# PART A

## Certificate details of the website ebay.com

| Field Name | Subject (CN) of certificate holder (website) | Subject (CN) of certificate holder (intermediate) | Subject (CN) of certificate holder (root) | Remarks/observations |
|---|---|---|---|---|
| **Issuer** | Country : US<br>Organization : DigiCert Inc<br>Common Name : DigiCert TLS RSA SHA256 2020 CA1 | CN = DigiCert Global Root CA<br>OU = www.digicert.com<br>O = DigiCert Inc<br>C = US | CN = DigiCert Global Root CA<br>OU = www.digicert.com<br>O = DigiCert Inc<br>C = US | We can notice that the issuer for both intermediate and root is the same because the root cert is self signed. |
| **Version No.** | 3 | 3 | 3 | Latest version of X.509 |
| **Signature Algo** | SHA-256 with RSA Encryption | SHA-256 with RSA Encryption | SHA-1 with RSA Encryption | Intermediate and end user certs are signed with the new algorithm whereas the root is signed with the old one still. |
| **Size of digest** | 256 bits | 256 bits | 160 bits | |
| **Signature Value** | 20 E6 20 36 28 59 6F 5A B0 B9 BB BA 54 34 3A 7E 6F C9 13 60 79 BB 8B CE 3B 3A 74 F2 32 DC 49 11 CE B3 84 49 08 BE 90 66 88 B9 A0 FE C5 DE 53 7F C0 E5 88 E4 FC 18 8D 7D 56 BD AC 5E 28 46 74 74 F8 80 12 14 76 2A 02 BD 73 73 1A 4A B7 D1 DA F8 53 AF 41 62 A2 59 59 0B 45 F7 E1 A0 31 8A 99 B4 C3 6A E7 EC 1B 5C EE B0 9E DE 5A 41 5D AA 90 D8 | 80 32 CE 5E 0B DD 6E 5A 0D 0A AF E1 D6 84 CB C0 8E FA 85 70 ED DA 5D B3 0C F7 2B 75 40 FE 85 0A FA F3 31 78 B7 70 4B 1A 89 58 BA 80 BD F3 6B 1D E9 7E CF 0B BA 58 9C 59 D4 90 D3 FD 6C FD D0 98 6D B7 71 82 5B CF 6D 0B 5A 09 D0 7B DE C4 | CB 9C 37 AA 48 13 12 0A FA DD 44 9C 4F 52 B0 F4 DF AE 04 F5 79 79 08 A3 24 18 FC 4B 2B 84 C0 2D B9 D5 C7 FE F4 C1 1F 58 CB B8 6D 9C 7A 74 E7 98 29 AB 11 B5 E3 70 A0 A1 CD 4C 88 99 93 8C 91 70 E2 AB 0F 1C BE 93 A9 FF 63 D5 E4 07 | |

| | | | | |
|---|---|---|---|---|
| | 86 D8 64 15 69 0B 32 D3 DD 27 99 5F 51 E9 28 A0 15 C5 E8 02 D7 7C 18 A1 EA 7C 19 2C 29 BC 4B 4F E2 88 FA E8 19 AF 50 1A 72 E8 9D 5F EB 8E 8A 14 C5 AE 5C 1C 8C 42 65 72 47 C7 15 86 26 34 43 5E E9 C6 91 AF 35 99 73 12 AD 27 37 63 75 39 5E 34 69 68 A4 D4 71 C9 B4 4C 4A 09 7E 27 23 1A A4 9E B6 A8 A2 EF 9C 16 27 80 20 85 6D FF 87 CB AA 81 B6 40 1D 54 D7 CC 28 F5 18 12 A1 7E F3 04 20 DF 20 F4 F9 5E CD 64 34 85 21 A9 A1 62 F2 90 D8 32 | 43 D8<br>2A A4 DE 9E 41 26 5F BB 8F 99 CB DD AE E1 A8 6F<br>9F 87 FE 74 B7 1F 1B 20 AB B1 4F C6 F5 67 5D 5D<br>9B 3C E9 FF 69 F7 61 6C D6 D9 F3 FD 36 C6 AB 03<br>88 76 D2 4B 2E 75 86 E3 FC D8 55 7D 26 C2 11 77 DF 3E 02 B6 7C F3 AB 7B 7A 86 36 6F B8 F7 D8 93<br>71 CF 86 DF 73 30 FA 7B AB ED 2A 59 C8 42 84 3B<br>11 17 1A 52 F3 C9 0E 14 7D A2 5B 72 67 BA 71 ED<br>57 47 66 C5 B8 02 4A 65 34 5E 8B D0 2A 3C 20 9C<br>51 99 4C E7 52 9E F7 6B 11 2B 0D 92 7E 1D E8 8A<br>EB 36 16 43 87 EA 2A 63 BF 75 3F EB DE C4 03 BB<br>0A 3C F7 30 EF EB AF 4C FC 8B 36 10 73 3E F3 A4 | 60 D3 A3 BF<br>9D 5B 09 F1 D5 8E E3 53 F4 8E 63 FA 3F A7 DB B4<br>66 DF 62 66 D6 D1 6E 41 8D F2 2D B5 EA 77 4A 9F<br>9D 58 E2 2B 59 C0 40 23 ED 2D 28 82 45 3E 79 54<br>92 26 98 E0 80 48 A8 37 EF F0 D6 79 60 16 DE AC<br>E8 0E CD 6E AC 44 17 38 2F 49 DA E1 45 3E 2A B9<br>36 53 CF 3A 50 06 F7 2E E8 C4 57 49 6C 61 21 18<br>D5 04 AD 78 3C 2C 3A 80 6B A7 EB AF 15 14 E9 D8<br>89 C1 B9 38 6C E2 91 6C 8A FF 64 B9 77 25 57 30<br>C0 1B 24 A3 E1 DC E9 DF 47 7C B5 B4 24 08 05 30<br>EC 2D BD 0B BF 45 BF 50 B9 A9 F3 EB 98 01 12 AD<br>C8 88 C6 98 34 5F 8D 0A 3C C6 E9 D5 95 95 6D DE | |
| **Validity period** | Tue, 25 Jan 2022 00:00:00 GMT to Wed, 25 Jan 2023 23:59:59 GMT | Wed, 14 Apr 2021 00:00:00 GMT to Sun, 13 Apr 2031 23:59:59 GMT | Fri, 10 Nov 2006 00:00:00 GMT to Mon, 10 Nov 2031 00:00:00 GMT | This might be one of the parameters used to determine if the certificate is legitimate or not. |
| **Is Subject field (CN), FQDN?** | yes | No | yes | FDQNs start with www or https etc. |
| **Certificate type: DV, IV, OV or EV? Tell also how you are able to determine the type!** | Organization Validation | Organization Validation | Organization Validation | Since for all the certificates common name as well as the organization name is present, it holds the |

| | | | | |
|---|---|---|---|---|
| | | | | organization validation and are more trustable |
| **Subject Alternative Name (SAN/UCC), if any** | DNS Name: hcptreportapi.ebay.com<br>DNS Name: hcptimgs.ebay.com<br>DNS Name: hcptassets.ebay.com<br>DNS Name: hcptjs.ebay.com<br>DNS Name: m.ebay.com<br>DNS Name: m.ebay.co.uk<br>DNS Name: info.ebayinc.com<br>DNS Name: image.edpn.ebay.com<br>DNS Name: www.ebay.com<br>DNS Name: ucpstatic.ebay.com<br>DNS Name: svcs.ebay.com<br>DNS Name: srwsvcs.ebay.com<br>DNS Name: srv.uk.ebayrtm.com<br>DNS Name: srv.main.ebayrtm.com<br>DNS Name: srv.it.ebayrtm.com<br>DNS Name: srv.in.ebayrtm.com<br>DNS Name: srv.fr.ebayrtm.com<br>DNS Name: srv.de.ebayrtm.com<br>DNS Name: srv.au.ebayrtm.com<br>DNS Name: sofe.express.ebay.com<br>DNS Name: sofe.ebay.it<br>DNS Name: sofe.ebay.in<br>DNS Name: sofe.ebay.fr<br>DNS Name: sofe.ebay.de<br>DNS Name: sofe.ebay.com.au<br>DNS Name: sofe.ebay.com<br>DNS Name: sofe.ebay.co.uk<br>DNS Name: sofe.ebay.at<br>DNS Name: shippingtool.ebay.cn<br>DNS Name: securepics.ebaystatic.com<br>DNS Name: secureir.sandbox.ebaystatic.com<br>DNS Name: secureir.ebaystatic.com<br>DNS Name: | NONE | NONE | There are no SANs for intermediate and root certificates whereas the end user holds many alias names and hence all these domains are certified under the same end-user certificate. |

| | | | | |
|---|---|---|---|---|
| | secureinclude.ebaystatic.com<br>DNS Name: rover.ebay.it<br>DNS Name: rover.ebay.in<br>DNS Name: rover.ebay.fr<br>DNS Name: rover.ebay.de<br>DNS Name: rover.ebay.com.au<br>DNS Name: rover.ebay.com<br>DNS Name: rover.ebay.co.uk<br>DNS Name: m.ebay.it<br>DNS Name: m.ebay.fr<br>DNS Name: m.ebay.de<br>DNS Name: m.ebay.com.au<br>DNS Name: identity-api.ebay.com<br>DNS Name: i.ebayimg.com<br>DNS Name: gh.ebaystatic.com<br>DNS Name: fundinginstrument.ebay.de<br>DNS Name: cdn.ebaymainstreet.com<br>DNS Name: apacshippingtool.ebay.com<br>DNS Name: apacshipping.ebay.com.hk<br>DNS Name: anywhere.ebay.pl<br>DNS Name: anywhere.ebay.nl<br>DNS Name: anywhere.ebay.it<br>DNS Name: anywhere.ebay.in<br>DNS Name: anywhere.ebay.ie<br>DNS Name: anywhere.ebay.es<br>DNS Name: anywhere.ebay.com.sg<br>DNS Name: anywhere.ebay.com.hk<br>DNS Name: anywhere.ebay.com<br>DNS Name: anywhere.ebay.ch<br>DNS Name: anywhere.ebay.ca<br>DNS Name: anywhere.ebay.be<br>DNS Name: anywhere.ebay.at<br>DNS Name: akamai.ebaycdn.net<br>DNS Name: akamai-static.ebaycdn.net<br>DNS Name: include.ebaystatic.com<br>DNS Name: hcpt.ebay.com | | | |

| | | | | |
|---|---|---|---|---|
| **Certificate category: Single domain, wildcard o Multi-domain SAN/UCC cert?** | Multi domain SAN | Single domain | Single domain | |
| **Public Key Info like key algo, key length, public exponent (e) in case of RSA** | Algo : RSA<br>Exponent : 65537 Key size : 2048 | Algorithm RSA<br>Key Size 2048<br>Exponent 65537 | Algorithm RSA<br>Key Size 2048<br>Exponent 65537 | We can observe that the exponent is backwards and forwards compatible with current hardware and software. |
| **Public key or modulus (n) in case of RSA** | Modulus :<br>BE:6D:2C:3D:E9:E9:42:12:85:0E:B2:07:70:7E:26:10:F3:CA:4A:A0:5C:AA:F4:8E:DC:B8:94:13:C9:68:BB:1A:E9:6B:02:DD:F0:85:C7:6B:4D:9E:68:A4:90:73:7D:D3:C6:CC:F6:68:1B:05:33:1A:11:2C:36:CF:AE:01:CF:34:C9:35:D6:C4:81:6B:3C:28:E1:90:1B:37:AE:E7:A5:D0:8D:53:81:3D:1D:17:3D:AFD5:33:C8:2B:26:05:DA:80:5C:36:53:4C:49:16:95:EE:0E:CE:62:44:C8:F8:51:53:78:97:D3:19:68:0C:FB:CF:07:44:54:54:B7:5A:2D:A4:D2:AE:17:90:7E:78:24:2A:BA:6E:A4:2C:91:EE:27:F3:39:77:2B:7B:49:4E:B0:94:4B:16:48:DE:7C:6A:48:DC:97:D9:B5:09:48:0C:82:EE:3A:96:C7:B1:14:6A:34:6A:8D:57:E6:C9:D1:A3:AA:03:E1:2C:58:F8:2E:55:9A:37:B5:ED:52:3B:E7:B6:24:C1:0D:9C:D3:D8:55:67:EB:DD:E2:94:56:3F:6D:AC:EB:0A:BA:5B:93:D2:EA:FA:4C:4D:1D:8C:75:55:23:C5:17:61:DF:4E:EE:C9:C7:AE:F1:54:9A:F0:D0:83:3D:55:36:2B:29:A7:17:7C:98:21:B0:DB | Modulus :<br>C1:4B:B3:65:47:70:BC:DD:4F:58:DB:EC:9C:ED:C3:66:E5:1F:31:13:54:AD:4A:66:46:1F:2C:0A:EC:64:07:E5:2E:DC:DC:B9:0A:20:ED:DF:E3:C4:D0:9E:9A:A9:7A:1D:82:88:E5:11:56:DB:1E:9F:58:C2:51:E7:2C:34:0D:2E:D2:92:E1:56:CB:F1:79:5F:B3:BBB:87:CA:25:03:7B:9A:52:41:66:10:60:4F:57:13:49:F0:E8:37:67:83:DF:E7:D3:4B:67:4C:22:51:A6:DF:0E:99:10:ED:57:51:74:26:E2:7D:C7:CA:62:2E:13:1B:7F:23:88:25:53:6F:C1:34:58:00:8B:84:FF:F8:BE:A7:58:49:22:7B:96:AD:A2:88:9B:15:BC:A0:7C:DF:E9:51:A8:D5:B0:ED:37:E2:36:B4:82:4B:62:B5:49:9A:EC:C7:67:D6:E3:3E:F5:E3:D6:12:5E:44:F1:BF:71:42:7D:58:84:03:80:B1:81:01:FA:F9:CA:32:BB:B4:8E:27:87:27:C5:2B:74:D4:A8:D6:97:DE:C3:64:F9:CA:CE:53:A2:56:BC:78:17:8E:49:03:29:AE:FB:49:4F:A4:15:B9:CE:F2:5C:19:57:6D:6B:79:A7:2B:A2:27:20:13:B5:D0:3D:40:D3:21:30:07:93:EA:99:F5 | Modulus :<br>E2:3B:E1:11:72:DE:A8:A4:D3:A3:57:AA:50:A2:8F:0B:77:90:C9:A2:A5:EE:12:CE:96:5B:01:09:20:CC:01:93:A7:4E:30:B7:53:F7:43:C4:69:00:57:9D:E2:8D:22:DD:87:06:40:00:81:09:CE:CE:1B:83:BF:DF:CD:3B:71:46:E2:D6:66:C7:05:B3:76:27:16:8F:7B:9E:1E:95:7D:EE:B7:48:A3:08:DA:D6:AF:7A:0C:39:06:65:7F:4A:5D:1F:BC:17:F8:AB:BE:EE:28:D7:74:7F:7A:78:99:59:85:68:6E:5C:23:32:4B:BF:4E:C0:E8:5A:6D:E3:70:BF:77:10:BF:FC:01:F6:85:D9:A8:44:10:58:32:A9:75:18:D5:D1:A2:BE:47:E2:27:6A:F4:9A:33:F8:49:08:60:8B:D4:5F:B4:3A:84:BF:A1:AA:4A:4C:7D:3E:CF:4F:5F:6C:76:5E:A0:4B:37:91:9E:DC:22:E6:6D:CE:14:1A:8E:6A:CB:FE:CD:B3:14:64:17:C7:5B:29:9E:32:BF:F2:EE:FA:D3:0B:42:D4:AB:B7:41:32:DA:0C:D4:EF: | |

| | | | | |
|---|---|---|---|---|
| | | | F8:81:D5:BB:8D:58:3F:B5:1B:E8:49:28:A2:70:DA:31:04:DD:F7:B2:16:F2:4C:0A:4E:07:A8:ED:4A:3D:5E:B5:7F:A3:90:C3:AF:27 | |
| **Key usages; how do they vary in the chain?** | Digital Signature, Key Encipherment<br>Extended : Server Authentication, Client Authentication | Digital Signature, Certificate Signing, CRL Signing<br>Extended : Server Authentication, Client Authentication | Digital Signature, Certificate Signing, CRL Signing | We can observe that the CRL signing and certificate signing authorities are given only to root and intermediate certificates meaning only they have the ability to verify other certificates. |
| **Basic constraints, how do they vary in the chain?** | Critical<br>Is not a Certification Authority | Critical<br>Is a Certification Authority<br>Maximum number of intermediate CAs: 0 | Critical<br>Is a Certification Authority<br>Maximum number of intermediate CAs: unlimited | We can observe that intermediate and root certificates are approved as CA. And clearly the root CA can have as many CAs as it wants below it while the intermediate one cannot have any CA under it. |
| **Name constraints (if any), how are these useful?** | - | - | - | |
| **Size of the certificate** | 3kb | 1.2kb | 947 bytes | Clearly , the size of the certificates is |

| | | | | |
|---|---|---|---|---|
| | | | | decreasing from end user to root certificate. |
| **Any other parameters that you found interesting?** | SHA2 fingerprint - 59 3A 89 6A BA 5F 3F 4C B2 05 ED E6 4E F0 3E 2D<br>7F E9 E0 47 60 B0 39 11 55 A7 86 CC 6F 6F 96 32<br><br>SHA1 fingerprint - 49 90 29 B2 46 80 64 C4 75 CF 4F E8 65 74 67 08 47 04 A2 59 | SHA2 fingerprint - 52 27 4C 57 CE 4D EE 3B 49 DB 7A 7F F7 08 C0 40<br>F7 71 89 8B 3B E8 87 25 A8 6F B4 43 01 82 FE 14<br><br>SHA1 fingerprint - 1C 58 A3 A8 51 8E 87 59 BF 07 5B 76 B7 50 D4 F2<br>DF 26 4F CD | SHA2 fingerprint - 43 48 A0 E9 44 4C 78 CB 26 5E 05 8D 5E 89 44 B4 D8 4F 96 62 BD 26 DB 25 7F 89 34 A4 43 C7 01 61<br><br>SHA1 fingerprint - A8 98 5D 3A 65 E5 E5 C4 B2 D7 D6 6D 40 C6 DD 2F<br>B1 9C 54 36 | To guarantee that the certificate is not tampered with, fingerprints are utilised as an additional level of security. |

# QUERRIES

1. **Which certificate type (DV/OV/IV/EV) is more trustable and expensive?**

   The extended validation certificate is the highest ranking ( in terms of trust ) and is the most expensive one as in essence, the issuer must do a background check on the requestor in order to confirm the requestor's identity and the existence of the requestor's operations.

2. **What is the role of the Subject Alternative Name (SAN) field in X.509 certificate?**

   Additional host names (sites, IP addresses, common names, and so on) can be secured by a single SSL Certificate using the Subject Alternative Name field. The most common reason y SANs are used is that few sites have a CNAME or alias.

   For example :

   www.abc.com

   abc.com

   If a user accesses https://abc.com but the certificate is issued to www.abc.com then the

user will receive a name mismatch warning. Instead if a SAN cert is used then both domains can be specified as valid and the user will not receive a warning.

3. **Why are key usages and basic constraints different for root, intermediate and end certificates?**
    a. The key usage extension defines the purpose of the key contained in the certificate. The purposes of the certificates vary along the chain from the root to the end user certificate. The key that every certificate owns has its own restrictions on what purposes it can serve. For example, only the root CA has the usage “encipherment” which means no other unauthorized entity can read the data. These restrictions on the certificates’ key usages helps the data stay more secure.
    b. Constraints are used to limit the number of certificate authority in your chain that we don’t trust. They take the form of rules imposed on the certificate authority, which allow or prohibit the CA from issuing certificates based on the criteria specified in the request.Basic Constraints limit the path length for a certificate chain. This type of constraint limits the number of CAs that exist below the CA (depth) where the constraint is defined.

4. **What is the difference between Signature value and Thumbprint of a digital certificate?**
    The basic difference between signatures and thumbprints is that signatures are used for security purposes whereas thumbprints are just used for reference.In reality, the thumbprint isn't part of the certificate at all. It's computed and shown for convenience. Thumbprint is solely used in the store to locate a needed certificate. The digital certificate's signature is used to validate the signature of the certificate.

5. **Why do RSA key lengths increase over the years? Why is ECDSA being preferred over RSA now-a-days?**
    The main reason behind increasing the lengths of RSA keys is to increase security. The time it takes to encrypt and decode data grows in lockstep with the size of the key. The suggested approach divides the file into blocks to speed up the encryption and decryption procedures and increases the algorithm's strength by increasing the key size. Hence longer the key length, the stronger the encryption would be and hence more secure.

    ECDSA offers the same level of security as RSA, but with significantly shorter key lengths. As a result, brute-forcing attacks on ECDSA will take significantly longer for longer keys. Another significant advantage of ECDSA over RSA is its superior speed and scalability. ECC uses less network and CPU power since it provides excellent security with lower key lengths. This is especially useful for devices with limited storage

and processing capabilities.

6. **What are pros and cons of pre-loading root and intermediate certificates in the root stores of browsers and OSes?**
   The advantage of preloading the root and intermediate certificates to the root store will not render so many errors of the users. When the correct intermediate CA certificates are not given, which is one of the most typically encountered difficulties when implementing TLS security, they will not see an error page.
   The disadvantage of preloading the certificates might expose the attackers to taint the certificates while they are getting pinned or in any other process. Hence for this reason , uploading certificates at the development time or in the first encounter is preferred over preloading them into root stores of Os/ browsers. And there are cases where it might happen that they issue fake certificates to some websites and we don't have an option but to trust them.

7. **Why are root CAs kept offline?**
   Root CA being at the top in the hierarchy, makes it an attractive target for potential attackers. If it were supposed to be replaced then it would impact the whole PKI as we would need to replace the root CA in every end entities' trust store which is a very time consuming process. Because of its importance and the potential for disruption if it is hacked, the root CA should be kept offline (not connected to the internet or deactivated), unavailable for use, and not enabling new certificates to be issued.

8. **List out names of OS/Browser/Company whose root stores pre-populated with Root and Intermediate CA certificates of the website #N?**

   Web Browsers
   AOL 5+ ,Boxee, Camino 1.0+,Chrome,Firefox 1.0+, Grandstream, Internet Explorer 5+, Konqueror 2.2.1+ ,Maxthon ,Microsoft Edge, Mozilla 7.0+, Netscape 4.5+,Opera 5+, Safari ,Sony Playstation, Nintendo Wii
   Operating Systems
   Access, Android, BlackBerry OS, Brew, Chrome OS, Debian, HP-UX, iOS, Mac OS X, Meego, Palm OS, Palm WebOS, SUSE Linux, Ubuntu, Windows (all versions)
   Server Platforms
   Apache, BEA WebLogic, C2Net Stronghold, Citrix, Cobalt RaQ3x/4x/XTR ,Courier, IMAP, cPanel / Web Host Manager, Ensim Control Panel, Hsphere, IBM HTTP Server

iPlanet Server, Java Web Server (Javasoft / Sun), Lighttpd, Lotus Domino, Microsoft IIS, Microsoft SQL Server 2005, Netscape Enterprise Server, Nginx, Novell ConsoleOne, Novell Web Server, OpenLDAP, Oracle HTTP Server, Plesk, Tomcat etc.

# PART B

1. **You have received the digital certificate of the website #N over email. How do you verify whether the certificate is valid without using any online tools or browsers? Write a psuedo-code of your verifier function named myCertChecker( ) and explain how it works by picking the entire chain of trust of an end-user cert (of the website #N) in PART-A of this assignment.**

```
int myCertChecker(Certificate * cert)
{
	valid=0;
	if(root certificate is reached)
		return valid;
	if(cert.validity >=startdate and cert.and cert.validity <= enddate)
	{
		if(cert.domainname= = dn_web || cert.san== dn_web)
		{
			if(cert not in CRL of browser)
			{
				if(cert.hash == digisignature)
				{
					valid =1;
					return valid;
				}
			}
	}
	valid=0;
	Return valid;

};
```

The explanation of the function is as follows :

- The first process of check will be to verify if the certificate lies in its expiry period by checking the start date and end date of the validity period . If it is expired, it is not feasible to trust.

- The next step would be to check its domain name or the alternative names if it is matching with that of the web server.
- Now check if the certificate lies in the CRL of the browser because if it does it is again not feasible to trust.
- Next step is the important one where we check if the hash that we obtain after applying the hashing algorithm of that certificate is the same as the digital signature of it. If not then it is compromised and not valid anymore.
- Repeat the same whole process in recursion until we reach the top most certificate in the hierarchy i.e the root certificate. If none of the cases fail till then, then the certificate is verified right and we return 1 as the result else 0.

2. **Consider the scenario in which evil Trudy has used the digital certificate of the website (Bob) named abc.com to launch her own web server with the domain name, xyz.com. Does your function myCertChecker( ) returns valid or invalid for this when someone like Alice tries to access Trudy's website xyz.com from a browser like Chrome/Edge/Firefox?**

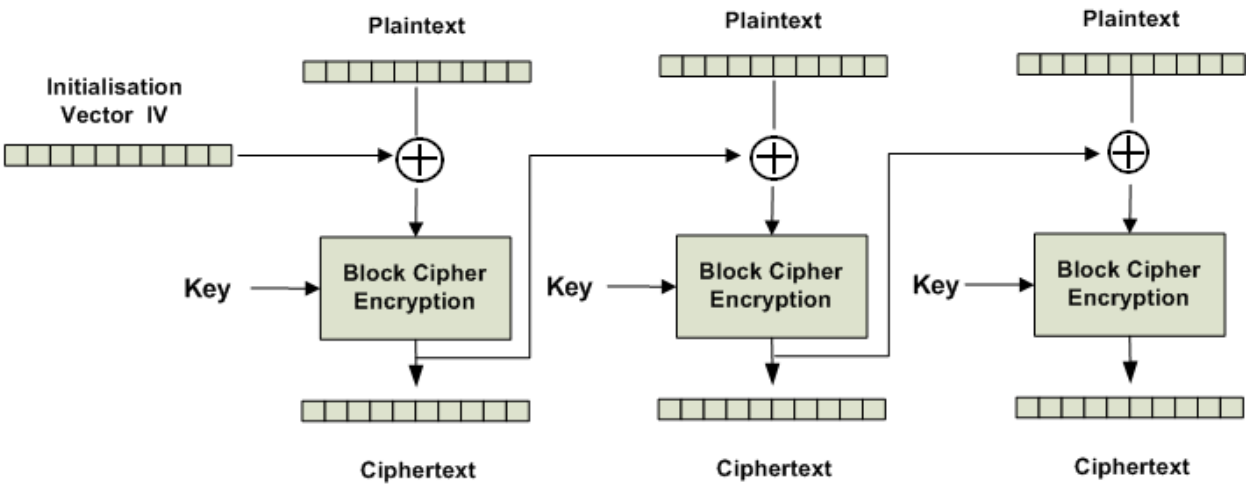
   If Trudy has launched her own web server with domain name xyz.com and Alice tries to access it , myCertChecker function would return 0 as in invalid. Since my function is checking if the domain name of the certificate matches with the CN of the server and this where the if condition return in the function fails and hence it will be treated as invalid.

3. **Consider the scenario in which evil Trudy has used the digital certificate of Bob's website abc.com to launch her own web server with the domain name, xyz.com. When a web client (Alice) tries to connect with Bob's website abc.com by sending a DNS query, Trudy responds with her IP address by launching MITM attack (What is DNS cache poisoning? | DNS spoofing | Cloudflare) Does your function myCertChecker( ) returns valid or invalid for this and what are the consequences? What kind of attacks can Trudy launch in this scenario?**
   The function being implemented here tries to stop the man in the middle attack. Since there is a step dedicated to check if the hash of the certificate and the digital signature of it are the same or not, if trudy responds with her own IP address then the digital signature won't match and hence the function returns 0 as in invalid.

# 7-zip

*7-zip uses AES-256 ( symmetric encryption algorithm) in CBC mode.*

*Working of CBC*




It makes use of a 256-bit encryption key. It is created with the aid of the PBKDF key derivation function (Password based key derivation function). The way it works is that it accepts the user's password and creates a key that is then utilised by the AES algorithm. To produce a cipher key from a password, it goes through several rounds. To reduce attack vulnerabilities, the key derivation function additionally employs a 512-bit salt. The process is usually completed in $2^{18}$ iterations.

Yes, the length of the password plays an important role in preventing the brute force attacks on encrypted files while decrypting them. Since it is directly proportional to the time ,computing resources and complexity involved in cracking the password by brute force, it might take years to crack it if the attacker tries all possible combinations of that lengthy password.

**PLAGIARISM STATEMENT**

*I certify that this assignment/report is my own work, based on my personal study and/or research and that I have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication. I also certify that this assignment/report has not previously been submitted for assessment in any other course, except where specific permission has been granted from all course instructors involved, or at any other time in this course, and that I have not copied in part or whole or otherwise plagiarised the work of other students and/or persons. I pledge to uphold the principles of honesty and responsibility at CSE@IITH. In addition, I understand my responsibility to report honour violations by other students if I become aware of it.*

Name: Nisha M
Date: 30/1/22
Signature: Nisha M