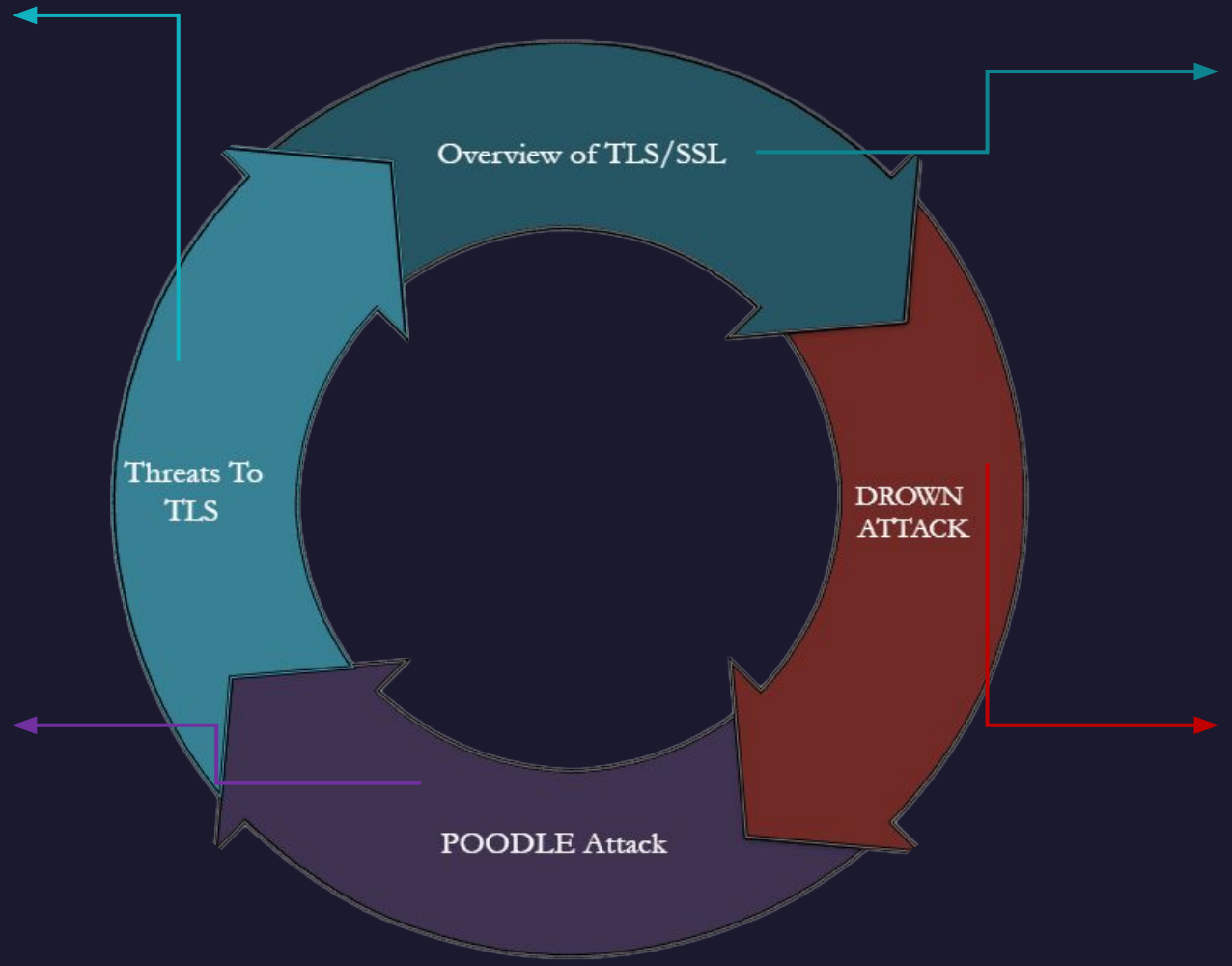
SSL/TLS Attacks

Presented By:

- Shayd Reeves (46444983)
- Lachlan Daly (44850647)
- Karthikeyan Venkatesan (46011910)

- Racoon Attack Requirements
- Future of Attacks on SSL/TLS



- What is SSL/TLS?
- A Brief History of SSL/TLS
- How SSL/TLS Protocol Works

- Downgrading attack
- POODLE TLS CBC
- Mitigations against POODLE

- Bleichanbachers' Attack
- DROWN Attack
- Mitigations against DROWN

What is SSL/TLS?

What does SSL/TLS aim to do?

- The primary goal of the TLS protocol is to provide privacy and data integrity between two communicating applications

What is the difference between SSL and TLS

- SSL is the predecessor to TLS
- Same protocol design, different crypto algorithms

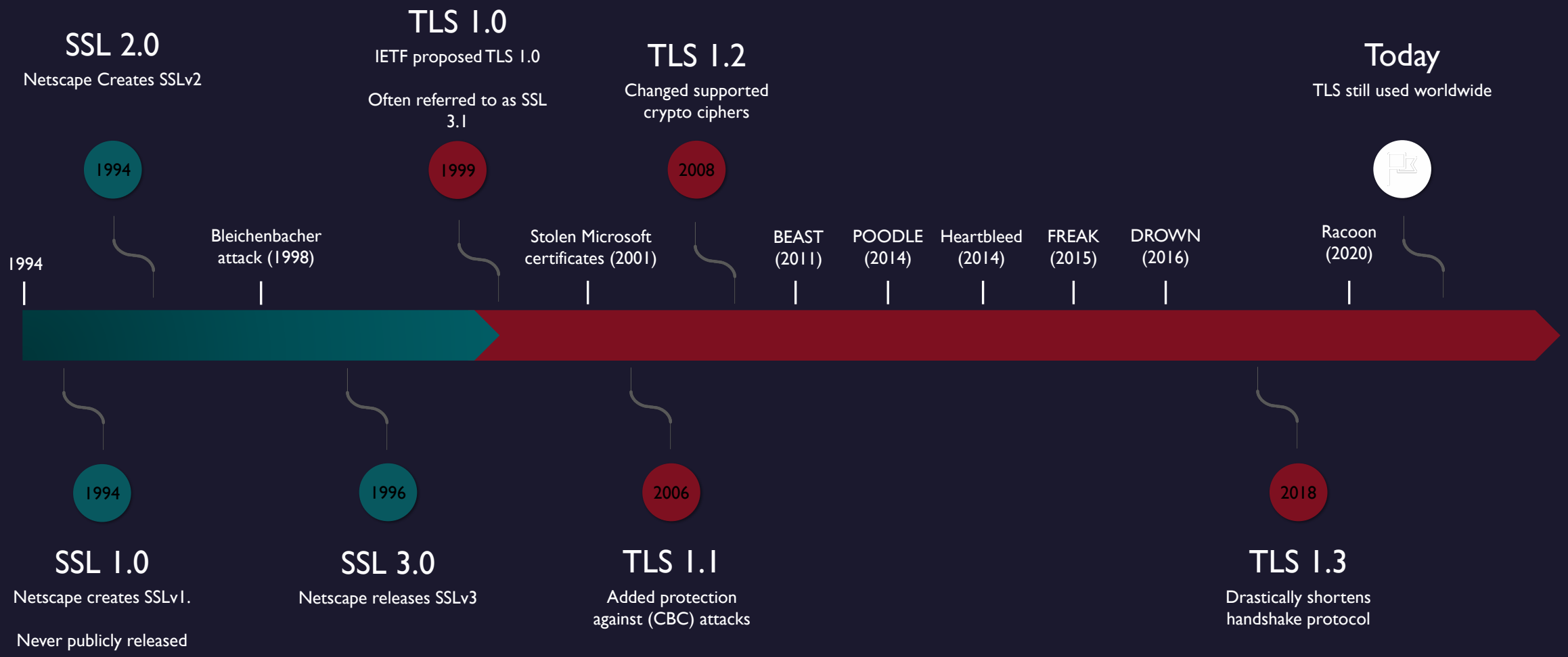
Where is TLS/SSL located?

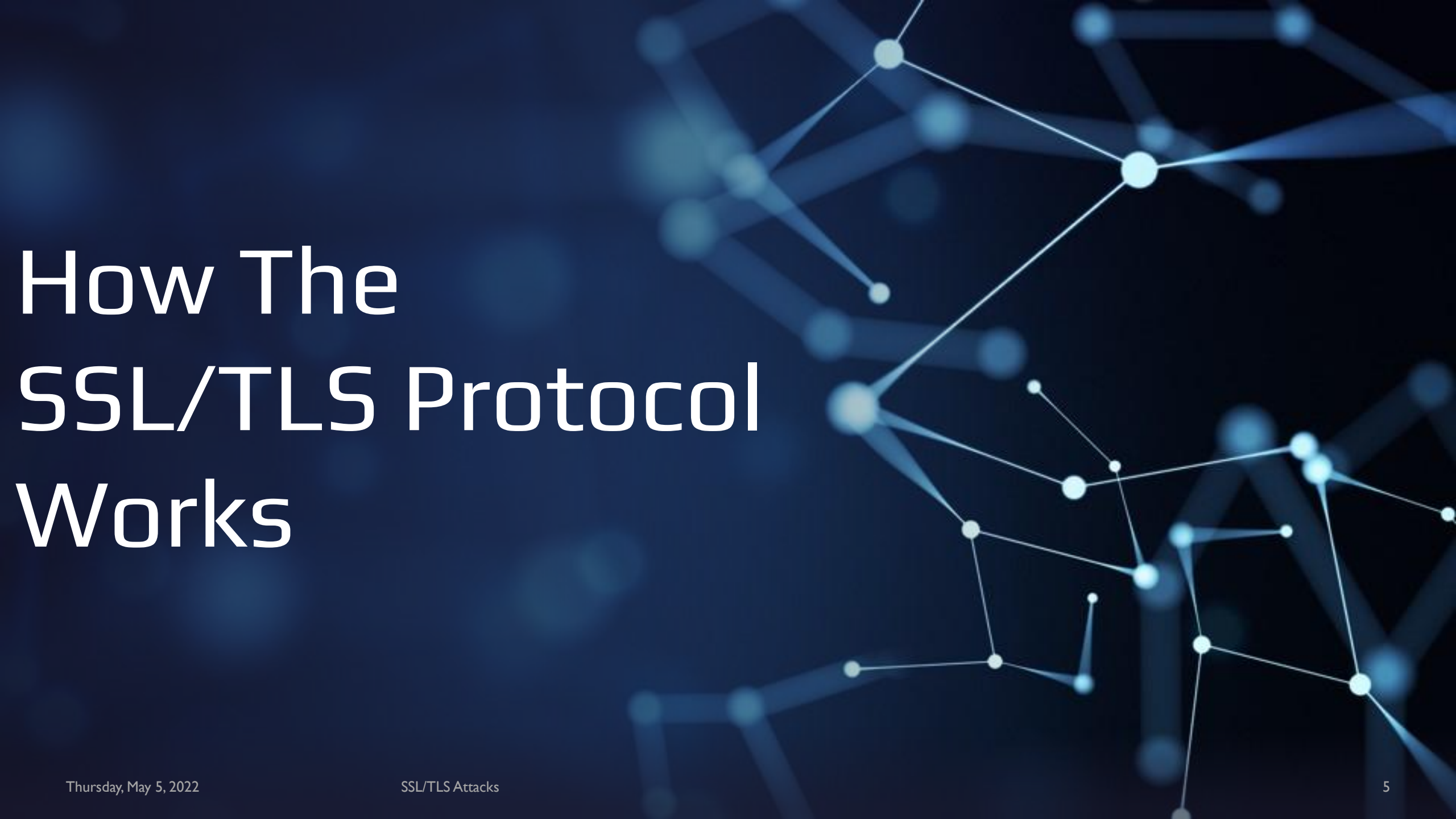
- Located on Application Layer of TCP/IP stack
- Uses TCP to send packets (needed for handshake protocol)

How is SSL/TLS used on the web?

- SSL/TLS is deployed in mostly every Web browser; also VoIP, payment systems, distributed systems ect
- In January 2021, 89% of pages loaded in Chrome were served over HTTPS (Google's Transparency Report)

Timeline of SSL/TLS Attacks and Versions





How The SSL/TLS Protocol Works

Handshake Protocol

https://www.a10networks.com/glossary/key-differences-between-tls-1-2-and-tls-1-3/

The ‘Client Hello' message

- Client sends what type of SSL/TLS versions and cipher suits it supports

The ‘Server Hello' message

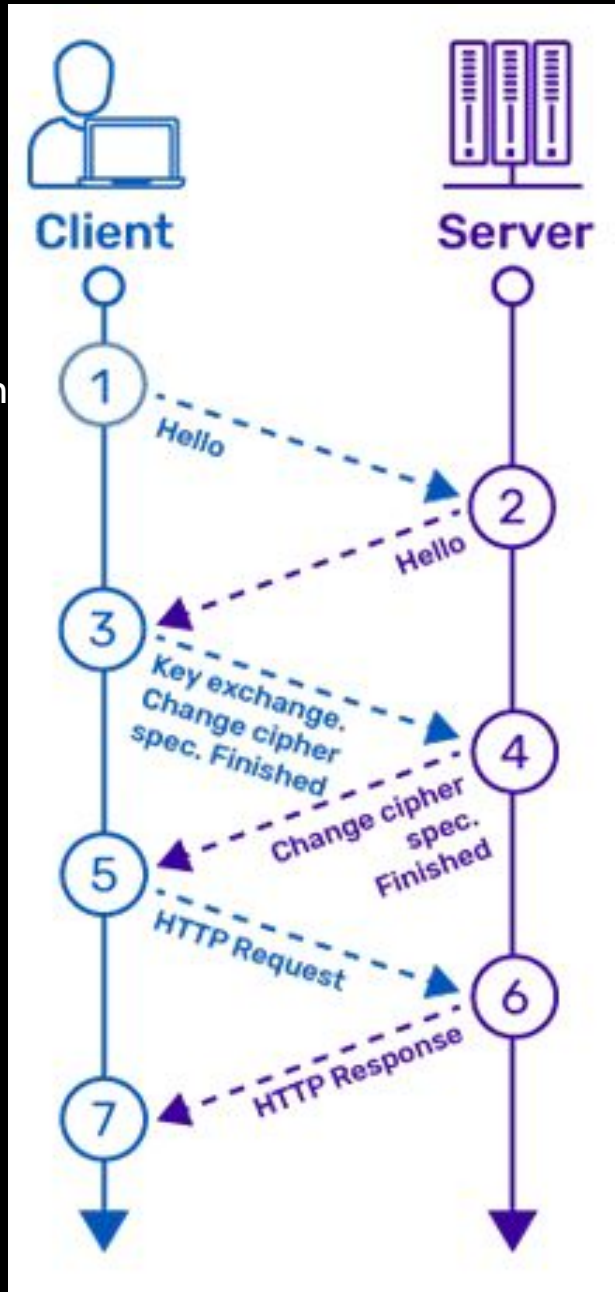
- the server sends a message containing the server’s SSL Certificate (X. 509), the server's chosen cipher suite from client
- Server can request for client Certificate if needed but usually doesn’t.

Client Authentication and Key exchange

- Client authenticates servers' certificate against Certificate authority
- Client creates “premaster secret” key (encrypts with servers public key) and sends to server

Session Keys

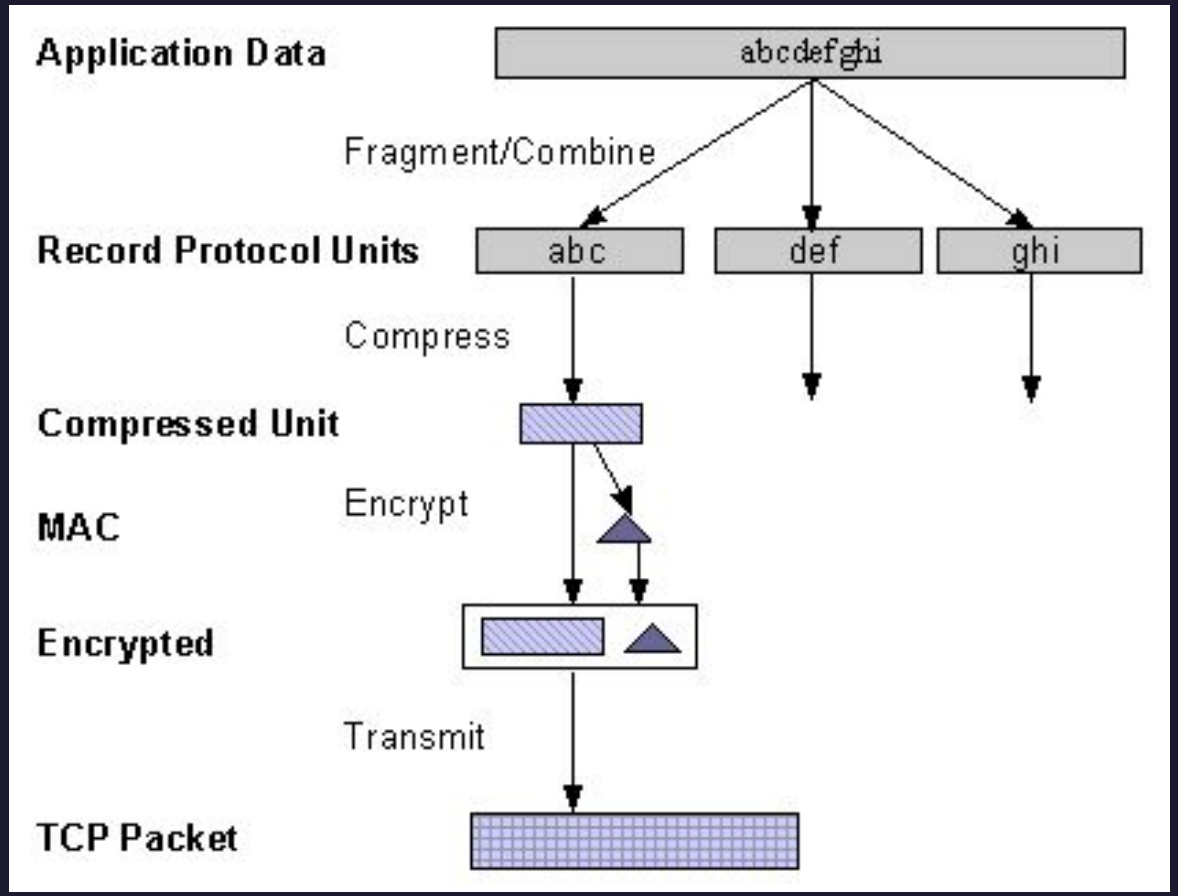
- Client and server generate session keys from premaster secret
- How they are derived and what type of keys depend on algorithm used



Record Protocol

- The Record Protocol is responsible for securing application data and verifying its integrity and origin
- Dividing outgoing messages into manageable blocks, and reassembling incoming messages.
- Applying a Message Authentication code(MAC) to outgoing messages and verifying incoming messages using the MAC.
- Encrypting outgoing messages and decrypting incoming messages.

SSL Vitaly Shmantikov



DROWN

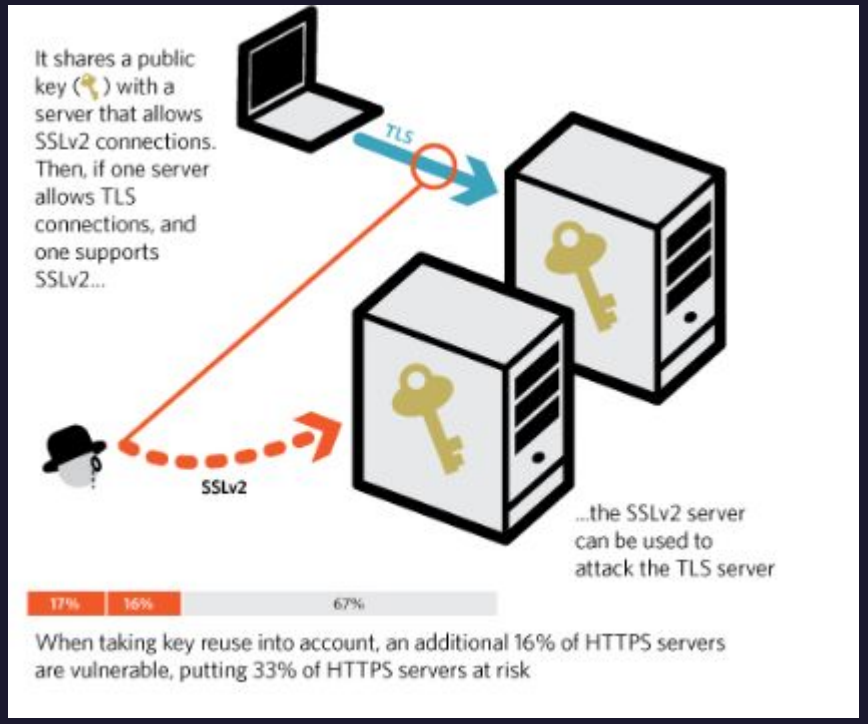
- **D**ecrypting **R**SA using **O**bsolete and **W**eakened e**N**cryption
- Publicly released March 2016
- Cross-Protocol Attack on TLS using SSL v2

- Breaks TLS encryption using SSL v2
- Vulnerable servers include:
  - Servers that implement TLS and SSL v2 (or out of date OpenSSL)
  - TLS servers that share a private key with a server implementing SSL v2

https://drownattack.com/media/img/DROWN_logo.svg

DROWN Components

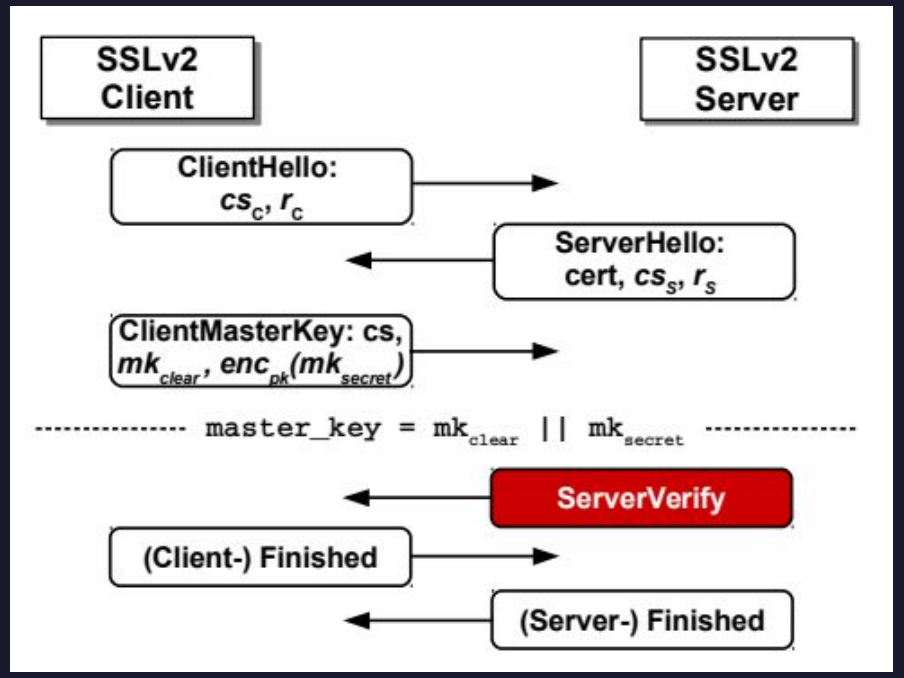
- Components of Practical Attack on SSL v2
  - Bleichenbacher's 1998 attack
  - Weak 40-bit export ciphers
  - Protocol flaw in SSL v2
- Component of Practical Attack on TLS
  - Shared keys amongst SSL and TLS
- Component of Trivial attack on TLS
  - Implementation flaws in OpenSSL



https://drownattack.com/

SSL v2 Handshake Flaw

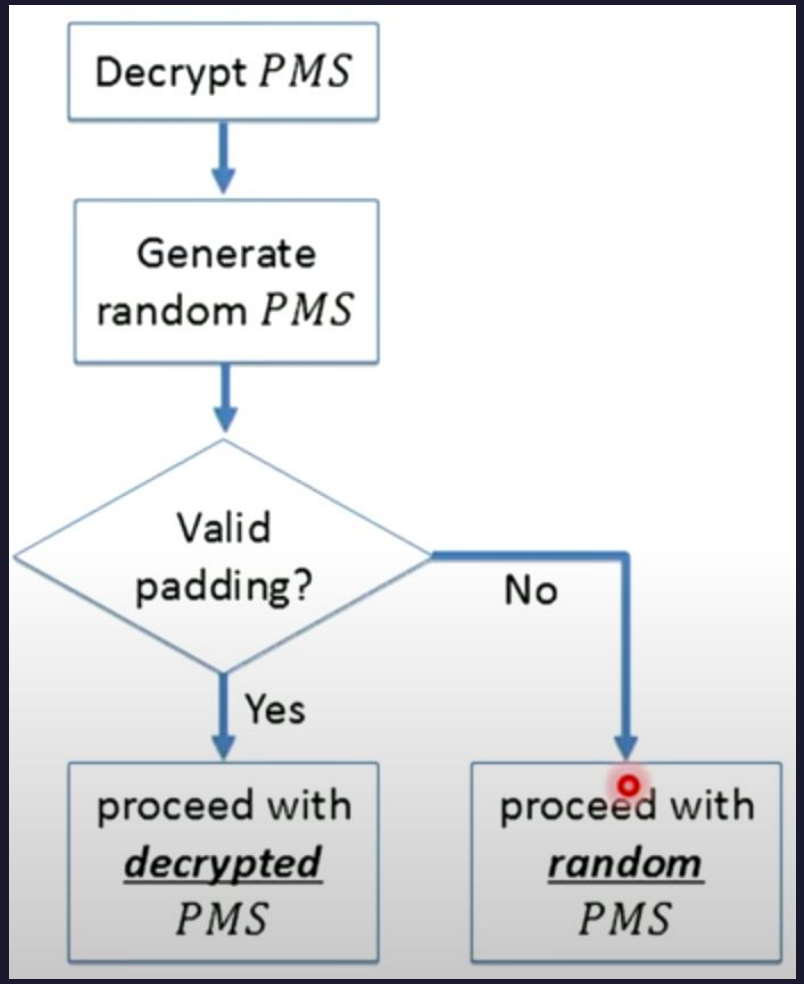
- Client sends ClientMasterKey with secret encrypted with the public key
- Server immediately sends back a challenge to the client encrypted with the newly formed secret key
- Attacker can replace the secret key with RSA ciphertext from collected victim TLS connections.
- Attacker can get information based on how SSL v2 server responds.



https://drownattack.com/drown-attack-paper.pdf

SSL v2 Oracle

- Connect twice to SSL v2 Oracle and send the same RSA ciphertext
- If master secret differs, PKCS#1 padding is wrong
- If master secret are the same, padding is correct



https://www.youtube.com/watch?v=chwtQI_xt8s

Mitigations

As a server owner:

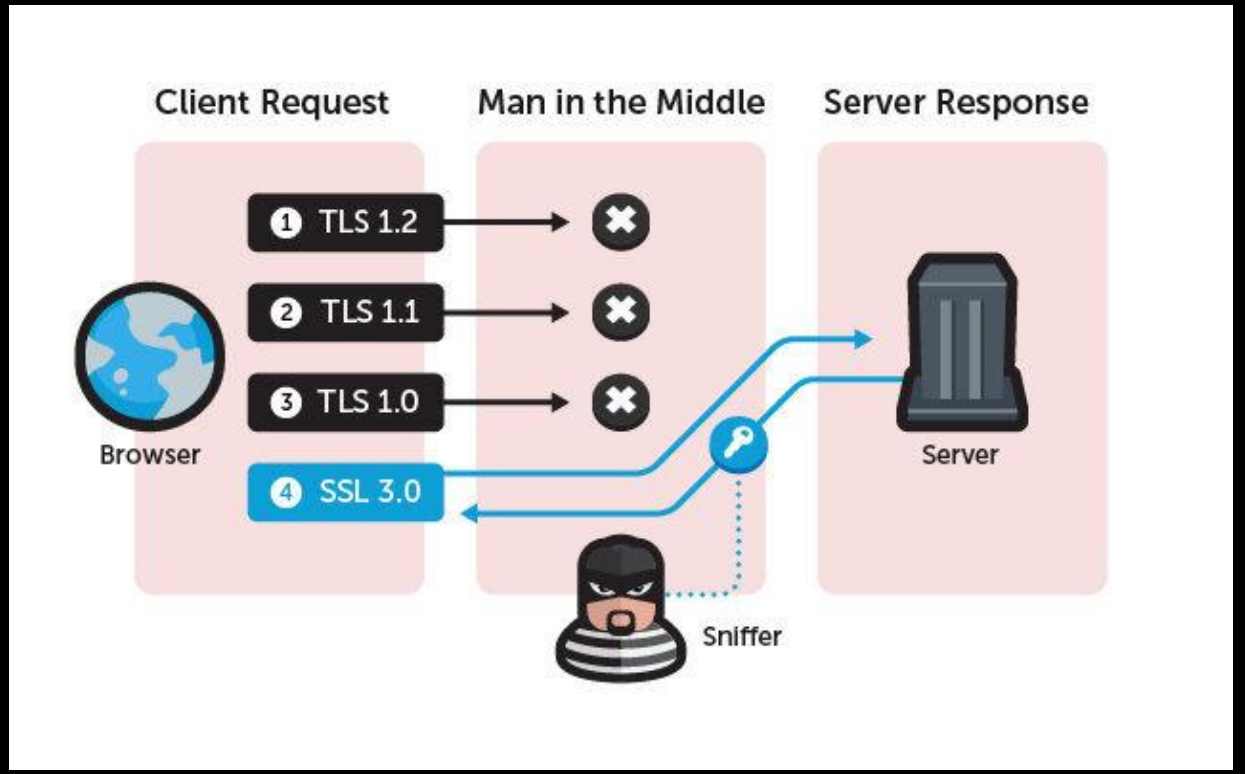
- Disable SSLv2 on servers
- Keep OpenSSL up to date
- Avoid sharing keys that must keep SSLv2 protocol

As a user:

- Avoid affected websites that are affected

Poodle Attack

- Padding Oracle On Downgraded Legacy Encryption
- Works in SSLv3 and below
- SSLv3 considered to be obsolete and insecure
- POODLE takes advantage of Downgrading to SSLv3
- Basically attacker sniffs the data from the server using the downgraded SSLv3 protocol



Padding

	BLOCK #1								BLOCK #2							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
Ex 1	F	I	C													
Ex 1 (Padded)	F	I	C	0x05	0x05	0x05	0x05	0x05								
Ex 2	B	A	N	A	N	A										
Ex 2 (Padded)	B	A	N	A	N	A	0x02	0x02								
Ex 3	A	V	O	C	A	D	O									
Ex 3 (Padded)	A	V	O	C	A	D	O	0x01								
Ex 4	P	L	A	N	T	A	I	N								
Ex 4 (Padded)	P	L	A	N	T	A	I	N	0x08	0x08	0x08	0x08	0x08	0x08	0x08	0x08
Ex 5	P	A	S	S	I	O	N	F	R	U	I	T				
Ex 5 (Padded)	P	A	S	S	I	O	N	F	R	U	I	T	0x04	0x04	0x04	0x04

Padding Oracle attack

Consider this example link:

https://www.cases.com/index.php?UID=8A219A434525535FF324D4G56AB8324 the UID parameters decrypts to 'anony'

- Case 1: Say you sent the value UID=8A219A434525535FF324D4G56AB8324 and it decrypts to a valid user anony

  Then the application would send a normal response.
- Case 2: Say you sent the value UID=998877PA434525535FF324D4G56AB83245 and it decrypts to bfmfy. The

  application might respond back with a 404 message saying no such page exists
- Case 3: • Say you sent the value UID=66IXS7IA434525535FF324D4G56Ab83245 and it decrypts to shayd. but with

  invalid padding. The application would return some exception
- Thus, with different cipher text and value combinations with valid padding we can decrypt cipher values.

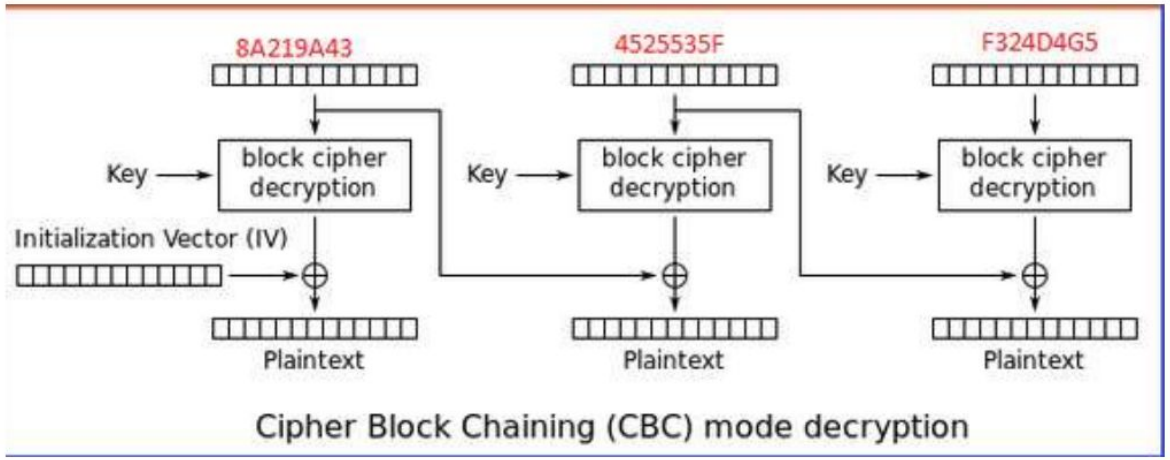
POODLE Against TLS

example url:

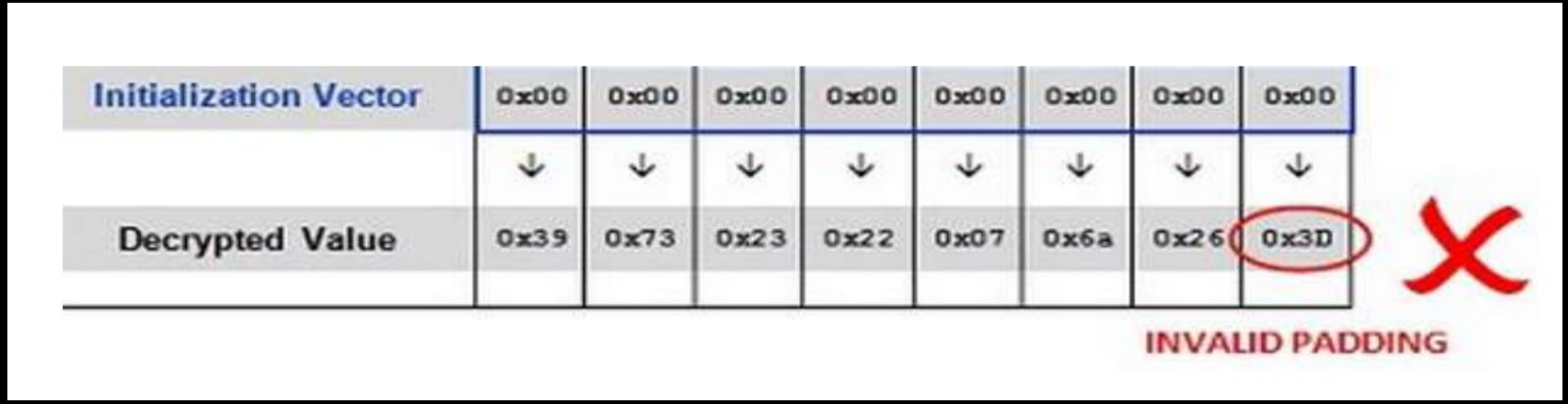
https://www.cases.com/index.php?UID=8A219A43|4525535F|F324

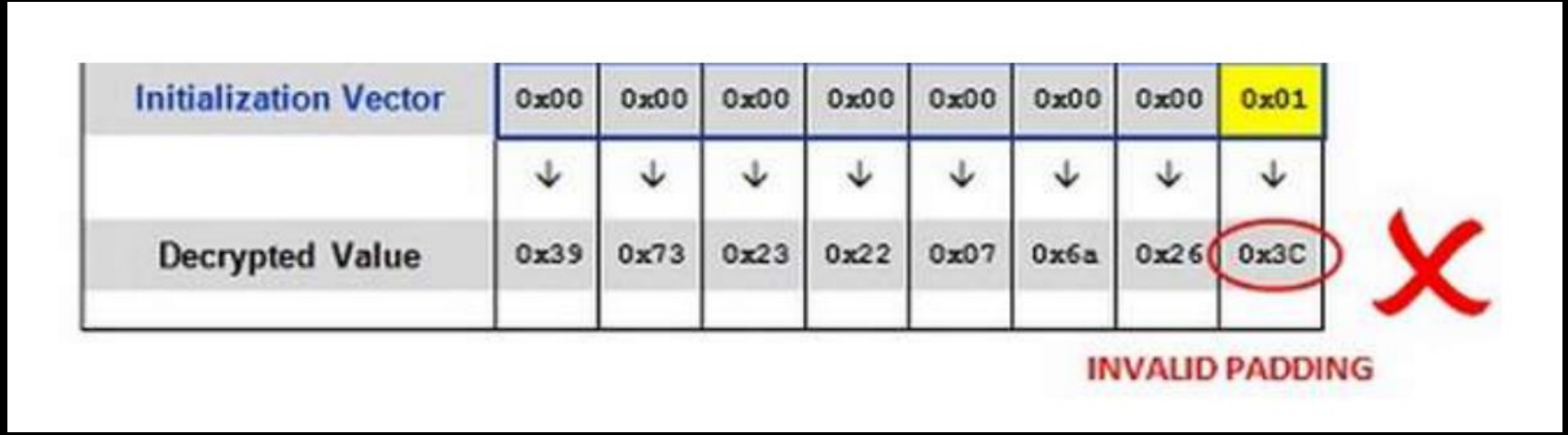
D4G5|6AB832488

cipher blocks: 8A219A43|4525535F|F324D4G5|6AB832488

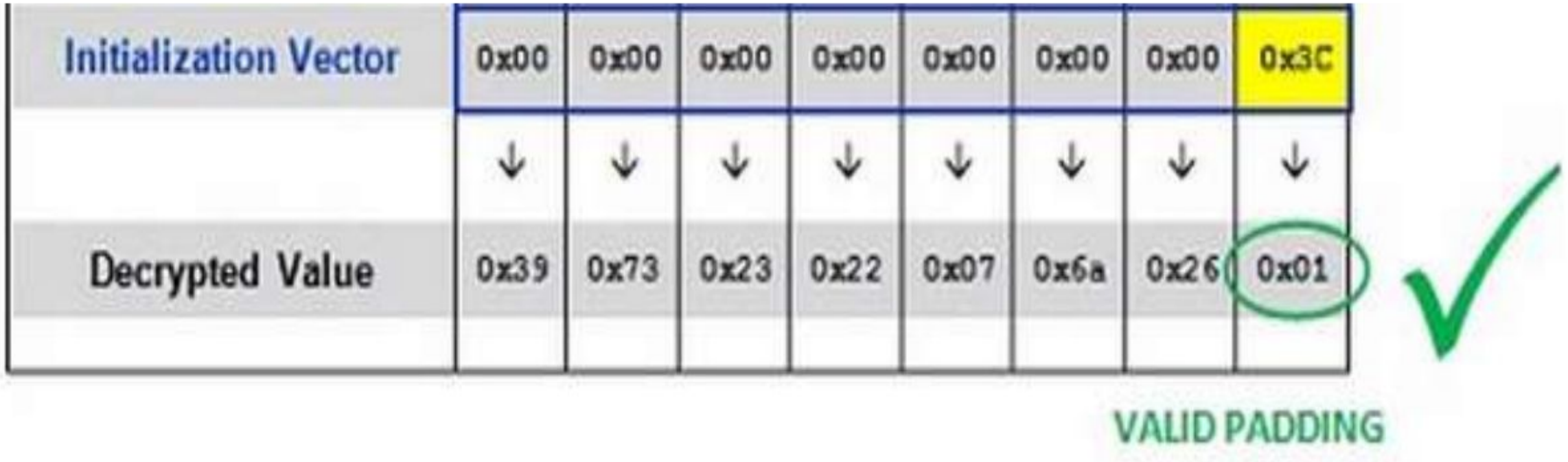


POODLE Against TLS contd.





POODLE Against TLS contd.



Mitigations against POODLE

- Best Solution: Avoid using SSLv3, Avoid downgrading the protocols upgrade to TLS protocol versions
- Temporary solution: in higher version of TLS we can include TLS_FALLBACK_SCSV to prevent downgrading

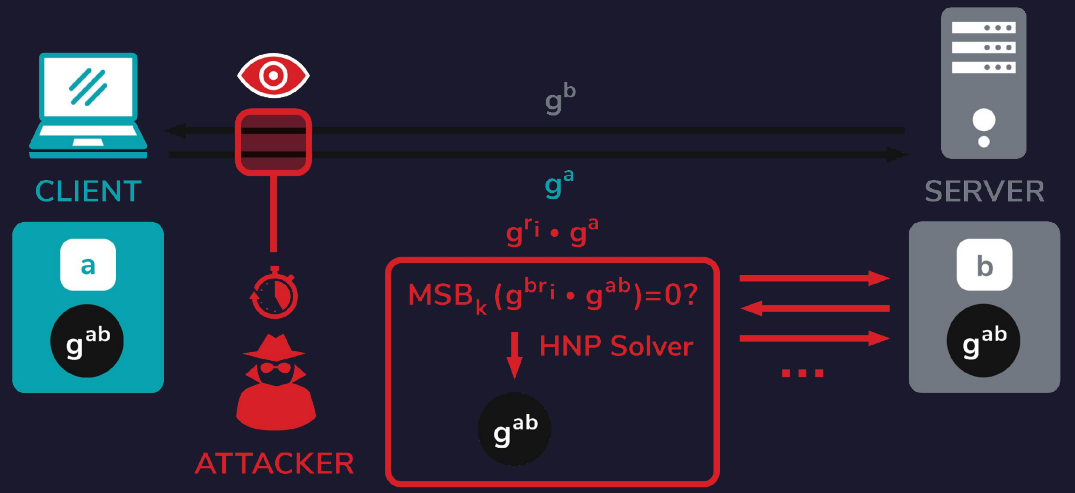

Threats to TLS

TLS 1.2 Attack Requirements

Racoon Attack

- Server uses Diffie-Hellman key exchange
- Server reuses keys for multiple connections
- needs to be close to the target server to perform high precision timing measurements
- the attacker needs to observe the original connection
- Compute complex mathematical calculations fast

Racoon Attack Report
https://raccoon-attack.com/RacoonAttack.pdf



The Future of TLS Attacks

- No attack on TLS 1.3 has been made known to the public
- TLS 1.3 has replaced most obsolete encryption algorithms with stronger ones
- TLS 1.3 have Prohibited SSL or RC4 negotiation for backwards compatibility
- Attacks will most likely come from breaking encryption rather than TLS protocol

Questions?

Sources

- https://resources.infosecinstitute.com/topic/padding-oracle-attack-2/
- https://resources.infosecinstitute.com/topic/end-ssl-poodle/
- https://www.openssl.org/~bodo/ssl-poodle.pdf
- **https://drownattack.com/**
- **https://drownattack.com/drown-attack-paper.pdf**
- **http://www.yaksman.org/~lweith/ssl.pdf**
- **https://crypto.stackexchange.com/questions/27131/differences-between-the-terms-pre-master-secret-master-secret-private-key?noredirect=1&lq=1**
- **https://www.cloudflare.com/en-au/learning/ssl/what-is-a-session-key/**
- **https://www.cryptologie.net/article/340/tls-pre-master-secrets-and-master-secrets/**
- **https://docs.microsoft.com/en-us/windows/win32/secauthn/tls-record-protocol#:~:text=The%20Transport%20Layer%20Security%20(TLS,verifying%20its%20integrity%20and%20origin.**
- **https://raccoon-attack.com/#:~:text=Raccoon%20is%20a%20timing%20vulnerability,able%20to%20read%20the%20communication.**
- https://raccoon-attack.com/RacoonAttack.pdf