Network Security Assignment - ABCs of Digital Certificates

cs23mtech14018 - Yash Shukla

PART - A

Field Name	Subject (CN) of certificate holder (website) = nytimes.com	Subject (CN) of certificate holder (intermediate)	Subject (CN) of certificate holder (root) = DigiCert Global Root CA	Remarks/observations
Issuer	Thawte RSA CA 2018	DigiCert Global Root CA	DigiCert Global Root CA	NA
Version No.	Version 3	Version 3	Version 3	All are using Version 3
Signature Algo	PKCS #1 SHA-256 With RSA Encryption	PKCS #1 SHA-256 With RSA Encryption	PKCS #1 SHA-1 With RSA Encryption	This information reveals the cryptographic algorithm employed to create the unique signature for the certificate by hashing and encrypting its contents.
Size of digest that is signed to generate Cert Sign	256 bits	256 bits	160 bits	Hash Length
Size of Cert Signature	2048 bits	2048 bits	2048 bits	Implies strength and efficiency
Validity period	22/03/2023 to 22/04/2024	06/11/2017 to 06/11/2027	10/11/2006 to 10/11/2031	More validity from top to bottom

Is Subject field (CN), FQDN?	**Nytimes.com , yes**	**CN = DigiCert Global Root CA , no**	**CN = Thawte RSA CA 2018 , No**	**FQDN stands for Fully Qualified Domain Name Allows us to see the location**
Certificate type: DV, IV, OV or EV? Tell also how you are able to determine the type!	**DV**	**OV**	**EV**	**Provides valuable insights into validation adn trustworthiness and intended use of certificate**
Subject Alternative Name(s) (SAN/UCC), if any	**DNS Name: www.homedelivery.nytimes.com DNS Name: *.api.dev.nytimes.com DNS Name: *.api.nytimes.com**	**NA**	**NA**	**Domain Names**
Certificate category: Single domain, wildcard, Multi-domain SAN/UCC cert?	**WildCard**	**Single Domain Certificate**	**Single Domain Cert**	**The end user is having many SANs whereas the intermediate and the root has a single domain certificate.**
Public Key Info like key algo, key length, public exponent (e) in case of RSA	**RSA , 2048 bits, Public Exponent (17 bits): 01 00 01**	**RSA , 2048 Bits, Public Exponent (17 bits): 01 00 01**	**RSA ,2048 Bits , Public Exponent (17 bits): 01 00 01**	**Information about the public key algorithm with its parameters**
Public key or modulus (n) in case of RSA	**E2 3B E1 11 72 DE A8 A4 D3 A3 57 AA 50 A2 8F 0B 77 90 C9 A2**	**CA 08 5E E5 53 8A 97 1C 1E 43 2F B6 8A A7 56 E9 8B 84 43 A8**	**B7 C8 EC BD 69 FE 11 63 D1 CF BF 82 3D 07 26 0E 89 F4 0D AC**	**The Public key**

	A5 EE 12 CE	AC 9D 7A 55	19 3A 1C 20	
	96 5B 01 09 20	82 7A 14 4B 86	2F A5 75 61 19	
	CC 01 93	B7 2F 8F	1A 0A B2	
	A7 4E 30 B7	52 9F 1C CA	07 65 8A A6 62	
	53 F7 43 C4 69	B1 20 5B 6F	A8 24 EF 0B	
	00 57 9D E2	BA 22 DD A6	8A AA E7 65	
	8D 22 DD	9C 2D 78 DA	3D F7 50	
	87 06 40 00 81	E9 06 08 4E	00 4C 55 56 18	
	09 CE CE 1B	BE 13 A6 EB	AD EF DC 59	
	83 BF DF CD	CB BB 3E B9	20 9B 7A 57 93	
	3B 71 46	05 0C 3E 4A	89 C2	
	E2 D6 66 C7	E1 F0 32 1F 13	77 1C 03 8A	
	05 B3 76 27 16	4E F5 06 C5 47	6C A5 BC 01	
	8F 7B 9E 1E	73 89 3E 80 A3	F3 48 46 0C 7F	
	95 7D EE	8B	4F 3C 00	
	B7 48 A3 08	F1 01 24 9B	3A 9C A0 47	
	DA D6 AF 7A	A3 99 66 92 6B	4F A2 E2 1B	
	0C 39 06 65 7F	68 AD 0D 2D	89 50 95 3D F7	
	4A 5D 1F	B4 CD 72	15 81 D2	
	BC 17 F8 AB	A2 F4 F9 38	33 FB 98 3B	
	BE EE 28 D7	5A 65 A6 B4	E7 7D E7 01	
	74 7F 7A 78 99	8C 53 C1 08	6F EA 7E 29	
	59 85 68	1A 84 F8 FD	15 EB 25 D6	
	6E 5C 23 32	2E F3 11 75 6E	0E 26 64 08 3F	
	4B BF 4E C0	DC 6A 31 29	25 7A AC C5	
	E8 5A 6D E3	AC 0D 87 CC	E4 FF 83 EA	
	70 BF 77 10	93 60 78	E4 B9 C7	
	BF FC 01 F6	DF 25 BA 26	54 E9 0F 8A 78	
	85 D9 A8 44 10	59 91 C6 83 52	34 9E 01 40 E4	
	58 32 A9 75 18	35 A6 CA 9C	A2 4D 52 FF	
	D5 D1	B8 28 1A	55 87	
	A2 BE 47 E2	CE D7 1C EE	35 00 80 D7 19	
	27 6A F4 9A	14 BF 76 5C	E6 61 D3 39 03	
	33 F8 49 08 60	65 AB 38 1E	B0 B1 5F F0	
	8B D4 5F	79 E9 7C CC	4B 39	
	B4 3A 84 BF	49 23 26 A2 52	65 BF 6B 51	
	A1 AA 4A 4C	50 66 D0 59 61	FE BC AA 25	
	7D 3E CF 4F	FF A0 FE 9A	14 98 87 A3 C7	
	5F 6C 76 5E	4C 0C	46 82 6A	
	A0 4B 37 91	9F F9 E8 8E	6A 18 4C B9	
	9E DC 22 E6	DE 09 8B B8	F4 2D 4E 07	
	6D CE 14 1A	15 C1 A4 08	B5 E4 C5 D5	
	8E 6A CB FE	4C 26 9C 7B	EC CE 44 14	
	CD B3 14 64	06 DB FD 8A	3F 81 95 5C E3	
	17 C7 5B 29	74 5B 58 7E	77 6C E3 A5	
	9E 32 BF F2	CD 63 A4 91	32 C6 A8 34	
	EE FA D3 0B	2F 45 F0 7A	D3 C5 E7	
	42 D4 AB B7	3C 94 0B 8A	3A A0 FE 9B	
	41 32 DA 0C	7C B2 05 A9	F7 63 5C E1 07	
	D4 EF F8 81	67 93 9F 68 E5	36 6F 52 29 3F	

	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	95 63 60 D8 58 95 5F E0 55 EF 93 A7 11 3B 7C E6 92 D8 66 44 E0 AB BD A7 8F CD A4 85 78 41 24 54 E7 D8 03	8E 59 9E 0B E7 E6 71 51 67 A4 C3 8E CD 40 73 ED 88 D2 35 84 94 90 7A E7 34 72 89 2C 46 2A 5C ED 5D 23	
Key usages; how do they vary in the chain, mention in the remarks?	Signing Key Encipherment	Signing Certificate Signer CRL Signer	Signing Certificate Signer CRL Signer	Key difference is just Certificate Signer adn Signer.
Basic constraints, how do they vary in the chain?	Not Critical Is not a Certification Authority	Subject Type=CA Path Length Constraint=0	Subject Type=CA Path Length Constraint=None	CA: No for end user cannot provide certificate for others
Size of the certificate	2.84 KB	1.62 KB	1.32 KB	NA
URI of CRL	URI: http://cdp.thawte.com/ThawteRSACA2018.crl	http://crl3.digicert.com/DigiCertGlobalRootCA.crl	NA	r location where the Certificate Revocation List (CRL) associated with the certificate can be found
URI of OCSP Responder	URI: http://status.thawte.com	URI: http://ocsp.digicert.com	NA	location where the OCSP responder service associated

				with the certificate
Any other parameters that you found interesting?	**NA**	**NA**	**NA**	**NA**

Answer the following queries after filling out the above table:

Q1. Which certificate type (DV/OV/IV/EV) is more trustable, secure, and expensive?

Answer 1) : EV certificates are the most trusted and have high security but they are expensive.

Q2. What is the role of the Subject Alternative Name (SAN) field in X.509 certificates?

Answer 2) : they just provide more flexibility and security and compatibility and secure the diverse environments efficiently while considering limitations.

Q3. Why are key usages and basic constraints different for root, intermediate and end certificates? What could go wrong if all of them have the same values?

Answer 3) :The root, intermediate, and end certificates have distinct key usages and essential limits based on the hierarchical structure and security assurance. Intermediates are limited in depth, root certificates cannot sign other certificates that further certify each other, and allowing the same values could compromise the PKI's trust model and lead to misuse, interoperability issues, hierarchy collapse, and security vulnerabilities.

Q4. What is the difference between Signature value and Thumbprint aka Fingerprint of a digital certificate?

Answer 3) : The CA creates signature value, a cryptographic value, certify the contents of the certificate and guarantee its validity and integrity. confirms the authenticity of the certificate.
On the other hand, the fingerprint or thumbprint is a condensed version of the certificate that is produced through the use of a hash function. It functions as a distinct identification that can be compared and verified. The thumbprint helps with identification and comparison.

Q5. Why do RSA key lengths increase over the years? Why is ECDSA being preferred over RSA now-a-days?

Answer 5) : RSA Key lengths increase over the years because To keep security against advances in computing power and cryptographic assaults, and ECDSA is preferred Because it requires shorter key lengths for equal security levels than RSA, it is preferable because it facilitates faster cryptographic operations and lowers computing cost and ECDSA is a more effective option for contemporary cryptographic applications since it also performs better in limited settings.

Q6. What are pros and cons of pre-loading root and intermediate certificates in the root stores of browsers and OSes?

Answer 6) : pro is pre-loading root and intermediate certificates in root stores improves security. This expedites the SSL/TLS handshake procedure and avoids security alerts. Cons is the root store larger and more complicated, which could make maintenance more difficult and update more slowly. Furthermore, it centralises judgements about trust, which might be dangerous if a compromised certificate is on the pre-loaded list.

Q7. Why are root CAs kept offline? How do they issue certificates to intermediate CAs?

Answer 7) : To reduce the possibility of compromise, root CAs are maintained offline Because to reduce the possibility of compromise.and they are issued through safe offline methods like hardware security modules (HSMs) or secure signing servers, they provide certificates to intermediate CAs. This lessens the possibility of unauthorised certificate issuing by guaranteeing the extreme security of the root CAs' private keys.

Q8. Why are root and intermediate certificates of new CAs cross-signed by the legacy CAs? Answer this by taking Let’s Encrypt and its parent organization as root and intermediate CAs

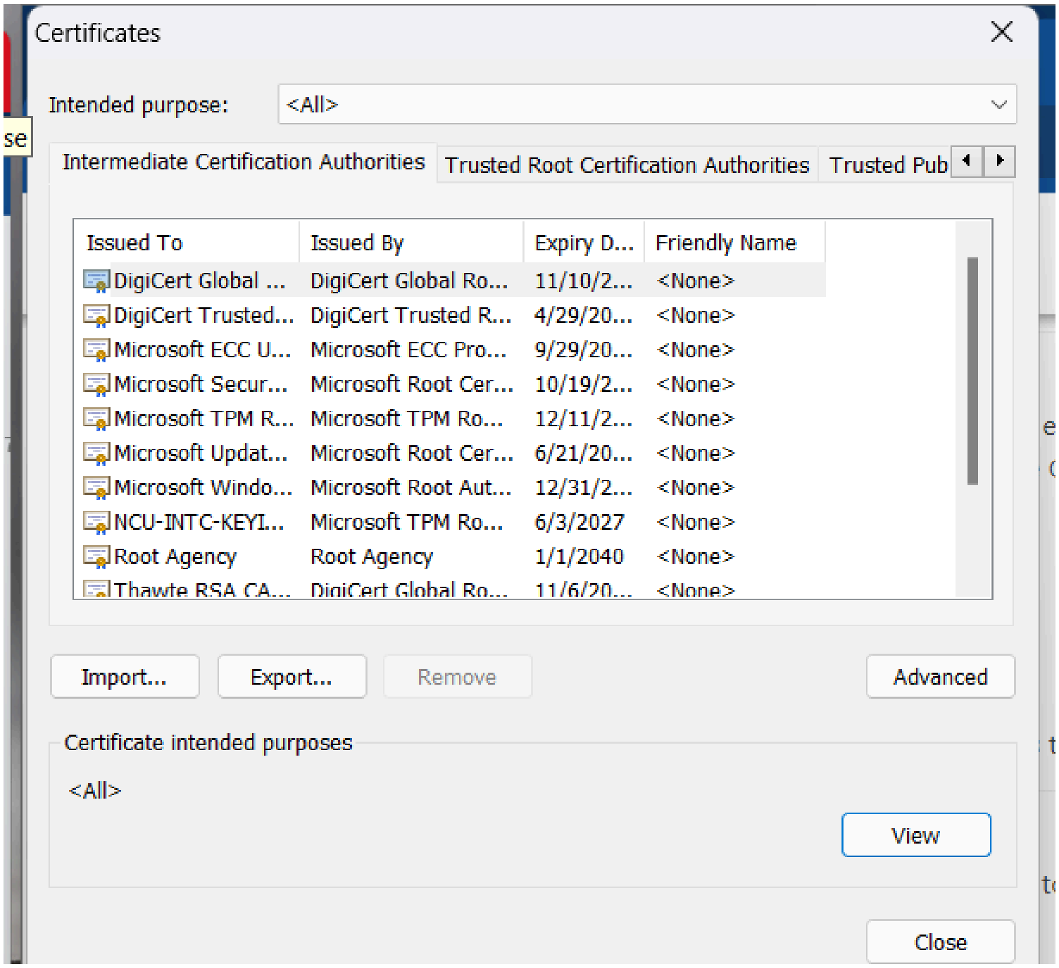
Answer 8) : In order to quickly build confidence with clients and systems that may not yet be aware of the new CA, traditional CAs cross-sign the root and intermediate certificates of new CAs, such Let's Encrypt. The Internet Security Research Group (ISRG), Let's Encrypt's parent organisation, cross-signs its intermediate certificates, and the IdenTrust root CA does the same.

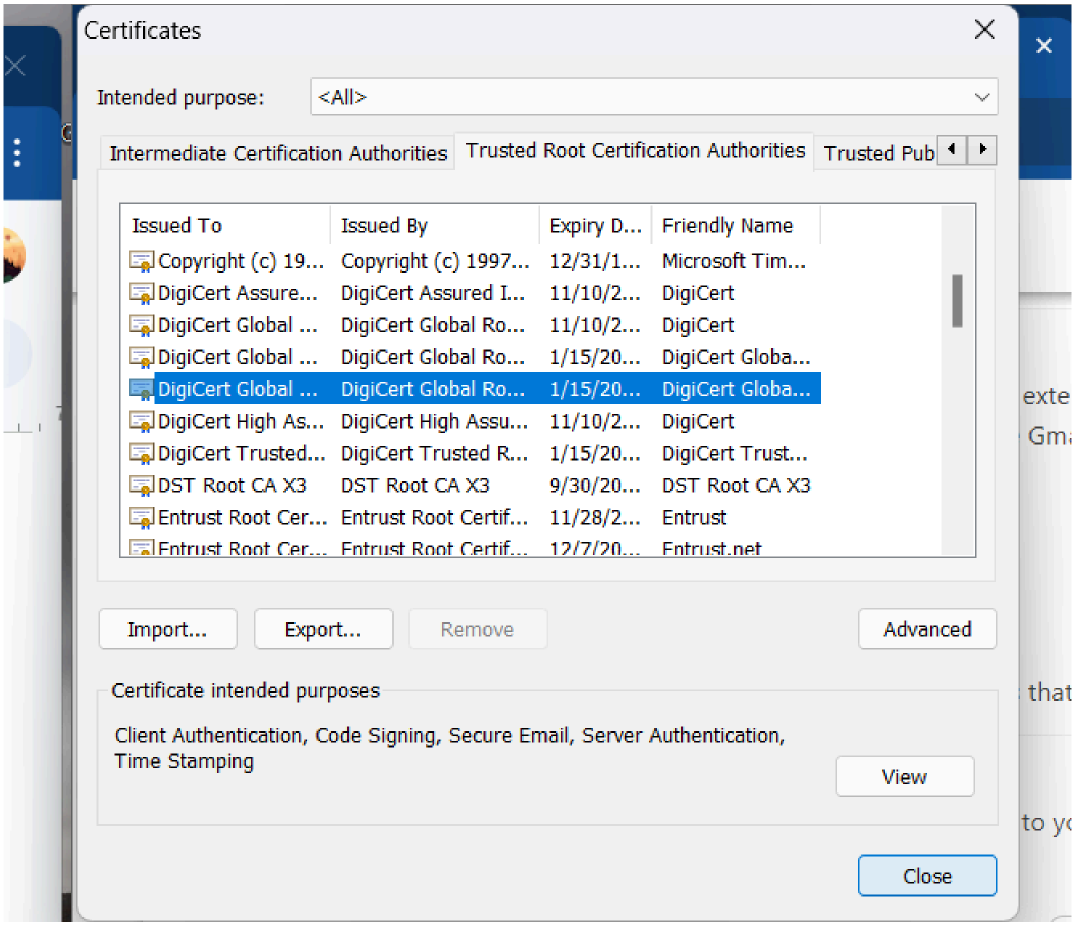
Q9. What challenge is posed to the certificate seekers (Alice) by Let’s Encrypt CA before issuing wildcard certificates? How does Alice respond to it so that she passes it?

Answer 9) : To Answer this question before providing wildcard certificates, Let's Encrypt challenges certificate seekers like Alice to validate their domains. By completing task, such adding a TXT record to the domain's DNS or putting a file with a specific content on her web server, and Alice must prove that she is the owner of the domain.so in order to successfully complete the challenge, Alice must demonstrate her control over the domain by responding to the validation request by doing the designated action within the allotted time.

Q10. List out names of OS/Browser/Company whose root stores were pre-populated with Root and Intermediate CA certificates of the website #N?

Answer 10) :


Certificates
Intended purpose: <All>
Intermediate Certification Authorities
Trusted Root Certification Authorities
Trusted Pub
Issued To | Issued By | Expiry D... | Friendly Name
DigiCert Global ... | DigiCert Global Ro... | 11/10/2... | <None>
DigiCert Trusted... | DigiCert Trusted R... | 4/29/20... | <None>
Microsoft ECC U... | Microsoft ECC Pro... | 9/29/20... | <None>
Microsoft Secur... | Microsoft Root Cer... | 10/19/2... | <None>
Microsoft TPM R... | Microsoft TPM Ro... | 12/11/2... | <None>
Microsoft Updat... | Microsoft Root Cer... | 6/21/20... | <None>
Microsoft Windo... | Microsoft Root Aut... | 12/31/2... | <None>
NCU-INTC-KEYI... | Microsoft TPM Ro... | 6/3/2027 | <None>
Root Agency | Root Agency | 1/1/2040 | <None>
Thawte RSA CA... | DigiCert Global Ro... | 11/6/20... | <None>
Import...
Export...
Remove
Advanced
Certificate intended purposes
<All>
View
Close



PART B

Q1 . A browser X has received the digital certificate of the website #N over a TLS connection. How does it verify whether the certificate is valid? Write a psuedo-code of browser X's verifier function named myCertVerifier( ) 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.

Answer 1) : A Browser has Received the Digital Certificate of the Website nytimes.com over a TLS Connection.

* To Verify whether the Certificate is valid or not : this are the potential reasons to declare invalid :

- Expired Certificate
- Invalid Signature
- Untrusted Certificate Authority (CA)
- Certificate Chain Issues (Incomplete Chain,Invalid Chain,Self-Signed Certificate)

- Domain Name Mismatch
- Revoked Certificate
- Critical Extensions Handling
- Fingerprint or Public Key Mismatch
- Revocation Status Check Failure

** Examination flow : Start -> myCertVerifier()
Signature Verification ->
Check Certificate Expiration ->
Verify Issuer Trustworthiness ->
Extract Certificate Chain ->
Chain Verification ->
Additional Checks ->
Overall Decision (Valid/Invalid)

Pseudo Code mentioned here as respective to the flow.

```
{.......................................
function myCertVerifier(cert):
   if not check_certificate_signature(cert):
      return False
   if not check_certificate_expiration(cert):
      return False
   if not check_trusted_CA(cert):
      return False
   if not verify_certificate_chain(cert):
      return False
   if not additional_checks(cert):
      return False
   return True

function check_certificate_signature(cert):
   # Verify the cryptographic signature of the certificate
   # using the public key of the issuer
   return signature_verified

function check_certificate_expiration(cert):
   # Check if the certificate has not expired
   return not expired

function check_trusted_CA(cert):
```

```
    # Check if the issuer (CA) of the certificate is trusted
    return trusted_CA

function verify_certificate_chain(cert):
    # Verify the entire certificate chain
    chain = extract_certificate_chain(cert)
            for cert in chain:
                  if not check_certificate_signature(cert):
                     return False
                  if not check_certificate_expiration(cert):
                     return False
                  if not check_trusted_CA(cert):
                     return False
       # Optionally, perform additional checks here
           return True

function extract_certificate_chain(cert):
    # Function to extract the certificate chain from the server certificate
    # This may involve retrieving additional certificates from the server
    return certificate_chain

function additional_checks(cert):
    # Perform additional checks such as domain name verification,
    # revocation status check, extension critical flag check, etc.
    return additional_checks_passed
```

……………………………………………………….}

Q2.) Consider the scenario in which evil Trudy has used the domain validated (DV) digital certificate of the website (Bob) named Bob.com to launch her own web server with the domain name, **xyz.com**. Does your function myCertVerify( ) returns valid or invalid for this when someone like Alice (browser) tries to access Trudy's website xyz.com?

Answer 2) : DV certificates are primarily used to verify control of domain,rather than the identity behind the domain, they are susceptible to misuse in scenarios where a certificate issued for one domain is used for another domain, this will return invalid.

Q3.) Consider another scenario in which evil Trudy has used the digital certificate of Bob's website Bob.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 by sending a DNS query, Trudy responds with her IP address by DNS cache poisoning (What is DNS cache poisoning? | DNS spoofing | Cloudflare) Does your function myCertVerifier( ) returns valid or invalid for this and what are the consequences? What kind of attacks can Trudy launch in this scenario?

Answer 3) : My Function checks during the verification of the entire "Chain of Trust," the function will detect that the certificate presented by Trudy does not belong to the domain for which the DNS query was made (Bob.com), but instead, it's being used for xyz.com.Trudy can launch Phishing ,Man in the middle attacks ,Data Theft ,Session Hijacking.

About : 7-zip

7-Zip Encryption:

- 7-Zip uses passwords to encrypt files
- Passwords are converted into encryption keys using a SHA.
- The keys are used with symmetric encryption algorithm (like AES) to encrypt the file data.

Role of Password Length in Brute Force Attacks:

- Longer passwords make brute force attacks harder.
- Each additional character exponentially increases the search space for possible passwords.
- Strong, lengthy passwords greatly improve the security of encrypted files.

References:

1. https://crt.sh/
2. https://ahrefs.com/blog/most-visited-websites/
3. http://lapo.it/asn1js/#
4. http://phpseclib.sourceforge.net/x509/decoder.php
5. https://www.ssl.com/article/dv-ov-and-ev-certificates/
6. https://www.ccadb.org/
7. DV, OV, IV, and EV Certificates - SSL.com
8. 7-Zip (7-zip.org)

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 plagiarized 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 honor violations by other students if I become aware of it.

Name: Yash Shukla
Date: 06 / 02 / 2024
Signature: Yash Shukla