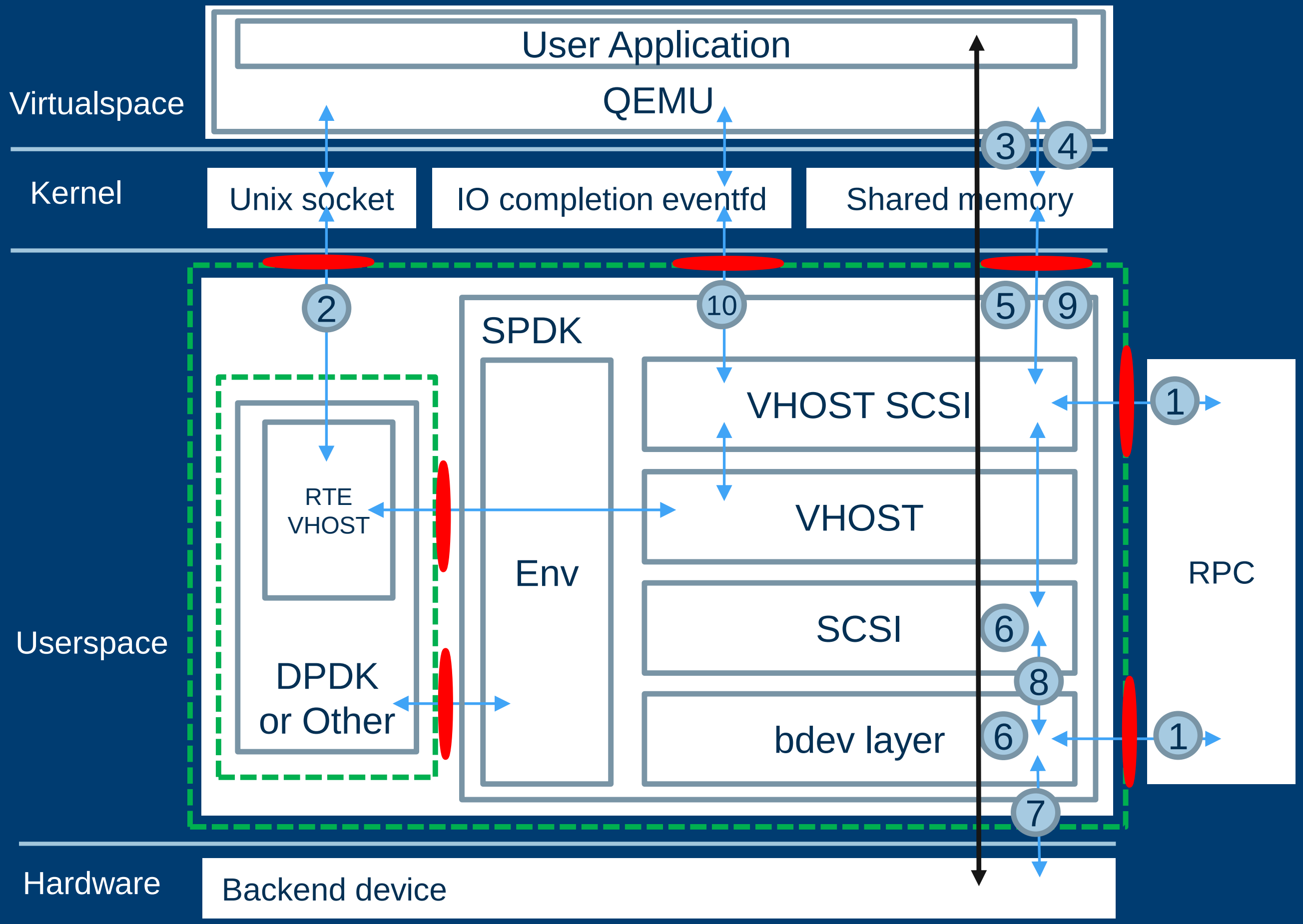
USE CASE: VHOST INTEGRATION

- User has decided to integrate SPDK vhost stack into his existing virtualization solution
- Related hardware and software components (including bdev interface) were configured correctly so they are assumed to be safe in this use case

SYSTEM DIAGRAM



Assets:

A. Data
B. SPDK Application
C. Shared Memory
D. Sockets
E. Env
F. QEMU
G. RTE_VHOST

High Level Flow:

1) SPDK gets configured with RPC
2) SPDK creates a unix socket and establishes connection with QEMU
3) QEMU shares memory and I\O queues
4) User application sends I\O request
5) SPDK detects request by continously polling shared I\O queues
6) SPDK processes the request, first in lib/scsi then in bdev layer
7) Request is sent to backend device
8) Callback is called from bdev layer
9) Vhost updates the request status by modifying shared queues
10) Vhost notifies the application about completion by writing to completion eventfd

Control/Function Calls	↔
Data	↔
Attack Surfaces	⬬
Trust Boundaries	- - -

ATTACK SURFACES

System Element	Compromise Type(s)	Assets exposed	Attack Method
QEMU socket interface	Invalid input	RTE_VHOST, Shared memory, QEMU, SPDK app	Malformed vhost-user commands
Completion eventfd	Invalid initialization	Sockets, RTE_VHOST	Bad target address, Cause loop
Shared memory	Invalid input	Data, RTE_VHOST, SPDK app	Malformed IO descriptors
ENV/RTE VHOST interface	DLL Injection	Env, RTE_VHOST, Data, QEMU, Sockets, Shared Memory, SPDK app	Replaced Env library calls
RPC interface	Invalid input	Sockets, RTE_VHOST, SPDK app	Malformed/invalid json-rpc requests

THREAT MATRIX

Assets Surface	Data	SPDK application	Shared memory	Sockets	Env	QEMU	RTE_VHOST
QEMU socket interface	No	Yes	No	Yes	No	Yes	Yes
Completion eventfd	No	No	No	No	No	No	Yes
Shared memory	Yes	Yes	Yes	No	No	No	Yes
ENV/RTE VHOST interface	Yes	Yes	Yes	Yes	Yes	Yes	Yes
RPC interface	No	Yes	No	Yes	No	No	Yes

ADVERSARIES IN SCOPE

Persona	Motivation	Attacker Type	Starting Privilege Level	Skill and Potential Effort level
Malicious VM User	Wants to snoop data/disrupt users on system	Software Adversary in a VM	None	Unskilled, gives up easily
Malicious Hypervisor	Wants to snoop data/disrupt users on system	Hypervisor Software Adversary	None	Proficient level of skill, does not give up easily
Malicious RPC Admin	Denial Of Service	Network Adversary	None	Proficient level of skill, does not give up easily

* host system software adversary is out of scope because such adversaries have permissions to defeat all mitigations. User needs to ensure appropriate deployment policies are in place to prevent system level software adversaries

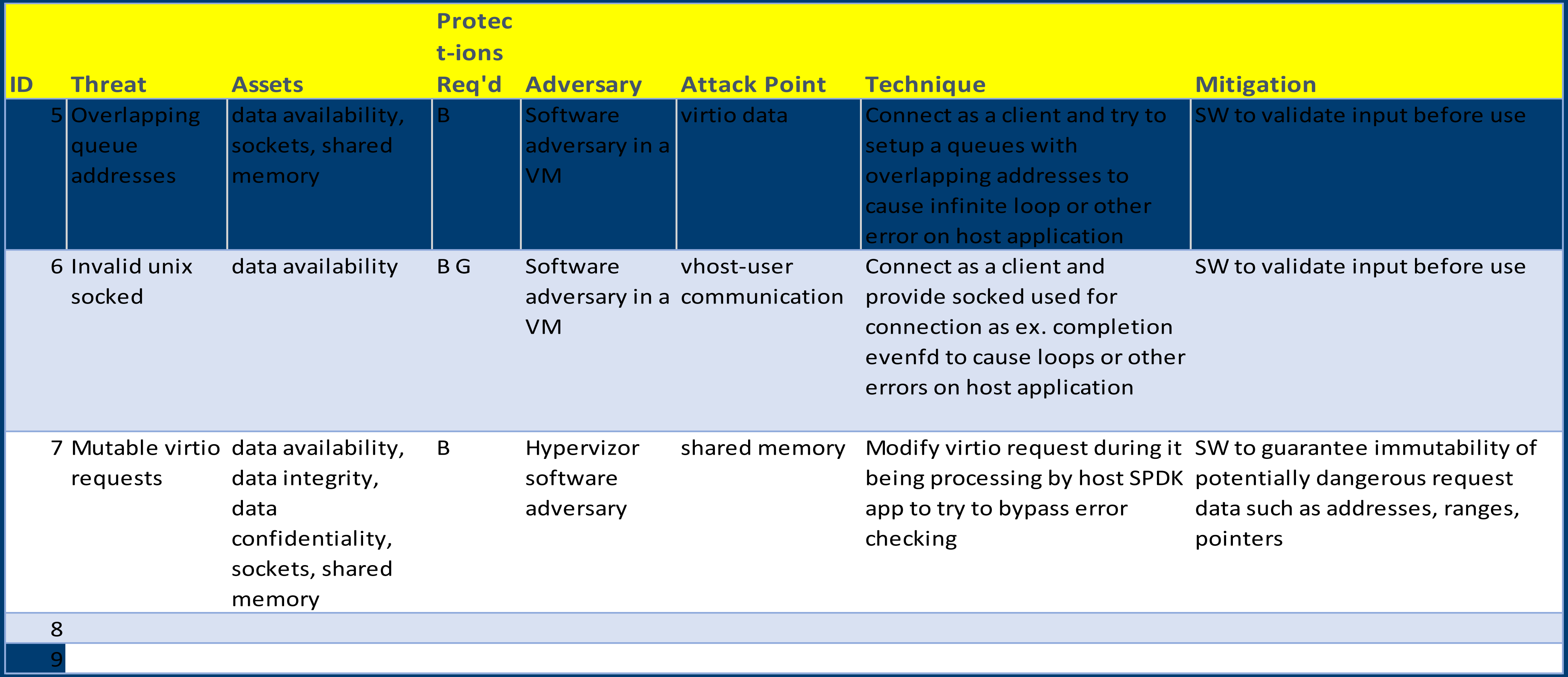
THREAT/ATTACK SURFACE MATRIX

Asset\Attack Surface	QEMU socket interface	Completion eventfd	Shared memory	ENV/ RTE_VHOST interface	RPC interface
Data availability	Y	Y	Y	Y	Y
Data confidentiality				Y	
Data integrity	Y	Y	Y	Y	Y
Shared memory resources				Y	Y
Unix sockets	Y	Y		Y	Y
App configuration file					

THREATS

ID	Threat	Assets	Protec t-ions Req'd	Adversary	Attack Point	Technique	Mitigation
1	Malformed vhost-user commands	data availability, sockets, shared memory	B G	Software adversary in a VM	vhost-user communication	Connect as a client and send malformed vhost-user messages to cause an error on host application	SW to validate input before use
2	Invalid memory setup	data availability, sockets, shared memory	B E	Software adversary in a VM	vhost-user communication	Connect as a client and try to setup invalid memory region to cause an error on host application	SW to validate input before use
3	Deinitialization of nonexisting virtqueues	data availability, sockets, shared memory	B G	Software adversary in a VM	vhost-user communication	Connect as a client and try to deinitialize nonexisting virtqueus to cause an error on host application	SW to validate input before use
4	Repeated reconnect	data availability	B G	Software adversary in a VM, Hypervizor software adversary	vhost-user communication	Repeatedly connect and disconnect causing SPDK to initialize new connections (most importantly map/unmap memory regions) which will result in delays for other users (DoS)	SW to implement smart QoS policy

THREATS



ID	Threat	Assets	Protec t-ions Req'd	Adversary	Attack Point	Technique	Mitigation
5	Overlapping queue addresses	data availability, sockets, shared memory	B	Software adversary in a VM	virtio data	Connect as a client and try to setup a queues with overlapping addresses to cause infinite loop or other error on host application	SW to validate input before use
6	Invalid unix socked	data availability	B G	Software adversary in a VM	vhost-user communication	Connect as a client and provide socked used for connection as ex. completion evenfd to cause loops or other errors on host application	SW to validate input before use
7	Mutable virtio requests	data availability, data integrity, data confidentiality, sockets, shared memory	B	Hypervizor software adversary	shared memory	Modify virtio request during it being processing by host SPDK app to try to bypass error checking	SW to guarantee immutability of potentially dangerous request data such as addresses, ranges, pointers
8							
9							