# Úvod do SSL

Jan Outrata

8. září 2002

(Překlad z *Introduction to SSL* [1])

Tento dokument představuje protokol Secure Sockets Layer (SSL). Původně vyvinutý Netscapem byl SSL přijat na WWW pro autentizovanou a šifrovanou komunikaci mezi klienty a servery.

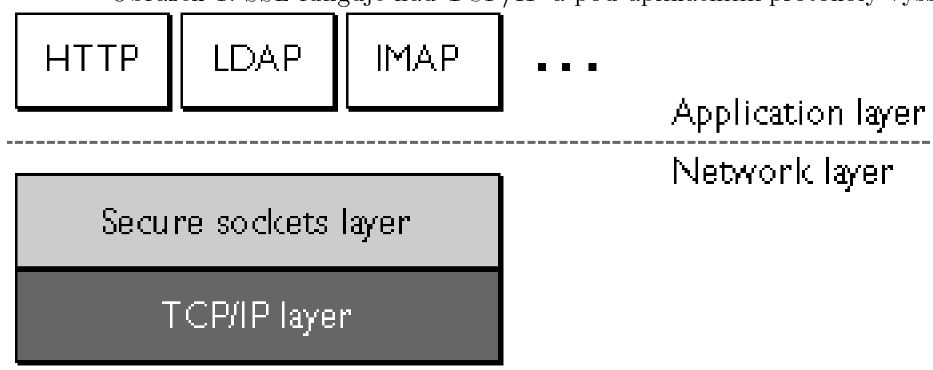
Nový standard Internet Engineering Task Force (IETF) nazvaný Transport Layer Security (TLS) je založen na SSL. Nedávno byl publikován jako IETF Internet-Draft, *The TLS Protocol Version 1.0* [2].

Dokument předpokládá znalost základních principů šifrování s veřejným klíčem.

## 1 Protokol SSL

Transmission Control Protocol/Internet Protocol (TCP/IP) zajišťuje transport a směrování dat přes Internet. Jiné protokoly, jako HyperText Transport Protocol (HTTP), Lightweight Directory Access Protocol (LDAP), nebo Internet Messaging Access Protocol (IMAP), fungují nad TCP/IP ve smyslu, že používají TCP/IP pro funkci typických aplikačních úloh jako zobrazování webových stránek nebo provoz emailového serveru.

Obrázek 1: SSL funguje nad TCP/IP a pod aplikačními protokoly vyšší úrovně




SSL protokol funguje nad TCP/IP a pod aplikačními protokoly vyšší úrovně jako HTTP nebo IMAP. Používá TCP/IP jménem protokolu vyšší úrovně a umožňuje SSL serveru se autentizovat SSL klientovi, klientovi se autentizovat serveru a oběma strojům vytvořit šifrované spojení.

Tyto možnosti adresují základní potřeby komunikace po Internetu a jiných TCP/IP sítích:

- **Autentizace SSL serveru** umožňuje uživateli potvrdit identitu serveru. SSL klient může použít standardní techniky šifrování s veřejným klíčem pro kontrolu, že certifikát serveru a veřejná identifikace jsou platné a byly potvrzeny certifikační autoritou (CA) uvedenou v klientově seznamu důvěryhodných CA. Toto potvrzení může být důležité, pokud uživatel např. posílá číslo kreditní karty po síti a chce si zkontrolovat identitu přijímacího serveru.

- **Autentizace SSL klienta** umožňuje serveru potvrdit identitu uživatele. Použitím stejným technik jako pro autentizaci serveru, SSL server může zkontrolovat, že certifikát klienta a veřejná identifikace jsou platné a byly potvrzeny certifikační autoritou (CA) uvedenou v serverově seznamu důvěryhodných CA. Toto potvrzení může být důležité, pokud je server např. banka, která posílá důvěrné finanční informace zákazníkovi, a chce si zkontrolovat identitu příjemce.
- **Šifrované SSL spojení** vyžaduje, aby všechny informace zasílané mezi klientem a serverem byly šifrované odesílatelem a dešifrované příjemcem, tedy poskytuje vysoký stupeň důvěryhodnosti. Důvěryhodnost je důležitá pro obě strany pro jakoukoliv privátní transakci. Navíc, všechna data zaslaná přes šifrované SSL spojení jsou chráněna mechanizmem pro detekci narušení – tj. pro automatické zjištění, zda byla data během přenosu pozměněna.

SSL protokol obsahuje dva pod-protokoly: SSL record protokol a SSL handshake protokol. SSL record protokol definuje formát používaný pro přenos dat. SSL handshake protokol zahrnuje použití SSL record protokolu pro výměnu řady zpráv mezi SSL serverem a SSL klientem když poprvé vytvoří SSL spojení. Tato výměna zpráv je navržena pro usnadnění následujících akcí:

- Autentizovat server klientovi.
- Umožnit klientovi i serveru vybrat šifrovací algoritmy (šifry), které oba podporují.
- Případně autentizovat klienta serveru.
- Použít techniky šifrování s veřejným klíčem pro generování sdílených secret.
- Vytvořit šifrované SSL spojení.

# 2 Šifry používané v SSL

SSL protokol podporuje použití spousty různých šifrovacích algoritmů (**šifer**) v operacích jako autentizace serveru a klienta navzájem, přenos certifikátů a vytvoření klíčů spojení. Klienti a servery mohou podporovat různé **sady šifer**, nebo množiny šifer, závisí to na faktorech jako verze SSL, firemní politika týkající se akceptovatelné síly šifrování, a vládní omezení na export SSL softwaru. Kromě dalších funkcí, SSL handshake protokol určuje, jak server a klient zjednají, kterou sadu šifer použijí pro vzájemnou autentizaci, přenos certifikátů a vytvoření klíčů spojení.

Popisy sad šifer, které následují, se odkazují na tyto algoritmy:

- **DES**. Data Encryption Standard, šifrovací algoritmus používaný vládou USA.
- **DSA**. Digital Signature Algorithm, část digitálního autentizačního standardu používaného vládou USA.
- **KEA**. Key Exchange Algorithm, algoritmus pro výměnu klíčů používaný vládou USA.
- **MD5**. Message Digest algoritmus vyvinutý Rivestem.
- **RC2 a RC4**. Rivestovy šifry vyvinuté pro RSA Data Security.
- **RSA**. Algoritmus s veřejným klíčem pro šifrování i autentizaci. Vyvinutý Rivestem, Shamirem a Adlemanem.
- **RSA key exchange**. Algoritmus na výměnu klíčů pro SSL založený na algoritmu RSA.
- **SHA-1**. Secure Hash Algorithm, hashovací funkce používaná vládou USA.

- **SKIPJACK**. Algoritmus s tajným klíčem implementovaný v hardwaru FORTEZZA používaném vládou USA.
- **Triple-DES**. DES aplikovaný třikrát.

Algoritmy na výměnu klíčů jako KEA a RSA key exchange poskytují způsob, jakým server a klient určí symetrické klíče, které použijí během SSL spojení. Nejvíce používané sady SSL šifer používají RSA key exchange.

SSL 2.0 a SSL 3.0 protokoly podporují překrývání množin sad šifer. Administrátoři mohou povolit nebo zakázat kteroukoliv z podporovaných sad šifer u klientů i serverů. Když si klient a server vyměňují informace během SSL handshake, zjistí nejsilnější povolené sady šifer, které mají společné, a ty použijí pro SSL spojení.

Rozhodnutí o tom, které sady šifer organizace povolí, závisí na citlivosti dat, rychlosti šifry a exportních zákonech.

Nějaké organizace mohou chtít zakázat slabší šifry pro zabránění SSL spojení se slabým šifrováním. Nicméně, kvůli omezením vlády USA na produkty, které podporují cokoliv silnějšího než 40-bitové šifrování, zakázáním podpory všech 40-bitových šifer efektivně omezují přístup pouze na internetové prohlížeče, které jsou dostupné v USA (pokud server nemá Global Server ID, které povoluje mezinárodním klientům použít silnější šifrování).

Pro obsloužení největšího počtu uživatelů mohou administrátoři povolit co nejširší výběr sad SSL šifer. Takto, když komunikují server nebo klient na území USA s jiným serverem nebo klientem taktéž na území USA, vyjednají si použití nejsilnějších možných šifer. A když komunikují server nebo klient na území USA s mezinárodním serverem nebo klientem, vyjednají si použití těch šifer, které jsou povoleny regulacemi exportu USA.

Ale protože 40-bitové šifry mohou být rozbity relativně rychle, administrátoři, kteří se obávají odposlouchávání a jejichž uživatelé mohou legálně používat silnější šifry, by měli zakázat 40-bitové šifry.

### 2.1 Sady šifer s RSA Key Exchange

V Tabulce 1 jsou uvedeny sady šifer podporované SSL, který používá algoritmus RSA key-exchange. Pokud není řečeno jinak, všechny šifry v tabulce jsou podporovány SSL 2.0 i SSL 3.0. Sady šifer jsou uvedeny od nejsilnějších k nejslabším.

### 2.2 Sady šifer FORTEZZA

V Tabulce 2 jsou uvedeny sady šifer podporované produkty s FORTEZZA pro SSL 3.0. FORTEZZA je šifrovací systém používaný vládními agenturami USA pro manipulaci s citlivými daty. Poskytuje hardwarovou implementaci dvou šifer vyvinutými vládou USA: FORTEZZA KEA a SKIPJACK. Šifry FORTEZZA pro SSL používají Key Exchange Algorithm (KEA) místo algoritmu RSA key-exchange a karty FORTEZZA a DSA pro autentizaci klienta.

## 3 SSL Handshake

SSL protokol používá kombinaci šifrování s veřejným a tajným klíčem. Šifrování s tajným klíčem je mnohem rychlejší než šifrování s veřejným klíčem, ale šifrování s veřejným klíčem poskytuje lepší techniky autentizace. SSL spojení vždy začíná výměnou zpráv nazývanou **SSL handshake**. Handshake umožňuje serveru se autentizovat klientovi použitím technik šifrování s veřejným klíčem, pak klient a server společně vytvoří tajné klíče pro rychlé šifrování, dešifrování a detekci narušení během spojení. Případně také handshake umožňuje klientovi se autentizovat serveru.

Tabulka 1: Sady šifer podporované SSL protokolem, který používá algoritmus RSA key-exchange

| Kategorie síly a doporučené použití | Sady šifer |
|---|---|
| **Nejsilnější sady šifer.** Povolené jen na území USA nebo jen mimo něj. Tyto sady šifer jsou vhodné pro banky a jiné instituce, které se starají o velmi citlivá data. | **Triple DES (168-bitové šifrování) a autentizace zpráv SHA-1.** Triple DES je nejsilnější šifra podporovaná SSL, ale není tak rychlá jako RC4. Triple DES používá třikrát klíč pro DES. Protože klíč je tak velký, je tu více možných klíčů než pro ostatní šifry – asi $3,7 * 10^{50}$. |
| **Silné sady šifer**. Povolené jen na území USA nebo jen mimo něj. Tyto sady šifer podporují šifrování, které je dost silné pro většinu komerčních nebo vládních potřeb. | **RC4 a RC2 (128-bitové šifrování) a autentizace zpráv MD5.** Protože šifry RC4 a RC2 jsou 128-bitové, jsou druhé nejsilnější po Triple DES. Poskytují asi $3,4 * 10^{38}$ možných klíčů. Šifry RC4 jsou nejrychlejší z podporovaných šifer, RC2 jsou pomalejší. SSL 3.0 nepodporuje sadu šifer RC2 s MD5.<br>**DES (56-bitové šifrování) a autentizace zpráv SHA-1.** DES je silnější než 40-bitové šifrování, poskytuje asi $7,2 * 10^{16}$ možných klíčů. SSL 2.0 používá MD5 pro autentizaci zpráv. |
| **Exportovatelné sady šifer.** Tyto sady šifer nejsou tak silné jako předešlé, ale mohou být exportovány mimo území USA. Poskytují nejsilnější šifrování dostupné pro exportovatelné produkty z USA. | **RC4 a RC2 (40-bitové šifrování) a autentizace zpráv MD5.** Poskytuje asi $1,1 * 10^{12}$ možných klíčů. |
| **Nejslabší sada šifer.** Tato sada šifer poskytuje autentizaci a detekci narušení, ale ne šifrování. Data zaslaná použitím této sady šifer nejsou zašifrovaná a mohou být odposloucháвána. | **Žádné šifrování a autentizace zpráv MD5.** Tato sada je typicky používána v případě, že server a klient nemají společné žádné jiné šifry. Není podporována SSL 2.0. |

Přesné programátorské detaily vyměněných zpráv během SSL handshake jsou mimo rámec tohoto dokumentu. Ale jednotlivé kroky mohou být shrnuty následovně (předpokládáme použití sad šifer uvedených v části 2.1):

1. Klient pošle serveru číslo verze klientova SSL, nastavení šifer, náhodně vygenerovaná data a další informace, které server potřebuje pro komunikaci s klientem pomocí SSL.

2. Server pošle klientovi číslo verze serverově SSL, nastavení šifer, náhodně vygenerovaná data a další informace, které klient potřebuje pro komunikaci se serverem pomocí SSL. Server také pošle svůj vlastní certifikát a, pokud klient požaduje prostředky serveru, které vyžadují autentizaci klienta, požaduje klientův certifikát.

3. Klient použije nějaké informace zaslané serverem pro autentizaci serveru (viz část 3.1). Pokud server nemůže být autentizován, uživatel je varován o problému a informován, že šifrované a autentizované spojení nemůže být navázáno.

4. Použitím všech dat dosud vygenerovaných při handshake, klient (se spoluprací serveru, to závisí na použité šifře) vytvoří **premaster secret** pro spojení, zašifruje ho serverovým veřejným klíčem (získaným ze serverově certifikátu) a pošle zašifrované premaster secret serveru.

Tabulka 2: Sady šifer FORTEZZA podporované produkty s FORTEZZA pro SSL 3.0

| Kategorie síly a doporučené použití | Sady šifer |
|---|---|
| **Silné sady šifer FORTEZZA.** Povolené jen pro zařízení v USA. Tato sada šifer poskytuje šifrování, které je dostatečně silné pro komerční a vládní potřeby. | **RC4 (128-bitové šifrování) a autentizace zpráv SHA-1.** Stejně jako RC4 (128-bitové šifrování) s autentizací zpráv MD5. Není podporována SSL 2.0.<br>**RC4 se SKIPJACK (80-bitové šifrování) a autentizace zpráv SHA-1.** SKIPJACK je šifrovací algoritmus s tajným klíčem implementovaný v hardwaru FORTEZZA. Není podporována SSL 2.0. |
| **Nejslabší sady šifer FORTEZZA.** Tato sada šifer poskytuje autentizaci a detekci narušení, ale ne šifrování. Data zaslaná použitím této sady šifer nejsou zašifrovaná a mohou být odposlouchávána. | **Žádné šifrování a autentizace zpráv SHA-1.** Není podporována SSL 2.0. |

5. Pokud server požadoval autentizaci klienta, klient také podepíše jiný kus dat, který je jedinečný pro tento handshake a známý klientovi i serveru. V tomto případě klient pošle podepsaná data a klientův vlastní certifikát serveru spolu se zašifrovaným premaster secret.

6. Pokud server požadoval autentizaci klienta, pokusí se autentizovat klienta (viz část 3.2). Pokud klient nemůže být autentizován, spojení je ukončeno. Pokud klient může být úspěšně autentizován, server použije jeho privátní klíč na dešifrování premaster secret, a pak provede řadu kroků (které klient také provede, počínaje stejným premaster secret) k vygenerování **master secret**.

7. Klient i server použijí master secret k vygenerování **klíčů spojení (session keys)**, což jsou tajné klíče používané k šifrování a dešifrování informací vyměněných během SSL spojení a pro ověření jeho neporušenosti – tj. pro detekci jakýchkoliv změn dat mezi jejich vysláním a příjmem přes SSL spojení.

8. Klient pošle zprávu serveru, informující ho, že další zprávy od něj budou šifrovány klíčem spojení. Pak pošle samostatnou zašifrovanou zprávu, že klientova část handshake je dokončena.

9. Server pošle zprávu klientovi, informující ho, že další zprávy od něj budou šifrovány klíčem spojení. Pak pošle samostatnou zašifrovanou zprávu, že serverová část handshake je dokončena.

10. SSL handshake je teď kompletní a SSL spojení je zahájeno. Klient a server používají klíče spojení pro šifrování a dešifrování dat, která si posílají a pro ověření jejich neporušenosti.

Před pokračováním spojení mohou být servery nakonfigurovány pro kontrolu, zda klientův certifikát je v uživatelově záznamu v LDAP adresáři. Tato konfigurace poskytuje jeden ze způsobů ujištění, že klientův certifikát nebyl zrušen.

Je důležité poznamenat, že autentizace klienta i serveru zahrnuje zašifrování nějakých dat jedním klíčem z páru veřejný-privátní klíč a jejich dešifrování druhým klíčem:

- V případě autentizace serveru, klient zašifruje premaster secret veřejným klíčem serveru. Jenom odpovídající privátní klíč ho může správně dešifrovat, takže klient má jakousi jistotu, že identita asociovaná s veřejným klíčem je fakticky server, ke kterému je klient připojen. Jinak server nemůže dešifrovat premaster secret a nemůže vygenerovat tajné klíče potřebné pro komunikaci a spojení bude ukončeno.

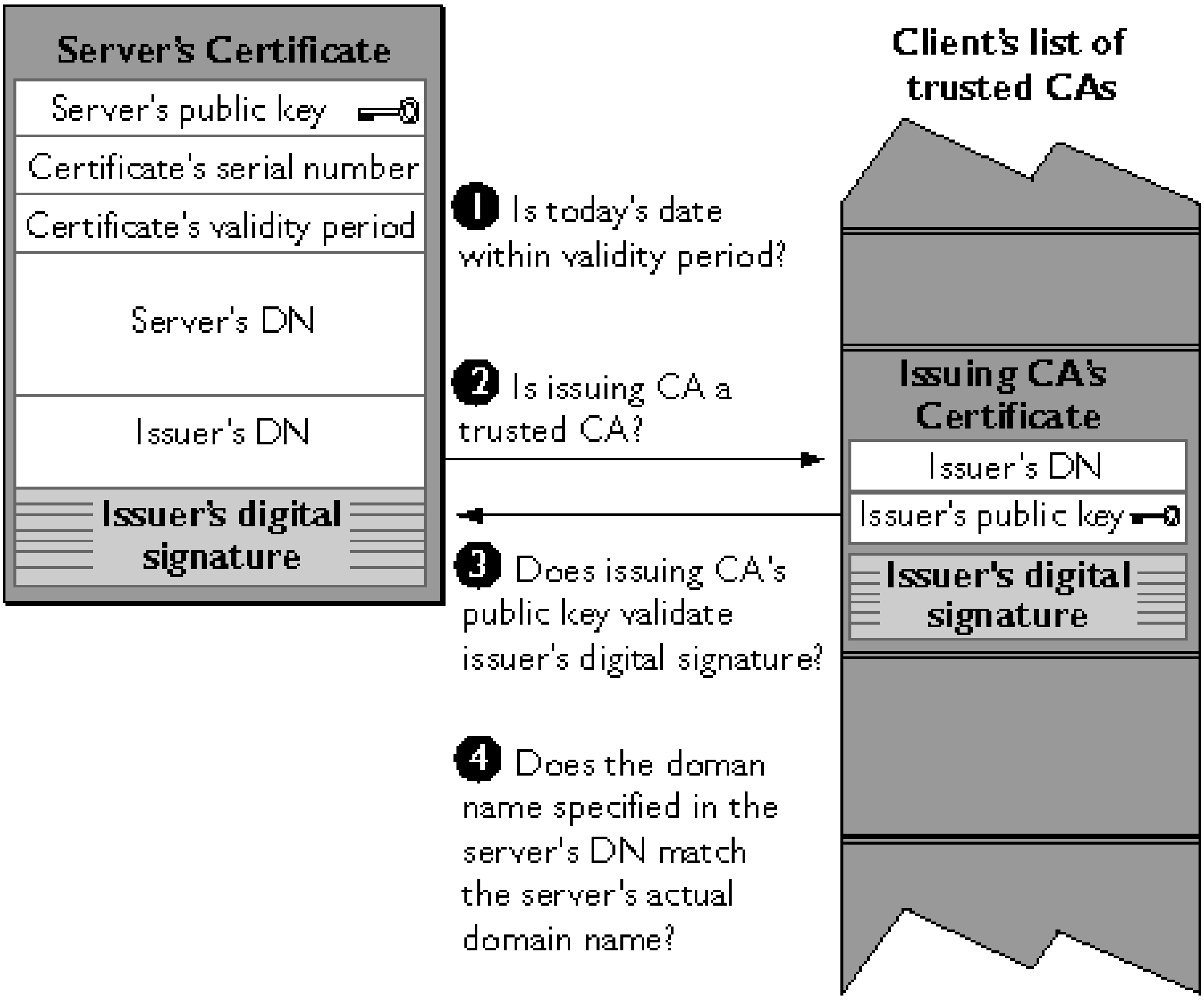
- V případě autentizace klienta, klient zašifruje nějaká náhodná data svým privátním klíčem – tzn. vytvoří digitální podpis. Veřejný klíč v klientově certifikátu může správně potvrdit digitální podpis jenom pokud byl použit odpovídající privátní klíč. Jinak server nemůže potvrdit digitální podpis a spojení je ukončeno.

## 3.1 Autentizace serveru

SSL klient vždy vyžaduje autentizaci serveru nebo šifrované potvrzení serverové identity. V kroku 2 SSL handshake pošle server klientovi certifikát k jeho autentizaci. Klient použije certifikát v kroku 3 k autentizaci identity, kterou certifikát reprezentuje.

K autentizaci vazby mezi veřejným klíčem a certifikátem identifikovaným serverem, který vlastní tento veřejný klíč, musí SSL klient dostat odpověď "ano" na čtyři otázky na obrázku 2. I když čtvrtá otázka není částí SSL protokolu, je odpovědností klienta provést tento požadavek, který představuje jakési ujištění identity serveru a tedy pomáhá chránit proti formě útoku známé jako "man in the middle".

Obrázek 2: Jak klient autentizuje certifikát serveru




SSL klient provede tyto kroky pro autentizaci identity serveru:

1. **Je dnešní datum v době platnosti?** Klient zkontroluje dobu platnosti certifikátu serveru. Pokud aktuální datum a čas jsou mimo tento rozsah, proces autentizace nebude dál pokračovat.
2. **Je vydávající CA důvěryhodnou CA?** Každý SSL klient si spravuje seznam certifikátů důvěryhodných CA. Tento seznam určuje, které certifikáty serveru klient přijme. Pokud je jméno (distinguished name - DN) vydávající CA obsaženo v klientově seznamu důvěryhodných CA, odpověď na tuto otázku je ano. Pokud vydávající CA není na seznamu, server nebude autentizován, pokud klient nemůže potvrdit řetěz certifikátů končící CA, která je na seznamu.
3. **Potvrdí veřejný klíč vydávající CA digitální podpis vydávajícího?** Klient použije veřejný klíč z certifikátu CA (který najde ve svém seznamu důvěryhodných CA), k ověření digitálního podpisu CA na certifikátu serveru. Pokud se změnila informace na certifikátu serveru od doby, co byl podepsán CA, nebo pokud veřejný klíč certifikátu CA neodpovídá privátnímu klíči, který CA použila k podepsání certifikátu serveru, klient neautentizuje identitu serveru. Pokud může být digitální podpis CA ověřen, klient považuje certifikát serveru za platný od té CA a pokračuje. V tomto bodě klient rozhodl, že certifikát serveru je platný.
4. **Je doménové jméno na certifikátu serveru stejné jako doménové jméno samotného serveru?** Tento krok potvrdí, že server má síťovou adresu specifikovanou doménovým jménem na certifikátu serveru. I když tento krok není částí SSL protokolu, poskytuje jedinou ochranu proti formě útoku známé jako Man-in-the-Middle útok (viz část 3.1.1). Klient musí provést tento krok a musí odmítnout autentizaci serveru nebo vytvoření spojení, pokud doménové jméno nesouhlasí.
5. **Server je autentizován.** Klient pokračuje v SSL handshake. Pokud se klient z jakéhokoliv důvodu nedostane do tohoto bodu, server identifikovaný certifikátem nemůže být autentizován a uživatel bude varován a informován, že šifrované a autentizované spojení nemůže být navázáno. Pokud server vyžaduje autentizaci klienta, provede kroky popsané v části 3.2.

Po těchto krocích musí server úspěšně použít svůj privátní klíč k dešifrování premaster secret, které mu klient poslal v kroku 4 SSL handshake. Jinak bude SSL spojení ukončeno. Toto představuje další ujištění, že identita asociovaná s veřejným klíčem na certifikátu serveru je server, se kterým je klient spojen.

### 3.1.1 Man-in-the-Middle útok

Jak bylo doporučeno v kroku 4 výše, klientská aplikace musí porovnat doménové jméno serveru specifikované na certifikátu serveru a skutečné doménové jméno serveru, se kterým se klient snaží komunikovat. Tento krok je nutný k ochraně proti man-in-the-middle útoku, který funguje následovně.

“Man in the middle” je program, který zadržuje veškerou komunikaci mezi klientem a serverem, se kterým se klient snaží komunikovat přes SSL. Program zadrží legitimní klíče, které jsou zasílány tam a zpět během SSL handshake, nahradí je svými vlastními, a pro klienta to vypadá tak, že on je server, a pro server, že je klient.

Šifrovaná informace vyměněná na začátku SSL handshake je vlastně zašifrovaná veřejným nebo privátním klíčem tohoto programu a ne klientovým nebo serverovým pravým klíčem. Program nakonec vytvoří jednu množinu klíčů spojení pro použití se skutečným serverem a druhou množinou klíčů spojení pro použití s klientem. To programu umožní nejen číst všechna data putující mezi klientem a skutečným serverem, ale také data měnit bez toho, že to bude detekováno. Tedy je velmi důležité, aby si klient ověřil, že doménové jméno na certifikátu serveru odpovídá doménovému jménu serveru, se kterým se klient snaží komunikovat – kromě ověření platnosti certifikátu provedením ostatních kroků.

## 3.2 Autentizace klienta

SSL server může být nakonfigurován tak, že vyžaduje autentizaci klienta, nebo šifrované potvrzení klientovy identity. Když takto nakonfigurovaný server vyžaduje autentizaci klienta (krok 6 SSL handshake), pošle klient serveru pro svoji autentizaci certifikát a ještě kousek digitálně podepsaných dat. Server použije podepsaná data k ověření platnosti veřejného klíče na certifikátu a na autentizaci identity, kterou certifikát reprezentuje.

SSL protokol vyžaduje, aby klient vytvořil digitální podpis jednocestným zahešováním dat, generovaných náhodně během SSL handshake a známých pouze klientovi a serveru. Hash dat je zašifrován privátním klíčem, který odpovídá veřejnému klíči na certifikátu, který je prezentován serveru.

K autentizaci vazby mezi veřejným klíčem a osobou nebo jinou entitou identifikovanou certifikátem, který obsahuje veřejný klíč, musí SSL server dostat odpověď "ano" na první čtyři otázky na obrázku 3. I když pátá otázka není součástí SSL protokolu, server může být nakonfigurován tak, aby využil uživatelova záznamu v LDAP adresáři jako součásti procesu autentizace.

SSL server provede tyto kroky k autentizaci uživatelovy identity:

1. **Potvrdí uživatelův veřejný klíč jeho digitální podpis?** Server zkontroluje, zda může být uživatelův digitální podpis potvrzen veřejným klíčem na certifikátu. Pokud ano, server zjistil, že veřejný klíč souhlasí s privátním klíčem použitým pro vytvoření podpisu a že data nebyla pozměněna od doby, co byla podepsána.

   V tomto bodě ale ještě nebyla zjištěna vazba mezi veřejným klíčem a jménem specifikovaným na certifikátu. Certifikát mohl být vytvořen někým vydávajícím se za uživatele.

2. **Je dnešní datum v době platnosti?** Server zkontroluje dobu platnosti certifikátu klienta. Pokud aktuální datum a čas jsou mimo tento rozsah, proces autentizace nebude dál pokračovat.

3. **Je vydávající CA důvěryhodnou CA?** Každý SSL server si spravuje seznam certifikátů důvěryhodných CA. Tento seznam určuje, které certifikáty klienta server přijme. Pokud je jméno vydávající CA obsaženo v serverově seznamu důvěryhodných CA, odpověď na tuto otázku je ano. Pokud vydávající CA není na seznamu, klient nebude autentizován, pokud server nemůže potvrdit řetěz certifikátů končící CA, která je na seznamu.

4. **Potvrdí veřejný klíč vydávající CA digitální podpis vydávajícího?** Server použije veřejný klíč z certifikátu CA (který najde ve svém seznamu důvěryhodných CA), k ověření digitálního podpisu CA na certifikátu klienta. Pokud se změnila informace na certifikátu klienta od doby, co byl podepsán CA, nebo pokud veřejný klíč certifikátu CA neodpovídá privátnímu klíči, který CA použila k podepsání certifikátu klienta, server neautentizuje identitu klienta. Pokud může být digitální podpis CA ověřen, server považuje certifikát uživatele za platný od té CA a pokračuje. V tomto bodě SSL protokol dovoluje serveru považovat klienta za autentizovaného a pokračovat ve spojení.

5. **Je uživatelův certifikát uveden v LDAP záznamu pro tohoto uživatele?** Tento nepovinný krok poskytuje jeden ze způsobů, jak může administrátor zrušit certifikát uživatele i když projde testy ve všech předchozích krocích. Server může také automaticky smazat zrušený certifikát z uživatelova záznamu v LDAP adresáři. Servery, které jsou nastaveny k provádění tohoto kroku, odmítnou autentizovat tento certifikát nebo vytvořit spojení. Pokud je uživatelův certifikát v adresáři stejný jako certifikát prezentovaný při SSL handshake, server pokračuje.

6. **Je autentizovaný klient autorizován pro přístup k požadovaným prostředkům?** Server zkontroluje, k jakým prostředkům má klient povolen přístup v souladu s access control lists (ACLs) serveru a vytvoří spojení s odpovídajícím přístupem. Pokud se server z jakéhokoliv důvodu nedostane do tohoto kroku, uživatel identifikovaný certifikátem nemůže být autentizován a není mu povolen přístup k jakýmkoliv prostředkům serveru, které vyžadují autentizaci.

## Reference


[1] *Introduction to SSL*
`http://developer.netscape.com/docs/manuals/security/sslin/`

[2] *The TLS Protocol Version 1.0*
`ftp://ds.internic.net/internet-drafts/draft-ietf-tls-protocol-05.txt`

Obrázek 3: Jak server autentizuje certifikát klienta