402 BİLGİSAYAR AĞLARI DERSİ

ARAŞTIRMA ÖDEVİ-IV

Ayben GÜLNAR-191180041

NİSAN 2023

İçindekiler Tablosu

Şekiller Listesi

Not: Şekiller tez formatında sola yaslı olduğu için sola yasladım.

1.GİRİŞ

İnternet, günlük yaşamımızın ayrılmaz bir parçası haline geldi ve hemen hemen her işlemimiz internet üzerinden gerçekleştiriliyor. Ancak bu iletişim kanalları aracılığıyla bilgi aktarımının güvenliği büyük bir endişe kaynağı haline gelmiştir. İşte bu noktada Secure Sockets Layer (SSL) ve Transport Layer Security (TLS) gibi protokoller, internet trafiğini şifreleyerek güvenli bir iletişim sağlarlar. SSL ve TLS, web sitelerinin kimlik doğrulama ve veri bütünlüğü gibi güvenlik sorunlarına çözümler sunar. Bu makale, SSL ve TLS'nin nasıl çalıştığı, hangi güvenlik özelliklerini sunduğu, hangi uygulamalarda kullanıldığı ve güvenlik açıkları hakkında detaylı bir inceleme yaparak, internet iletişimini daha güvenli hale getirme konusunda okuyuculara bilgi sağlamayı amaçlamaktadır.

Bir mağazaya girdiğimizde, kiminle iş yaptığımızı biliriz. Ürünleri, markalamayı ve mağaza görevlisini görürüz. Satın aldığımız üründe bir sorun olursa, mağaza müdürüne veya sahibine başvurabileceğimizden emin olabiliriz. Ancak internet ortamında, web sitesini (sanal mağazayı) kimin yönettiğini genellikle güvenilir bir şekilde bilemeyiz. Müşteriler online bir satın alma yapmak amacıyla bir web sitesini ziyaret ettiklerinde, kimin ödeme alacağını bilmek isterler. Web sitesi sahibinin kimliğinin kanıtını ve gönderdikleri kişisel bilgilerinin diğer internet kullanıcıları tarafından ele geçirilemeyeceğini bilmek isterler. İşte bu noktada SSL sertifikaları önem kazanıyor. SSL (Güvenli Soket Katmanı), bir web tarayıcısı ve web sunucusunun güvenli bir şekilde iletişim kurmasını sağlayan Netscape tarafından geliştirilen bir protokoldür. SSL protokolü, bir SSL bağlantısının yapılabilmesi için web sunucusunun üzerinde bir dijital sertifika yüklü olmasını gerektirir. SSL, SSL bağlantısı üzerinden aktarılan verileri şifrelemek için bir genel anahtar kullanarak çalışır. Hem Netscape Navigator hem de Internet Explorer SSL'yi destekler ve birçok web sitesi kredi kartı numaraları gibi gizli kullanıcı bilgilerini elde etmek için bu protokolü kullanır. SSL bağlantısı gerektiren URL'ler, http yerine https ile başlarlar [1].

TLS protokolü, çevrimiçi işlemlerde iletişimi güvence altına almak için kullanılan bir yöntemdir ve finansal, sağlık ve sosyal işlemler gibi çeşitli alanlarda kullanılır. Bu işlemler sırasında işlenen kişisel kimlik bilgileri, finansal veriler veya oturum açma bilgileri gibi hassas verilerin korunması önemlidir ve TLS bu verilerin güvenli bir şekilde iletilmesini sağlar [2].

Bu araştırma ödevimde Secure Sockets Layer (SSL) ve Transport Layer Security (TLS) nedir, nasıl çalışır, güvenlik özellikleri gibi çeşitli konuları detaylıca ele aldım.

2. SECURE SOCKETS LAYER (SSL) VE TRANSPORT LAYER SECURİTY (TLS) TANIMLARI

2.1 Secure Socket Layer (SSL)

SSL, Secure Sockets Layer protokolünün kısaltmasıdır ve Netscape tarafından geliştirilmiştir. Güvenli HyperText transfer protokolü (HTTPS), bilgisayarlar arasında şifrelenmiş bilgi transferi için tasarlanmış bir iletişim protokolüdür. HTTPS, SSL kullanan bir http'dir. Güvenli bir soket katmanı, HTTPS kullanan bir Web sunucusunda devreye giren bir şifreleme protokolüdür. SSL, soket iletişiminin bir türüdür ve TCP/IP ve üst katman uygulamaları arasında yer alır; uygulama katmanına herhangi bir değişiklik gerektirmez. Genellikle sunucu ve istemci arasında bağlantıyı güvence altına almak için kullanılır. SSL soketlerine bir çağrı yapmak için tipik bir TCP/IP soketleri çağrısı yerine kullanılır ve bir dizi uygulama programlama arayüzü (API) sunulur. Soket katmanında güvenliği "takmak", aynı güvenliği sağlamak için gerekli şifreleme bileşenlerinin inşa edilmesi ve entegre edilmesine kıyasla geliştirme süresini önemli ölçüde azaltabilir. Verinin taşınması ve yönlendirilmesini İnternet üzerinden yöneten protokol, Transmission Control Protocol/Internet Protocol (TCP/IP)'dir. Diğer protokoller, HyperText Transfer Protocol (HTTP), Lightweight Directory Access Protocol (LDAP) veya Internet Messaging Access Protocol (IMAP) gibi, TCP/IP'nin "üstünde" çalışır; hepsi, web sayfaları görüntülemek veya e-posta sunucuları çalıştırmak gibi tipik uygulama görevlerini desteklemek için TCP/IP'yi kullanır. SSL protokolü iki alt protokol içerir: SSL kayıt protokolü ve SSL el sıkışma protokolü. SSL kayıt protokolü, veri iletmek için kullanılan formatı tanımlar. SSL el sıkışma protokolü, SSL etkinleştirilmiş bir sunucu ve SSL etkinleştirilmiş bir istemci arasında ilk kez bir SSL bağlantısı kurulduğunda bir dizi mesajın SSL kayıt protokolünü kullanarak değiş tokuşunu içerir. Bu mesajların değiş tokuşu, aşağıdaki işlemleri kolaylaştırmak için tasarlanmıştır:

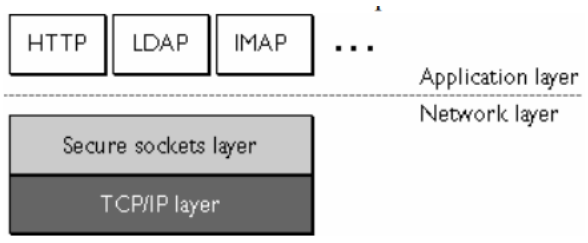


Figure 1. Location of Secure Socket Layer

Şekil 2.1.1 Socket Layer

• Sunucunun istemciye kimliğini doğrulaması.

• İstemci ve sunucunun desteklediği şifreleme algoritmalarını veya şifrelerini seçmelerine izin verir.

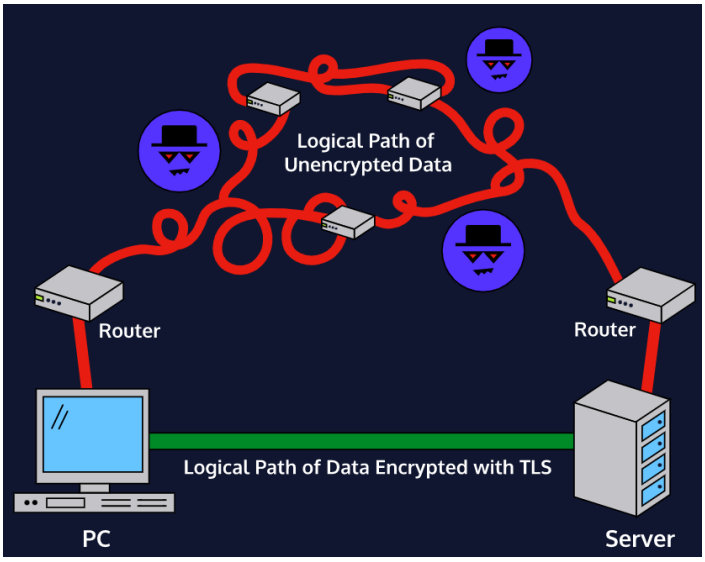
• İstemcinin sunucuya isteğe bağlı olarak kimliğini doğrulaması.

• Paylaşılan sırlar oluşturmak için açık anahtarlı şifreleme tekniklerini kullanır.

• Şifrelenmiş bir SSL bağlantısı kurar [1].

2.2. Transport Layer Security (TLS)

Transport Layer Security (TLS), bilgisayarlar arasında güvenli bağlantılar kurmak için kullanılan bir protokoldür. TLS'nin en önemli özelliği, webde güvenli bir şekilde gezinmemizi sağlayan HTTPS protokolünü kullanmasıdır.



Şekil 2.2.1 TLS

Adından da anlaşılacağı gibi, TLS, taşıma katmanı protokolleri aracılığıyla gönderilen verilerin güvenliğini sağlar. Bu, verilerin hedefine iletilmesi için güvenli bir bağlantı (sıklıkla bir tünel olarak kavramsallaştırılır) oluşturarak gerçekleştirilir. TLS, taşıma katmanı protokollerine bir kılıf olarak düşünebilirsiniz. TLS, şifreleme ve anahtar değişimi gibi şeyleri halletmek için diğer algoritmaları ve protokolleri kullanır. Ancak, TLS kendisi bir şifreleme algoritması değildir. TLS, sunucuların (ve bazen istemcilerin) kim olduklarını doğrulamak için genel anahtar sertifikalarını kullanır. Bu sertifikalar, asimetrik şifreleme yeteneğini kullanarak verileri dijital olarak imzalayarak, bunların gerçekliğini ve kaynağını doğrulayarak oluşturulur [3].

3. SSL ve TLS NASIL ÇALIŞIR?

3.1 SSL Nasıl Çalışır?

Bir istemci ve sunucu iletişim kurduğunda, SSL kimlik doğrulama, şifreleme ve bütünlük denetimleri sağlayarak bağlantının özel ve güvenli olmasını sağlar. Kimlik doğrulama, sunucunun ve isteğe bağlı olarak istemcinin, söyledikleri kişi olduklarını doğrular. Bir anahtar değişimi aracılığıyla gerçekleştirilen şifreleme, izinsiz herhangi bir sistemin verileri okumasını engelleyen güvenli bir "tünel" oluşturur. Bütünlük denetimleri ise şifrelenmiş akışın izinsiz bir sistem tarafından değiştirilmesini tespit eder.

SSL özellikli istemciler (örneğin, Mozilla™ veya Microsoft Internet Explorer™ web tarayıcıları) ve SSL, dijital sertifikaları kullanarak birbirlerinin kimliklerini doğrular. Dijital sertifikalar, güvenilir üçüncü taraf Sertifika Otoriteleri (CA'lar) tarafından verilir ve bir bireyin iddia ettiği kimlik hakkında bilgi sağlar, ayrıca halka açık anahtarlarını da içerir. Halka açık anahtarlar, açık anahtarlı şifreleme sistemlerinin bir bileşenidir. Bir mesajın göndericisi, verileri şifrelemek için bir halka açık anahtar kullanır. Mesajın alıcısı, sadece karşılık gelen özel anahtarla verileri deşifre edebilir. Halka açık anahtarlar herkes tarafından bilinirken, özel anahtarlar gizlidir ve sadece sertifikanın sahibi tarafından bilinir. Sertifikalardaki CA dijital imzasını doğrulayarak, her iki taraf da bir sahtekarın iletimi araya girmiş olup doğru özel anahtarına sahip sahte bir halka açık anahtar sağladığından emin olabilirler. SSL, hem açık anahtarlı hem de simetrik anahtarlı şifreleme kullanır. Simetrik anahtarlı şifreleme, açık anahtarlı şifrelemeye göre çok daha hızlıdır, ancak açık anahtarlı şifreleme daha iyi kimlik doğrulama teknikleri sağlar. Bu nedenle, SSL, kimlik doğrulaması için açık anahtarlı şifrelemeyi ve daha sonra toplu veri şifrelemesi için kullanılan simetrik anahtarları değiştirmek için de açık anahtarlı şifrelemeyi kullanır. SSL'nin oluşturduğu güvenli tünel, SSL özellikli bir istemci ve SSL özellikli bir sunucu arasında gönderilen tüm bilgilerin özel kalmasını sağlayan şifreli bir bağlantıdır. SSL, verilerin aktarımı sırasında herhangi birinin verileri değiştirdiğinde bunu algılayarak bağlantının güvenilirliğini sağlar. Bu, mesaj bütünlük kontrollerinin yardımıyla yapılır. Bu bütünlük kontrolleri bağlantının güvenilir olduğunu garanti eder. Aktarım sırasında herhangi bir noktada SSL, bağlantının güvenli olmadığını tespit ederse, bağlantıyı sonlandırır ve istemci ve sunucu yeni bir güvenli bağlantı kurarlar.

Sertifika tabanlı karşılıklı kimlik doğrulama kullanarak bir şifreli kanalın doğrulanması ve kurulması süreci açısından, SSL aşağıdaki adımları içerir:

1. Bir istemci, korunan bir kaynağa erişim isteği gönderir.

2. Sunucu, sertifikasını istemciye sunar.
3. İstemci, sunucunun sertifikasını doğrular.
4. Başarılı ise, istemci kendi sertifikasını sunucuya gönderir.
5. Sunucu, istemcinin sertifikasını doğrular.
6. Başarılı ise, sunucu istemcinin talep ettiği korunan kaynağa erişim izni verir [4].

3.1.1 Symmetric Encryption ve Asymmetric Encryption

Şifreleme genel olarak iki türe ayrılabilir: simetrik ve asimetrik (ayrıca açık anahtarlı şifreleme olarak da bilinir). Simetrik şifreleme, bir iletiyi şifreleme yeteneği otomatik olarak çözme yeteneği verir. Bu, aynı gizli anahtar kullanılarak şifreler ve çözer. Örneğin, antik bir simetrik şifreleme düzeni olan Sezar Şifresi vardır. Modern simetrik şifreleme düzenleri şunları içerir: Veri Şifreleme Standardı (DES) ve Gelişmiş Şifreleme Standardı (AES).

Asimetrik şifreleme, açık anahtar ve özel anahtar kullanarak şifreleme yapan bir şifreleme düzenidir. Açık anahtar şifreleme için kullanılırken, özel anahtar çözme için kullanılır. Simetrik şifreleme gibi, asimetrik şifreleme de otomatik olarak çözme yeteneği vermez. Modern asimetrik şifreleme düzenleri, büyük asal sayıların çarpanlarının bulunmasının zorluğuna dayanan RSA ve ayrık logaritma çözmenin zorluğuna dayanan Diffie-Hellman gibi yöntemleri içerir.

Asimetrik şifreleme ayrıca dijital imzalar oluşturmak için de kullanılabilir. Analog karşıtları gibi, dijital imzaların da bir belgeyi onaylayan tarafı doğrulama işlevi vardır. İmza oluşturma özel anahtar kullanılarak yapılırken, imzanın kimliğinin doğrulanması açık anahtar kullanılarak yapılır. SSL, web sunucularının kimliğini doğrulamak için bu dijital imzaları kullanır [5].

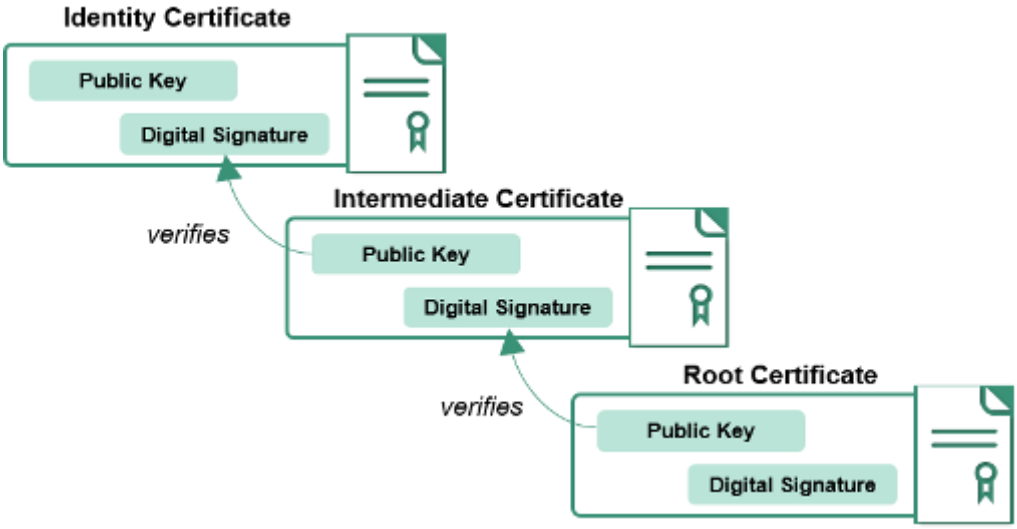
3.1.2 SSL kurulumunda sertifikalar nasıl çalışır?

SSL bağlantısı kurarken sertifikaların kullanılması, iki taraf arasındaki güveni sağlar ve birinin paylaştığı genel anahtarın gerçekten onlara ait olduğuna başka birisinin sahip olmadığına inanılır. Nasıl yapıldığına bakalım:

Web tarayıcısı sunucuyla iletişim kurmak istediğinde sunucu sertifikaları web tarayıcısına gönderir. Ardından istemci bu sertifikaların doğruluğunu doğrular ve sertifikalar doğrulanırsa, istemci SSL sertifikasındaki sunucunun genel anahtarının gerçekten sunucuya ait olduğunu ve sunucunun genel anahtarını kullanarak şifrelenmiş verileri sunucuya göndermeye başlar. İstemci tarafından sunucuya gönderilen tüm veriler, bu genel anahtar kullanılarak şifrelenir. Şimdi istemci, SSL el sıkışmasını tamamlamak için sunucunun genel anahtarını kullanarak sunucuya ön paylaşılan anahtar gönderir ve güvenli bir bağlantı başlatır. Tarayıcıda adres

çubuğunda bir kilit düğmesi gösterilir. Sunucu tarafından sertifikalar gönderilmediyse veya sertifikalar doğrulanmadıysa, tarayıcı güvenli bağlantıyı başlatmaz ve adres çubuğunda "Güvensiz" gösterilir [6].

3.1.3 Tarayıcı, Sertifikaların otantikliğini nasıl doğrular?



Şekil 3.1.3.1 Certificate

Sunucu tarafından gönderilen sertifikalar, Ara Sertifika Yetkilisi (CA) Sertifikası ve sunucunun SSL sertifikasıdır. SSL sertifikası, Ara CA'nın özel anahtarı ile dijital olarak imzalanır ve Ara CA'nın sertifikası, Kök Sertifika Yetkilisi (CA) tarafından kendi Özel Anahtarı kullanılarak imzalanır. Her sertifikada ayrıntılar bulunur. Örneğin, bir SSL sertifikası, alan adı, genel anahtar, kuruluş adı, ülke, bölge, dijital imza vb. gibi kuruluş bilgilerini içerir.

Benzer şekilde, ara sertifika yetkilisi sertifikası, kök sertifika imzası, bölge, ülke, veren, son kullanma tarihi, geçerlilik tarihinden başlayarak geçerli olan sertifika yetkilisinin adı vb. gibi bilgiler içerir. Benzer şekilde, Kök sertifikalar da benzer türde bilgiler içerir, örneğin son kullanma tarihi, geçerlilik tarihi, kuruluş adı, şifreleme algoritması ayrıntıları vb.

Sunucu, Kök sertifikasını web tarayıcısına göndermez çünkü Kök Sertifikaları zaten işletim sistemiyle birlikte sağlanır ve işletim sistemiyle birlikte gelir. Ve web sunucusu tarafından gönderilen sertifikaların doğruluğu kontrol edilen sertifika yetkilisi aracılığıyla doğrulanır.

Neden Ara Sertifikayı kullanıyoruz? SSL sertifikasını doğrulamak için Kök Sertifikasını doğrudan kullanamaz mıyız?

Ara sertifikayı doğrudan Kök Sertifikası yerine kullanıyoruz, bir katman daha güvenlik eklemek için. Genellikle, Kök Sertifika Yetkilileri offline olarak saklanır ve Özel Anahtarı,

kısıtlı erişimli birçok fiziksel güvenlik katmanı ile offline olarak tutulur, çünkü Kök Sertifika Yetkilisi'nin özel anahtarı tehlikeye düşerse, İnternet'in çoğu güvensiz hale gelir. Bu, düşünebileceğiniz en kötü güvenlik tehdididir. Çünkü bu özel anahtar, yetkisiz sertifikaları imzalamak için kullanılabilir ve ağın ortasındaki biri bu yetkisiz sertifikaları gönderebilir ve tarayıcı onları doğrular ve gerçekten güvenli bir bağlantı olduğunu düşünürken aslında öyle değildir. Eğer biri, ağın ortasında sertifikaları değiştirirse veya değiştirirse, tarayıcı tarafından doğrulanmaz çünkü Kök Sertifika genel anahtarı ara CA sertifikasının imzasını doğrulamaz ve doğrulanmadığı için güven zinciri bozulur ve SSL bağlantısı kurulamaz ve web tarayıcıları adres çubuğunda güvensiz bir bağlantı gösterir.SSL sertifikaları nasıl verilir?

SSL sertifikaları Ara CA tarafından verilir. Bir SSL sertifikası vermek için organizasyon belge doğrulaması, alan doğrulaması vb. Gereklidir.

Kullanılan Farklı Türlerde Sertifikalar:

1. Kök Sertifika Otoritesi sertifikası
2. Ara Sertifika Yetkilisi Sertifikası
3. SSL sertifikası

Kök Sertifika Otoritesi, en üst düzey Sertifika Otoritesidir. Kök sertifika, Ara CA sertifikasındaki dijital imzayı doğrulayarak Ara CA sertifikasını doğrular ve ara sertifika otoritesi, SSL sertifikasını doğrulayarak imzasını doğrular. Bu, güven zinciri olarak adlandırılır. Zincirdeki herhangi bir sertifika doğrulanmazsa, güven zinciri bozulur ve tarayıcı güvenli bir bağlantı sağlayamaz [6].

3.1.4 SSL Handshake

Client Hello: Sunucunun SSL kullanarak istemciyle iletişim kurmak için gereken bilgiler. Bu, SSL sürüm numarası, şifreleme ayarları, oturumla ilgili verileri içerir.

Server Hello: İstemciyle SSL kullanarak iletişim kurmak için gereken bilgiler. Bu, SSL sürüm numarası, şifreleme ayarları, oturumla ilgili verileri içerir.

Kimlik Doğrulama ve Ön Anahtar Gizliliği: İstemci, sunucu sertifikasını doğrular. (Örn. Ortak Adı / Tarih / İhraç Eden) İstemci (şifreleme yöntemine bağlı olarak) oturum için ön anahtar gizliliğini oluşturur, sunucunun genel anahtarıyla şifreler ve şifrelenmiş ön anahtar gizliliğini sunucuya gönderir.

Şifre Çözme ve Anahtar Gizliliği: Sunucu, ön anahtar gizliliğini çözmek için özel anahtarını kullanır. Her iki taraf da kabul edilen şifreleme yöntemiyle anahtar gizliliği oluşturmak için adımlar atar.

Oturum Anahtarıyla Şifreleme: Her iki taraf da gelecekteki mesajların şifreleneceğini bildirmek için mesajlar alışverişi yapar. SSL el sıkışmasından sonra, hem istemci hem de sunucu simetrik şifreleme kullanarak güvenli bir kanal üzerinden iletişim kurar.

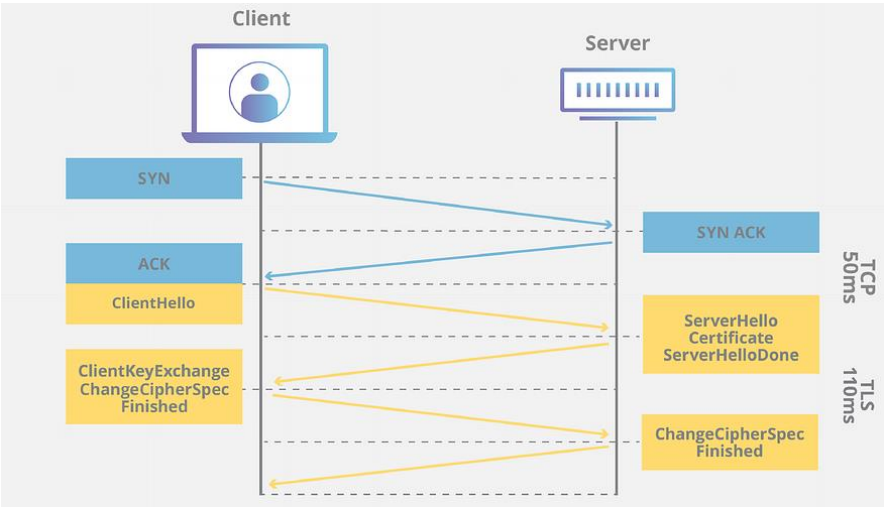
3.1.5 Web tarayıcıları ve sunucularının dışında SSL'i kimler kullanır?

SSL, HTTPS'in SSL destekli sürümü ile web tarayıcısı ve sunucu iletişimini güvence altına alma bağlamında ele aldık, ancak birçok diğer protokolde de kullanılır.

SSL, dosya transfer protokolünü (FTPS) ve e-posta protokolünü (SMTPS) güvence altına almak için kullanılabilir. Windows Güncelleme uygulaması, Windows yamalarının gerçekliğini sağlamak için SSL kullanır. VPN, SSL üzerine kurulabilir. SSL, MYSQL istemcileri ve sunucuları arasındaki bağlantıları güvence altına almak için kullanılabilir.

Ancak en genel anlamda, SSL, TCP'nin üzerine kurulmuştur, bu da TCP/IP tabanlı herhangi bir protokolün SSL'yi gerektiğinde içerebileceği anlamına gelir [6].

3.2 TLS Nasıl Çalışır?



Şekil 3.2.1 Tls Çalışma Mantığı

TLS, verilerin güvenli bir şekilde iletilmesi için simetrik ve asimetrik şifreleme yöntemlerinin bir kombinasyonunu kullanır. Simetrik şifrelemede, veri gönderen ve alıcının bildiği gizli bir anahtar ile şifrelenir ve çözümlenir. Genellikle 128 bit uzunluğunda olmasına rağmen, tercihen 256 bit uzunluğunda olmalıdır (80 bit'ten daha kısa olanlar artık güvensiz olarak kabul edilir).

Simetrik şifreleme hesaplama açısından verimlidir, ancak ortak bir gizli anahtarın kullanılması nedeniyle güvenli bir şekilde paylaşılması gerekmektedir.

Asimetrik şifreleme ise, bir çift anahtar kullanır - biri halka açık anahtar, diğeri ise özel anahtar. Halka açık anahtar, özel anahtarla matematiksel olarak ilgilidir, ancak yeterli anahtar uzunluğu verildiğinde, özel anahtarın halka açık anahtardan türetilmesi hesaplamalı olarak imkansızdır. Bu, göndericinin, alıcının halka açık anahtarını kullanarak göndermek istediği verileri şifrelemesine izin verir, ancak bu veriler sadece alıcının özel anahtarı ile çözülebilir.

Asimetrik şifrelemenin avantajı, şifreleme anahtarlarının paylaşım işleminin güvenli olmak zorunda olmamasıdır, ancak halka açık ve özel anahtarların matematiksel ilişkisi nedeniyle daha büyük anahtar boyutları gereklidir. Tavsiye edilen minimum anahtar uzunluğu 1024 bit olmasına rağmen, tercihen 2048 bit olmalıdır. Ancak, bu, eşdeğer güçlükteki simetrik anahtarların bin katına kadar daha fazla hesaplama gerektirir (örneğin, 2048 bit asimetrik bir anahtar, yaklaşık olarak 112 bit simetrik bir anahtara eşdeğerdir) ve asimetrik şifrelemenin birçok amaç için çok yavaş olmasına neden olur.

Bu nedenle, TLS, oturum anahtarının güvenli bir şekilde oluşturulup değiştirilmesi için asimetrik şifreleme kullanır. Oturum anahtarı, bir tarafın ilettiği verileri şifrelemek ve diğer tarafta alınan verileri deşifrelemek için kullanılır. Oturum bittiğinde, oturum anahtarı atılır.

RSA, Diffie-Hellman (DH), Ephemeral Diffie-Hellman (DHE), Elliptic Curve Diffie-Hellman (ECDH) ve Ephemeral Elliptic Curve Diffie-Hellman (ECDHE) gibi farklı anahtar oluşturma ve değiştirme yöntemleri kullanılabilir. DHE ve ECDHE ayrıca, bir özel anahtar elde edilirse bile bir oturum anahtarı tehlikeye düşmeyecek şekilde ileriye dönük gizlilik sunar, ancak zayıf rastgele sayı üretimi ve/veya sınırlı bir asal sayı aralığının kullanımı durumunda bile 1024-bit DH anahtarlarının kırılabilmesi olasılığı öne sürülmüştür. Ancak bunlar, uygulama sorunları olarak kabul edilebilir ve daha zayıf şifreleme süitleri için test yapmak için araçlar mevcuttur.

TLS ile bir sunucuya bağlanan bir istemcinin, sunucunun genel anahtarının sahipliğini doğrulayabilmesi istenir. Bu genellikle, genel anahtarın otantikliğini doğrulayan bir üçüncü taraf olan bir Sertifika Otoritesi (CA) tarafından verilen X.509 dijital bir sertifika kullanılarak gerçekleştirilir. Bazı durumlarda, bir sunucu, istemci tarafından açıkça güvenilir olarak belirtilmesi gereken bir kendinden imzalı sertifika kullanabilir (tarayıcılar, güvenilmeyen bir sertifika ile karşılaştıklarında bir uyarı mesajı görüntülemelidir), ancak bu, özel ağlarda ve/veya güvenli sertifika dağıtımı mümkün olduğunda kabul edilebilir olabilir. Ancak, genellikle, kamu tarafından güvenilir CA'lar tarafından verilen sertifikaların kullanılması önerilir [7].

3.2.1 CA Nedir?

Bir Sertifika Otoritesi (CA), Kamu Anahtar Altyapıları (PKI) için ITU-T'nin X.509 standardına uygun dijital sertifikaları veren bir kuruluştur. Dijital sertifikalar, sertifikanın sahibinin (konu olarak bilinen) halka açık anahtarını ve sahibin sertifika ile korunan alanı kontrol ettiğini doğrular. Bu nedenle, CA, güvenilir bir üçüncü taraf olarak hareket eder ve müşterilere (güvenen taraflar olarak bilinir) doğrulanmış bir varlık tarafından işletilen bir sunucuya bağlandıklarından emin olmaları için güvence sağlar.

Bitiş varlık sertifikaları, kök sertifikasından kaynaklanan bir güven zinciri aracılığıyla doğrulanır ve güven çıpası olarak da bilinen bir kök sertifikasından kaynaklanır. Asimetrik şifreleme ile, kök sertifikasının özel anahtarını kullanarak diğer sertifikaları imzalamak mümkündür, böylece kök sertifikasının halka açık anahtarını kullanarak doğrulama yapılabilir ve bu nedenle sertifikanın verildiği CA'nın güvenini miras alabilirler. Uygulamada, bitiş varlık sertifikaları genellikle bir veya daha fazla ara sertifika (bazen alt CA veya alt kuruluş olarak da bilinir) tarafından imzalanır çünkü bu, bir bitiş varlık sertifikası yanlış verildiğinde veya tehlikeye girdiğinde kök sertifikayı korur.

Kök sertifikası güveni genellikle kök sertifikalarının işletim sistemlerinde veya tarayıcılarda fiziksel dağıtımı yoluyla kurulur. Ana sertifika programları, Microsoft (Windows ve Windows Phone), Apple (OSX ve iOS) ve Mozilla (Firefox ve Linux) tarafından yürütülür ve CA'ların sıkı teknik gereksinimlere uymalarını ve WebTrust, ETSI EN 319 411-3 (eski adıyla TS 102 042) veya ISO 21188: 2006 denetimini tamamlamalarını gerektirirler. WebTrust, Amerikan Sertifikalı Kamu Muhasebecileri Enstitüsü ve Kanadalı Şirket Muhasebecileri Enstitüsü tarafından geliştirilen bir programdır, ETSI Avrupa Telekomünikasyon Standartları Enstitüsü'dür, ISO ise Uluslararası Standartlar Örgütü'dür.

Büyük işletim sistemleri ve tarayıcılarla birlikte dağıtılan kök sertifikaları genellikle halka açık veya küresel olarak güvenilir olarak adlandırılır ve teknik ve denetim gereksinimleri, ihraç eden CA'ların çok uluslu şirketler veya hükümetler olmaları anlamına gelir. Şu anda yaklaşık 50 halka açık olarak güvenilen CA bulunmaktadır, ancak çoğu / tümü birden fazla kök sertifikaya sahiptir ve çoğu sertifika verme ve yönetme endüstri yönergeleri geliştiren CA / Tarayıcı Forumu üyesidir.

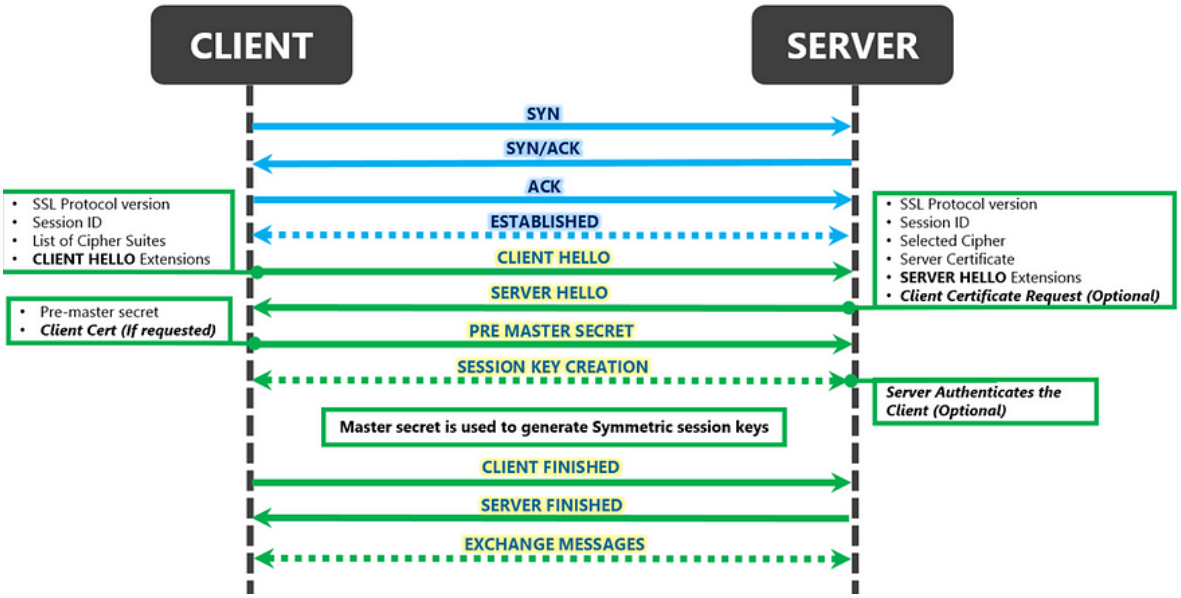
Ancak, özel CA'lar kurmak ve kök sertifikalarını istemci sistemlerinde güvenli dağıtım ve kurulum yoluyla sağlamak da mümkündür. Örnekler, IP adreslerini ve AS numaralarını tuttuklarına dair bildirimde bulunan Yerel İnternet Kayıt Defterlerine sertifika veren Bölgesel

İnternet Kayıt Defterleri (AfriNIC, APNIC, ARIN, LACNIC ve RIPE NCC) tarafından işletilen RPKI CA'larıdır; dağıtılmış bilimsel hesaplama için kullanılan sunucu ve istemci sertifikaları vermek için güvenlik çerçevesi sağlayan Uluslararası Izgara Güveni Federasyonu (IGTF) gibi. Bu durumlarda, kök sertifikaları, halka açık olarak güvenilir CA tarafından verilen bir sertifika kullanan sitelerden güvenli bir şekilde indirilip yüklenir.

X.509 PKI sisteminin bir zayıf noktası, üçüncü tarafların (CA'lar) talep eden kuruluşun gerçekten sahibi veya kontrol ettiği herhangi bir etki alanı için sertifikalar çıkarabilmesidir. Genellikle alan doğrulama yoluyla doğrulama yapılır, yani etki alanının yönetimsel olarak sorumlu bir adresine kimlik doğrulama bağlantısı içeren bir e-posta gönderilir. Bu genellikle 'hostmaster@domain' gibi standart iletişim adreslerinden biri veya WHOIS veritabanında listelenen teknik temas adreslerinden biridir, ancak DNS veya BGP protokollerindeki adam-in-the-orta saldırılarına veya daha basitçe, ayrılmamış alanlarda yönetim adresleri kaydeden kullanıcılara açık kalır. Belki daha da önemlisi, Alan Doğrulama (DV) sertifikaları, bir alanın herhangi bir yasal varlıkla bir ilişkisi olduğunu iddia etmez, hatta bir etki alanının öyle görünse bile. Bu nedenle, CA'lar artık Kuruluş Doğrulanmış (OV) ve Genişletilmiş Doğrulama (EV) sertifikalarının kullanımını teşvik etmektedir. OV sertifikaları ile, talep eden kuruluş kamu veritabanları kullanılarak organizasyon adı, adres ve telefon numarasının onaylanmasına ek olarak ek kontrollerden geçirilir. EV sertifikaları ile, yasal kuruluşun, fiziksel konumun ve talep eden kuruluş adına hareket ettiğini iddia eden kişilerin kimlikleri konusunda ek kontroller yapılır. Geçerli bir EV sertifikasıyla karşılaşıldığında tarayıcılar doğrulanmış kuruluş adını yeşil olarak görüntüler, ancak maalesef bir OV sertifikasını bir DV sertifikasından ayırt etmenin kolay bir yolu yoktur.

Tabii ki, bu hala CA'ların yanlışlıkla veya dolandırıcılıkla yanlış sertifikalar çıkarmasını engellemez ve CA'ların sahte sertifikalar çıkarmaya kandırıldığı güvenlik ihlalleri vakaları da mevcuttur. Birkaç yüksek profilli olayın ardından güvenlik prosedürlerinin önemli ölçüde sıkılaştırılmasına rağmen, sistem üçüncü taraf güvenine bağımlı kalmaya devam etmektedir. Bu, RFC 6698, 7671, 7672 ve 7673'te belirtildiği gibi DNS tabanlı Adlandırılmış Varlıkların Doğrulanması (DANE) protokolünün geliştirilmesine yol açmıştır [8].

3.2.2 TLS handshake?



Şekil 3.2.2.1 TLS Handshake

TLS el sıkışmaları, bir istemci ve bir sunucu arasında değiştirilen bir dizi veri gramı veya mesajlardır. TLS el sıkışması, el sıkışmasını tamamlamak ve ileriye dönük konuşmaya izin vermek için gereken bilgilerin istemci ve sunucu arasında değiştirilmesini içeren birden fazla adımdan oluşur.

TLS el sıkışması içindeki adımlar, kullanılan anahtar değişim algoritmasına ve her iki tarafın desteklediği şifreleme kümesine bağlı olarak değişebilir. En sık kullanılan anahtar değişim algoritması RSA'dır ve aşağıdaki şekilde gerçekleşir:

İstemci merhaba mesajı: İstemci, sunucuya "merhaba" mesajı göndererek el sıkışmasını başlatır. Mesaj, istemcinin desteklediği TLS sürümünü, şifreleme kümesini ve "istemci rasgele" olarak bilinen bir dizi rasgele baytı içerir.

Sunucu merhaba mesajı: Sunucu, istemci merhaba mesajına yanıt olarak, sunucunun SSL sertifikasını, sunucunun seçtiği şifreleme kümesini ve sunucu rasgelesini içeren bir mesaj gönderir. Sunucu rasgelesi, sunucu tarafından oluşturulan başka bir rasgele bayt dizisidir.

Kimlik doğrulama: İstemci, sunucunun SSL sertifikasını, sertifikayı veren sertifika otoritesiyle doğrular. Bu, sunucunun iddia ettiği kişi olduğunu ve istemcinin etkileşimde bulunduğu alan adının gerçek sahibiyle etkileşimde olduğunu doğrular.

Ön anahtar sırrı: İstemci bir başka rasgele bayt dizisi olan "ön anahtar sırrısını" gönderir. Ön anahtar sırrısı, genel anahtarla şifrelenir ve yalnızca sunucu tarafından özel anahtarla şifre çözülebilir. (İstemci genel anahtarı, sunucunun SSL sertifikasından alır.)

Kullanılan özel anahtar: Sunucu, ön anahtar sırrısını şifre çözer.

Oturum anahtarları oluşturuldu: İstemci ve sunucu, istemci rasgelesinden, sunucu rasgelesinden ve ön anahtar sırrısından oturum anahtarları oluştururlar. Aynı sonuçlara ulaşmaları gerekir. Müşteri hazır: RSA el sıkışmasıyla aynıdır.

Sunucu hazır: Güvenli simetrik şifreleme sağlandı.

DH parametresi: DH, Diffie-Hellman için kısaltılmıştır. Diffie-Hellman algoritması, aynı ön anahtarı bulmak için üstel hesaplamalar kullanır. Sunucu ve istemci, hesaplama için bir parametre sağlar ve birleştirildiğinde her tarafta farklı bir hesaplama sonucu elde edilir ve sonuçlar eşittir [8].

3.4 SSL/TLS'nin tüm sürümleri nelerdir?

SSL 1.0	Unpublished	Unpublished
SSL 2.0	1995	Deprecated in 2011 (RFC 6176)
SSL 3.0	1996	Deprecated in 2015 (RFC 7568)
TLS 1.0	1999	Deprecated in 2021 (RFC 8996)
TLS 1.1	2006	Deprecated in 2021 (RFC 8996)
TLS 1.2	2008	In use since 2008
TLS 1.3	2018	In use since 2018 [9]

3.5 Temel Farkları

SSL ve TLS arasındaki ana farklılıklar şunlardır:

Protokol geliştirme: SSL, Netscape Communications Corporation tarafından 1990'ların ortalarında geliştirildi. İlk halka açık sürümü SSL 2.0'dı, ardından SSL 3.0 izledi. TLS ise SSL ile ilgili güvenlik endişelerini ele almak için standartlaştırılmış bir protokol olarak Internet Engineering Task Force (IETF) tarafından geliştirildi. TLS'nin ilk sürümü olan TLS 1.0, SSL 3.0'ın bir yükseltmesi olarak 1999'da piyasaya sürüldü.

Güvenlik iyileştirmeleri: TLS, SSL'de bulunan güvenlik açıklarını ele almak için geliştirildi. Zaman içinde, TLS birçok revizyondan geçti ve her biri yeni güvenlik özellikleri ve iyileştirmeler sundu. Özellikle TLS 1.2 ve 1.3, SSL 3.0'a kıyasla daha güçlü şifreleme algoritmaları, daha iyi karma işlevleri ve gelişmiş güvenlik mekanizmaları sağlar.

Kriptografik algoritmalar: TLS, SSL'ye kıyasla daha geniş bir kriptografik algoritma yelpazesi destekler. Örneğin, TLS 1.2, Advanced Encryption Standard (AES) ve authenticated encryption with associated data (AEAD) şifre paketleri desteğini tanıttı. TLS 1.3, desteklenen algoritmaların listesini daha da basitleştirdi ve zayıf veya güvensiz olarak kabul edilenleri kaldırdı.

El sıkışma işlemi: Hem SSL hem de TLS, istemci ve sunucu arasında güvenli bir bağlantı kurmak için bir el sıkışma işlemi kullanır. Ancak TLS, güvenliği artırmak için ek kontroller ve mekanizmalar tanıtarak SSL'nin el sıkışma işlemine iyileştirmeler getirdi. TLS 1.3, el sıkışma işlemini daha da basitleştirdi ve gerekli tur sayısını azaltarak daha hızlı ve daha güvenli bağlantılar sağladı.

İleri gizlilik: Modern TLS sürümleri, Diffie-Hellman veya eliptik eğri Diffie-Hellman gibi geçici anahtar değişim algoritmalarını kullanarak ileri gizlilik sağlayabilir. Bu, bir saldırganın bir sunucunun özel anahtarını ele geçirse bile, geçmiş iletişim oturumlarını şifreleyemez anlamına gelir. SSL 3.0, ileri gizliliği varsayılan olarak desteklemedi.

Terminoloji: SSL ve TLS terimleri bazen birbirinin yerine kullanılsa da, aslında protokolün farklı versiyonlarını ifade ederler. Teknik olarak, SSL, Netscape tarafından geliştirilen eski versiyonları (SSL 2.0 ve SSL 3.0) ifade ederken, TLS, IETF tarafından geliştirilen versiyonları (TLS 1.0 ve sonrası) ifade eder [10].

3.6 Güvenlik Zafiyetleri ve Tehditleri

Bazı eski şifreleme özellikleri, zayıflıklara neden oldu veya belirli türde siber saldırılara olanak tanıdı. İşte TLS 1.2 şifreleme zayıflıklarının ve bunlarla ilişkili zayıflıkların veya saldırıların non-eksiksiz bir listesi:

- RSA anahtar taşıma: Gelecekteki gizlilik sağlamaz
- CBC modu şifreleri: BEAST ve Lucky 13 saldırıları
- RC4 akım şifresi: HTTPS kullanımı için güvenli değil
- Rastgele Diffie-Hellman grupları: CVE-2016-0701
- İhracat şifreleri: FREAK ve LogJam saldırıları

Yukarıda listelenenlere ek olarak, birçok TLS 1.2 özelliği kaldırılmıştır. Amacı, TLS 1.2'nin zayıf yönlerini etkinleştirmek için kimseye imkân vermemektir. Bu, hükümetin emniyet kemerleri olmadan yeni araç üretmeyi yasa dışı kıldığı duruma benzer: Düzenlemelerin amacı, emniyet kemeri olmayan arabaların aşamalı olarak kaldırılmasını sağlamak ve herkesin daha güvende olmasını sağlamaktı. Bir süre daha eski araç modellerini kullanarak daha az güvende olma seçeneğine sahip olan sürücüler olsa da, daha tehlikeli arabalar sonunda yollardan kayboldu [11].

4. SSL VE TLS'NİN ALTERNATİFLERİ VE REKABETÇİ TEKNOLOJİLERİ

SSL ve TLS gibi güvenli bağlantı protokolleri, internet üzerinden yapılan iletişimlerde yaygın bir şekilde kullanılmaktadır. Ancak, alternatif ve rekabetçi teknolojiler de mevcuttur. İşte bazı örnekler:

IPSec: Internet Protocol Security (IPSec), ağ katmanında çalışan bir güvenlik protokolüdür. Verilerin güvenli bir şekilde iletilmesini sağlar ve SSL/TLS gibi uygulama katmanı protokollerine göre daha düşük bir işlem yüküne sahiptir. IPSec, özellikle sanal özel ağlar (VPN) gibi ağ bazlı güvenlik senaryolarında kullanılır.

SSH: Secure Shell (SSH), ağ protokolüdür ve özellikle uzaktan erişim senaryolarında kullanılır. SSH, kullanıcıların uzak bir bilgisayara güvenli bir şekilde erişmesine izin verir. SSH, TLS'ye benzer şekilde, çift yönlü kimlik doğrulama ve şifreleme sağlar.

SFTP: Secure File Transfer Protocol (SFTP), dosya transferi senaryolarında kullanılan bir protokoldür. SSH üzerinden çalışır ve güvenli dosya transferi sağlar. SFTP, FTPS (FTP over SSL/TLS) gibi diğer dosya transfer protokollerine göre daha güvenlidir.

HTTPS: Hypertext Transfer Protocol Secure (HTTPS), web siteleri için kullanılan bir protokoldür. HTTPS, SSL/TLS üzerinden çalışır ve web sayfalarının güvenli bir şekilde iletilmesini sağlar. HTTPS, kullanıcıların verilerinin güvenliği için önemlidir ve birçok web tarayıcısı tarafından şifrelenmemiş bağlantılarda uyarı verilir.

5.TEST VE DOĞRULAMA ARAÇLARI

SSL Server Test (https://www.ssllabs.com/ssltest/): Qualys tarafından sunulan ücretsiz bir web uygulamasıdır. SSL sertifikaları ve sunucu yapılandırmalarını test eder, güncel güvenlik standartlarına uygun olup olmadığını değerlendirir.

SSL Checker (https://www.sslshopper.com/ssl-checker.html): Ücretsiz bir araçtır, SSL sertifikalarının doğruluğunu, doğru şekilde kurulup kurulmadığını ve sunucu yapılandırmasının uygunluğunu kontrol eder.

OpenSSL (https://www.openssl.org/): OpenSSL, açık kaynak kodlu bir kriptografi kütüphanesidir. SSL/TLS protokollerini uygulamak için kullanılabilir. Ayrıca, sertifika otoriteleri ve anahtarları yönetmek için kullanılabilir.

TestSSLServer (https://github.com/drwetter/testssl.sh): Linux ve Unix sistemler için açık kaynak kodlu bir araçtır. SSL/TLS bağlantılarını tarar, protokollerin uygunluğunu ve güvenliği kontrol eder.

SSL Diagnostics Tool (https://www.microsoft.com/en-us/download/details.aspx?id=46899): Microsoft tarafından sunulan ücretsiz bir araçtır. SSL/TLS bağlantıları, sertifikalar ve sunucu yapılandırması hakkında bilgi sağlar.

SSL/TLS Capabilities of Your Browser (https://www.howsmyssl.com/): Bu web sitesi, kullandığınız web tarayıcısının SSL/TLS desteği hakkında bilgi verir.

SSLScan (https://github.com/rbsec/sslscan): Açık kaynak kodlu bir araçtır ve sunucuların SSL/TLS yapılandırmalarını taramak için kullanılabilir.

SSL/TLS Secure Renegotiation Test (https://github.com/syncsynchalt/ssltest): Bu araç, sunucuların güvenli yeniden müzakereye uygunluğunu test etmek için kullanılabilir.

Bu araçlar, SSL ve TLS'nin doğru şekilde yapılandırıldığından emin olmak için kullanılabilir. Ancak, bu listeye ek araçlar da bulunabilir.

6. SSL VE TLS İLE İLGİLİ DÜNYA GENELİNDEKİ YASAL VE UYGULAMA FARKLILIKLARI

SSL ve TLS protokolleri, dünya genelinde birçok ülkenin yasal düzenlemeleri ve uygulama politikaları ile farklılık göstermektedir. Bu farklılıklar, şifreleme yöntemleri, sertifika otoriteleri, anahtar yönetimi ve diğer konuları kapsamaktadır. Bazı önemli farklılıklar şunlardır:

Şifreleme Yöntemleri: Bazı ülkeler, belirli şifreleme algoritmalarını yasaklamakta veya kısıtlamaktadır. Örneğin, Rusya ve Çin gibi ülkeler, belirli şifreleme algoritmalarının kullanımını kısıtlamakta veya yasaklamaktadır.

Sertifika Otoriteleri: Farklı ülkelerde, farklı sertifika otoriteleri bulunmaktadır. Bazı ülkeler, kendi sertifika otoritelerini kullanmayı zorunlu kılmaktadır. Örneğin, Çin'de web siteleri, Çin'in devlet destekli sertifika otoritesi olan CNNIC tarafından onaylanmalıdır.

Anahtar Yönetimi: Bazı ülkeler, anahtar yönetimini sıkı bir şekilde düzenlemektedir. Örneğin, Almanya'da, anahtar yönetimi, Federal Güvenlik Ofisi (BSI) tarafından belirlenen standartlara uymalıdır.

Yasal Düzenlemeler: Farklı ülkelerde, SSL ve TLS protokolleri ile ilgili yasal düzenlemeler farklılık göstermektedir. Bazı ülkeler, özel anahtarların yurt dışına çıkarılmasını yasaklamaktadır. Diğer ülkeler, şifreli iletişim trafiğinin izlenmesine veya şifreli iletişim araçlarının kullanımının kısıtlanmasına izin vermektedir.

7.SONUÇ

Sonuç olarak, SSL ve TLS, internet üzerindeki güvenli iletişim için kritik öneme sahip olan iki protokoldür. SSL, 1990'larda Netscape tarafından geliştirilmiştir ve TLS, SSL'nin güvenlik açıklarını gidermek için IETF tarafından standardize edilmiştir. TLS, güçlü şifreleme algoritmaları, daha iyi karma işlevleri, gelişmiş güvenlik mekanizmaları ve ileri güvenlik özellikleri sağlar. SSL ve TLS, çeşitli sınav ve doğrulama araçları ile test edilebilir ve doğrulanabilir. Bununla birlikte, TLS 1.2 ve önceki sürümlerinde bazı güvenlik açıkları vardı. Bu nedenle, TLS 1.3, bazı özellikleri çıkararak daha güvenli bir protokol haline getirdi.

SSL ve TLS, dünya çapındaki farklı yasal ve uygulama farklılıklarından etkilenir. Bazı ülkeler, yasal olarak belirli şifreleme algoritmalarının kullanımını kısıtlayabilir veya yasaklayabilir. Bazı ülkeler, sertifikaların doğrulanmasına ilişkin standartları farklı uygulayabilir. Farklı ülkelerdeki internet servis sağlayıcıları ve web siteleri, farklı protokoller, algoritmalar ve güvenlik ayarları kullanabilir. Bu nedenle, farklı ülkelerdeki kullanıcılar, farklı düzeylerde güvenlik ve gizlilik seviyeleriyle karşılaşabilirler.

SSL ve TLS, internet üzerindeki güvenli iletişim için önemlidir ve birçok kullanıcı, web sitelerinde SSL/TLS kullanılmasını bekler. Web sitelerinin SSL/TLS kullanarak verileri şifrelemesi ve doğrulaması, kullanıcıların bilgilerini kötü amaçlı kullanımlardan korur. SSL/TLS protokollerinin güvenli bir şekilde kullanılması, internet üzerindeki güvenliği artırır ve kullanıcıların bilgilerinin gizliliğini korur [13].

8.KAYNAKÇA

1. Ma, J., Luo, X., Liu, Y., & Jiang, Z. (2008). A Secure Data Sharing Scheme in the Web of Data. International Conference on Convergence and Hybrid Information Technology, 318-325. doi: 10.1109/ICHIT.2008.289
2. National Institute of Standards and Technology (NIST). (2014). Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations, NIST Special Publication 800-52 Revision 1. Retrieved from https://cdn.atraining.ru/docs/NIST.SP.800-52r1.pdf
3. Codecademy. “Transport Layer Security (TLS).” Codecademy, 15 Aug. 2019, www.codecademy.com/articles/what-is-tls.
4. Kumar, R., Singh, Y., & Agrawal, R. (2013). A Comparative Study of Symmetric Key Cryptography Algorithms. International Journal of Computer Science and Applications, 6(3), 243-250. Retrieved from https://www.researchpublications.org/IJCSA/NCAICN-13/245.pdf
5. "What is SSL and How Does It Work?" by user3141592, Medium, 2019. Link: https://user3141592.medium.com/what-is-ssl-and-how-does-it-work-a5465d19b494
6. Gupta, N. (2021, January 21). How SSL Works. Medium. https://namangupta01.medium.com/how-ssl-works-23d8e5ed0cfa
7. Hostinger. (2021, March 30). What is TLS? Retrieved from https://www.hostinger.com/tutorials/what-is-tls
8. Karakaya, B. (2020). Deep Dive into TLS. DevOps Dudes. https://medium.com/devops-dudes/deep-dive-into-tls-a9798ac1763a
9. https://en.wikipedia.org/wiki/Transport_Layer_Security
10. Cloudflare. (2022, February 1). Why use TLS 1.3? Cloudflare. https://www.cloudflare.com/learning/ssl/why-use-tls-1.3/
11. Jiang, R. (2019). A Comprehensive Guide: SSL and TLS Protocols, Key Features, and Differences. Retrieved from https://medium.com/@ramseyjiang_22278/a-comprehensive-guide-ssl-and-tls-protocols-key-features-and-differences-6558d9629b93.

12. https://www.cloudflare.com/learning/ssl/alternatives-to-ssl/

13. "SSL and TLS: A Beginner's Guide to Secure Communication", Cloudflare, https://www.cloudflare.com/learning/ssl/ssl-tls-legal-considerations/