Encrypting OVN tunnels with IPsec

Qiuyu Xiao

qiuyu.xiao.qyx@gmail.com

The University of North Carolina at Chapel Hill

Motivations

Why do we need encryption?

- VMs compute and communicate sensitive data
  - Financial data
  - Health records
- Physical network devices (e.g., router, switch) cannot be trusted or might be compromised
  - Traffic across datacenters
  - Router misconfiguration
  - Attackers breaking into internal network
  - Phishing or social engineering attacks on administrators

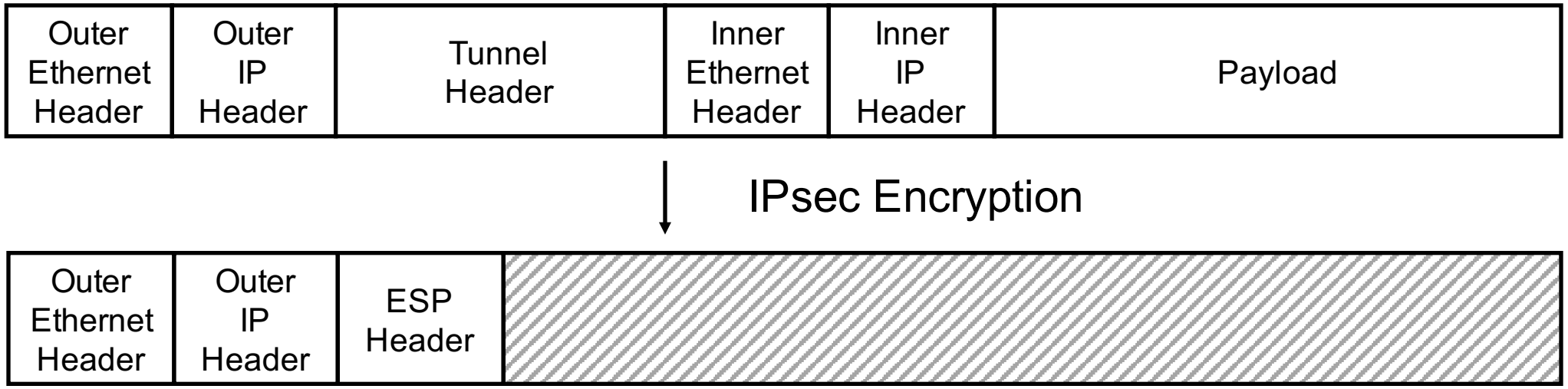
Motivations

IPsec configuration is complicated

- Many configuration fields
- Various cryptographic algorithms and parameters
- Different configuration interfaces from different IKE daemons
- Verifying security configuration is hard

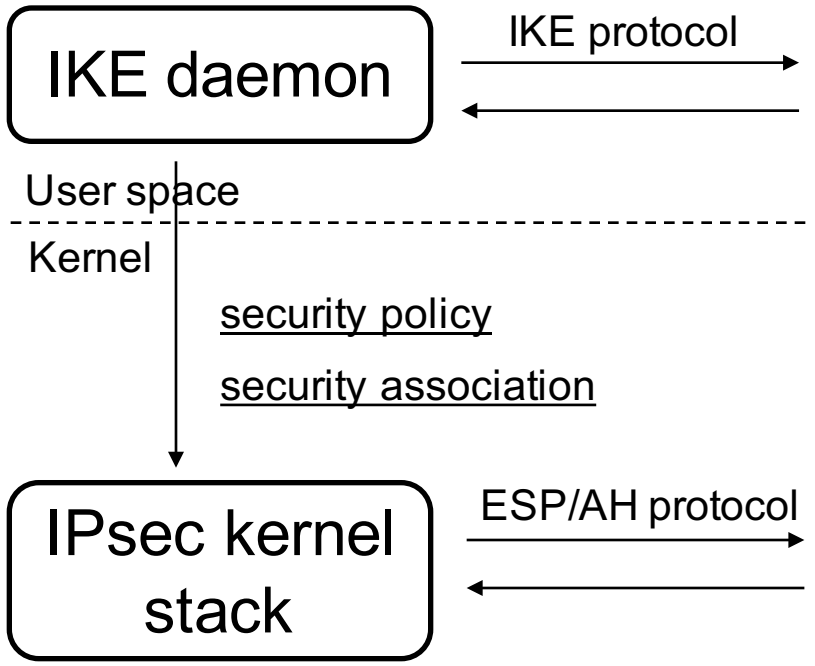
OVS/OVN IPsec

Offer an easy-to-use interface to configure IPsec encryption for tunnel traffic



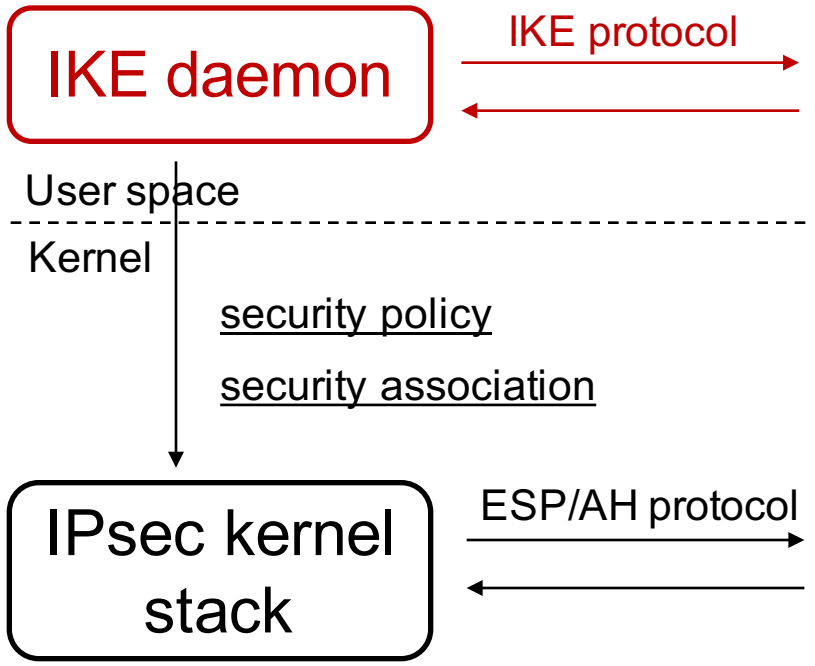
- Confidentiality
- Integrity
- Authenticity

IPsec in Linux



IPsec in Linux

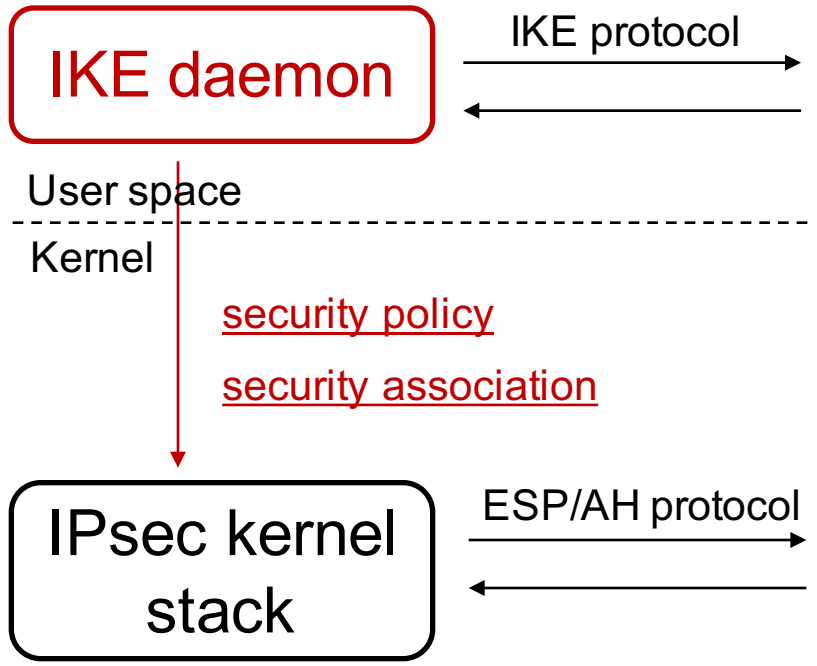
IKE daemon
- Authentication
- Negotiates cryptographic algorithms
- Generates keying material



IPsec in Linux

IKE daemon

- Authentication
- Negotiates cryptographic algorithms
- Generates keying material
- Installs security policy and security association

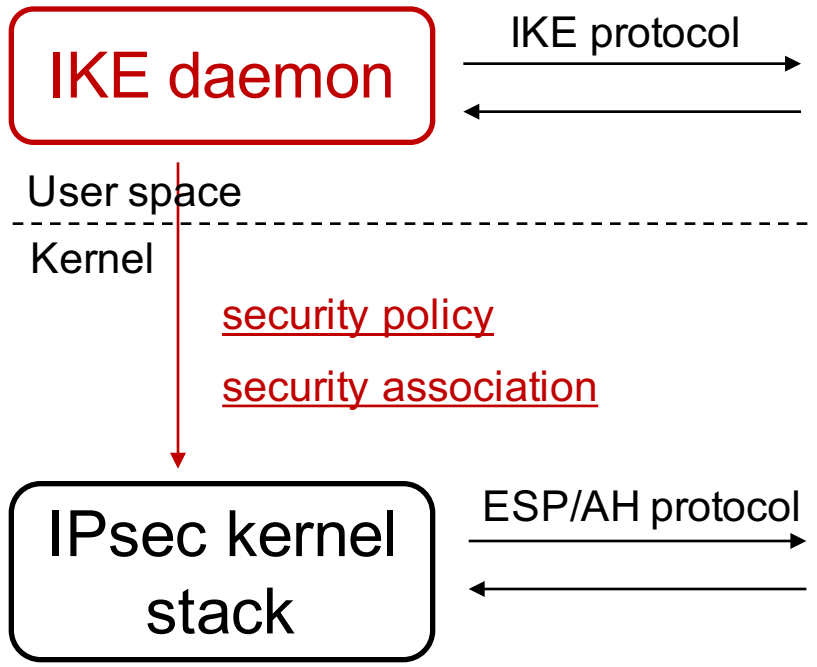


IPsec in Linux

IKE daemon

- Authentication
- Negotiates cryptographic algorithms
- Generates keying material
- Installs security policy and security association

Which traffic to protect



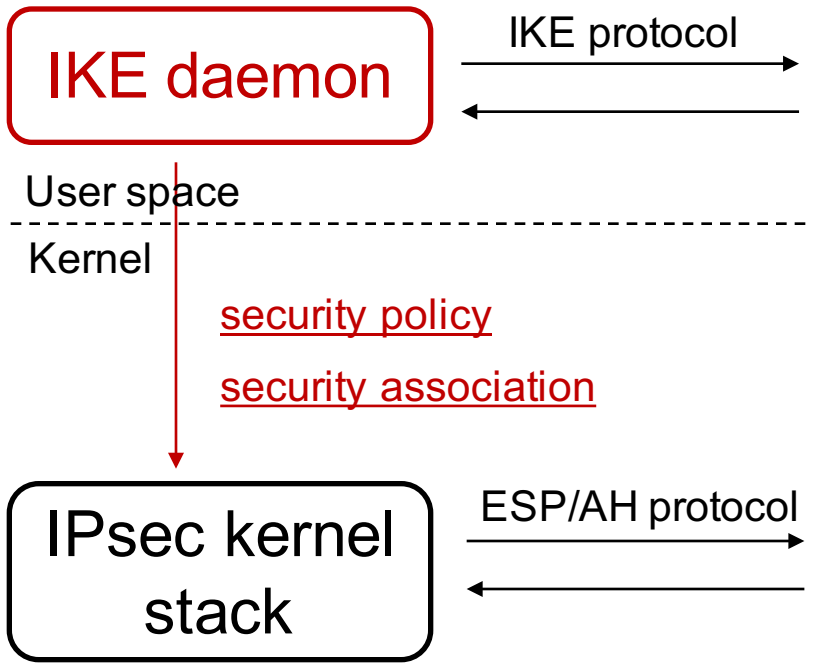
IPsec in Linux

IKE daemon

- Authentication
- Negotiates cryptographic algorithms
- Generates keying material
- Installs security policy and security association

↓

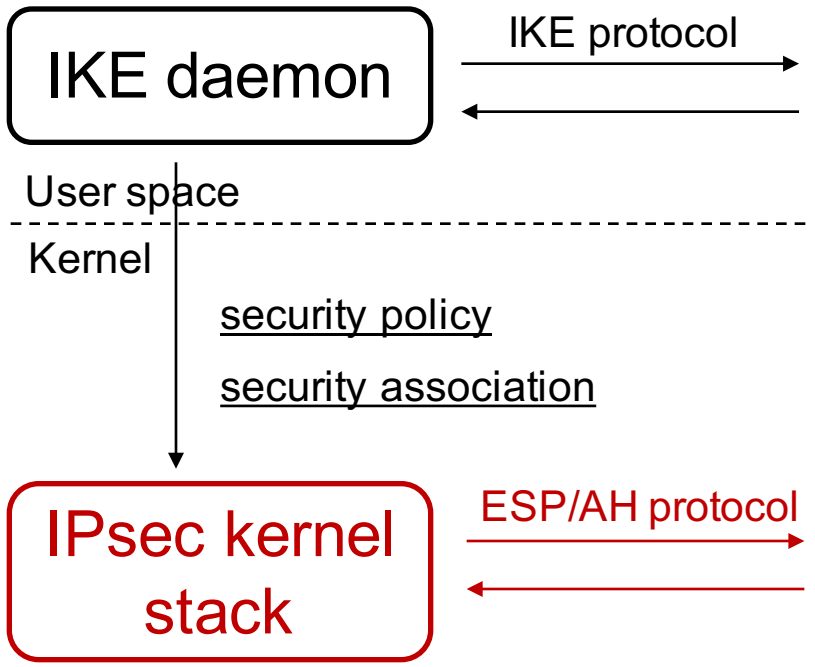
How to protect the selected traffic



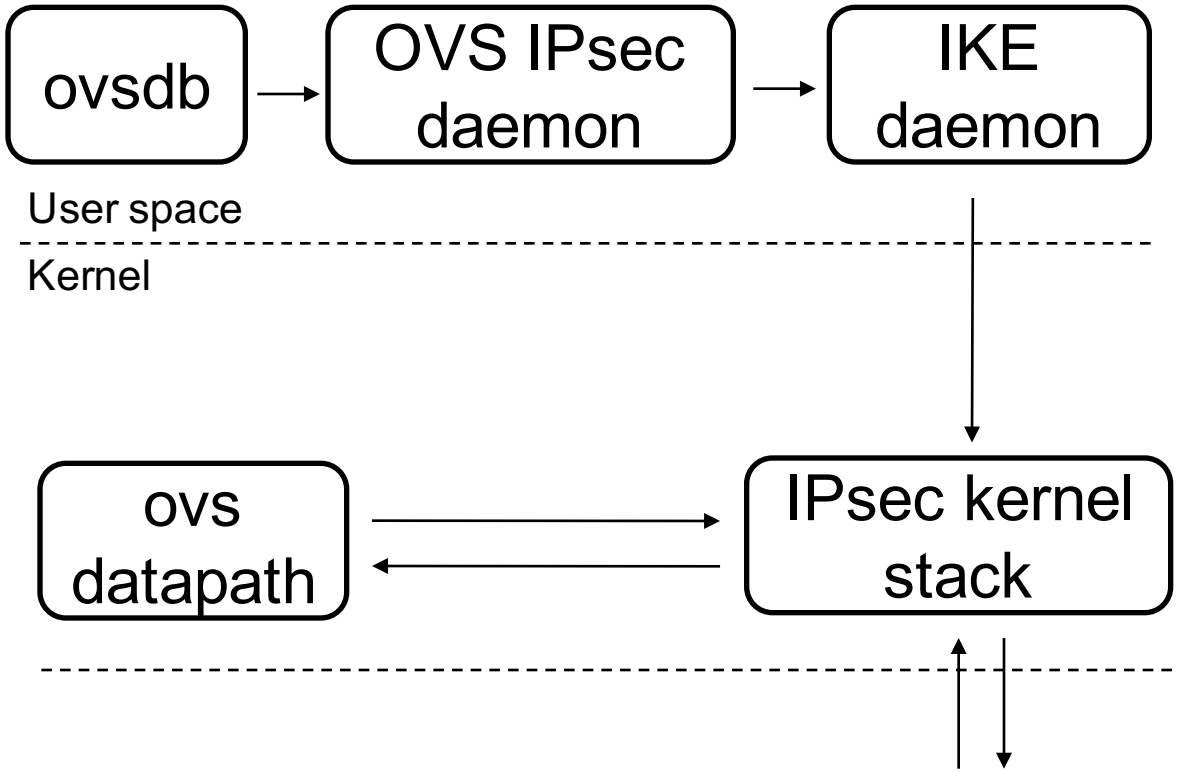
IPsec in Linux

IPsec kernel stack

- Encryption and decryption
- Checks integrity and authenticity



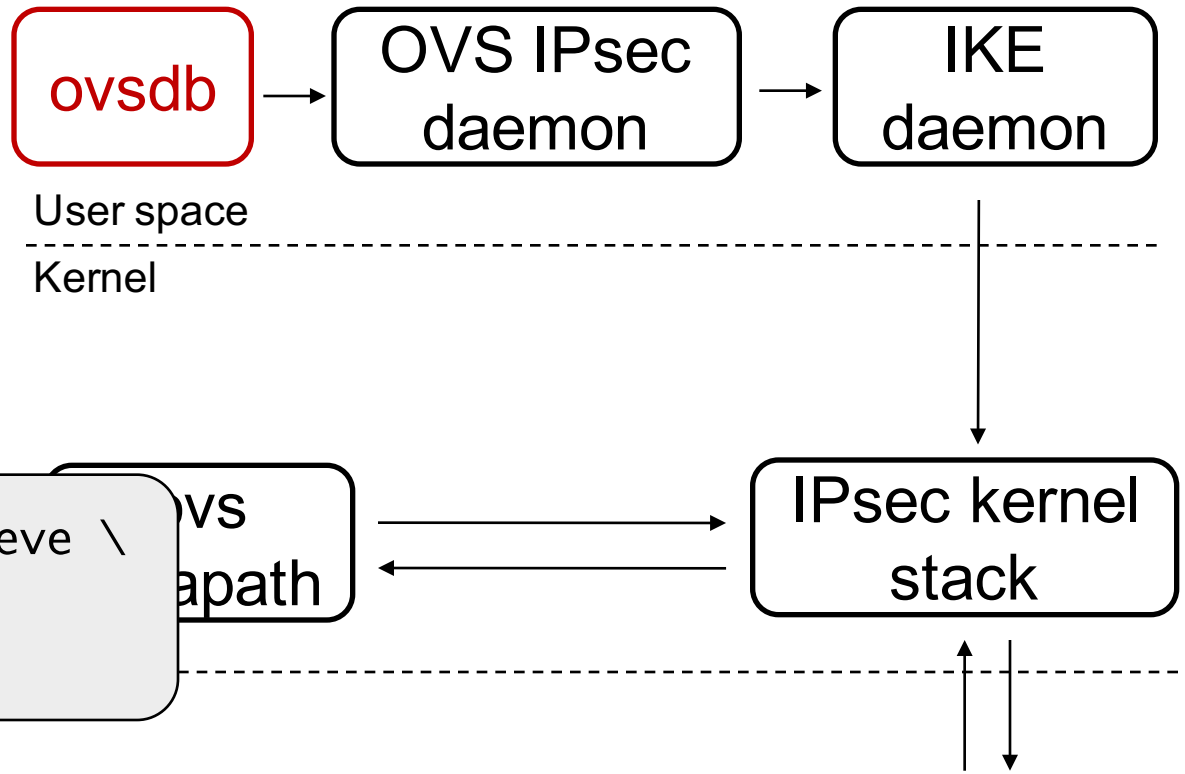
OVS IPsec Tunnel



OVS IPsec Tunnel

Configuring IPsec tunnel via ovsdb

- Using pre-shared key



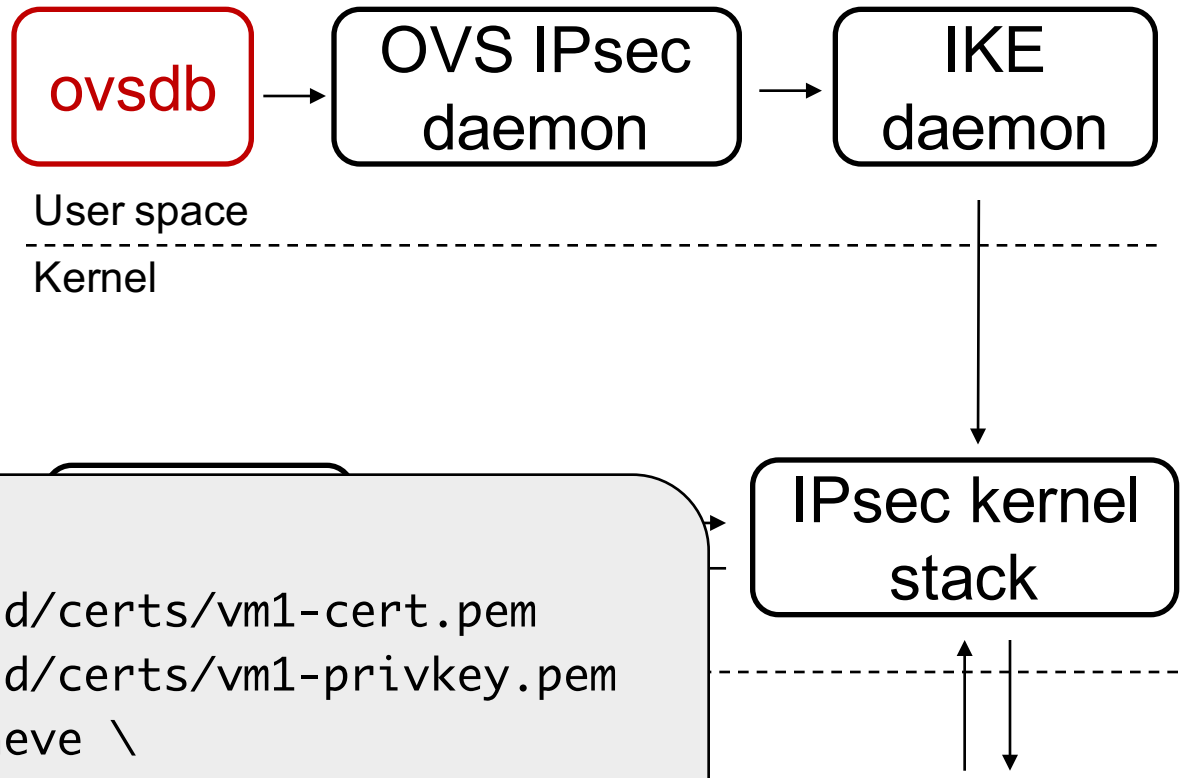
For example:

```
$ ovs-vsctl set interface tun type=geneve \
  options:remote_ip=10.33.79.149 \
  options:psk=swordfish
```

OVS IPsec Tunnel

Configuring IPsec tunnel via ovsdb

- Using self-signed certificate



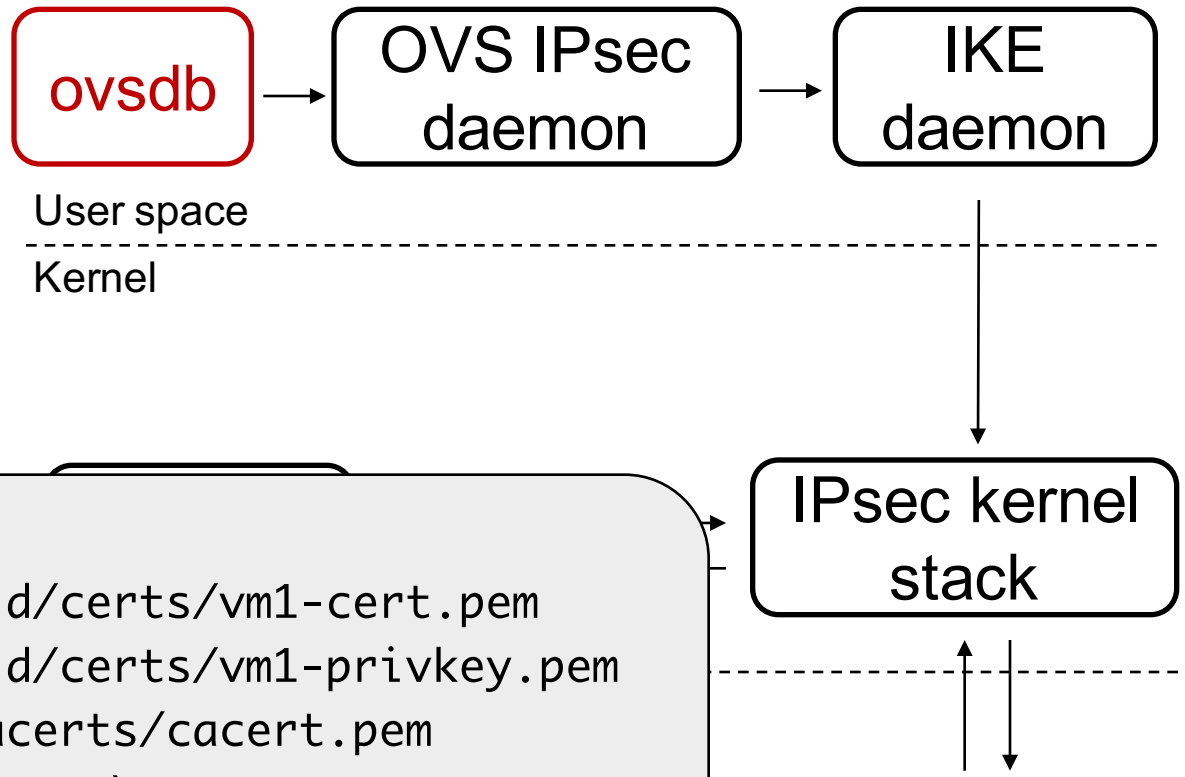
For example:

```
$ ovs-vsctl set Open_vSwitch . \
  other_config:certificate=/etc/ipsec.d/certs/vm1-cert.pem
  other_config:private_key=/etc/ipsec.d/certs/vm1-privkey.pem
$ ovs-vsctl set interface tun type=geneve \
  options:remote_ip=10.33.79.149 \
  options:remote_cert=/etc/ipsec.d/certs/vm2-cert.pem
```

OVS IPsec Tunnel

Configuring IPsec tunnel via ovsdb

- Using CA-signed certificate



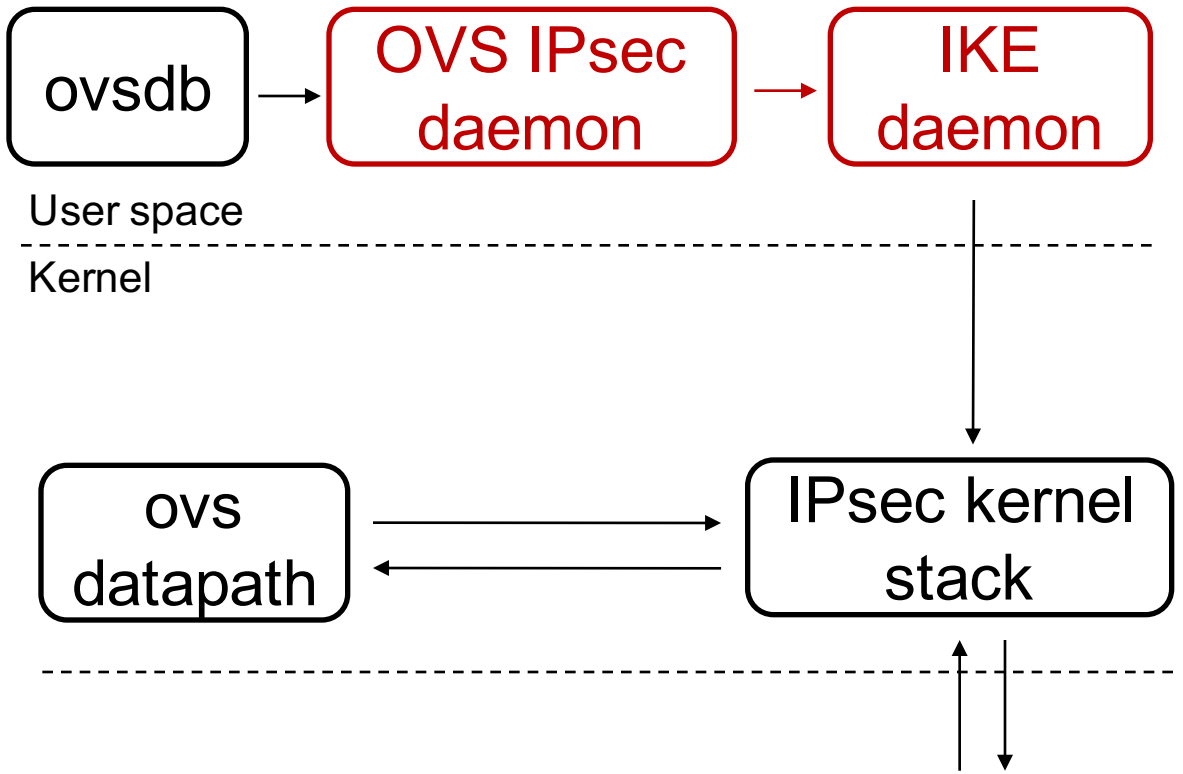
For example:

```
$ ovs-vsctl set Open_vSwitch . \
  other_config:certificate=/etc/ipsec.d/certs/vm1-cert.pem
  other_config:private_key=/etc/ipsec.d/certs/vm1-privkey.pem
  other_config:ca_cert=/etc/ipsec.d/cacerts/cacert.pem
$ ovs-vsctl set interface tun type=geneve \
  options:remote_ip=10.33.79.149 \
  options:remote_name=vm2
```

OVS IPsec Tunnel

Establishing IPsec tunnel

- OVS IPsec daemon configures IKE daemon



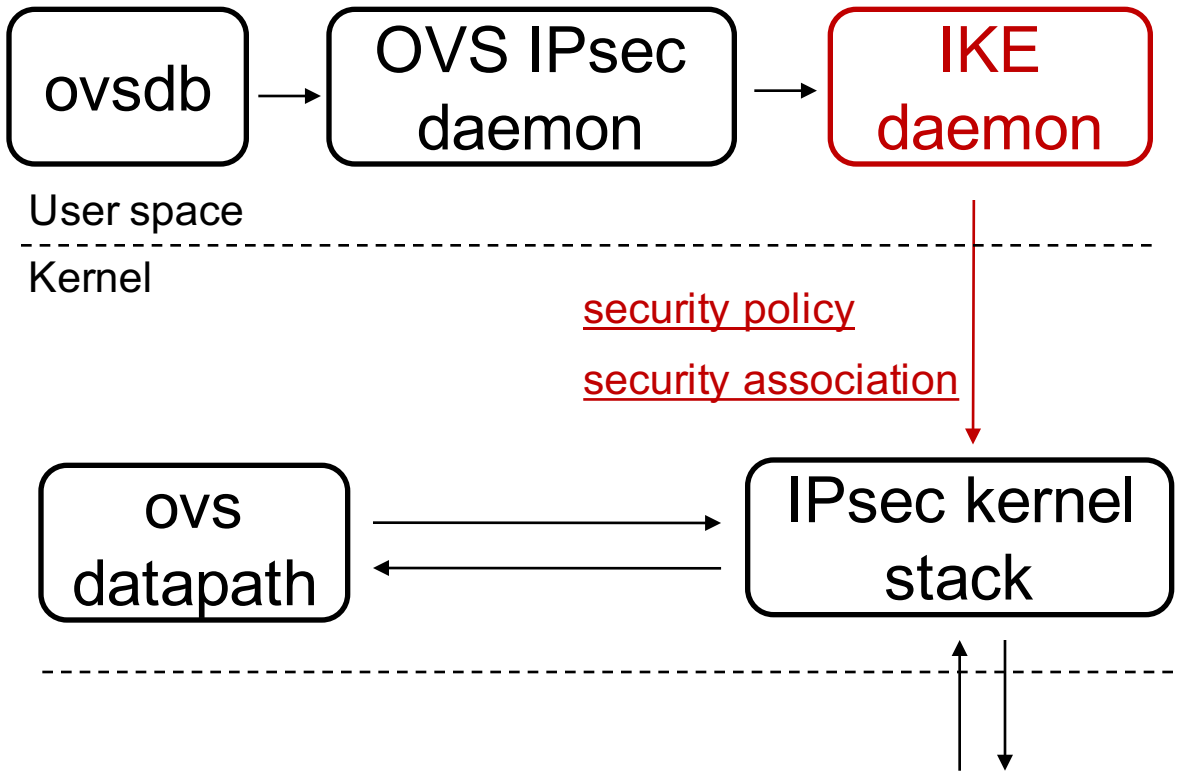
OVS IPsec Tunnel

Establishing IPsec tunnel

- OVS IPsec daemon configures IKE daemon
- IKE daemon sets up security policy and security association

For example (geneve tunnel):

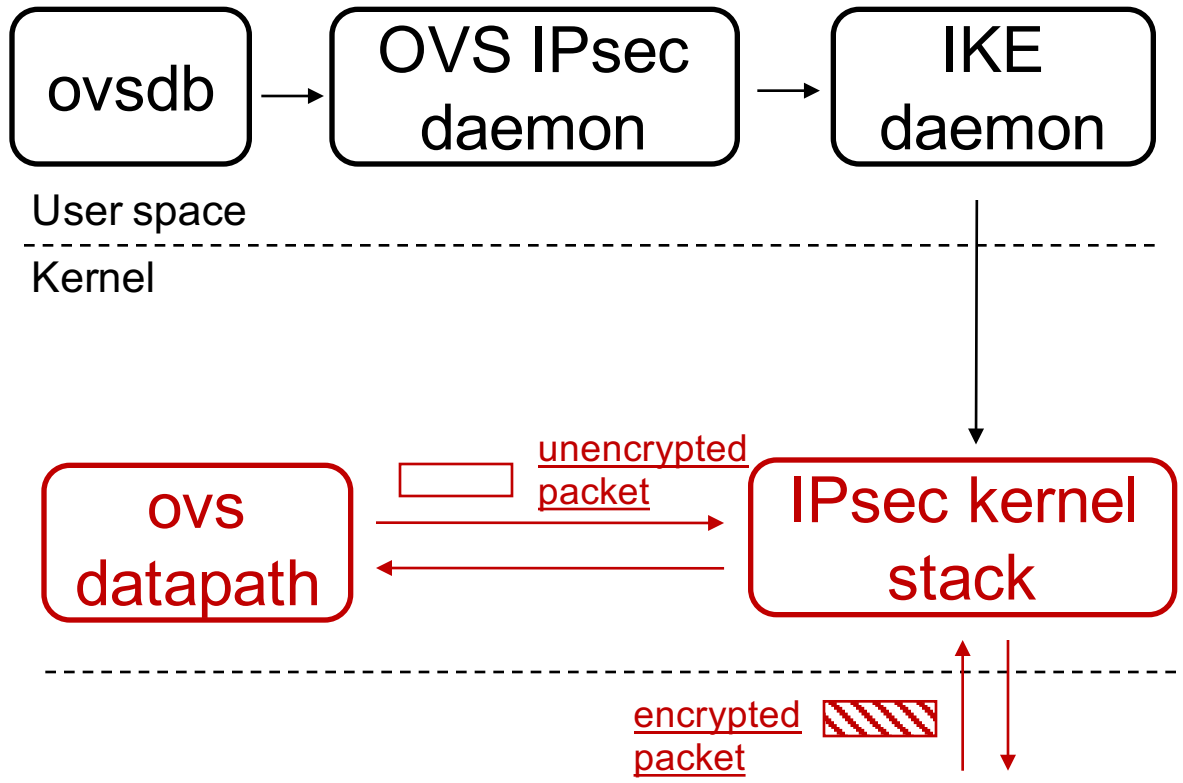
```
root@ubuntu:~/debian/4.13# ip xfrm policy show
src 10.33.78.172/32 dst 10.33.79.149/32 proto udp sport 6081
        dir in priority 5888
        tmpl src 0.0.0.0 dst 0.0.0.0
                proto esp reqid 2 mode transport
src 10.33.79.149/32 dst 10.33.78.172/32 proto udp dport 6081
        dir out priority 5888
        tmpl src 0.0.0.0 dst 0.0.0.0
                proto esp reqid 2 mode transport
src 10.33.78.172/32 dst 10.33.79.149/32 proto udp dport 6081
        dir in priority 5888
        tmpl src 0.0.0.0 dst 0.0.0.0
                proto esp reqid 1 mode transport
src 10.33.79.149/32 dst 10.33.78.172/32 proto udp sport 6081
        dir out priority 5888
        tmpl src 0.0.0.0 dst 0.0.0.0
                proto esp reqid 1 mode transport
```



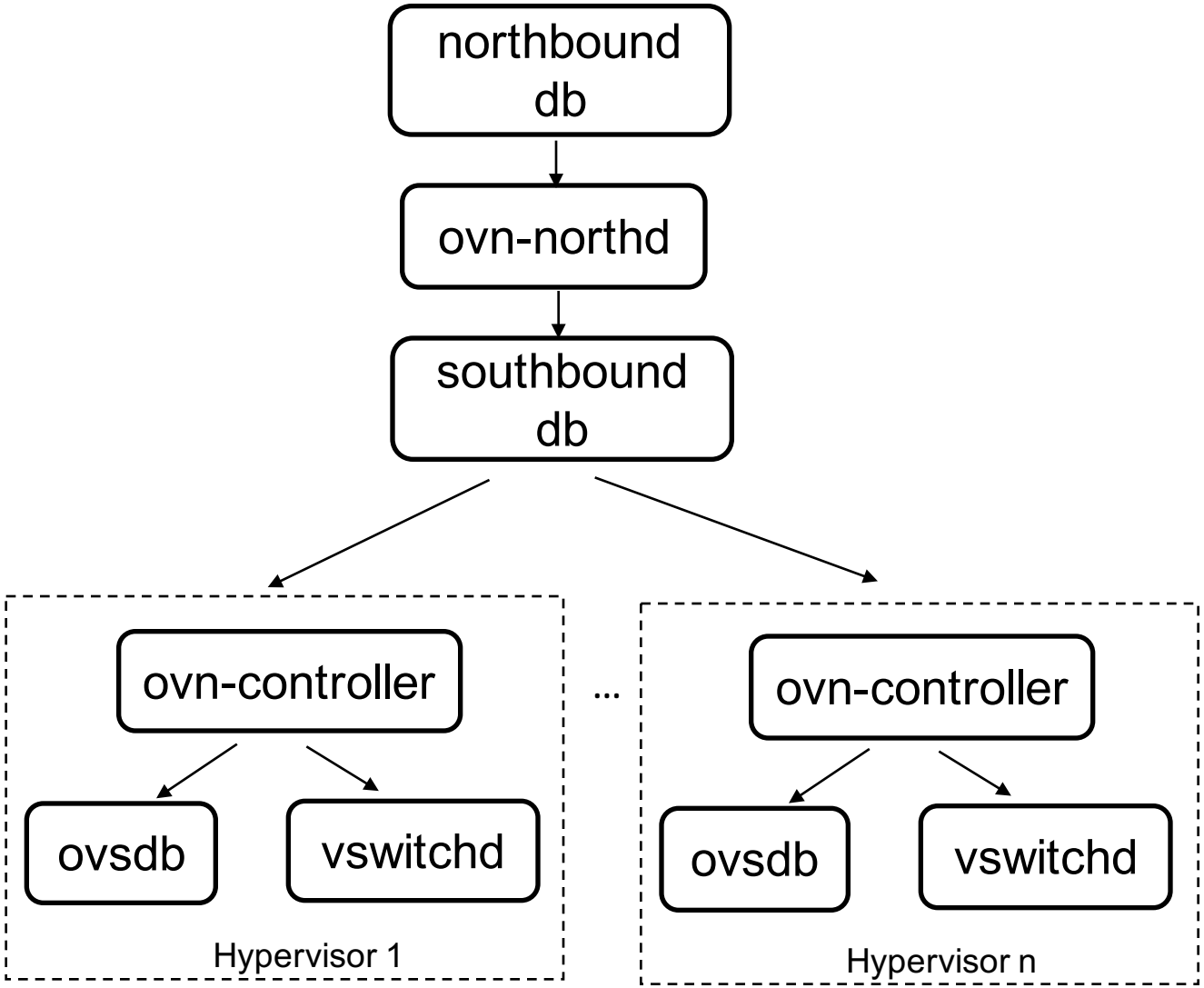
OVS IPsec Tunnel

IPsec kernel stack

- Encryption and decryption
- Checks integrity and authenticity



OVN IPsec

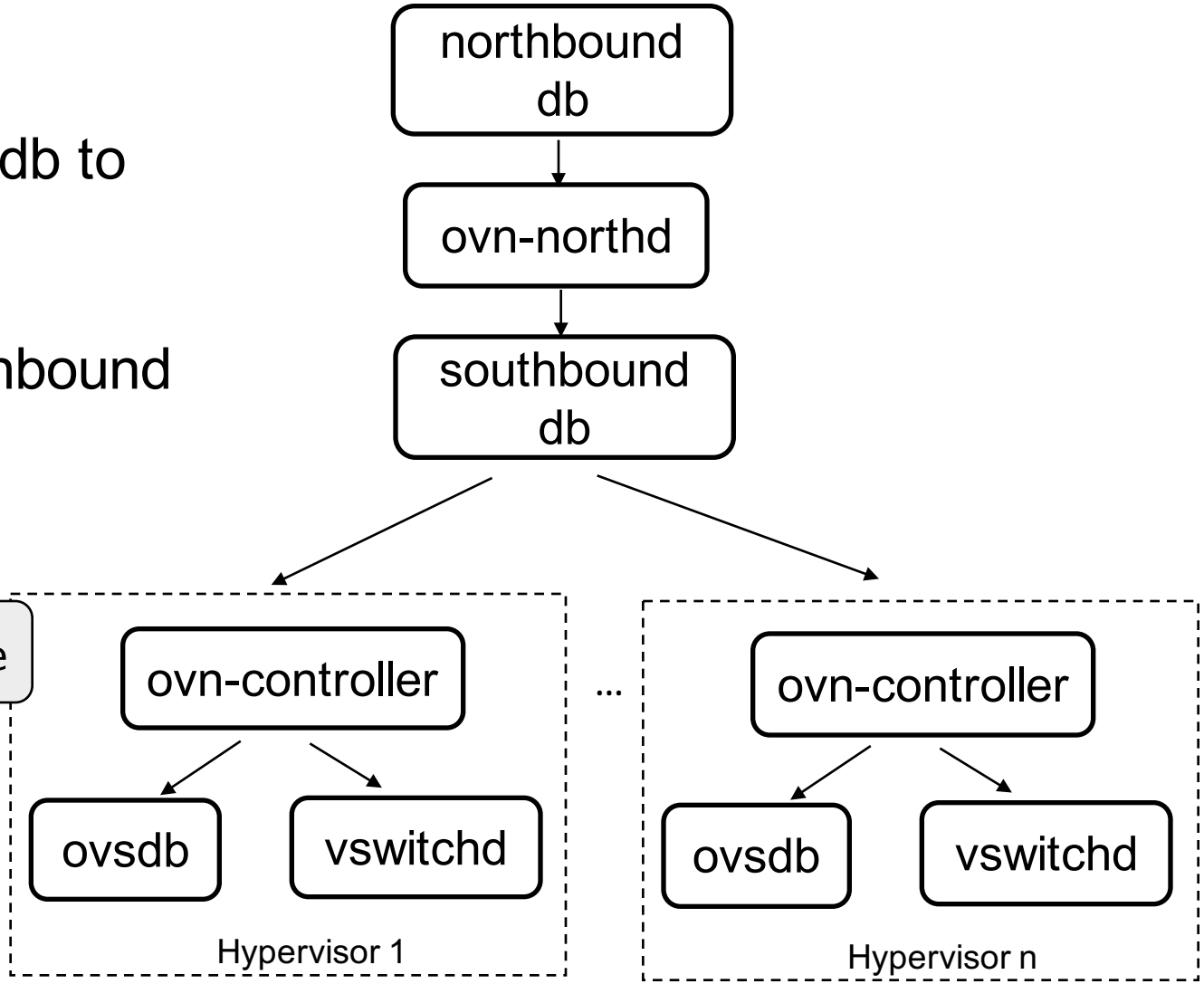

northbound db
ovn-northd
southbound db
ovn-controller
ovsdb
vswitchd
Hypervisor 1
...
ovn-controller
ovsdb
vswitchd
Hypervisor n

OVN IPsec

- In each hypervisor, configure ovsdb to use CA-signed certificate for authentication
- Enable IPsec by configuring northbound database

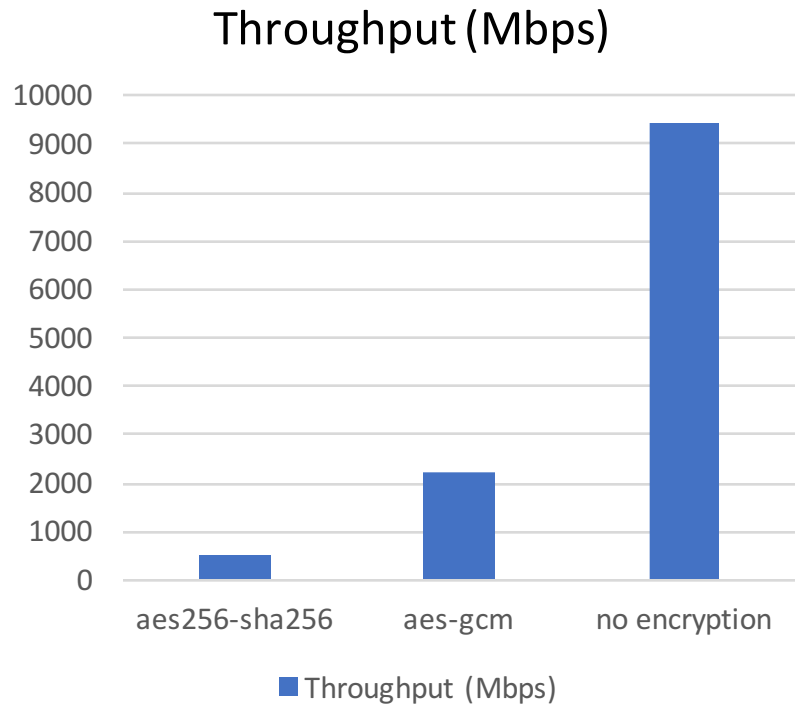
For example:

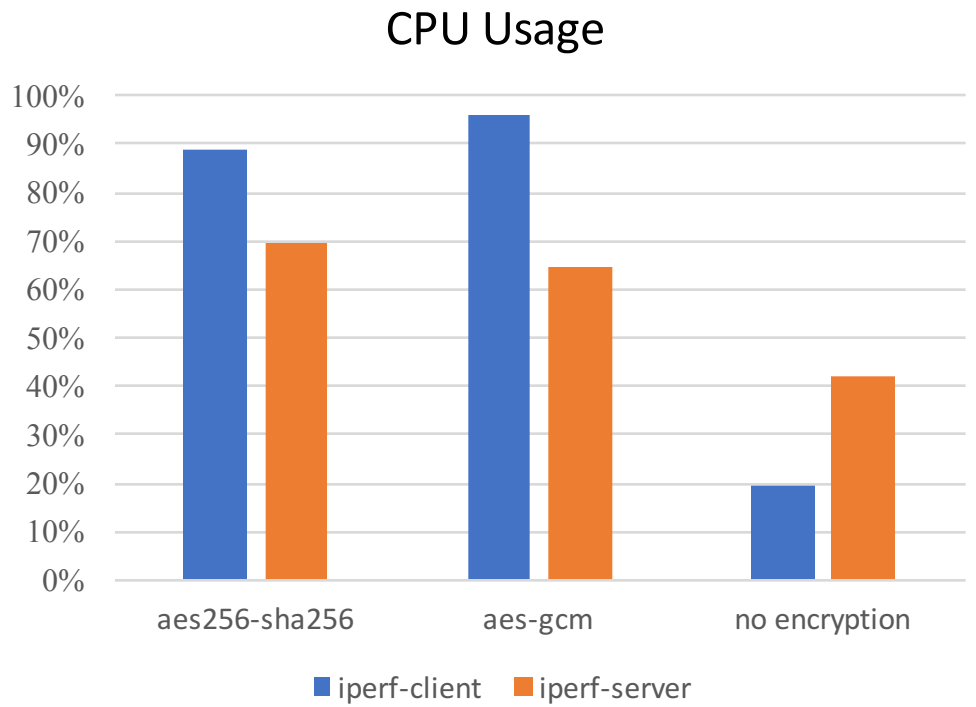
```
$ ovn-nbctl set nb_global . ipsec=true
```



IPsec Evaluation

- Environment: StrongSwan 5.3.5, Linux 4.4.0, Intel Xeon 2 GHz, 10 Gbps NIC
- iperf generates TCP stream (window size: 85KB), which is encrypted in a single core





Current Status

- Compatible with StrongSwan and LibreSwan IKE daemon
- Packages for Ubuntu and Fedora
- Tutorials on using OVS/OVN IPsec
- Need to use OVS out-of-tree kernel module

Possible Extensions

More flexible tunnel encryption policies:

- Only encrypting tunnel traffic between certain hypervisors
- Only encrypting tunnel traffic from certain logical network

Q&A