

# BPF dynamic pointers

```
struct bpf_dynptr_kern {  
    void *data;  
    u32 size;  
    u32 offset;  
} __aligned(8);
```

# Applications

- Dynamic memory allocations that can be persisted
- Dynamically sized ring buffer reservations
- Parsing packet data dynamically + ergonomically
- Dynamically sized strings

# Examples of use cases APIs

## malloc

```
/* Dynamically allocate memory */  
bpf_dynptr_alloc(size, 0, &ptr);  
  
/* Release the memory allocation */  
bpf_dynptr_put(&ptr);
```

## Packet parsing

```
/* Get a dynptr to __sk_buff data */  
bpf_dynptr_from_skb(skb, &skb_ptr);  
  
/* Get a dynptr to xdp_md data */  
bpf_dynptr_from_xdp(ctx, &xdp_ptr);  
  
/* Advance the ptr into skb->data */  
bpf_dynptr_advance(&skb_ptr, sizeof(struct ethhdr));
```

## Ring buffer

```
/* Reserve a dynamically-sized ringbuf record */  
bpf_ringbuf_reserve_dynptr(&ringbuf, size, 0, &ptr);  
  
/* Write data to reservation */  
bpf_dynptr_write(&ptr, 0, buf, sizeof(buf));  
  
/* Commits the record */  
bpf_ringbuf_submit_dynptr(&ptr, 0);
```