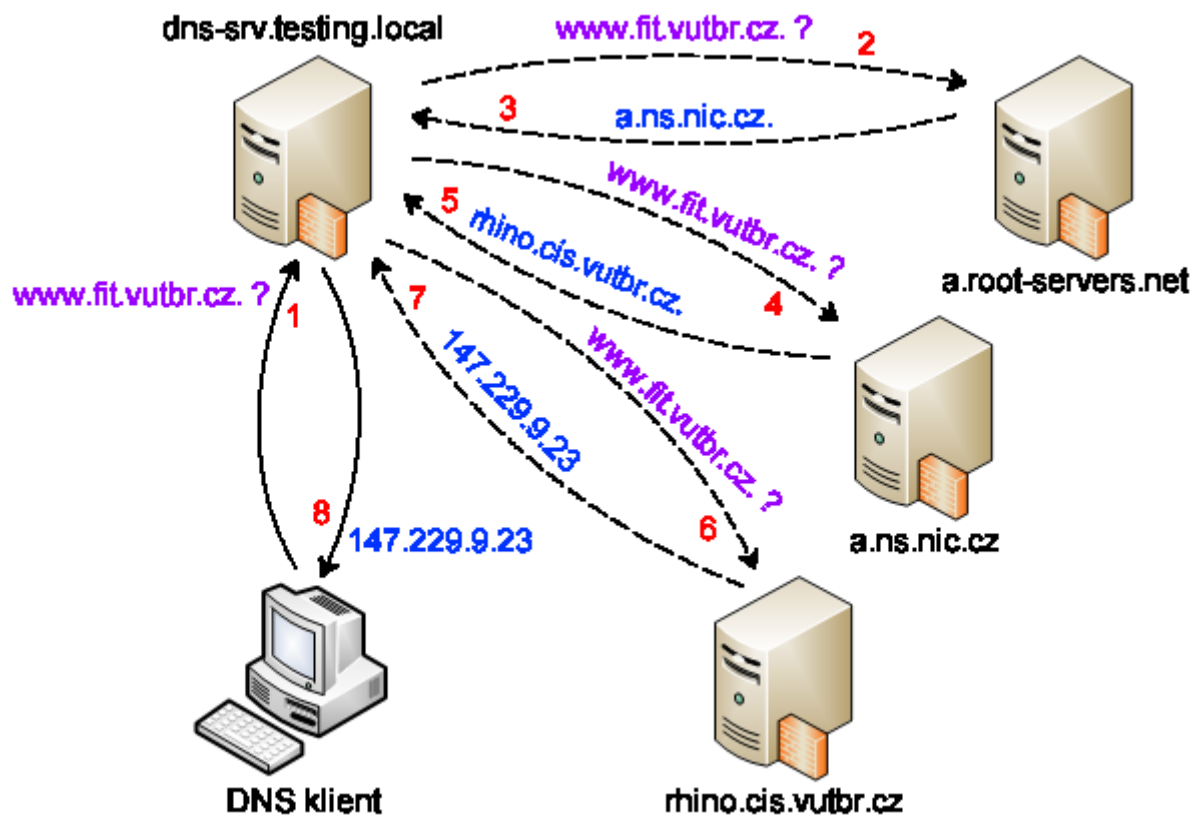
Jak již bylo řečeno dříve, systém **DNS** je založen na modelu klient-server. Klient tedy zasílá serveru požadavky a server se jim snaží vyhovět, tedy vrací odpovídající IP adresu nebo doménové jméno (při reverzním mapování), pokud disponuje těmito informacemi. Každý dotaz (požadavek) zaslaný **DNS** serveru musí obsahovat tři hlavní informace:

- **Plně kvalifikované doménové jméno** (FQDN), i v případě reverzního mapování je IP adresa zapsána jako FQDN (doména *in-addr.arpa.* nebo *ip6.arpa.*).
- **Typ dotazu**, což může být požadovaný typ záznamu (viz dále) nebo speciální typ operace.
- **Třída doménového jména**, kde se prakticky používá pouze třída **IN** (*Internet*).

Pokud se **DNS** klientovi nepodaří převést doménové jméno či IP adresu lokálně (např. využitím vyrovnávací paměti **DNS**), musí kontaktovat server. Obecně existují dva typy **DNS** dotazů, jenž může klient použít:

- **Rekurzivní dotaz**, kdy **DNS** klient zašle dotaz na určitý záznam danému **DNS** serveru a server musí na tento dotaz odpovědět buď požadovanými daty, nebo chybovou hláškou.
- **Iterativní dotaz**, kdy **DNS** server vrátí **DNS** klientovi nejlepší možnou odpověď, kterou může poskytnout. Dotazovaný **DNS** server se podívá do své databáze a pokud nenajde odpověď, vrátí jména a adresy **DNS** serverů, které by mohly znát požadovanou odpověď (tedy takové, jenž jsou nejbližší hledané adrese v kontextu hierarchie domén).

V obou případech dotazů uvedených výše může sám **DNS** server plnit i funkci **DNS** klienta a ptát se dalších **DNS** serverů na požadované informace. Nejčastěji se používá přístup, kdy se **DNS** klient ptá pomocí rekurzivního dotazu a pokud **DNS** server nedokáže na zasláný dotaz odpovědět, nejprve se zkusí zeptat dalších **DNS** serverů pomocí iterativních dotazů. Příklad tohoto přístupu je znázorněn na obrázku 3.



Obrázek 3. Příklad průběhu DNS dotazu mezi klientem a serverem

Na obrázku 3 výše je vidět průběh překladu doménového jména **www.fit.vutbr.cz.** na odpovídající IP adresu. **DNS** klient nejprve zašle rekurzivní dotaz (plná čára) lokálnímu **DNS** serveru, který ovšem není schopen na tento dotaz odpovědět. Lokální **DNS** server tedy zašle iterativní dotaz (čárkovaná čára) nejvyššímu (*root*) **DNS** serveru se stejným požadavkem pro překlad. Nejvyšší **DNS** server sice taky neví odpověď, ale odpoví, že server **a.ns.nic.cz.** pro doménu **.cz.** by mohl znát odpověď. Lokální server tedy zašle stejný požadavek také serveru pro doménu **.cz.**, jenž ale také nezná odpověď, poradí ovšem, že server **rhino.cis.vutbr.cz.** pro doménu **vutbr.cz.** by mohl znát odpověď. Lokální server tedy opět zopakuje zaslání dotazu i pro tento server, který již zná odpověď a vrátí lokálnímu **DNS** serveru odpovídající IP adresu. Lokální **DNS** server nakonec zašle tuto informaci **DNS** klientovi.

System DHCP

[Povinné]

Zajištění konektivity mezi počítači v malé (např. domácí) síti je celkem jednoduché, stačí manuálně nastavit jednotlivá síťová rozhraní. Často to není ani nutné, jelikož lze využít automatické přidělování IPv4 adres pomocí systému **APIPA**. Pokud ovšem pracujeme ve větší síti, což je v případě existence serverů pravděpodobné, nejsou předchozí možnosti příliš použitelné. Manuální konfigurace je pracná a náchylná na chyby způsobené uživatelem (chybně zadané údaje apod.). Použití systému **APIPA** je nepřijatelné, jelikož pravděpodobnost, že se podaří každému počítači vygenerovat do deseti pokusů unikátní IPv4 adresu je v tomto počtu mizivá.

Systém DHCP slouží k automatické konfiguraci síťových rozhraní. Umožňuje nastavit IPv4 adresy, masky podsítě, výchozí brány, adresy **DNS** a **WINS** serverů a další informace. Konfigurace jednotlivých rozhraní je realizována pomocí protokolu **DHCP** (*Dynamic Host Configuration Protocol*), který vzniknul jako rozšíření protokolu **BOOTP**, jenž sloužil pro *bootování* bezdiskových stanic. **BOOTP** byl schopen přidělovat jen IPv4 adresu, masku podsítě, adresu **TFTP** (*Trivial File Transfer Protocol*) serveru, který obsahoval *bootovací* obraz, a cestu k tomuto obrazu. Protože **DHCP** je rozšíření protokolu **BOOTP**, je s ním také zpětně kompatibilní.

Služba DHCP

[Povinné]

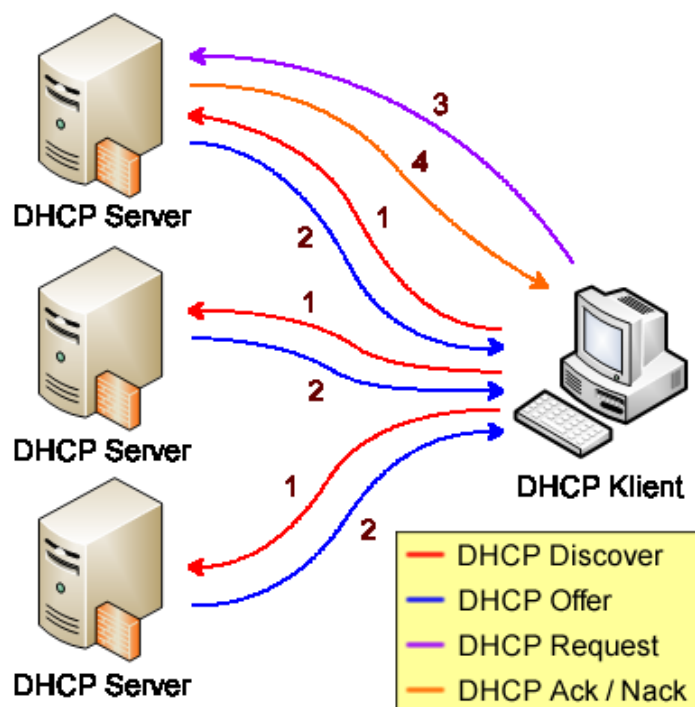
Stejně jako v případě **DNS** lze i zde **službu DHCP** rozdělit na dvě části. První část tvoří **DHCP** server, jenž obsahuje informace o IPv4 adresách přidělených jednotlivým rozhraním. Druhou částí je **DHCP** klient, který zjišťuje informace potřebné pro konfiguraci jednotlivých síťových rozhraní.

Jak již bylo řečeno výše, pro komunikaci se využívá **DHCP** protokol, jenž běží na protokolem **UDP** na portech **67** (server) a **68** (klient). Komunikace je vždy realizována pomocí všesměrového vysílání (*broadcast*), jelikož ten jediný lze použít i v případě, že rozhraní ještě nemá přidělenou IP adresu.

Přidělování IPv4 adres pomocí DHCP

[Povinné]

Princip přidělování IPv4 adres je znázorněn na obrázku 4 níže. Postup lze shrnout do následujících několika kroků:



Obrázek 4. Průběh přidělování IPv4 adres pomocí DHCP (DORA)

1. **DHCP** klient zašle všesměrovou (*broadcast*) zprávu **DHCP Discover** všem **DHCP** serverům na dané síti, touto zprávou žádá o přidělení IPv4 adresy.
2. Každý **DHCP** server zašle zpět všesměrovou (*broadcast*) zprávu **DHCP Offer**, která obsahuje IPv4 adresu, jenž server nabízí k použití. V případě, že **DHCP** server nemá již k dispozici žádnou volnou IPv4 adresu pro zapůjčení, nijak na žádosti nereaguje.
3. **DHCP** klient čeká na nabídky od **DHCP** serverů. Z přijatých nabídek vybere jedinou (nejčastěji první příchozí) a odpoví na ni opět všesměrovou (*broadcast*) zprávou **DHCP Request**, kterou stvrzuje svůj zájem o použití nabízené IPv4 adresy.

4. **DHCP** server, jenž nabídl danou IPv4 adresu, ověří, zda je možné tuto IPv4 adresu opravdu zapůjčit a v případě, že ano, zašle zpět všesměrovou (*broadcast*) zprávou **DHCP Ack**, kterou potvrzuje zapůjčení této IPv4 adresy. Pokud IPv4 adresu již z nějakého důvodu nelze zapůjčit (např. již zatím byla zapůjčena jinému rozhraní), odpoví všesměrovou (*broadcast*) zprávou **DHCP Nack** a **DHCP** klient musí požádat o novou (jinou) IPv4 adresu.

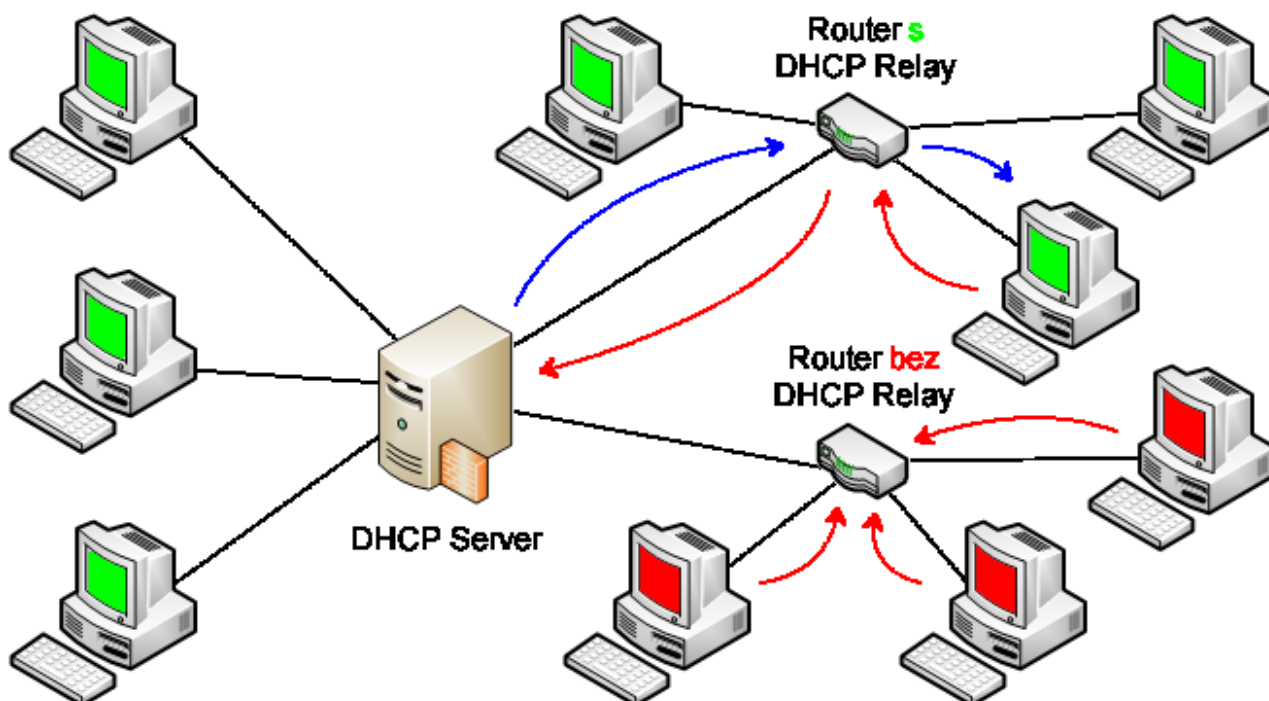
Tento postup je často označován jako **DORA** (*Discover, Offer, Request, Ack*) a popisuje přidělování IPv4 adres klientům, jenž ještě nemají přidělenou žádnou IPv4 adresu. Adresa je vždy zapůjčována jen na určitou dobu, kterou určuje **DHCP** server. **DHCP** klient pak musí pravidelně tuto dobu prodlužovat zasíláním žádosti o prodloužení výpůjčky (*lease renewal*).

Po vypršení 50% doby platnosti výpůjčky IPv4 adresy se začne **DHCP** klient pokoušet prodloužit dobu její platnosti. Prodloužení je realizováno zasláním normální (*unicast*) zprávy **DHCP Request** **DHCP** serveru, jenž zapůjčil danou IPv4 adresu. Ten buď potvrdí prodloužení výpůjčky pomocí zprávy **DHCP Ack** nebo zamítne pomocí **DHCP Nack**. Pokud je prodloužení zamítnuto, klient si ponechá IPv4 adresu do konce doby její platnosti a pak si požádá o novou. V případě, že se **DHCP** klientovi nepodaří prodloužit výpůjčku do 87,5% doby její platnosti, pokusí se klient kontaktovat jakýkoliv **DHCP** server, který ji může prodloužit. Prodloužení se provede jako v předchozím případě, jen **DHCP Request** je zaslán všesměrově (*broadcast*) všem serverům. Pokud se **DHCP** klientovi vůbec nepodaří prodloužit výpůjčku do vypršení její doby platnosti, znovu požádá po vypršení o novou IPv4 adresu.

DHCP relay

[Povinné]

Hlavní nevýhodou **DHCP** je jeho závislost na všesměrovém vysílání (*broadcast*), **DHCP** zprávy tedy nelze standardně šířit za hranice směrovačů do jiných (pod)sítí. **DHCP relay** slouží k přeposílání **DHCP** zpráv do jiných sítí, přesněji k směrování **DHCP** zpráv z dané (pod)sítě na **DHCP** server v jiné (pod)síti a naopak. Funkce **DHCP relay** je ilustrována na obrázku 5 níže.



Obrázek 5. Ilustrace funkce DHCP relay

Z obrázku 5 výše je vidět, že klienti z jiné (pod)sítě, jenž jsou spojeni se směrovačem (*router*), který má podporu **DHCP relay**, jsou schopni získat IPv4 adresu od **DHCP** serveru, zatímco jiní klienti za směrovačem bez podpory **DHCP relay** ne, jelikož jejich požadavky se nedostanou k **DHCP** serveru.

Lektorské úkoly

- Pro přístup na server **yetti** přes síť *Internal* je nutné použít jeho plně kvalifikované doménové jméno **yetti.nepal.aps**

Lab L00 – konfigurace virtuálních stanic

[\[Provést \]](#)

Připojte síťové adaptéry stanic k následujícím virtuálním přepínačům:

Adaptér (MAC suffix)	LAN1 (-01)	LAN2 (-02)	LAN3 (-03)	LAN4 (-04)
w10-base	Nepřipojeno	Private1	Nepřipojeno	Nepřipojeno
w2016-base	Nepřipojeno	Private1	Nepřipojeno	Nepřipojeno
w10-domain	Nepřipojeno	Private1	Nepřipojeno	Nepřipojeno
w2016-dc	Nepřipojeno	Private1	Nepřipojeno	Nepřipojeno

- v případech, kdy je potřeba přistupovat na externí síť, připojte adaptér **LAN1** k přepínači *Internal*.

Lab L01 – Instalace a základní nastavení DHCP serveru

[\[Provést \]](#)

Cíl cvičení

Povýšit server do role **DHCP** serveru a nastavit nový rozsah (*scope*) pro přidělování IPv4 adres, ověřit funkčnost připojením klienta do sítě

Potřebné virtuální stroje

w10-base

w2016-base

- Na **w2016-base** nastavte statickou IPv4 adresu **192.168.1.1**
 - Otevřete okno **Network Connections** (Settings – Network & Internet – Ethernet – Change adapter options), zvolte **LAN2** a pak **Properties**
 - Zvolené síťové rozhraní musí odpovídat *Private1*, standardně to je **LAN2**
 - Vyberte Internet **Protocol Version 4 (TCP/IPv4)** a zvolte **Properties**
 - Zvolte **Use the following IP address** a jako **IP address** zadejte **192.168.1.1**
 - Klikněte do zadávacího pole u **Subnet mask**, maska podsítě bude doplněna automaticky
 - Potvrďte **OK**
- Spustíte **Server Manager**
 - Start** → **Server Manager**
- Nainstalujte roli **DHCP** server
 - Vyberte **Add Roles and Features** z nabídky **Manage**
 - Pokračujte **Next >**
 - Vyberte **Role-based or feature-based installation** a pokračujte **Next >**
 - Vyberte aktuální server a pokračujte **Next >**
 - V seznamu rolí vyberte **DHCP Server**, potvrďte přidání potřebných funkcí **Add Features** a pokračujte třikrát **Next >**
 - Potvrďte instalaci **Install**

- g. Po dokončení instalace najdete v notifikacích **Server Manageru** odkaz na **Post-deployment Configuration** DHCP serveru (Complete DHCP configuration)
- h. V poinstalační průvodci potvrďte bezpečnostní skupiny pomocí **Commit** a následně **Close**
4. Nakonfigurujte **DHCP server** - vytvořte nový rozsah **192.168.1.10 – 192.168.1.100** pro přidělování IPv4 adres klientům (rozhraním)
 - a. Spusťte **DHCP Manager**
 - Buď z nabídky **Tools – DHCP** nebo vyberte v levém sloupci roli DHCP a z kontextové nabídky nad jménem serveru zvolte **DHCP Manager**
 - b. V DHCP konzoli zvolte **Add/Remove Bindings...** z nabídky nad **w2016-base**
 - c. V DHCP konzoli rozbalte uzel **w2016-base – IPv4**
 - d. Z kontextové nabídky nad **IPv4** zvolte **New scope**
 1. U **Scope name** zvolte název **Private1**, pokračujte **Next >**
 2. **Start IP address** nastavte na **192.168.1.10** a **End IP address** na **192.168.1.100**
 3. Zkontrolujte, že automaticky vyplněné pole **Subnet mask** obsahuje správnou masku podsítě **255.255.255.0**, pokračujte **Next >**
 4. Zmiňte, k čemu slouží **Exclusions** a pokračujte **Next >**
 5. Nastavte **Lease Duration** na **8 hodin** (zmiňte, k čemu slouží, kdy je dobré nastavit kratší interval apod.), pokračujte **Next >**
 6. V nastavení **Configure DHCP Options** vysvětlete, k čemu slouží, zvolte **Yes,...** a **Next >**
 7. **Default Gateway** nenastavujte, **Next >**
 8. V nastavení **Domain Name and DNS Servers** zadejte jako Parent Domain doménu **testing2.local** a do **DNS server IP address** adresu **192.168.1.1** a pokračujte **Next >**
 9. V části **WINS Servers** nic nenastavujte a pokračujte **Next >**
 10. Zvolte **Yes, I want to activate this scope now** pro aktivaci vytvořeného rozsahu a pokračujte **Next >** a **Finish**
 - e. Na **w10-base** vynutíte obnovení IPv4 adresy
 - a. Spusťte příkaz **ipconfig /renew**
 - f. Ověřte, že **w10-base** obdržel od **DHCP** serveru IPv4 adresu z nastaveného rozsahu

Lab L02 – Pokročilé nastavení DHCP serveru

[Na cvičeních]

Lab L03 – Instalace a základní nastavení DNS serveru

[Provést]

Cíl cvičení

Povýšit server do role **DNS** serveru a vytvořit novou zónu, ověřit funkčnost pomocí klienta

Potřebné virtuální stroje

w10-base

w2016-base

Další prerekvizity

Dokončený úkol **Lab L01** (stačí z hlediska konfigurace síťových rozhraní)

1. Na **w2016-base** spusťte **Server Manager**
 - a. Start → (All Programs) → Administrative Tools → Server Manager
2. Povýšte server do role DNS serveru
 - a. Vyberte **Add Roles and Features** z nabídky **Manage**

- b. Pokračujte [Next >](#)
 - c. Vyberte [Role-based or feature-based installation](#) a pokračujte [Next >](#)
 - d. Vyberte aktuální server a pokračujte [Next >](#)
 - e. V seznamu rolí vyberte [DNS Server](#), potvrďte přidání potřebných funkcí [Add Features](#) a pokračujte třikrát [Next >](#)
 - f. Zaškrtněte [Restart the destination server automatically if required](#) a potvrďte instalaci [Install](#)
 - g. Po dokončení instalace zkontrolujte notifikace [Server Manageru](#)
3. Spustíte **DNS Manager**
 - Buď z nabídky [Tools – DNS](#) nebo vyberte v levém sloupci roli DNS a z kontextové nabídky nad jménem serveru zvolte [DNS Manager](#)
4. Vytvořte novou zónu **testing2.local**
 - a. Klikněte pravým na [Forward Lookup Zones](#) a zvolte [New Zone...](#)
 - b. Pokračujte [Next >](#)
 - c. Vyberte [Primary Zone](#) a pokračujte [Next >](#)
 - d. V části pojmenování zóny zadejte jako [Zone name](#) **testing2.local** a pokračujte [Next >](#)
 - e. V části uložení zónového souboru ponechte výchozí nastavení [Create a new file with this file name](#) s vygenerovaným názvem souboru podle názvu vytvářené zóny a pokračujte [Next >](#)
 - f. V části nastavení dynamických aktualizací ponechte výchozí volbu [Do not allow dynamic updates](#) a pokračujte [Next >](#)
 - g. Potvrďte vytvoření nové zóny pomocí [Finish](#)
5. Přidejte do zóny **testing2.local** nový **A** záznam pro **w2016-base**
 - a. Klikněte pravým na zónu **testing2.local** a zvolte [New Host \(A or AAAA\)...](#)
 - b. Jako [Name](#) zvolte **w2016-base** a jako [IP address](#) nastavte na **192.168.1.1**
 - c. Přidejte záznam pomocí [Add Host](#)
6. Na **w10-base** ověřte správný překlad doménového jména serveru na odpovídající IP adresu
 - a. Spustíte příkaz **nslookup**
 - b. Zadejte **w2016-base.testing2.local**
 - c. Ověřte, že vrácená IP adresa je opravdu IP adresa serveru

Lab L04 – Pokročilá nastavení DNS serveru a klienta

[Na cvičeních]

Studentské úkoly

- Na všech stanicích zakažte *Internal* síťové rozhraní (**LAN1**)

Lab S01 – Vytvoření rezervace pro DHCP klienta

[Povinné]

Cíl cvičení

Na **DHCP** serveru vytvořit rezervaci pro klienta a ověřit, v jakém pořadí se aplikují nastavení obsažená v **DHCP** zprávách, pokud jsou definována na různých úrovních

Potřebné virtuální stroje

w10-base**w2016-base**

Další prerekvizity

Dokončený úkol **Lab L01**

- Na **w2016-base** nastavte na úrovni *serveru* a *rozsahu* **Private1** různé výchozí brány
 - Spustíte **DHCP**
 - Klikněte pravým na **Server Options** a zvolte **Configure Options...**
 - V záložce **General** zaškrtněte možnost **003 Router**
 - Do **IP address** zadejte **192.168.1.1** a zvolte **Add**
 - Potvrďte **OK**
 - Zopakujte body **b - e** tentokrát pro **Scope Options** a adresu **192.168.1.2**
- Na **w10-base** obnovte přidělenou IPv4 adresu pomocí příkazu **ipconfig /renew**
 - Obnovení IPv4 adresy zároveň obnoví veškerá nastavení, jenž při přidělování poskytuje **DHCP** server klientům
- Ověřte, že **w10-base** má nastavenou jako výchozí bránu IPv4 adresu **192.168.1.2**
 - Nastavení na úrovni *rozsahu* mají vždy přednost před nastaveními na úrovni *serveru*
- Vytvořte pro **w10-base** rezervaci u **DHCP** serveru
 - Na **w10-base** zjistěte pomocí příkazu **ipconfig /all** fyzickou (MAC) adresu rozhraní **LAN2**
 - Síťové rozhraní musí odpovídat *Private1*, standardně to je **LAN2**
 - Na **w2016-base** otevřete **DHCP**
 - Klikněte pravým na **Reservations** a zvolte **New Reservation...**
 - Rezervaci pojmenujte **w10-base** (pole **Reservation Name**)
 - Jako **IP address** zvolte **192.168.1.50** a do **MAC address** zadejte zjištěnou fyzickou adresu
 - Cílová IP adresa musí být samozřejmě vždy z rozsahu poskytovaných IPv4 adres
 - Fyzickou (MAC) adresu lze zadat s i bez pomlček, interně se ukládá bez
 - Přidejte rezervaci pomocí **Add**
- Nastavte výchozí bránu **192.168.1.3** pro rezervaci **w10-base**
 - Klikněte pravým na rezervaci **w10-base** a zvolte **Configure Options...**
 - V záložce **General** zaškrtněte možnost **003 Router**
 - Do **IP address** zadejte **192.168.1.3** a zvolte **Add**
 - Potvrďte **OK**
- Na **w10-base** opět obnovte přidělenou IPv4 adresu pomocí příkazu **ipconfig /renew**
- Ověřte, že **w10-base** má přidělenou IPv4 adresu **192.168.1.50** a nastavenou výchozí bránu **192.168.1.3**
 - IP adresa, jenž je zarezervována pro určitého klienta, nemůže být nikdy přidělena jinému klientovi

- Nastavení na úrovni jednotlivých rezervací mají vždy přednost před stejnými nastaveními na ostatních úrovních

Lab S02 – Aplikace nastavení DHCP na základě user class

[Volitelné]

Cíl cvičení

Přidat rozhraní do vytvořené *user class* a ověřit aplikaci nastavení pouze na její členy

Potřebné virtuální stroje

w10-base

w2016-base

Další prerekvizity

Dokončeny úkol Lab S01

1. Na **w2016-base** vytvořte novou *user class* s názvem **ugtest**
 - a. Spustíte **DHCP**
 - b. Klikněte pravým na **IPv4** a zvolte **Define User Classes...**
 - c. Zvolte **Add**
 - d. Do **Display Name** a části **ASCII** zadejte **ugtest** (**Binary** část bude automaticky doplněna)
 - e. Potvrďte vytvoření pomocí **OK**
2. Nastavte adresu **192.168.1.4** jako výchozí bránu pro všechny klienty v *user class* **ugtest**
 - a. V **DHCP** klikněte pravým na rezervaci **w10-base** a zvolte **Configure Options...**
 - b. V záložce **Advanced** zvolte u **User class** **ugtest** a zaškrtněte možnost **003 Router**
 - c. Do **IP address** zadejte **192.168.1.4** a zvolte **Add**
 - d. Potvrďte **OK**
3. Přidejte síťové rozhraní **LAN2** na **w10-base** do *user class* **ugtest**
 - Síťové rozhraní musí odpovídat *Private1*, standardně to je **LAN2**
 - a. Na **w10-base** spustíte příkaz **ipconfig /setclassid "LAN2" ugtest**
 - Nastavení *user class* pro dané rozhraní vyžaduje administrátorské oprávnění
4. Ověřte, že **w10-base** má nastavenou jako výchozí bránu IPv4 adresu **192.168.1.4**
 - Nastavení pro konkrétní *user class* má vždy přednost před nastavením pro všechny

Lab S03 – Vytvoření reverzního mapování pro IPv4

[Povinné]

Cíl cvičení

Vytvořit reverzní zónu pro překlad IPv4 adres na odpovídající doménová jména, ověřit pomocí klienta funkčnost

Potřebné virtuální stroje

w10-base

w2016-base

Další prerekvizity

Dokončeny úkol Lab L03

1. Na **w2016-base** vytvořte IPv4 reverzní zónu pro doménu **testing2.local**
 - a. Spustíte **DNS Manager**
 - b. Klikněte pravým na **Reverse Lookup Zones** a zvolte **New Zone...**
 - c. Na úvodní obrazovce pokračujte **Next >**
 - d. Jako typ zóny vyberte **Primary Zone** a pokračujete **Next >**
 - e. V další části ponechte výchozí nastavení **IPv4 Reverse Lookup Zone** a pokračujete **Next >**

- Specifikace verze protokolu IP přímo určuje část názvu domény, pro kterou danou zónu vytváříme, v případě IPv4 vytváříme subdoménu domény **in-addr.arpa**
- f. Nastavte **Network ID** na **192.168.1** a pokračujte **Next >**
 - **Network ID** určuje (pod)síť, jenž obsahuje IPv4 adresy, pro které se má provádět reverzní mapování, také se z něj určuje název vytvářené domény, v tomto případě je to doména **1.168.192.in-addr.arpa**
- g. V části uložení zónového souboru ponechte výchozí nastavení **Create a new file with this file name** s vygenerovaným názvem souboru podle názvu vytvářené zóny a pokračujte **Next >**
- h. V části nastavení dynamických aktualizací ponechte výchozí volbu **Do not allow dynamic updates** a pokračujte **Next >**
- i. Potvrďte vytvoření nové reverzní zóny pomocí **Finish**
- 2. Přidejte do reverzní zóny **1.168.192.in-addr.arpa** nový **PTR** záznam pro **1**
 - a. Klikněte pravým na reverzní zónu **1.168.192.in-addr.arpa** a zvolte **New Pointer (PTR)...**
 - b. Doplňte **Host IP address** na **192.168.1.1** a u **Name** zadejte **w2016-base.testing2.local**
 - c. Potvrďte přidání záznamu pomocí **OK**
- 3. Na **w10-base** ověřte správný překlad IPv4 adresy serveru na odpovídající doménové jméno
 - a. Spusťte příkaz **nslookup**
 - b. Zadejte **192.168.1.1**
 - c. Ověřte, že vrácené doménové jméno je opravdu doménové jméno serveru
 - Všimněte si, že **nslookup** již hlásí jako **Default Server w2016-base.testing2.local** místo **Unknown** předtím, nyní již totiž může díky *reverse lookup* zóně provádět také reverzní mapování IPv4 adres na odpovídající doménová jména, může tedy zjistit název serveru z jeho IPv4 adresy

Lab S04 – Vytvoření zóny globálních jmen

[Volitelné]

Cíl cvičení

Vytvořit zónu globálních jmen a otestovat funkčnost pomocí klienta

Potřebné virtuální stroje

w10-domain

w2016-dc

1. Na **w2016-dc** nastavte statickou IPv4 adresu **192.168.1.2**
 - a. Spusťte následující příkaz **netsh interface ip set address name="LAN2" source=static addr=192.168.1.2 mask=255.255.255.0**
 - Název **name** musí odpovídat síťovému rozhraní *Private1*, standardně to je **LAN2**
2. Na **w10-domain** nastavte statickou IPv4 adresu **192.168.1.200** a také IPv4 adresu primárního **DNS** serveru **192.168.1.2**
 - a. Spusťte následující příkaz **netsh interface ip set address name="LAN2" source=static addr=192.168.1.200 mask=255.255.255.0**
 - b. Spusťte další příkaz **netsh interface ip set dnsservers name="LAN2" source=static address=192.168.1.2**
 - Oba názvy **name** musí odpovídat síťovému rozhraní *Private1*, standardně to je **LAN2**
 - Oba příkazy vyžadují administrátorské oprávnění

3. Na **w2016-dc** vytvořte zónu globálních jmen (zóna **GlobalNames**)
 - a. Spusťte **DNS Manager**
 - b. Klikněte pravým na **Forward Lookup Zones** a zvolte **New Zone...**
 - c. Na úvodní obrazovce pokračujte **Next >**
 - d. Jako typ zóny vyberte **Primary Zone** a dole zaškrtněte **Store The Zone In Active Directory**, pokračujte **Next >**
 - Zóna globálních jmen musí být integrována v Active Directory
 - e. V části rozsahu replikace zóny zvolte **To all DNS servers running on domain controllers in this forest: testing.local**
 - Zóna globálních jmen musí být replikována v rámci celého *forestu*
 - f. V části pojmenování zóny zadejte do **Zone name** **GlobalNames** a pokračujte **Next >**
 - Zóna globálních jmen je ve své podstatě normální zóna jako všechny ostatní, pouze speciálně pojmenovaná (tak ji také systém rozeznává od zbylých zón)
 - g. V části nastavení dynamických aktualizací zvolte možnost **Do not allow dynamic updates** a pokračujte **Next >**
 - Zóna globálních jmen nepodporuje dynamické aktualizace, všechny záznamy musí být přidány manuálně
 - h. Potvrďte vytvoření zóny globálních jmen pomocí **Finish**
4. Na **w2016-dc** povolte podporu zóny globálních jmen
 - a. Spusťte příkaz **dnscmd /config /enableglobalnamesupport 1**
 - Povolení podpory zóny globálních jmen vyžaduje administrátorské oprávnění
 - b. Restartujte službu **DNS**
 1. Přejděte do **DNS Manageru**
 2. Klikněte pravým na **w2016-dc**, vyberte **All Tasks** a pak **Restart**
5. Přidejte do zóny globálních jmen nový **CNAME** záznam odkazující **wdc** na **w2016-dc**
 - a. Spusťte příkaz **dnscmd w2016-dc.testing.local /recordadd GlobalNames wdc cname w2016-dc.testing.local**
 - Přidání záznamu do zóny vyžaduje administrátorské oprávnění
 - Jelikož zóna globálních jmen slouží k náhradě **WINS** serverů, obsažená jména jsou **NetBIOS** jmény a musí mít tedy délku maximálně 15 znaků
6. Na **w10-domain** ověřte správný překlad globálních jmen na odpovídající doménová jména
 - a. Spusťte příkaz **nslookup**
 - b. Zadejte **wdc**
 - c. Ověřte, že vrácené doménové jméno je opravdu doménové jméno odpovídající serveru **w2016-dc** a IPv4 adresa taky náleží severu
 - Pokud není počítač připojen do domény ležící ve *forestu*, ve kterém se zóna globálních jmen replikuje, nebude výše popsany postup fungovat! Pro použití globálních jmen mimo takovou doménu je potřeba použít **<globální jméno>.GlobalNames** nebo přidat zónu **GlobalNames** do seznamu používaných *suffixů* pro dané rozhraní. Můžete si ověřit, že na **w10-base** nebude **wdc** fungovat, ale **wdc.GlobalNames** ano