Protocols and Servers 2

Task 1 Introduction (Görev 1 Giriş)

Protokoller ve Sunucular odası birçok protokolü kapsamaktadır:

- Telnet
- HTTP
- FTP
- SMTP
- POP3
- IMAP

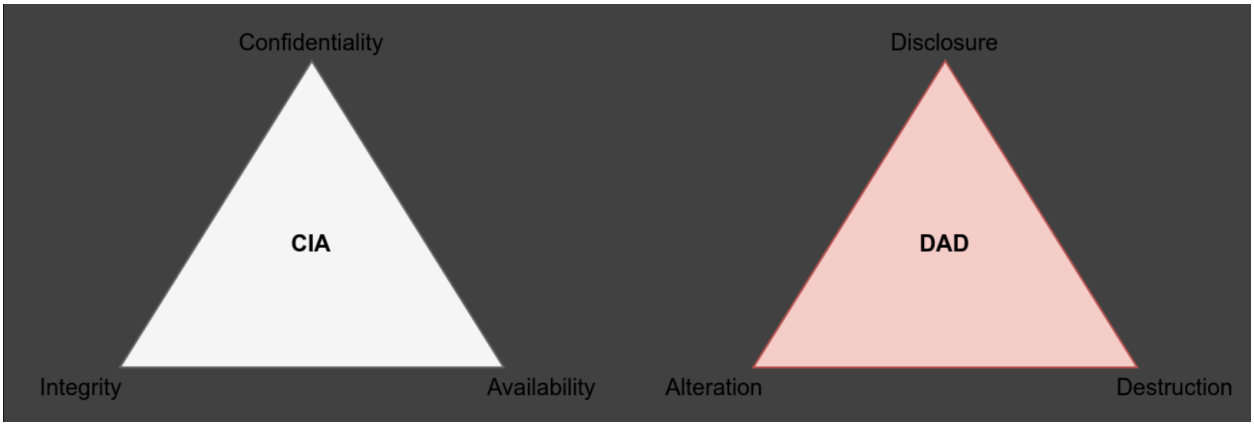
Bu protokolleri uygulayan sunucular farklı saldırı türlerine maruz kalmaktadır. Birkaç isim vermek gerekirse, düşünün:

1. **Sniffing Attack (Network Packet Capture)** (Sniffing Saldırısı (Ağ Paket Yakalama))
2. **Man-in-the-Middle (MITM) Attack** (Man-in-the-Middle (MITM) Saldırısı)
3. **Password Attack (Authentication Attack)** (Şifre Saldırısı (Kimlik Doğrulama Saldırısı))
4. **Vulnerabilities** (Güvenlik Açıkları)

Güvenlik perspektifinden bakıldığında, her zaman neyi korumayı amaçladığımızı düşünmemiz gerekir; güvenlik üçlüsünü göz önünde bulundurun: Gizlilik, Bütünlük ve Kullanılabilirlik (CIA). Gizlilik, iletişimin içeriğinin amaçlanan taraflar için erişilebilir olmasını ifade eder. Bütünlük, gönderilen herhangi bir verinin hedefine ulaştığında doğru, tutarlı ve eksiksiz olmasını sağlama fikridir. Son olarak, kullanılabilirlik, ihtiyaç duyduğumuzda hizmete erişebilmeyi ifade eder. Farklı taraflar bu üç konuya farklı vurgular yapacaktır. Örneğin, bir istihbarat teşkilatı için gizlilik en yüksek öncelik olacaktır. Çevrimiçi bankacılık, işlemlerin bütünlüğüne en

fazla önem verecektir. Reklam sunarak para kazanan herhangi bir platform için kullanılabilirlik en yüksek öneme sahiptir.

Gizlilik, Bütünlük ve Kullanılabilirliği (CIA) koruduğumuzu bilerek, bir saldırı İfşa, Değişiklik ve Tahribata (DAD) neden olmayı hedefler. Aşağıdaki şekiller bunu yansıtmaktadır.



Bu saldırılar sistemin güvenliğini doğrudan etkiler. Örneğin, ağ paketlerinin ele geçirilmesi gizliliği ihlal eder ve bilgilerin ifşa edilmesine yol açar. Başarılı bir şifre saldırısı da ifşaya yol açabilir. Öte yandan, bir Ortadaki Adam (MITM) saldırısı, iletilen verileri değiştirebileceğinden sistemin bütünlüğünü bozar. Bu saldırılar protokol tasarımı ve sunucu uygulamasının ayrılmaz bir parçası olduğu için bu odada bu üç saldırıya odaklanacağız.

Güvenlik açıkları daha geniş bir spektruma sahiptir ve istismar edilen güvenlik açıklarının hedef sistemler üzerinde farklı etkileri vardır. Örneğin, bir Hizmet Reddi (DoS) güvenlik açığından faydalanmak sistemin kullanılabilirliğini etkileyebilirken, bir Uzaktan Kod Yürütme (RCE) güvenlik açığından faydalanmak daha ciddi zararlara yol açabilir. Bir güvenlik açığının kendi başına bir risk oluşturduğuna dikkat etmek önemlidir; hasar yalnızca güvenlik açığı istismar edildiğinde ortaya çıkabilir. Kendi modülleri olan Güvenlik Açığı Araştırması'na sahip oldukları için bu odada güvenlik açıklarını ele almıyoruz.

Bu oda, bir protokolün ifşa ve değiştirmeye karşı, yani iletilen verilerin gizliliğini ve bütünlüğünü korumak için nasıl yükseltilebileceğine veya değiştirilebileceğine odaklanacaktır. Ek konuları kapsayan başka modüller de önereceğiz.

Ayrıca, zayıf şifreleri bulmak için Hydra'yı tanıtıyoruz.

Soru ⇒ Aşağıdaki görevleri yerine getirmeye devam ederken AttackBox'ı ve sanal makineyi başlatmanızı öneririz. Daha iyi pratik ve öğrenme deneyimi için Telnet veya Netcat üzerinden farklı hizmetlere bağlanabilirsiniz.

Cevap ⇒ Cevap Gerekmemektedir.

Task 2 Sniffing Attack (Görev 2 Koklama Saldırısı)

Sniffing saldırısı, hedef hakkında bilgi toplamak için bir ağ paketi yakalama aracının kullanılması anlamına gelir. Bir protokol açık metin olarak iletişim kurduğunda, değiş tokuş edilen veriler analiz etmek için üçüncü bir tarafça yakalanabilir. Basit bir ağ paketi yakalama işlemi, veriler aktarım sırasında şifrelenmemişse, özel mesajların içeriği ve oturum açma kimlik bilgileri gibi bilgileri ortaya çıkarabilir.

Kullanıcının uygun izinlere sahip olması koşuluyla (Linux'ta root izinleri ve MS Windows'ta yönetici ayrıcalıkları) bir Ethernet (802.3) ağ kartı kullanılarak bir koklama saldırısı gerçekleştirilebilir. Ağ paketlerini yakalamak için birçok program mevcuttur. Aşağıdakileri dikkate alıyoruz:

1. Tcpdump, birçok işletim sisteminde çalışmak üzere taşınmış ücretsiz bir açık kaynak komut satırı arayüzü (CLI) programıdır.
2. Wireshark, Linux, macOS ve MS Windows dahil olmak üzere çeşitli işletim sistemleri için kullanılabilen ücretsiz bir açık kaynaklı grafik kullanıcı arayüzü (GUI) programıdır.
3. Tshark, Wireshark'a bir CLI alternatifidir.

Parolaları ve hatta mesajların tamamını yakalamak için birkaç özel araç vardır; ancak, bu yine de biraz ek çaba ile Tcpdump ve Wireshark ile elde edilebilir.

POP3 kullanarak e-posta mesajlarını kontrol eden bir kullanıcı düşünün. İlk olarak, kullanıcı adı ve şifreyi yakalamak için Tcpdump kullanacağız. Aşağıdaki terminal çıktısında sudo tcpdump port 110 -A komutunu kullandık. Bu komutu açıklamadan önce, bu saldırının ağ trafiğine erişim gerektirdiğini belirtmeliyiz, örneğin bir telefon dinleme veya port yansıtma özelliğine sahip bir anahtar aracılığıyla. Alternatif olarak, başarılı bir Man-in-the-Middle (MITM) saldırısı başlatırsak değiş tokuş edilen trafiğe erişebiliriz.

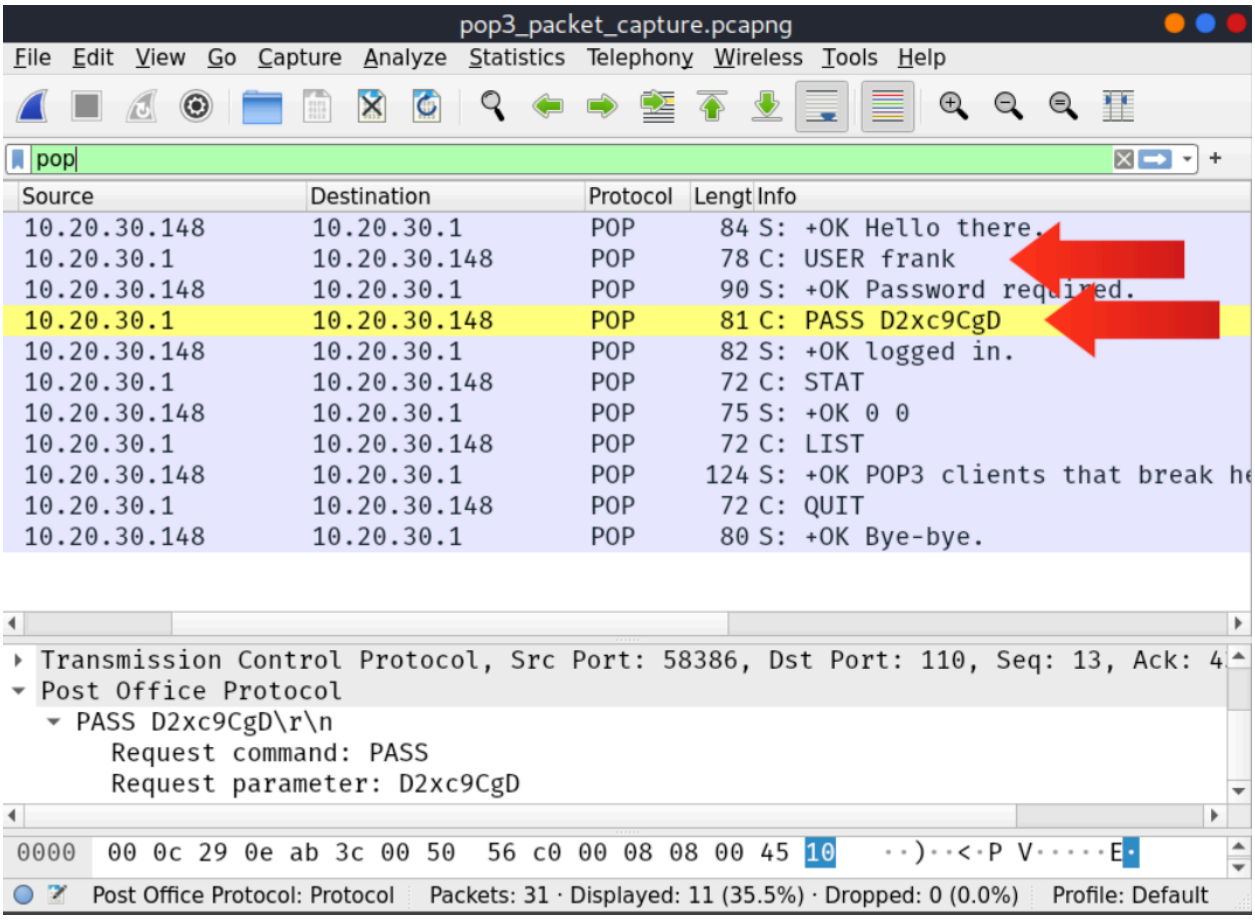
Paket yakalama root ayrıcalıkları gerektirdiği için sudo'ya ihtiyacımız var. Yakalanan ve görüntülenen paketlerin sayısını POP3 sunucusuyla değiş tokuş edilenlerle sınırlamak istedik. POP3'ün 110 numaralı portu kullandığını biliyoruz, bu yüzden paketlerimizi 110 numaralı portu kullanarak filtreledik. Son olarak, yakalanan paketlerin içeriğini ASCII formatında görüntülemek istedik, bu yüzden -A ekledik.

```
pentester@TryHackMe$ sudo tcpdump port 110 -A[...]
09:05:15.132861 IP 10.20.30.1.58386 > 10.20.30.148.pop3: Flags [P.], seq 1:13,
ack 19, win 502, options [nop,nop,TS val 423360697 ecr 3958275530], length
12
E..@.V@.@.g.
...
......n......".....
.;....}.USER frank

09:05:15.133465 IP 10.20.30.148.pop3 > 10.20.30.1.58386: Flags [.], ack 13, wi
n 510, options [nop,nop,TS val 3958280553 ecr 423360697], length 0
E..4..@.@.O~
...
....n...."........?P.....
...i.;..
09:05:15.133610 IP 10.20.30.148.pop3 > 10.20.30.1.58386: Flags [P.], seq 19:4
3, ack 13, win 510, options [nop,nop,TS val 3958280553 ecr 423360697], len
gth 24
E..L..@.@.Oe
...
....n...."........←.....
...i.;..+OK Password required.

09:05:15.133660 IP 10.20.30.1.58386 > 10.20.30.148.pop3: Flags [.], ack 43, wi
n 502, options [nop,nop,TS val 423360698 ecr 3958280553], length 0
E..4.W@.@.g.
...
......n......".....??.....
.;.....i
```

```
09:05:22.852695 IP 10.20.30.1.58386 > 10.20.30.148.pop3: Flags [P.], seq 13:2
8, ack 43, win 502, options [nop,nop,TS val 423368417 ecr 3958280553], len
gth 15
E..C.X@.@.g.
...
.....n....."....6.....
.<.....iPASS D2xc9CgD
[...]
```

Yukarıdaki terminal çıktısında, önemli olanlara daha iyi odaklanmanıza yardımcı olmak için önemsiz paketleri kaldırdık. Özellikle, kullanıcı adı ve parolanın her biri kendi paketinde gönderilmiştir. İlk pakette açıkça "USER frank" ifadesi yer alırken, son pakette "PASS D2xc9CgD" şifresi yer almaktadır.

Aynı sonuçları elde etmek için Wireshark'ı da kullanabiliriz. Aşağıdaki Wireshark penceresinde, filtre alanına pop girdiğimizi görebiliriz. Artık sadece ilgilendiğimiz trafiği filtrelediğimize göre, bir kullanıcı adı ve parolanın yakalandığını görebiliriz.



Kısacası, açık metin iletişimi kullanan her protokol bu tür bir saldırıya açıktır. Bu saldırının başarılı olması için tek şart, iletişim kuran iki sistem arasındaki bir sisteme erişime sahip olmaktır. Bu saldırı dikkat gerektirir; hafifletme, herhangi bir ağ protokolünün üzerine bir şifreleme katmanı eklemekte yatmaktadır. Özellikle HTTP, FTP, SMTP, POP3, IMAP ve diğerlerine Taşıma Katmanı Güvenliği (TLS) eklenmiştir. Uzaktan erişim için Telnet, güvenli alternatif Secure Shell (SSH) ile değiştirilmiştir.

Wireshark hakkında daha fazla bilgi edinmek isterseniz, Wireshark 101 odasını tavsiye ederiz.

Sorular

Soru ⇒ Yalnızca Telnet trafiğini yakalamak için sudo tcpdump komutuna ne eklemeniz gerekir?

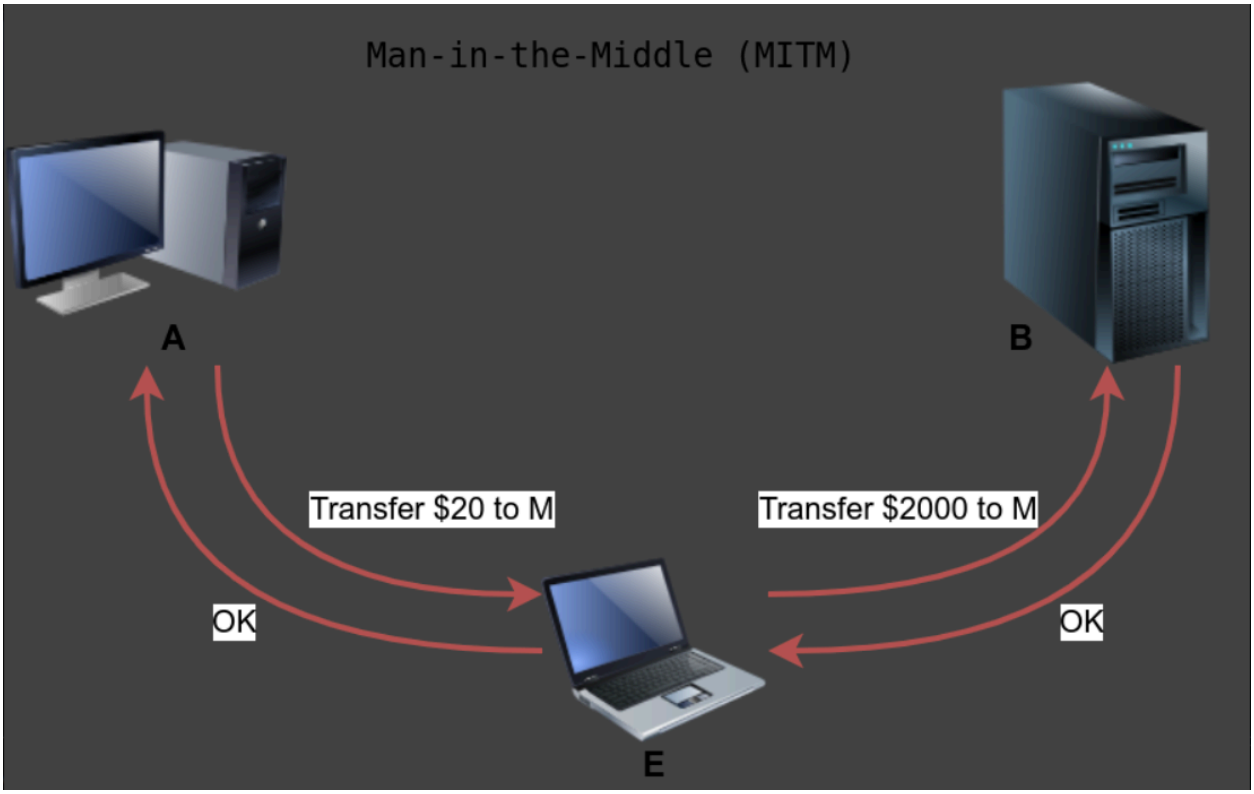
Cevap ⇒ port 23

Soru ⇒ Yalnızca IMAP trafiğini göstermek için Wireshark ile kullanabileceğiniz en basit görüntüleme filtresi nedir?

Cevap ⇒ imap

Task 3 Man-in-the-Middle (MITM) Attack (Görev 3 Ortadaki Adam (MITM) Saldırısı)

Ortadaki Adam (MITM) saldırısı, bir kurban (A) meşru bir hedefle (B) iletişim kurduğuna inandığında ancak farkında olmadan bir saldırganla (E) iletişim kurduğunda meydana gelir. Aşağıdaki şekilde A, M'ye 20$ transfer edilmesini talep etmektedir; ancak E bu mesajı değiştirmiş ve orijinal değeri yeni bir değerle değiştirmiştir. B değiştirilmiş mesajı aldı ve ona göre hareket etti.



İki taraf her bir mesajın gerçekliğini ve bütünlüğünü teyit etmezse bu saldırının gerçekleştirilmesi nispeten kolaydır. Bazı durumlarda, seçilen protokol güvenli

kimlik doğrulama veya bütünlük kontrolü sağlamaz; dahası, bazı protokoller kendilerini bu tür saldırılara açık hale getiren içsel güvensizliklere sahiptir.

HTTP üzerinden her gezindiğinizde, bir MITM saldırısına maruz kalabilirsiniz ve korkutucu olan şey, bunu fark edememenizdir. Ettercap ve Bettercap gibi birçok araç böyle bir saldırıyı gerçekleştirmenize yardımcı olabilir.

MITM ayrıca FTP, SMTP ve POP3 gibi diğer açık metin protokollerini de etkileyebilir. Bu saldırıya karşı hafifletme kriptografi kullanımını gerektirir. Çözüm, değiş tokuş edilen mesajların şifrelenmesi veya imzalanması ile birlikte uygun kimlik doğrulamasında yatmaktadır. Açık Anahtar Altyapısı (PKI) ve güvenilir kök sertifikaların yardımıyla Taşıma Katmanı Güvenliği (TLS) MITM saldırılarına karşı koruma sağlar.

Sorular

Soru ⇒ Ettercap kaç farklı arayüz sunuyor(İpucu ⇒ Bakınız https://www.ettercap-project.org/about.html)?

Cevap ⇒ 3

Soru ⇒ Bettercap'i kaç şekilde çağırabilirsiniz(İpucu ⇒ https://www.bettercap.org/usage/ adresini ziyaret edin)?

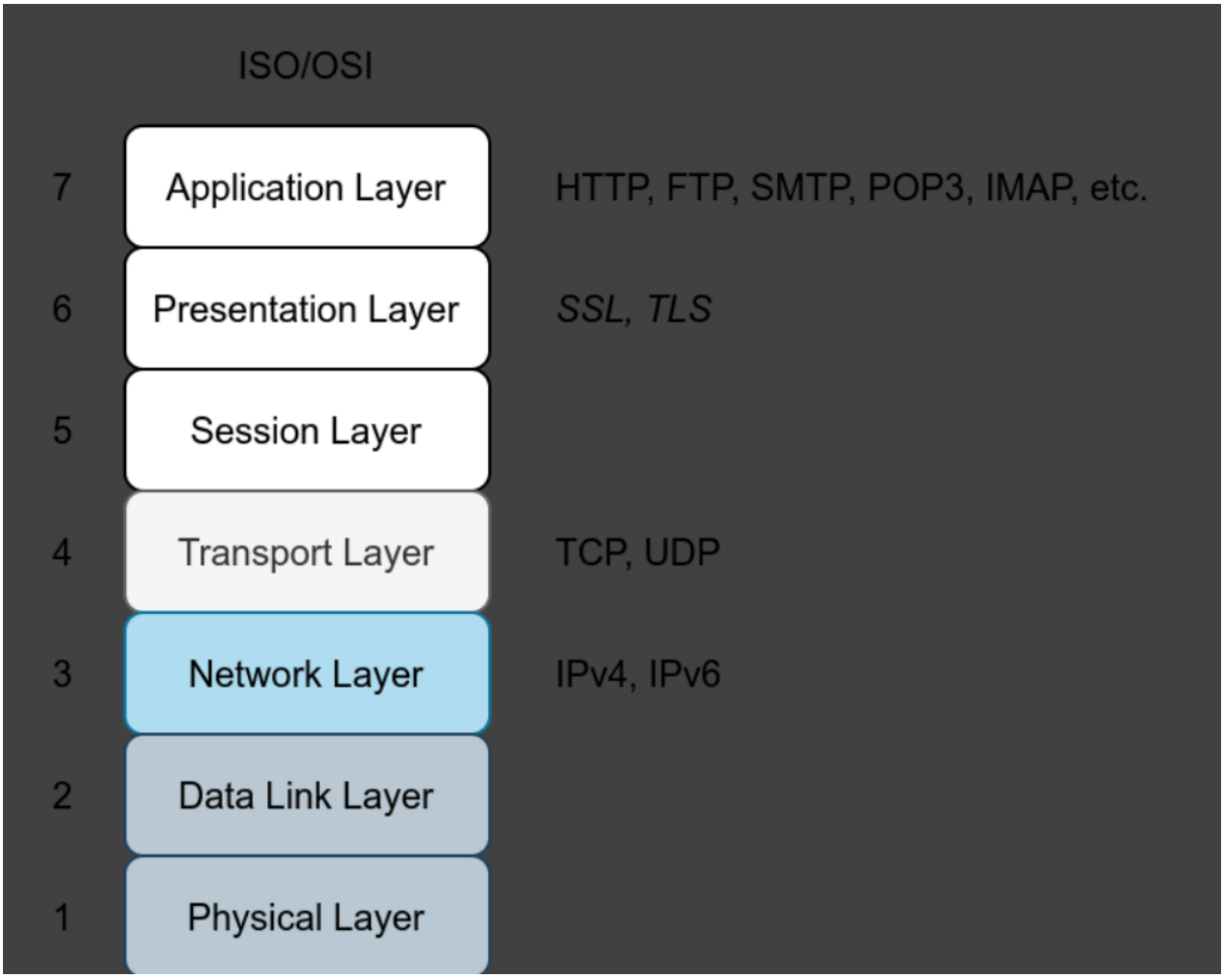
Cevap ⇒ 3

Task 4 Transport Layer Security (TLS) (Görev 4 Aktarım Katmanı Güvenliği (TLS))

Bu görevde, değiş tokuş edilen paketlerin gizliliğini ve bütünlüğünü korumak için standart bir çözüm hakkında bilgi ediniyoruz. Aşağıdaki yaklaşım parola koklama ve MITM saldırılarına karşı koruma sağlayabilir.

SSL (Secure Sockets Layer), world wide web'de online alışveriş ve ödeme bilgilerinin gönderilmesi gibi yeni uygulamalar görülmeye başlandığında ortaya çıkmıştır. Netscape SSL'i 1994 yılında tanıttı ve SSL 3.0 1996 yılında piyasaya sürüldü. Ancak sonunda daha fazla güvenliğe ihtiyaç duyuldu ve 1999 yılında TLS (Transport Layer Security) protokolü tanıtıldı. TLS ve SSL'in ne sağladığını açıklamadan önce, ağ modeline nasıl uyduklarını görelim.

Şimdiye kadar ele aldığımız yaygın protokoller verileri açık metin olarak gönderir; bu da ağa erişimi olan herkesin değiş tokuş edilen mesajları yakalamasını, kaydetmesini ve analiz etmesini mümkün kılar. Aşağıdaki resim ISO/OSI ağ katmanlarını göstermektedir. Bu odada şimdiye kadar ele aldığımız protokoller uygulama katmanında yer almaktadır. ISO/OSI modelini düşünün; protokollerimize sunum katmanı üzerinden şifreleme ekleyebiliriz. Sonuç olarak, veri orijinal formu yerine şifrelenmiş bir formatta (şifreli metin) sunulacaktır.



SSL ve TLS arasındaki yakın ilişki nedeniyle, biri diğerinin yerine kullanılabilir. Ancak TLS, SSL'den daha güvenlidir ve pratikte SSL'in yerini almıştır. SSL'i bırakabilir ve SSL/TLS yerine sadece TLS yazabilirdik, ancak SSL terimi hala yaygın olarak kullanıldığından herhangi bir belirsizliği önlemek için ikisinden bahsetmeye devam edeceğiz. Ancak, tüm modern sunucuların TLS kullanmasını bekleyebiliriz.

Mevcut bir açık metin protokolü SSL/TLS aracılığıyla şifreleme kullanacak şekilde yükseltilebilir. TLS'yi HTTP, FTP, SMTP, POP3 ve IMAP gibi protokolleri yükseltmek için kullanabiliriz. Aşağıdaki tabloda, SSL/TLS aracılığıyla şifreleme yükseltmesinden önce ve sonra ele aldığımız protokoller ve bunların varsayılan bağlantı noktaları listelenmektedir. Liste kapsamlı değildir; ancak amaç süreci daha iyi anlamamıza yardımcı olmaktır.

Protocol	Default Port	Secured Protocol	Default Port with TLS
HTTP	80	HTTPS	443
FTP	21	FTPS	990
SMTP	25	SMTPS	465
POP3	110	POP3S	995
IMAP	143	IMAPS	993

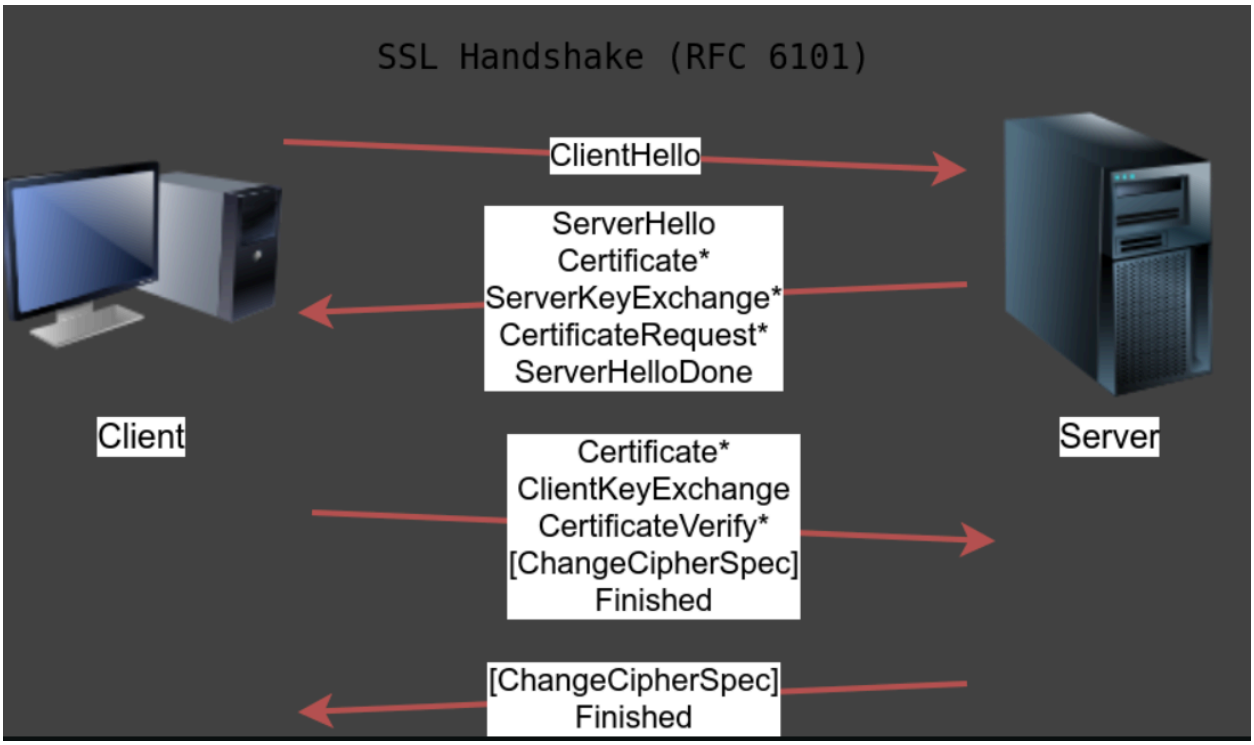
HTTP örneğini ele alalım. Başlangıçta, HTTP üzerinden bir web sayfasını almak için web tarayıcısının en azından aşağıdaki iki adımı gerçekleştirmesi gerekir:

1. Uzak web sunucusuyla bir TCP bağlantısı kurun
2. Web sunucusuna GET ve POST istekleri gibi HTTP istekleri gönderin.

HTTPS trafiği şifrelemek için ek bir adım gerektirir. Bu yeni adım TCP bağlantısı kurulduktan sonra ve HTTP istekleri gönderilmeden önce gerçekleşir. Bu ekstra adım, daha önce sunulan resimdeki ISO/OSI modelinden çıkarılabilir. Sonuç olarak, HTTPS en az aşağıdaki üç adımı gerektirir:

1. Bir TCP bağlantısı kurun
2. SSL/TLS bağlantısı kurun
3. Web sunucusuna HTTP istekleri gönderme

Bir SSL/TLS bağlantısı kurmak için istemcinin sunucu ile uygun el sıkışmasını gerçekleştirmesi gerekir. RFC 6101'e göre, SSL bağlantı kuruluşu aşağıdaki şekildeki gibi görünecektir.



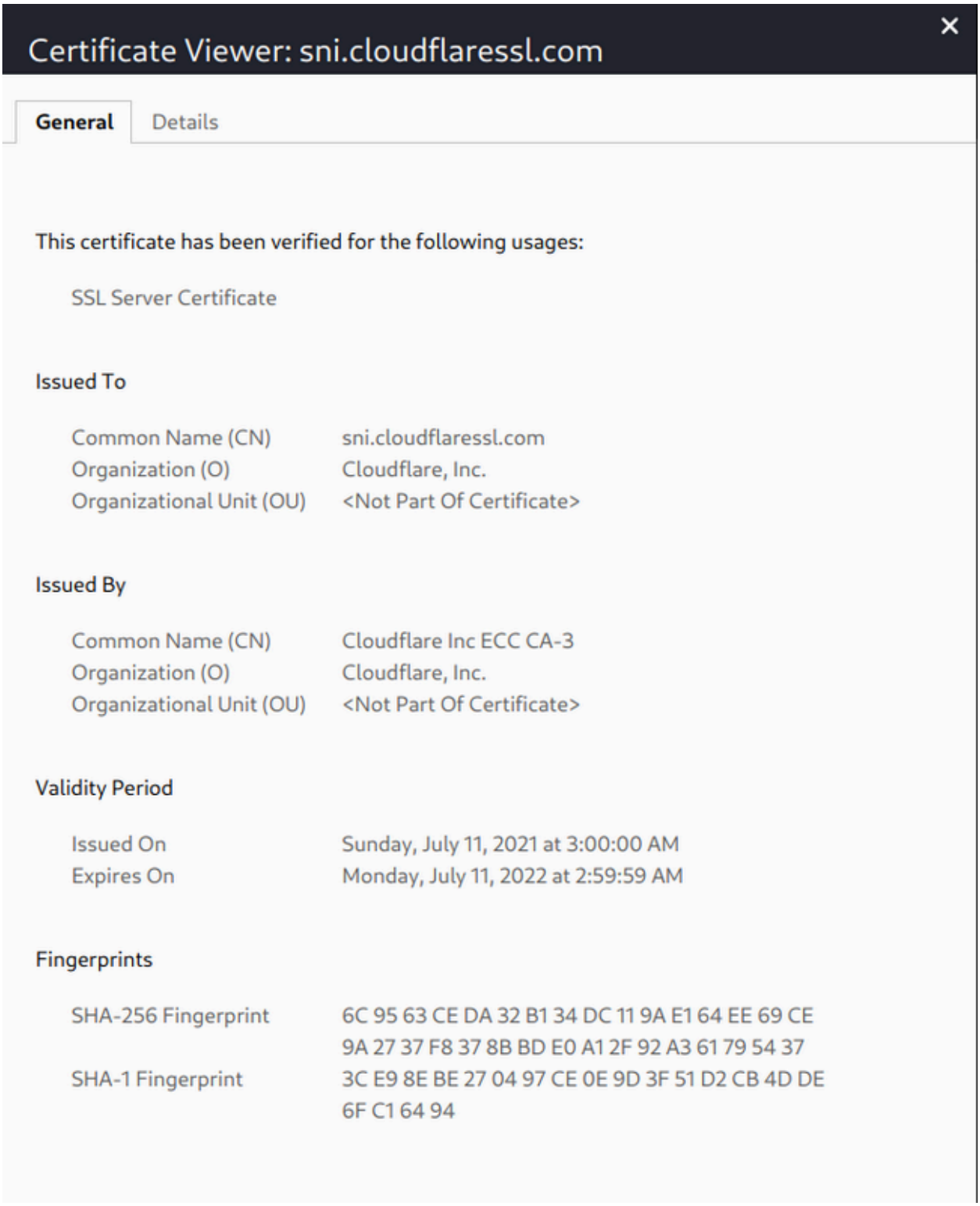
Sunucu ile bir TCP bağlantısı kurduktan sonra, istemci yukarıdaki şekilde gösterildiği gibi bir SSL/TLS bağlantısı kurar. Kriptografi bilginize bağlı olarak terimler karmaşık görünebilir, ancak dört adımı şu şekilde basitleştirebiliriz:

1. İstemci, desteklenen algoritmalar gibi yeteneklerini belirtmek için sunucuya bir ClientHello gönderir.
2. Sunucu, seçilen bağlantı parametrelerini belirten bir ServerHello ile yanıt verir. Sunucu kimlik doğrulaması gerekiyorsa sunucu sertifikasını sağlar. Sertifika kendini tanımlamak için dijital bir dosyadır; genellikle üçüncü bir tarafça dijital olarak imzalanır. Ayrıca, müzakereyi tamamladığını belirtmek için ServerHelloDone mesajını göndermeden önce ServerKeyExchange mesajında ana anahtarı oluşturmak için gerekli ek bilgileri gönderebilir.
3. İstemci, ana anahtarı oluşturmak için gereken ek bilgileri içeren bir ClientKeyExchange ile yanıt verir. Ayrıca, şifreleme kullanmaya geçer ve ChangeCipherSpec mesajını kullanarak sunucuyu bilgilendirir.
4. Sunucu da şifreleme kullanmaya geçer ve istemciyi ChangeCipherSpec mesajında bilgilendirir.

Bu hala karmaşık geliyorsa, endişelenmeyin; sadece özüne ihtiyacımız var. Bir istemci, genel sertifikaya sahip bir sunucu ile gizli bir anahtar üzerinde anlaşmaya varabilmiştir. Bu gizli anahtar, kanalı izleyen üçüncü bir tarafın bunu keşfedemeyeceği şekilde güvenli bir şekilde oluşturulmuştur. İstemci ve sunucu arasındaki diğer iletişim, oluşturulan anahtar kullanılarak şifrelenecektir.

Sonuç olarak, bir SSL/TLS el sıkışması kurulduktan sonra, HTTP istekleri ve değiş tokuş edilen veriler iletişim kanalını izleyen herhangi biri tarafından erişilebilir olmayacaktır.

Son bir not olarak, SSL/TLS'nin etkili olabilmesi için, özellikle HTTPS üzerinden web'de gezinirken, sistemlerimiz tarafından güvenilen sertifika yetkilileri tarafından imzalanmış genel sertifikalara güveniriz. Başka bir deyişle, TryHackMe'ye HTTPS üzerinden göz attığımızda, tarayıcımız TryHackMe web sunucusunun aşağıdaki örnekte olduğu gibi güvenilir bir sertifika yetkilisinden imzalı bir sertifika sağlamasını bekler. Bu şekilde, tarayıcımız doğru sunucu ile iletişim kurduğundan emin olur ve bir MITM saldırısı gerçekleşemez.



Yukarıdaki şekilde aşağıdaki bilgileri görebiliriz:

1. Sertifika kime veriliyor? Bu sertifikayı kullanacak olan şirketin adıdır.

2. Sertifikayı kim verdi? Bu, bu sertifikayı veren sertifika yetkilisidir.
3. Geçerlilik süresi. Örneğin, geçerlilik süresi dolmuş bir sertifikayı kullanmak istemezsiniz.

Neyse ki, ziyaret ettiğimiz her site için sertifikayı manuel olarak kontrol etmek zorunda değiliz; web tarayıcımız bunu bizim için yapacaktır. Web tarayıcımız doğru sunucu ile konuştuğumuzdan emin olacak ve sunucunun sertifikası sayesinde iletişimimizin güvenli olduğundan emin olacaktır.

Soru ⇒ DNS TLS kullanılarak da güvence altına alınabilir. TLS kullanan DNS protokolünün üç harfli kısaltması nedir (İpucu ⇒ Araştırma gerektirir.)?

Cevap ⇒ DoT

Task 5 Secure Shell (SSH) (Görev 5 Güvenli Kabuk (SSH))

Secure Shell (SSH) uzaktan sistem yönetimi için güvenli bir yol sağlamak amacıyla oluşturulmuştur. Başka bir deyişle, ağ üzerinden başka bir sisteme güvenli bir şekilde bağlanmanızı ve uzak sistemde komutlar çalıştırmanızı sağlar. Basitçe söylemek gerekirse, SSH'deki "S" güvenli anlamına gelir ve bu da basitçe şu şekilde özetlenebilir:

1. Uzak sunucunun kimliğini doğrulayabilirsiniz
2. Değiş tokuş edilen mesajlar şifrelenir ve yalnızca hedeflenen alıcı tarafından çözülebilir
3. Her iki taraf da mesajlardaki herhangi bir değişikliği tespit edebilir

Yukarıdaki üç nokta kriptografi ile sağlanır. Daha teknik bir ifadeyle, bunlar gizlilik ve bütünlüğün bir parçasıdır ve farklı şifreleme algoritmalarının uygun şekilde kullanılmasıyla mümkün olur.

SSH kullanmak için bir SSH sunucusuna ve bir SSH istemcisine ihtiyacınız vardır. SSH sunucusu varsayılan olarak 22 numaralı bağlantı noktasını dinler. SSH istemcisi aşağıdakileri kullanarak kimlik doğrulaması yapabilir:

- Bir kullanıcı adı ve şifre

- Bir özel ve genel anahtar (SSH sunucusu ilgili genel anahtarı tanıyacak şekilde yapılandırıldıktan sonra)

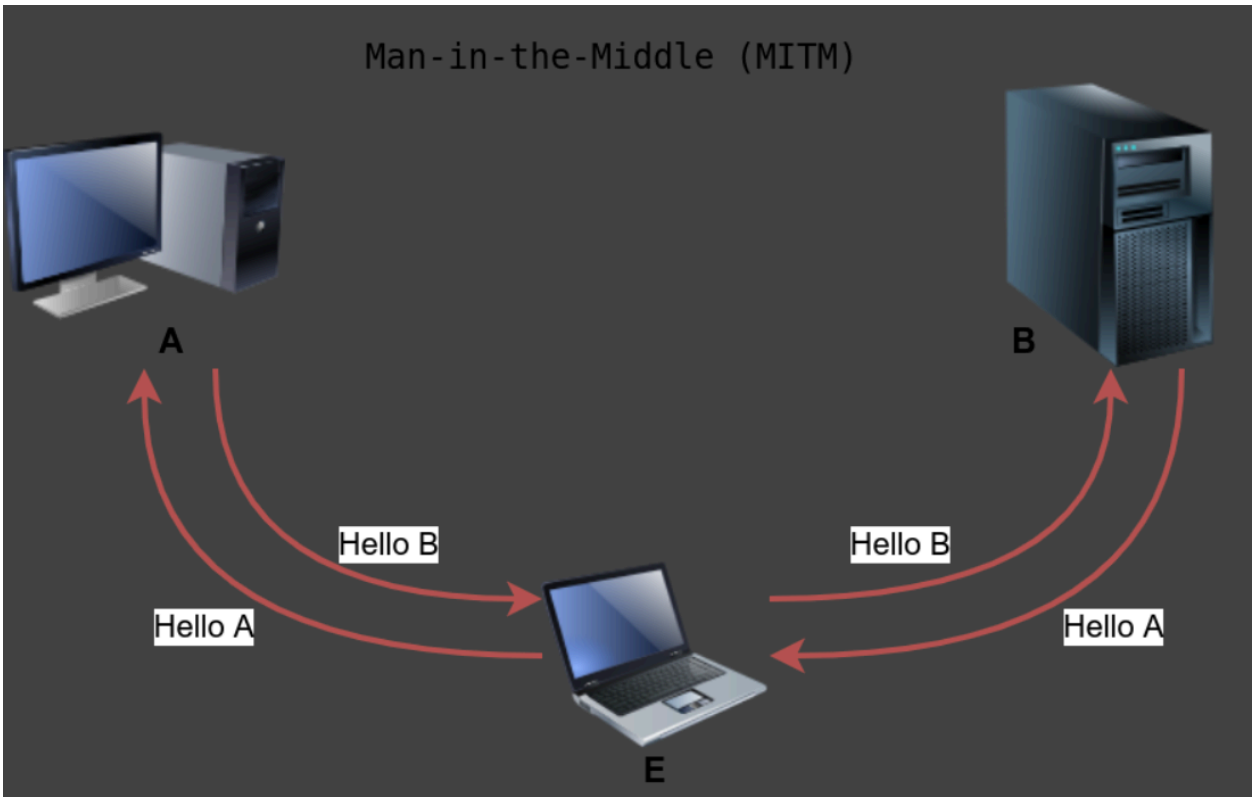
Linux, macOS ve 2018 sonrası MS Windows sürümlerinde, aşağıdaki ssh username@MACHINE_IP komutunu kullanarak bir SSH sunucusuna bağlanabilirsiniz. Bu komut MACHINE_IP IP adresindeki sunucuya kullanıcı adı ile bağlanmaya çalışacaktır. Eğer bir SSH sunucusu varsayılan bağlantı noktasını dinliyorsa, sizden kullanıcı adı için parola girmenizi isteyecektir. Kimlik doğrulandıktan sonra, kullanıcı hedef sunucunun terminaline erişebilecektir. Aşağıdaki terminal çıktısı, Debian Linux sunucusuna erişmek için SSH kullanımına bir örnektir.

```
user@TryHackMe$ ssh mark@MACHINE_IPmark@MACHINE_IP's password:
XBtc49AB

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Sep 20 13:53:17 2021
mark@debian8:~$
```

Yukarıdaki örnekte, ssh mark@MACHINE_IP komutunu verdik. Ardından, doğru parolayı girdiğimizde, uzaktaki sistemin terminaline erişim sağladık. SSH uzaktan yönetim için çok güvenilirdir çünkü kullanıcı adımız ve parolamız şifreli olarak gönderilir; ayrıca, uzaktaki sistemde çalıştırdığımız tüm komutlar şifreli bir kanal üzerinden gönderilecektir.

Bu sisteme ilk kez bağlanıyorsak, ortadaki adam (MITM) saldırılarından kaçınmak için SSH sunucusunun açık anahtarının parmak izini doğrulamamız gerekeceğini unutmayın. Daha önce açıklandığı gibi, MITM, kötü niyetli bir taraf olan E'nin kendisini A ve B arasına yerleştirmesi ve B gibi davranarak A ile iletişim kurması ve A ve B birbirleriyle doğrudan iletişim kurduklarını düşünürken A gibi davranarak B ile iletişim kurması durumunda gerçekleşir. SSH durumunda, genellikle açık anahtarın geçerli olup olmadığını kontrol edecek üçüncü bir tarafımız yoktur, bu

nedenle bunu manuel olarak yapmamız gerekir. Bu saldırı aşağıdaki resimde gösterilmektedir.



SSH protokolüne dayalı SCP (Güvenli Kopyalama Protokolü) kullanarak dosya aktarmak için SSH kullanabiliriz. Sözdiziminin bir örneği aşağıdaki gibidir: scp mark@MACHINE_IP:/home/mark/archive.tar.gz ~. Bu komut, /home/mark dizininde bulunan archive.tar.gz adlı bir dosyayı uzak sistemden ~ dizinine, yani oturum açmış olan kullanıcının ev dizininin kök dizinine kopyalayacaktır.

Başka bir örnek sözdizimi scp backup.tar.bz2 mark@MACHINE_IP:/home/mark/ şeklindedir. Bu komut backup.tar.bz2 dosyasını yerel sistemden uzak sistemdeki /home/mark/ dizinine kopyalayacaktır.

```
user@TryHackMe$ scp document.txt mark@MACHINE_IP:/home/markmark@
MACHINE_IP's password:
document.txt                                  100% 1997KB  70.4MB/s   00:00
```

Son bir not olarak, FTP, 990 numaralı bağlantı noktasını kullanan FTPS protokolü kullanılarak SSL/TLS ile güvence altına alınabilir. FTP'nin SFTP protokolü olan SSH

protokolü kullanılarak da güvence altına alınabileceğini belirtmek gerekir. Varsayılan olarak bu hizmet tıpkı SSH gibi 22 numaralı bağlantı noktasını dinler.

Sorular

Soru ⇒ XBtc49AB parolasıyla işaret olarak MACHINE_IP'ye bağlanmak için SSH kullanın. uname -r kullanarak Çekirdek sürümünü bulun?

Cevap ⇒ 5.15.0-119-generic

Soru ⇒ Uzak sistemden book.txt dosyasını indirmek için SSH kullanın. Scp indirme boyutu olarak kaç KB gösterdi?

Cevap ⇒ 415

Task 6 Password Attack (Görev 6 Parola Saldırısı)

Ağ paketi yakalama ve MITM saldırılarının yanı sıra TLS ve SSH kullanarak bu saldırıların nasıl azaltılabileceğini tartıştık. Bu odada ele alacağımız üçüncü saldırı türü şifre saldırısıdır.

Birçok protokol kimlik doğrulaması yapmanızı gerektirir. Kimlik doğrulama, kim olduğunuzu iddia ettiğinizi kanıtlamaktır. POP3 gibi protokolleri kullanırken, kimliğimizi doğrulamadan önce posta kutusuna erişim izni verilmemelidir. Protokoller ve Sunucular odasındaki POP3 örneği size kolaylık sağlamak için aşağıda tekrarlanmıştır. Bu örnekte, frank kullanıcısı olarak tanımlandık ve doğru şifreyi verdiğimiz için sunucu bizi doğruladı. Başka bir deyişle, parola kimlik doğrulamanın bir yoludur.

```
pentester@TryHackMe$ telnet MACHINE_IP 110Trying MACHINE_IP...
Connected to MACHINE_IP.
Escape character is '^]'.
+OK MACHINE_IP Mail Server POP3 Wed, 15 Sep 2021 11:05:34 +0300
USER frank
+OK frank
PASS D2xc9CgD
+OK 1 messages (179) octets
STAT
```

```
+OK 1 179
LIST
+OK 1 messages (179) octets
1 179
.
RETR 1
+OK
From: Mail Server
To: Frank
subject: Sending email with Telnet
Hello Frank,
I am just writing to say hi!
.
QUIT
+OK MACHINE_IP closing connection
Connection closed by foreign host.
```

Kimlik doğrulama veya kimliğinizi kanıtlama, aşağıdakilerden biri veya ikisinin bir kombinasyonu yoluyla gerçekleştirilebilir:

1. Şifre ve PIN kodu gibi bildiğiniz bir şey.
2. SIM kart, RFID kartı ve USB dongle gibi sahip olduğunuz bir şey.
3. Parmak izi ve iris gibi olduğunuz bir şey.

Bu görev parolalara, yani hedefin bildiği bir şeye yönelik saldırılara odaklanacaktır. Telnet, SSH, POP3 ve IMAP gibi protokolleri kullanarak önceki birkaç sunucuyla iletişimi tekrar gözden geçirirseniz, erişim sağlamak için her zaman bir parolaya ihtiyacımız vardır. 2013'teki Adobe ihlalinden sızan 150 milyon kullanıcı adı ve şifreye dayanarak, ilk on şifre şunlardır:

- 123456
- 123456789
- password
- adobe123
- 12345678

- qwerty
- 1234567
- 111111
- photoshop
- 123123

Yalnızca iki parola Adobe ve ürünleriyle ilgilidir, ancak geri kalanı geneldir. Bunun son on yılda değiştiğini düşünebilirsiniz; ancak 123456, 1234567, 12345678 ve 123456789 hala birçok kullanıcı için yaygın seçeneklerdir. Diğerleri henüz qwerty'nin gizli olmadığını fark etmedi ve birçok kişi tarafından şifre olarak kullanılıyor.

Parolalara yönelik saldırılar genellikle şu şekilde gerçekleştirilir:

1. Şifre Tahmin Etme: Bir parolayı tahmin etmek, hedef hakkında evcil hayvanının adı ve doğum yılı gibi bazı bilgiler gerektirir.
2. Sözlük Saldırısı: Bu yaklaşım, parola tahminini genişletir ve bir sözlük veya kelime listesindeki tüm geçerli kelimeleri dahil etmeye çalışır.
3. Kaba Kuvvet Saldırısı: Bu saldırı, bir saldırganın tüm olası karakter kombinasyonlarını denemeye kadar gidebileceği en kapsamlı ve zaman alıcı saldırıdır ve hızlı bir şekilde büyür (karakter sayısı ile üstel büyüme).

Sözlük saldırılarına odaklanalım. Zaman içinde, bilgisayar korsanları veri ihlallerinden sızan şifreleri içeren liste üzerine liste derlediler. Bunun bir örneği, AttackBox'ta /usr/share/wordlists/rockyou.txt adresinde bulabileceğiniz RockYou'nun ihlal edilen şifreler listesidir. Kelime listesinin seçimi hedef hakkındaki bilginize bağlı olmalıdır. Örneğin, Fransız bir kullanıcı İngilizce bir kelime yerine Fransızca bir kelime kullanabilir. Sonuç olarak, Fransızca bir kelime listesi daha umut verici olabilir.

Ortak şifreleri veya bir kelime listesindeki girişleri denemek için otomatik bir yol istiyoruz; işte THC Hydra geliyor. Hydra, FTP, POP3, IMAP, SMTP, SSH ve HTTP ile ilgili tüm yöntemler dahil olmak üzere birçok protokolü destekler. Genel komut satırı sözdizimi şöyledir: hydra -l username -P wordlist.txt server service burada aşağıdaki seçenekleri belirtiriz:

- -l kullanıcı adı: -l kullanıcı adından, yani hedefin oturum açma adından önce gelmelidir.
- -P wordlist.txt: -P, verilen kullanıcı adıyla denemek istediğiniz parolaların listesini içeren bir metin dosyası olan wordlist.txt dosyasından önce gelir.
- sunucu, hedef sunucunun ana bilgisayar adı veya IP adresidir.
- service, sözlük saldırısını başlatmaya çalıştığınız hizmeti belirtir.

Aşağıdaki somut örnekleri göz önünde bulundurun:

- hydra -l mark -P /usr/share/wordlists/rockyou.txt MACHINE_IP ftp, FTP sunucusuna karşı sağlanan şifreler üzerinde yineleme yaparken kullanıcı adı olarak mark'ı kullanacaktır.
- hydra -l mark -P /usr/share/wordlists/rockyou.txt ftp://MACHINE_IP önceki örnekle aynıdır. MACHINE_IP ftp ftp://MACHINE_IP ile aynıdır.
- hydra -l frank -P /usr/share/wordlists/rockyou.txt MACHINE_IP ssh, farklı parolalar kullanarak SSH üzerinden oturum açmaya çalışırken kullanıcı adı olarak frank'ı kullanacaktır.

Ekleyebileceğiniz bazı ekstra isteğe bağlı argümanlar vardır:

- -Söz konusu hizmet için varsayılan olmayan bir bağlantı noktası belirtmek için -s PORT.
- Verbose için -V veya -vV, Hydra'nın denenen kullanıcı adı ve parola kombinasyonlarını göstermesini sağlar. Bu ayrıntı düzeyi, özellikle komut satırı sözdiziminizden hala emin değilseniz, ilerlemeyi görmek için çok kullanışlıdır.
- -t n burada n hedefe paralel bağlantı sayısıdır. -16, hedefe bağlanmak için kullanılan 16 iş parçacığı oluşturacaktır.
- -d, hata ayıklama için, neler olup bittiği hakkında daha ayrıntılı bilgi almak için. Hata ayıklama çıktısı sizi büyük bir hayal kırıklığından kurtarabilir; örneğin, Hydra kapalı bir bağlantı noktasına bağlanmaya çalışır ve zaman aşımına uğrarsa, -d bunu hemen ortaya çıkaracaktır.

Parola bulunduğunda, işlemi sonlandırmak için CTRL-C komutunu verebilirsiniz. TryHackMe görevlerinde, herhangi bir saldırının beş dakikadan daha kısa sürede bitmesini bekliyoruz; ancak gerçek hayat senaryolarında saldırı genellikle daha

uzun sürer. Hydra'nın ilerleyişi hakkında sizi bilgilendirmesini istiyorsanız, ayrıntı veya hata ayıklama seçenekleri oldukça yararlı olabilir.

Özetle, oturum açma sistemlerine yönelik saldırılar, uygun bir kelime listesi ile birlikte THC Hydra gibi bir araç kullanılarak verimli bir şekilde gerçekleştirilebilir. Bu tür saldırılara karşı önlemler karmaşık olabilir ve hedef sisteme bağlıdır. Yaklaşımlardan birkaçı şunlardır:

- **Password Policy** (Parola İlkesi): Kullanıcı tarafından belirlenen parolalar üzerinde minimum karmaşıklık kısıtlamalarını zorlar.
- **Account Lockout** (Hesap Kilitleme): Belirli sayıda başarısız denemeden sonra hesabı kilitler.
- **Throttling Authentication Attempts** (Kimlik Doğrulama Girişimlerini Azaltma): Bir oturum açma girişimine verilen yanıtı geciktirir. Parolayı bilen biri için birkaç saniyelik gecikme tolere edilebilir, ancak otomatik araçları ciddi şekilde engelleyebilir.
- **Using CAPTCHA** (CAPTCHA Kullanımı): Makineler için zor olan bir sorunun çözülmesini gerektirir. Oturum açma sayfası bir grafik kullanıcı arayüzü (GUI) üzerinden yapılıyorsa iyi çalışır. (CAPTCHA'nın Bilgisayarlar ve İnsanları Ayırmak için Tamamen Otomatikleştirilmiş Halka Açık Turing testi anlamına geldiğini unutmayın).
- Kimlik doğrulama için genel bir sertifika kullanılmasını gerektirir. Bu yaklaşım örneğin SSH ile iyi çalışır.
- **Two-Factor Authentication** (İki Faktörlü Kimlik Doğrulama): Kullanıcıdan e-posta, akıllı telefon uygulaması veya SMS gibi diğer yollarla kullanılabilen bir kod sağlamasını isteyin.
- IP tabanlı coğrafi konum belirleme gibi daha sofistike veya kullanıcı hakkında bazı yerleşik bilgiler gerektirebilecek başka birçok yaklaşım vardır.

Yukarıdaki yaklaşımların bir kombinasyonunu kullanmak, parola saldırılarına karşı korunmak için mükemmel bir yaklaşımdır.

Soru ⇒ E-posta hesaplarından birinin lazie olduğunu öğrendik. MACHINE_IP üzerinde IMAP hizmetine erişmek için kullanılan parola nedir?

İpucu ⇒ butterfly

Task 7 Summary (Görev 7 Özet)

Bu oda, çeşitli protokolleri, kullanımlarını ve kaputun altında nasıl çalıştıklarını ele aldı. Üç yaygın saldırı vardır:

1. Sniffing Attack (Koklama Saldırısı)
2. MITM Attack (MITM Saldırısı)
3. Password Attack (Şifre Saldırısı)

Yukarıdakilerin her biri için hem saldırı detaylarına hem de hafifletme adımlarına odaklandık.

Belirli sunuculara ve protokollere karşı başka birçok saldırı gerçekleştirilebilir. Bazı ilgili modüllerin bir listesini vereceğiz.

- **Vulnerability Research** (Güvenlik Açığı Araştırması): Bu modül güvenlik açıkları ve istismarlar hakkında daha fazla bilgi sağlar.
- **Metasploit** (Metasploit): Bu modül, hedef sistemleri istismar etmek için Metasploit'in nasıl kullanılacağı konusunda sizi eğitir.
- Burp Suite (Burp Suite): Bu modül, HTTP trafiğini kesmek ve web ile ilgili saldırılar başlatmak için Burp Suite'i nasıl kullanacağınızı öğretir.

Yaygın protokoller için varsayılan bağlantı noktası numarasını hatırlamak iyidir. Kolaylık olması açısından, ele aldığımız hizmetler aşağıdaki tabloda alfabetik sıraya göre listelenmiştir.

Protocol	TCP Port	Application(s)	Data Security
FTP	21	File Transfer	Cleartext
FTPS	990	File Transfer	Encrypted
HTTP	80	Worldwide Web	Cleartext
HTTPS	443	Worldwide Web	Encrypted
IMAP	143	Email (MDA)	Cleartext
IMAPS	993	Email (MDA)	Encrypted

POP3	110	Email (MDA)	Cleartext
POP3S	995	Email (MDA)	Encrypted
SFTP	22	File Transfer	Encrypted
SSH	22	Remote Access and File Transfer	Encrypted
SMTP	25	Email (MTA)	Cleartext
SMTPS	465	Email (MTA)	Encrypted
Telnet	23	Remote Access	Cleartext

Hydra, farklı şifreleri denemek için terminalden başlatabileceğiniz çok etkili bir araç olmaya devam ediyor. Ana seçeneklerini aşağıdaki tabloda özetliyoruz.

Option	Explanation
`-l username`	Provide the login name
`-P WordList.txt`	Specify the password list to use
`server service`	Set the server address and service to attack
`-s PORT`	Use in case of non-default service port number
`-V` or `-vV`	Show the username and password combinations being tried
`-d`	Display debugging output if the verbose output is not helping

Soru ⇒ Şu ana kadar Ağ Güvenliği modülünün sekizinci odasını tamamladınız. Becerilerinizi test etmek için lütfen bu modülün son odasına geçin.

Cevap ⇒ Cevap Gerekmemektedir.