DEVCORE
SECURITY
CONSULTING

# 挑戰百萬賞金！

## 虛擬世界之密室逃脫

Meh

戴夫寇爾股份有限公司

contact@devco.re

2023.03.11
DEVCORE Conference

# 血與淚的故事

# 虛擬世界之密室逃脫

DEVCORE SECURITY CONSULTING

# Virtual Machine Escape

## 虛擬機逃逸

DEVCORE SECURITY CONSULTING


PWN
2
OWN


PWN
2
OWN
ORACLE
VirtualBox

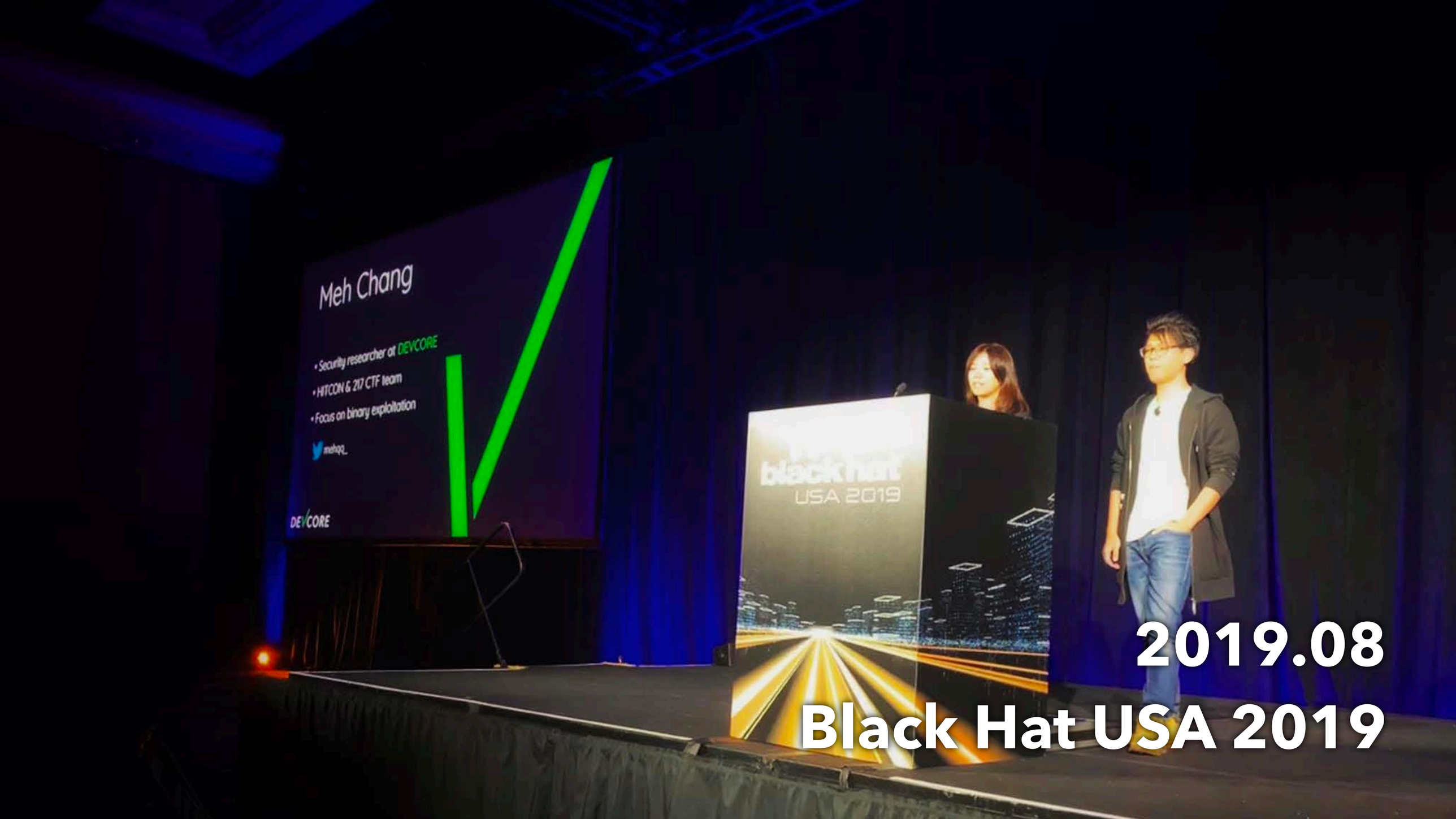
TECH EDITORIALS

Base Score: 9.8 CRITICAL

# Exim Off-by-one RCE: Exploiting CVE-2018-6789 with Fully Mitigations Bypassing

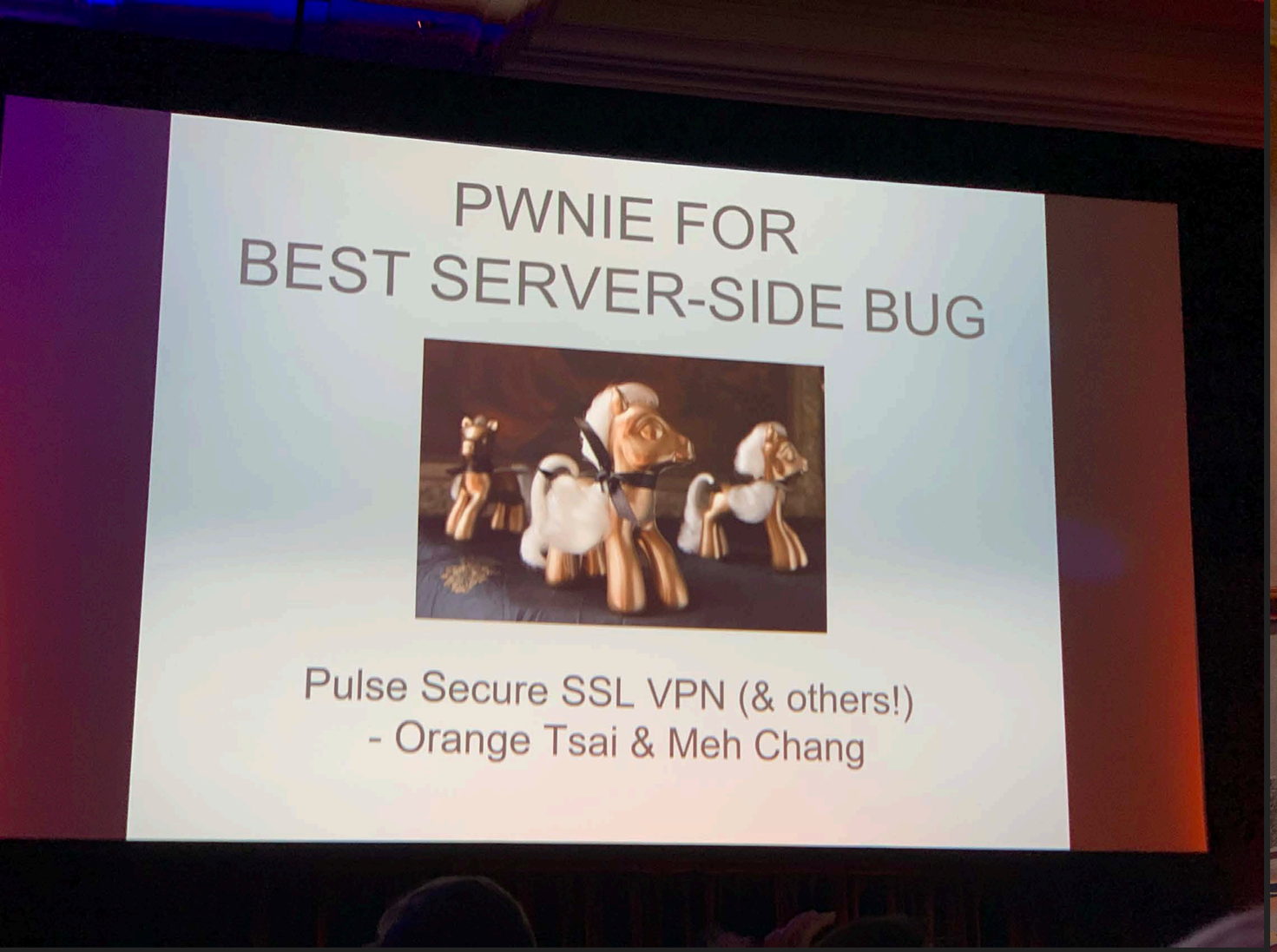
Advisory, Exim, RCE, CVE

By **Meh** on 2018-03-06




# 2019.08
# Black Hat USA 2019

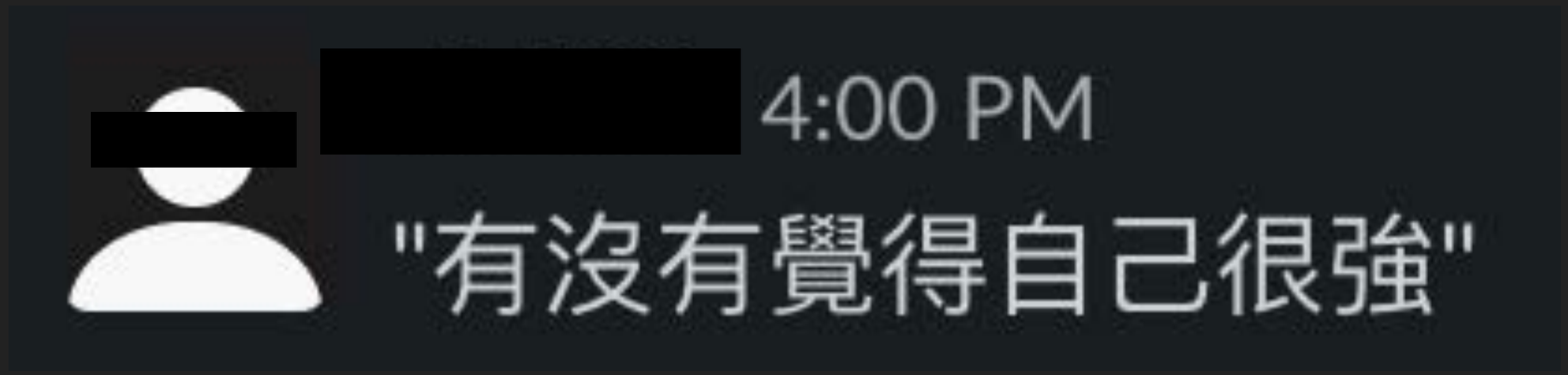

# 2019.08
# DEF CON 2019





**2019.08**

**Pwnie Awards 最佳伺服器漏洞獎**


[redacted]
[redacted]
4:00 PM
"有沒有覺得自己很強"


PWN
2
OWN
ORACLE
VirtualBox

# 有多少時間？

一兩個月小研究案、半年一年大目標？

# 研究的目的？

## 找到影響世界的漏洞、拓展技能樹？

# 想要的攻擊情境？

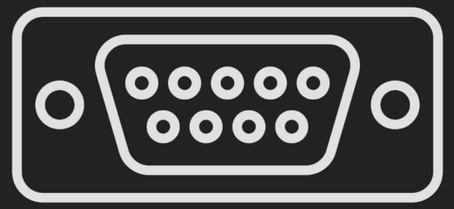
遠端攻擊伺服器、本地端提升權限？


PWN
2
OWN
ORACLE
VirtualBox

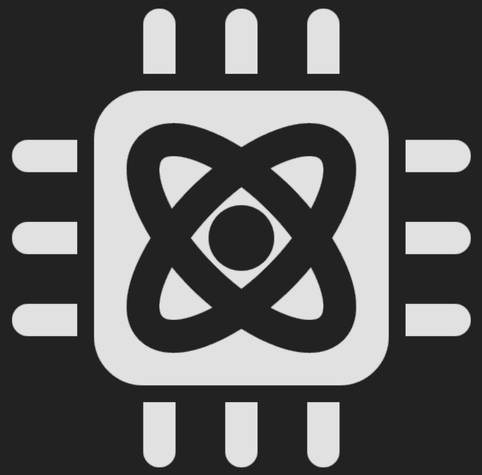

DEVCORE SECURITY CONSULTING

| Target | Prize | Master of Pwn Points | Eligible for Add-on Prize |
|---|---|---|---|
| Oracle VirtualBox | $40,000 | 4 | Yes |
| VMware Workstation | $80,000 | 8 | Yes |
| VMware ESXi | $150,000 | 15 | No |
| Microsoft Hyper-V Client | $250,000 | 25 | Yes |

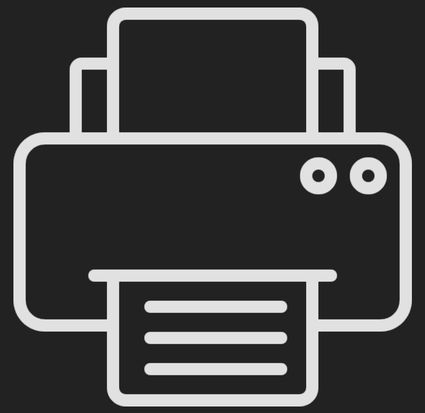
# VMware Workstation Attack Surface



各種接口
(SCSI/EHCI/UHCI/SVGA ...)

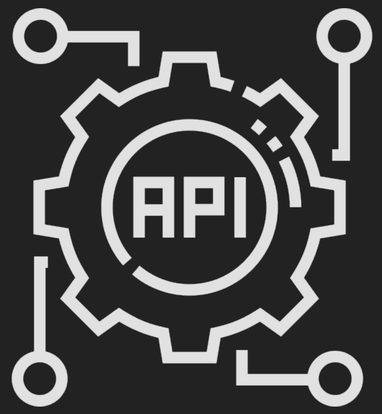


Kernel Module



Virtual Printer

Daemon




RPC

# 熱門度、熟悉度、難易度

# 到底難不難？

- IDB 大小
- Symbol、字串數量
- Coding style
- 可控 input 影響範圍
- 有文件、研究資料、RFC 可參考
- Open source project 改寫
- 過往漏洞多寡

# Evaluating Attack Surface

| | |
|---|---|
| 熱門度 | ★★★★★ |
| 熟悉度 | ★★★ |
| 難易度 | ★★★★ |

各種接口
(SCSI/EHCI/UHCI/SVGA …)

| | |
|---|---|
| 熱門度 | ★★★★ |
| 熟悉度 | ★ |
| 難易度 | ★★★★★ |

Kernel Module

| | |
|---|---|
| 熱門度 | ★★★ |
| 熟悉度 | ★★ |
| 難易度 | ★★★ |

Virtual Printer

| | |
|---|---|
| 熱門度 | ★★ |
| 熟悉度 | ★★★★★ |
| 難易度 | ★★★ |

Daemon

| | |
|---|---|
| 熱門度 | ★★★★★ |
| 熟悉度 | ★★ |
| 難易度 | ★★★ |

RPC

# 冷門、熟悉度高、難度適中

# VMware daemons

- dhcpd
- natd
- authd

# CVE-2020-3947

March 13th, 2020

## VMware Workstation vmnetdhcp Use-After-Free Privilege Escalation Vulnerability

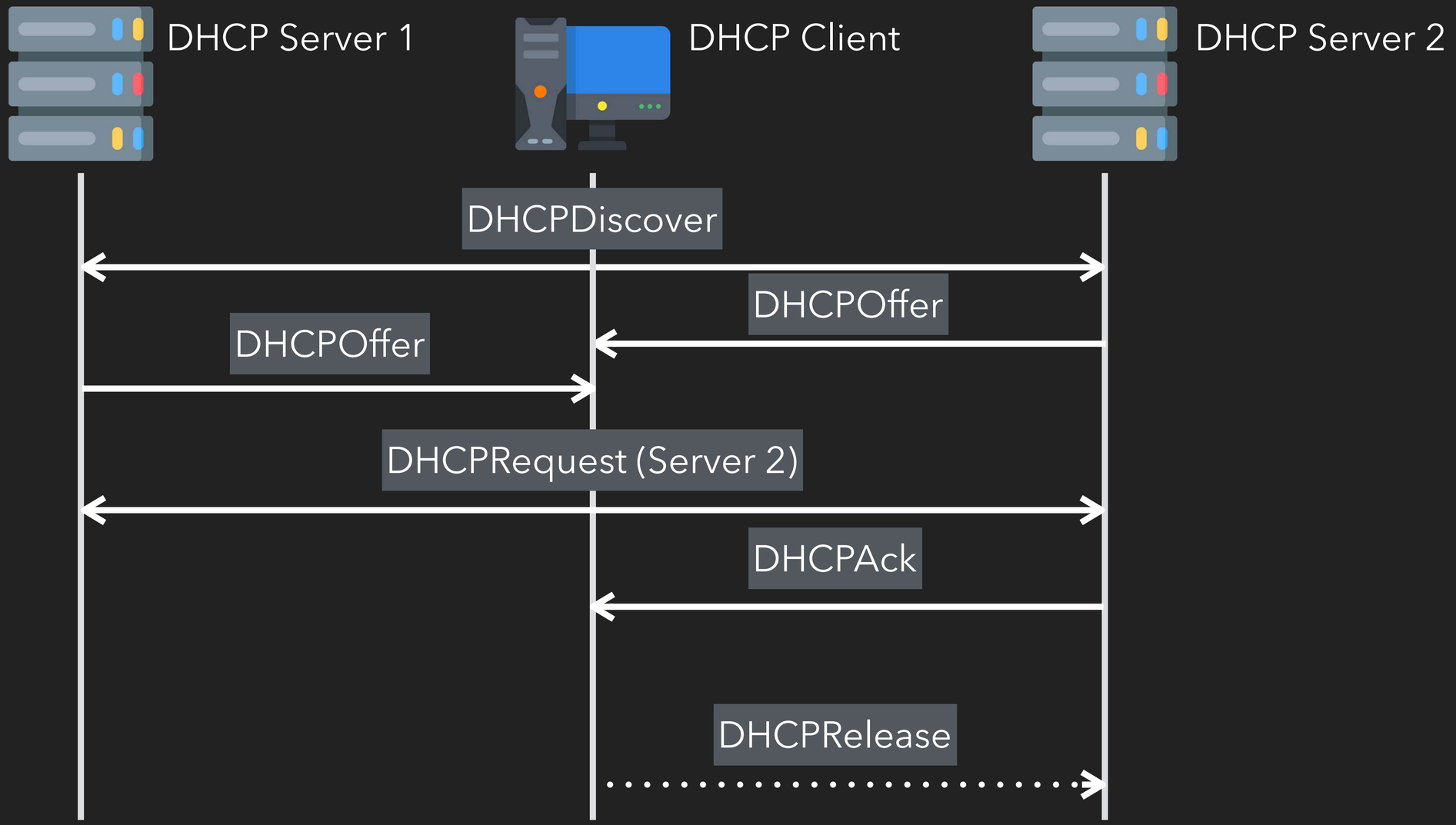
ZDI-20-298
ZDI-CAN-9292

DEVCORE SECURITY CONSULTING

# VMware vmnetdhcp

- Handle DHCP packets from guests
  - Broadcast to UDP port 67
- Enabled for NAT mode
- Based on open source project: isc-projects/dhcp v2
  - Written in C language

# DHCP Protocol



# DHCP lease

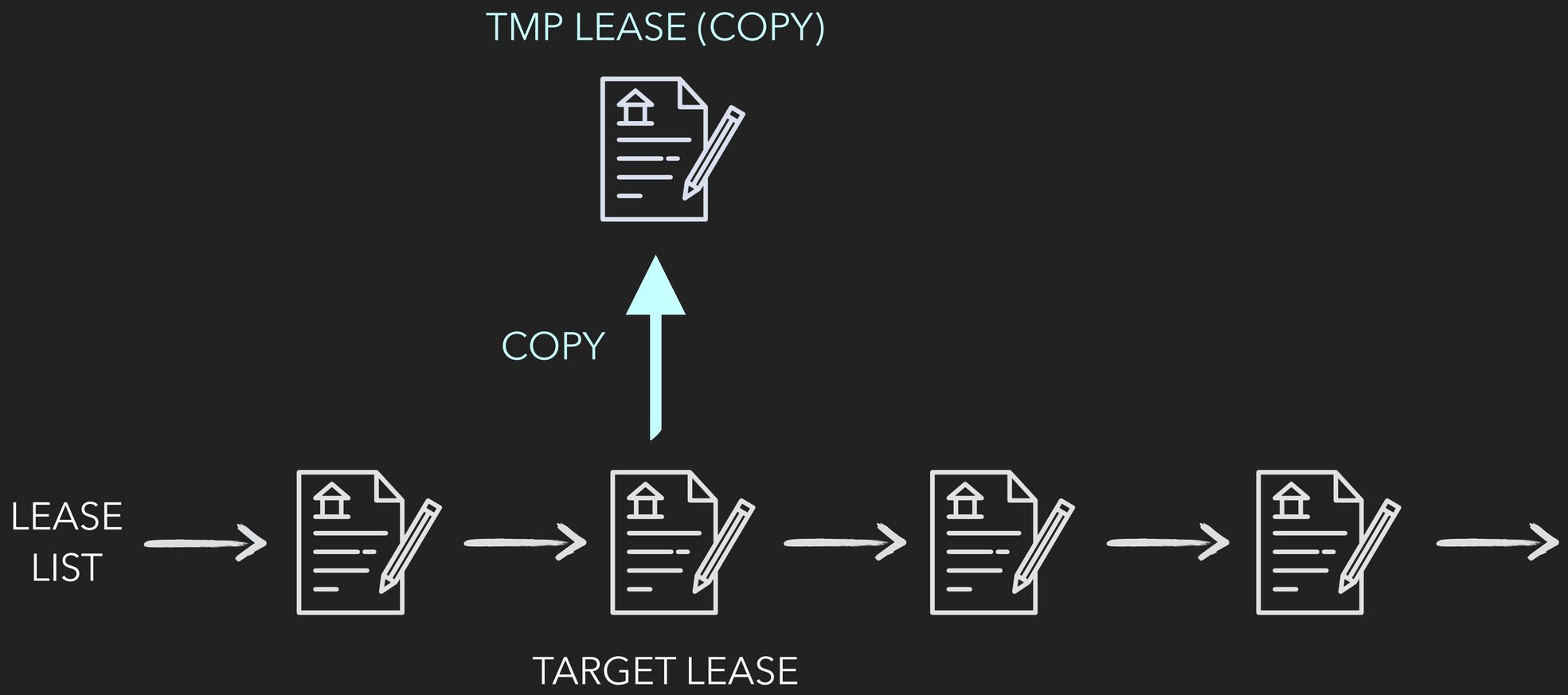
```
Offset  Size  #pragma pack(push, 4)
              struct lease
              {
  0000  0008    struct lease *next;
  0008  0008    struct lease *prev;
  0010  0008    struct lease *n_uid;
  0018  0008    struct lease *n_hw;
  0020  0008    struct lease *waitq_next;
  0028  0014    struct iaddr ip_addr;
  003C  0004    int pad;
  0040  0008    __int64 starts;
  0048  0008    __int64 ends;
  0050  0008    __int64 timestamp;
  0058  0008    char *uid;
  0060  0004    int uid_len;
  0064  0004    int uid_max;
  0068  0020    char uid_buf[32];
  0088  0008    char *hostname;
  0090  0008    char *client_hostname;
  0098  0008    struct host_decl *host;
  00A0  0008    struct subnet *subnet;
  00A8  0008    struct shared_network *shared_network;
  00B0  0012    struct hardware hardware_addr;
  00C4  0004    int flags;
  00C8  0008    struct lease_state *state;
        00D0  };
              #pragma pack(pop)
```

# release_lease

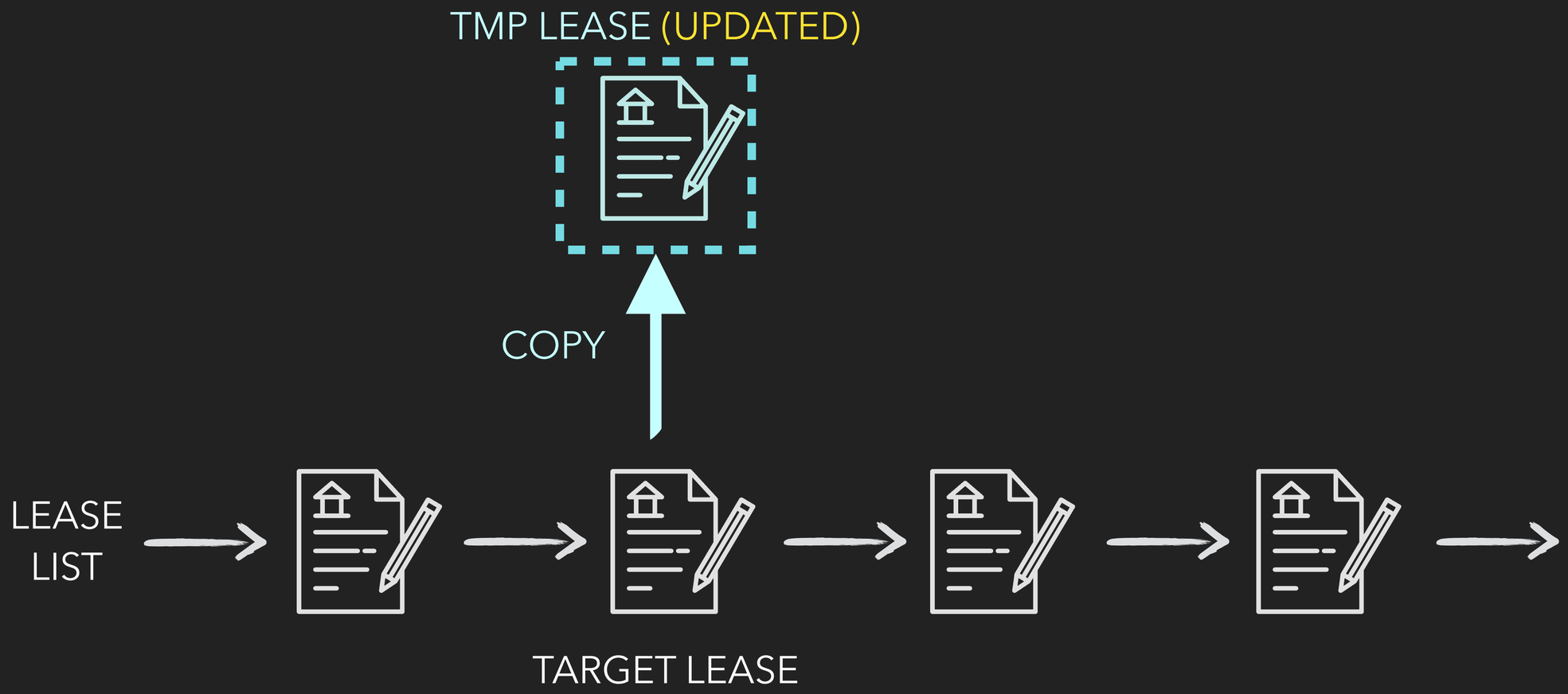
```
memcpy(&tmp_lease, lease, sizeof(tmp_lease));
if ( tmp_lease.ends > g_start_time )
{
  tmp_lease.ends = g_start_time;
  supersede_lease(lease, &tmp_lease, 1);
}
```

# release_lease

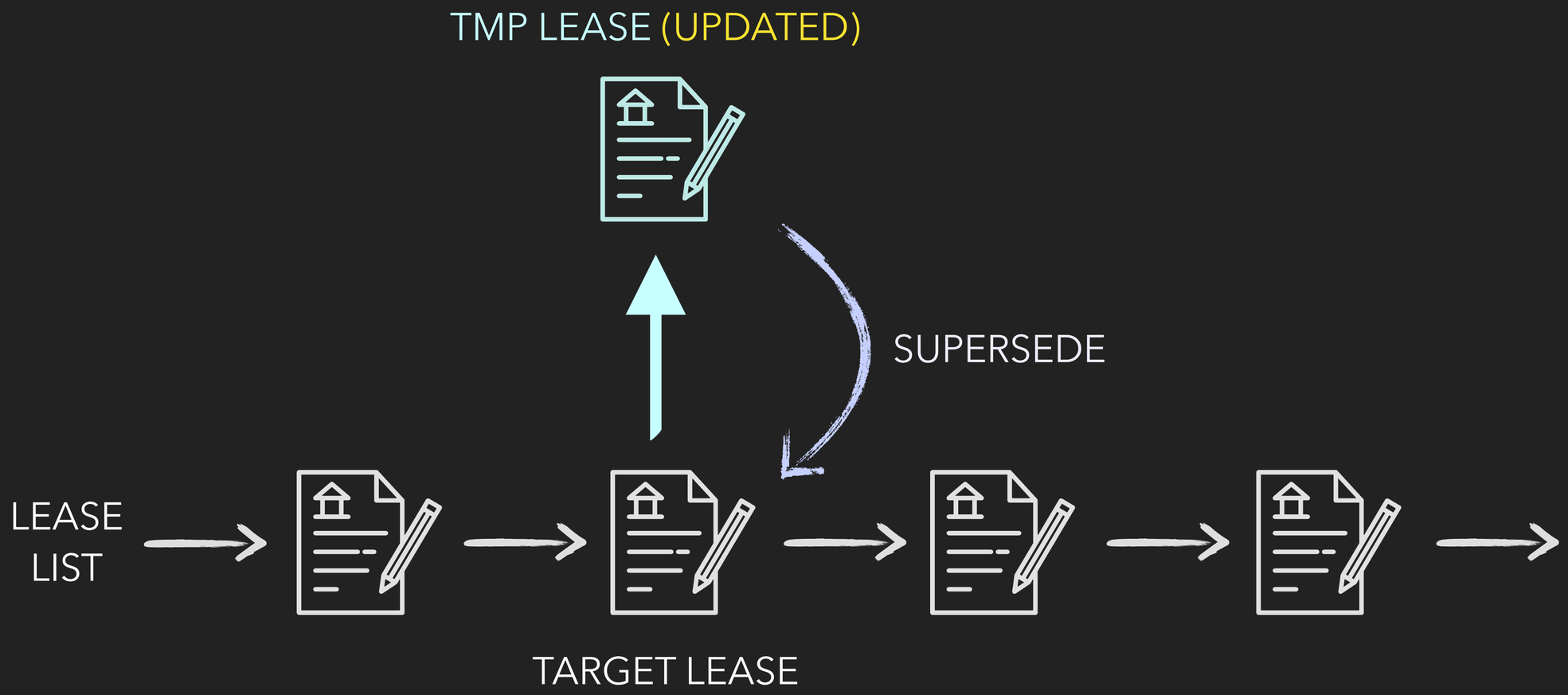
# release_lease



# release_lease



# release_lease



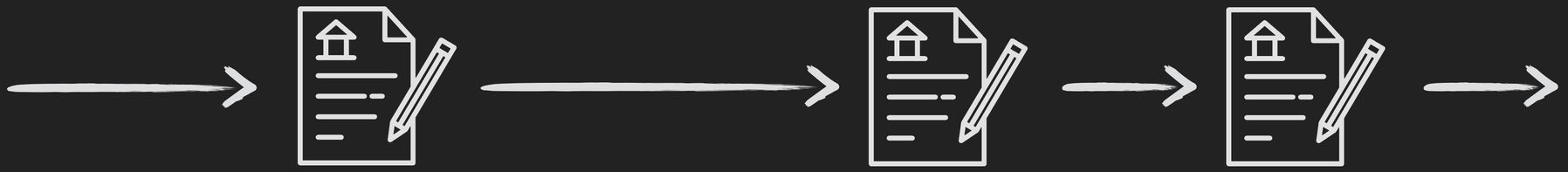
# supersede_lease


# supersede_lease

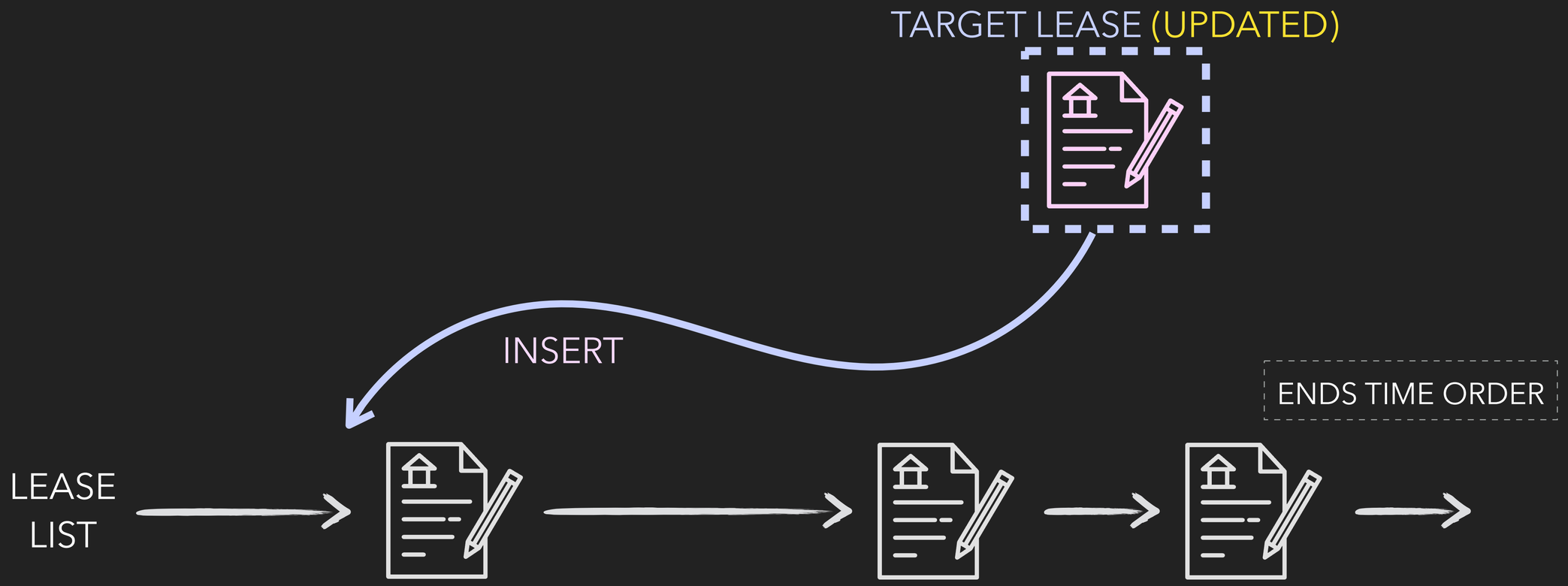
TMP LEASE (UPDATED)

TARGET LEASE (UPDATED)

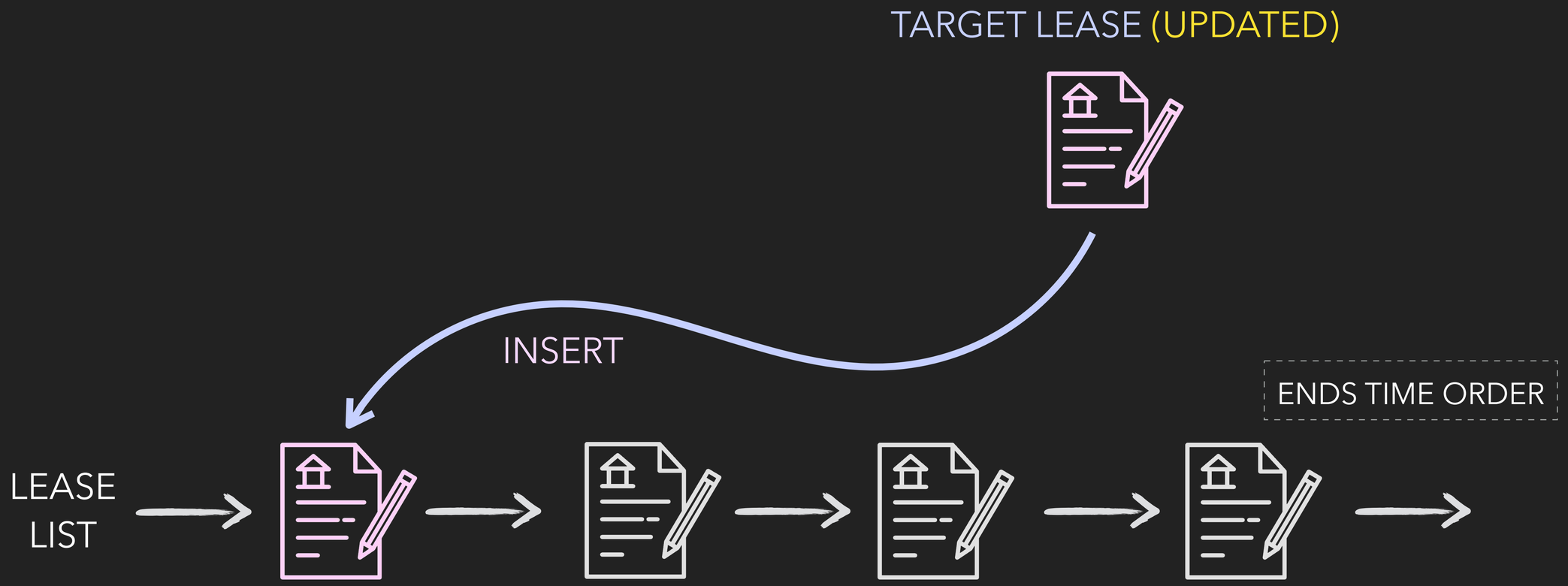
LEASE LIST



# supersede_lease



# supersede_lease



# Vulnerability

TMP LEASE (UPDATED)

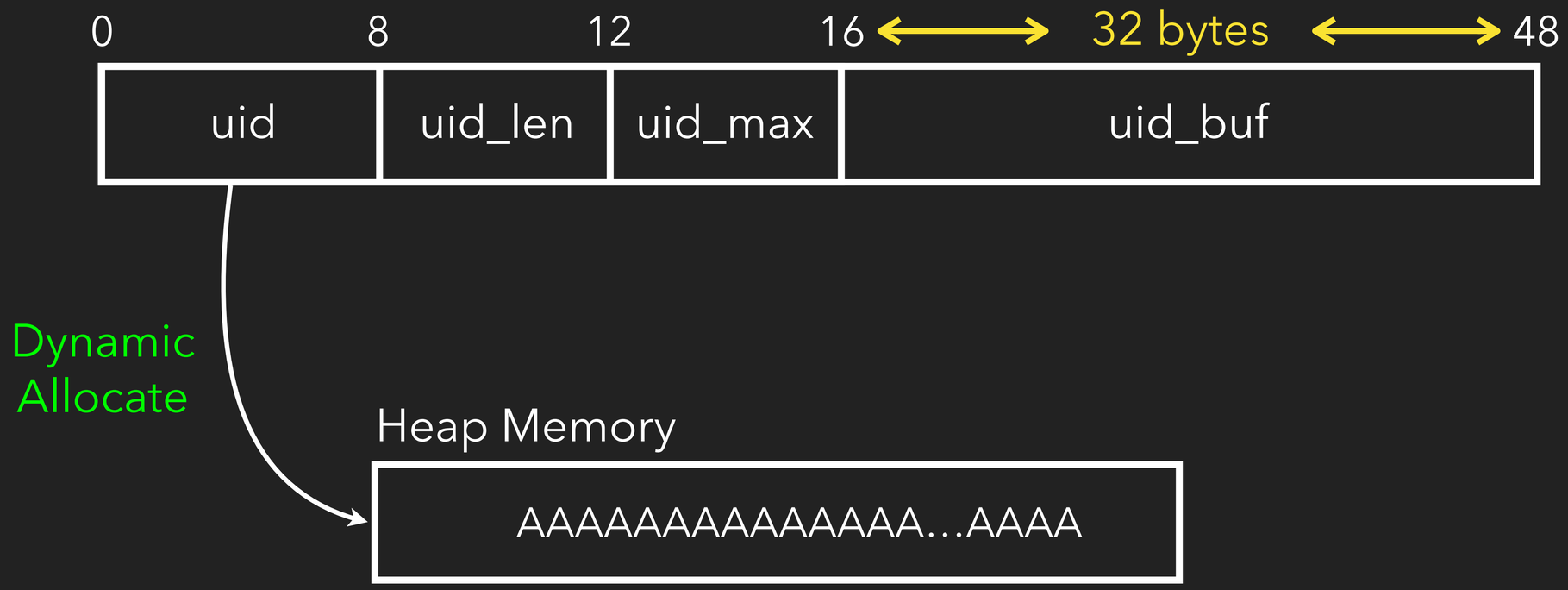
SUPERSEDE

TARGET LEASE

# DHCP lease

```
003C 0004   int pad;
0040 0008   __int64 starts;
0048 0008   __int64 ends;
0050 0008   __int64 timestamp;
0058 0008   char *uid;
0060 0004   int uid_len;
0064 0004   int uid_max;
0068 0020   char uid_buf[32];
0088 0008   char *hostname;
0090 0008   char *client_hostname;
0098 0008   struct host_decl *host;
00A0 0008   struct subnet *subnet;
00A8 0008   struct shared_network *shared_network;
00B0 0012   struct hardware hardware_addr;
```

# UID String


0
8
12
16
32 bytes
48
uid
uid_len
uid_max
uid_buf
Dynamic
Allocate
Heap Memory
AAAAAAAAAAAAAA…AAAA

# Lease Copy


TARGET LEASE
uid
uid_len
uid_max
uid_buf
TMP LEASE
uid
uid_len
uid_max
uid_buf
Heap Memory
AAAAAAAAAAAAAA…AAAA

# release_lease

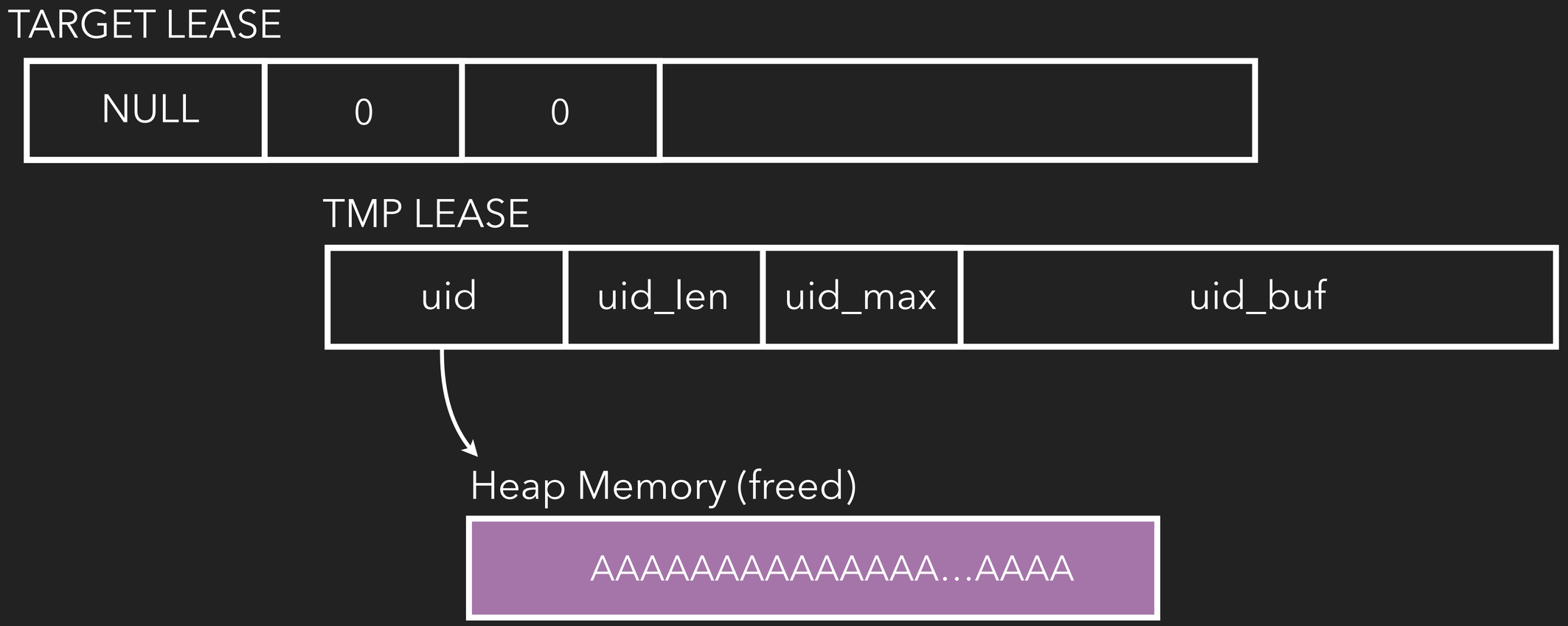
```
memcpy(&tmp_lease, lease, sizeof(tmp_lease));
if ( tmp_lease.ends > g_start_time )
{
  tmp_lease.ends = g_start_time;
  supersede_lease(lease, &tmp_lease, 1);
}
```

# Clear Old UID

```
if(l->uid != l->uid_buf)
{
  free(l->uid);
}
l->uid = 0;
…
if(lt->uid)
  l->uid = lt->uid;
```

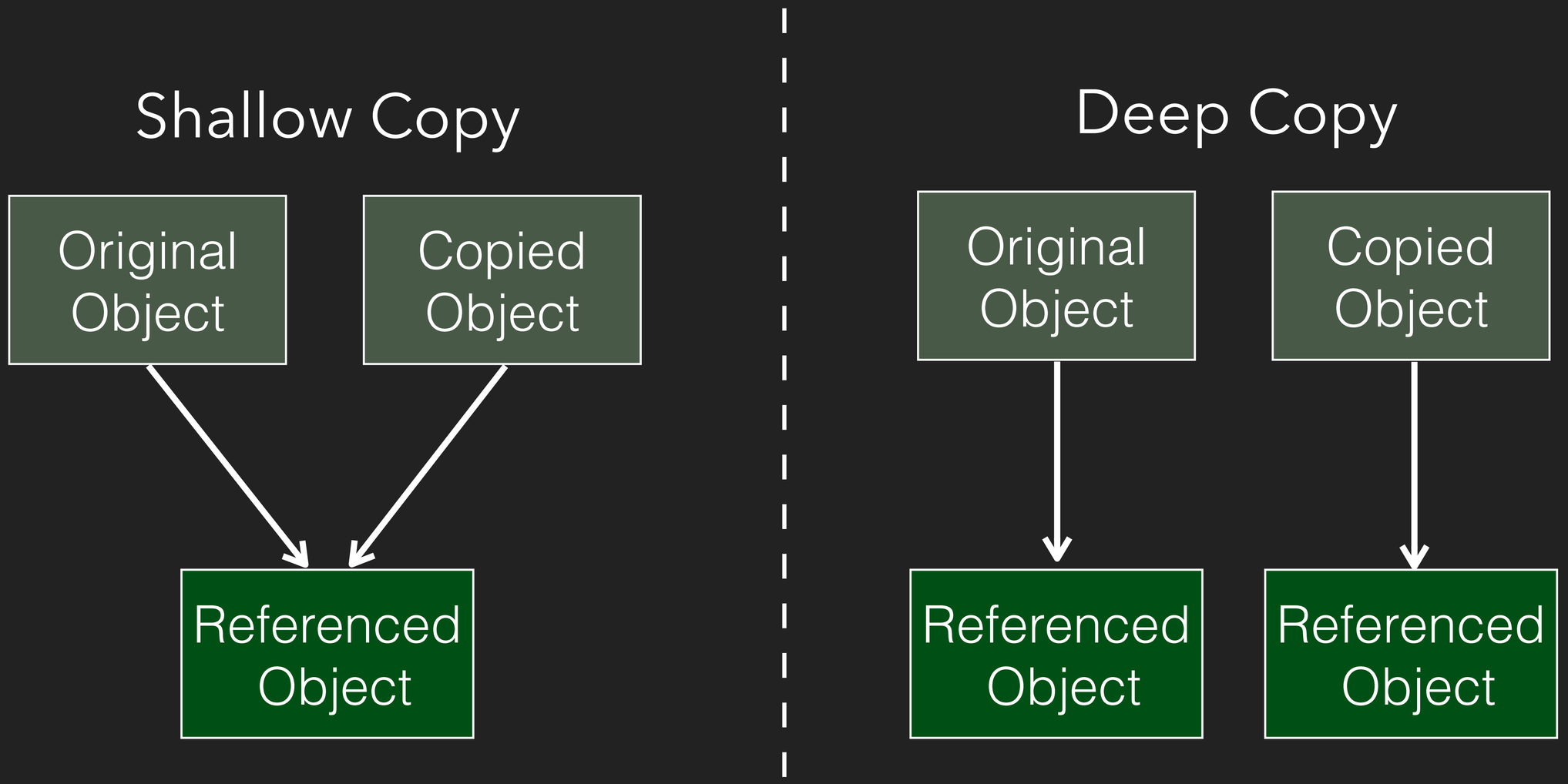
# Free Old uid pointer

# Clear Old uid structure



# Update target uid

# Shallow Copy & Deep Copy



# Trigger Vulnerability

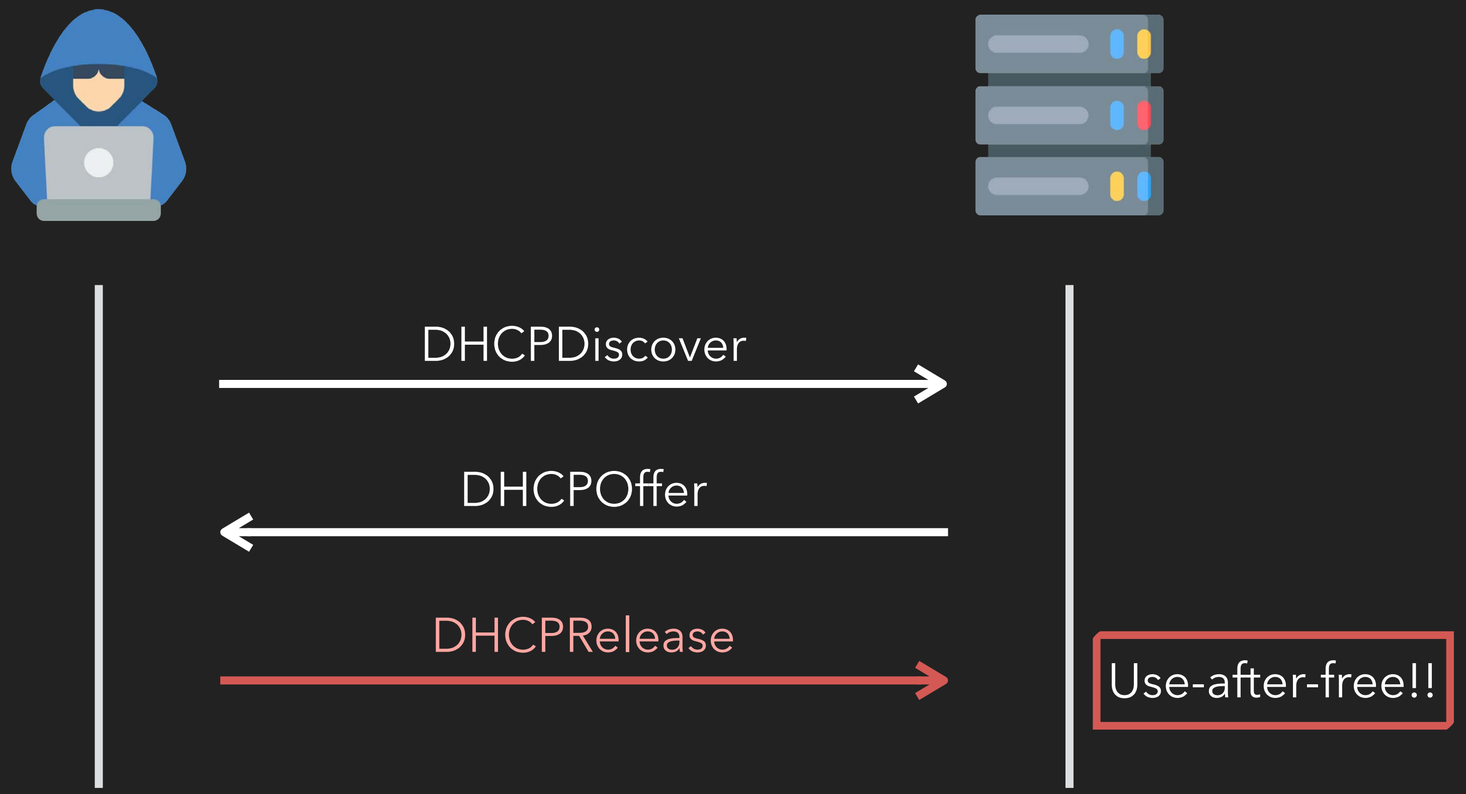





Image Source:
Flaticon.com

March 13th, 2020

**meh** 5:15 PM

我現在可以讓他 daemon 消失惹

[redacted] 5:16 PM

賀

DOS

# CVE-2020-3947

March 13th, 2020

## VMware Workstation vmnetdhcp Use-After-Free Privilege Escalation Vulnerability

ZDI-20-298
ZDI-CAN-9292

DEVCORE SECURITY CONSULTING

March 13th, 2020

**meh** 5:15 PM

我現在可以讓他 daemon 消失惹

[redacted] 5:16 PM

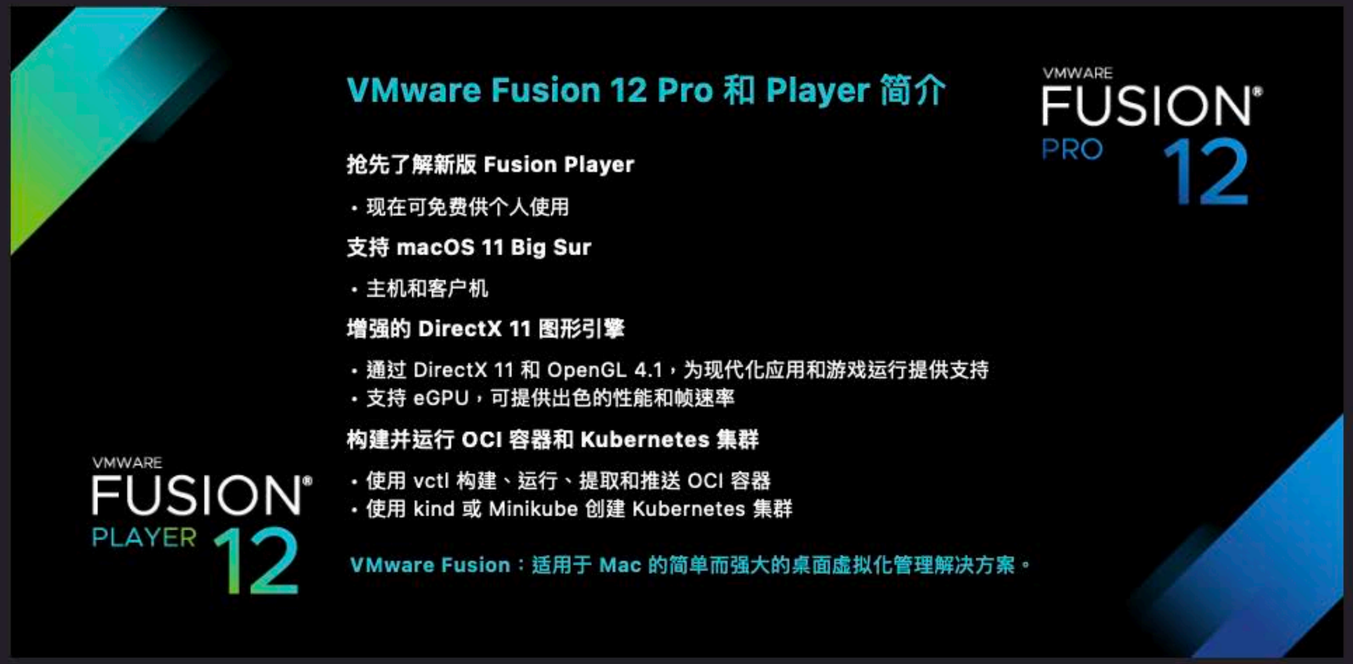
賀

DOS

軟體更新

**A new version of VMware Fusion is available!**

VMware Fusion 12.0.0 is now available (you have 11.5.3).




略過此版本　　稍後提醒我　　Learn More...

DEVCORE SECURITY CONSULTING

# 撞洞，被修了QQ

# “排 heap 就是浪費時間，不如再找一個洞”

*- Angelboy 10.17.2022*

# 痛定思痛

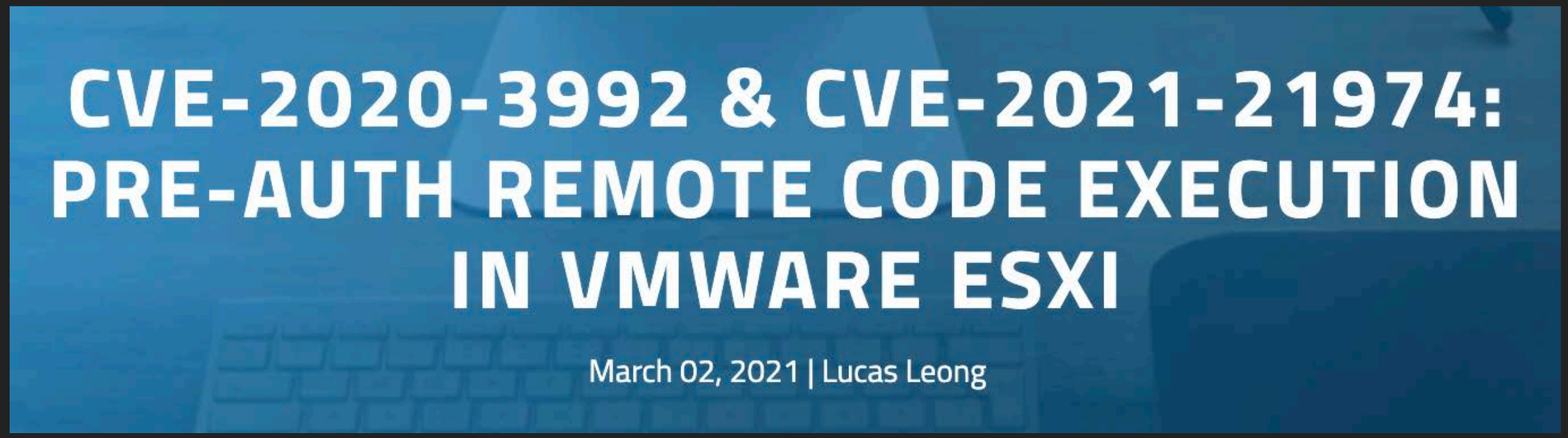
# 換個顏色

vmware®
ESXi

DEVCORE SECURITY CONSULTING

# 修了又修的洞

- CVE-2020-3992 Use-after-free
  - Python backdoor (2022)
- CVE-2021-21974 Heap overflow
  - Ransomware ESXiArgs (2023)



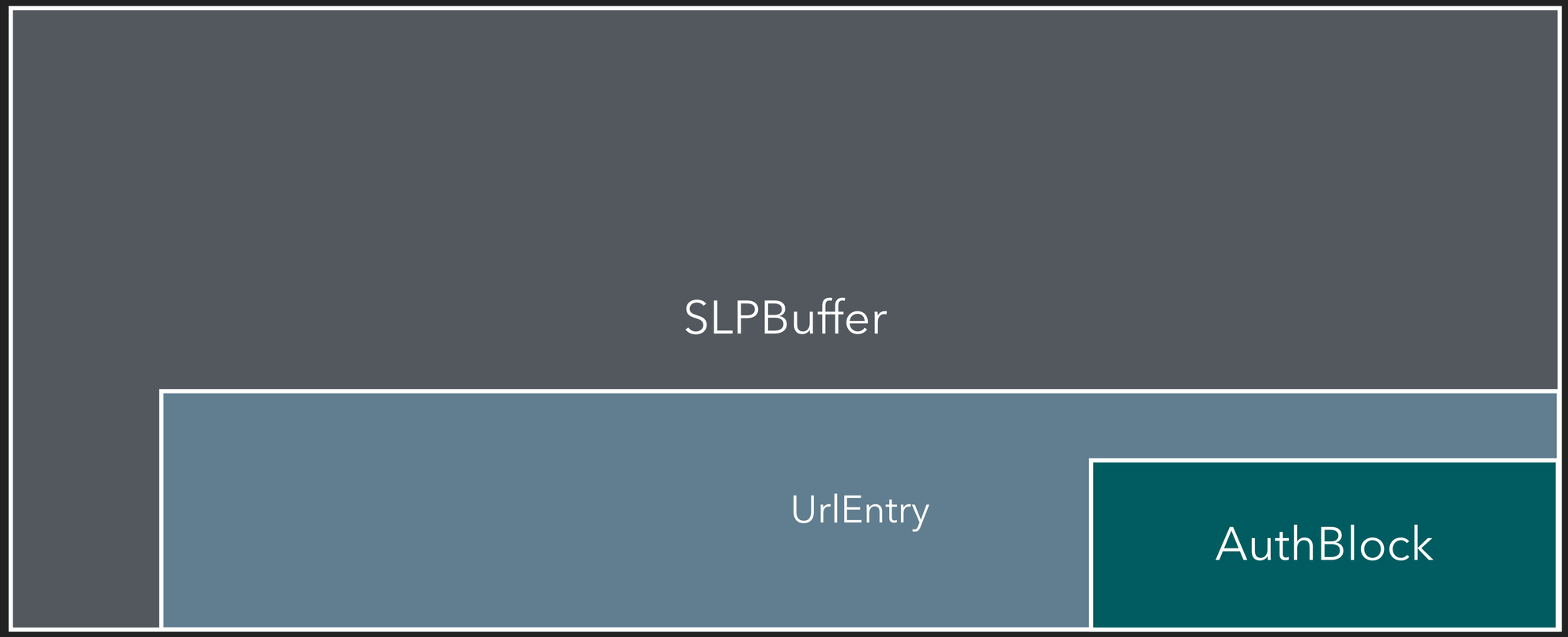
# VMware slpd

- Service Location Protocol
  - Service discovery protocol for local area network
  - Host TCP / UDP port 427
- Modified from OpenSLP v1.0.1 (2001)
- Full Protection

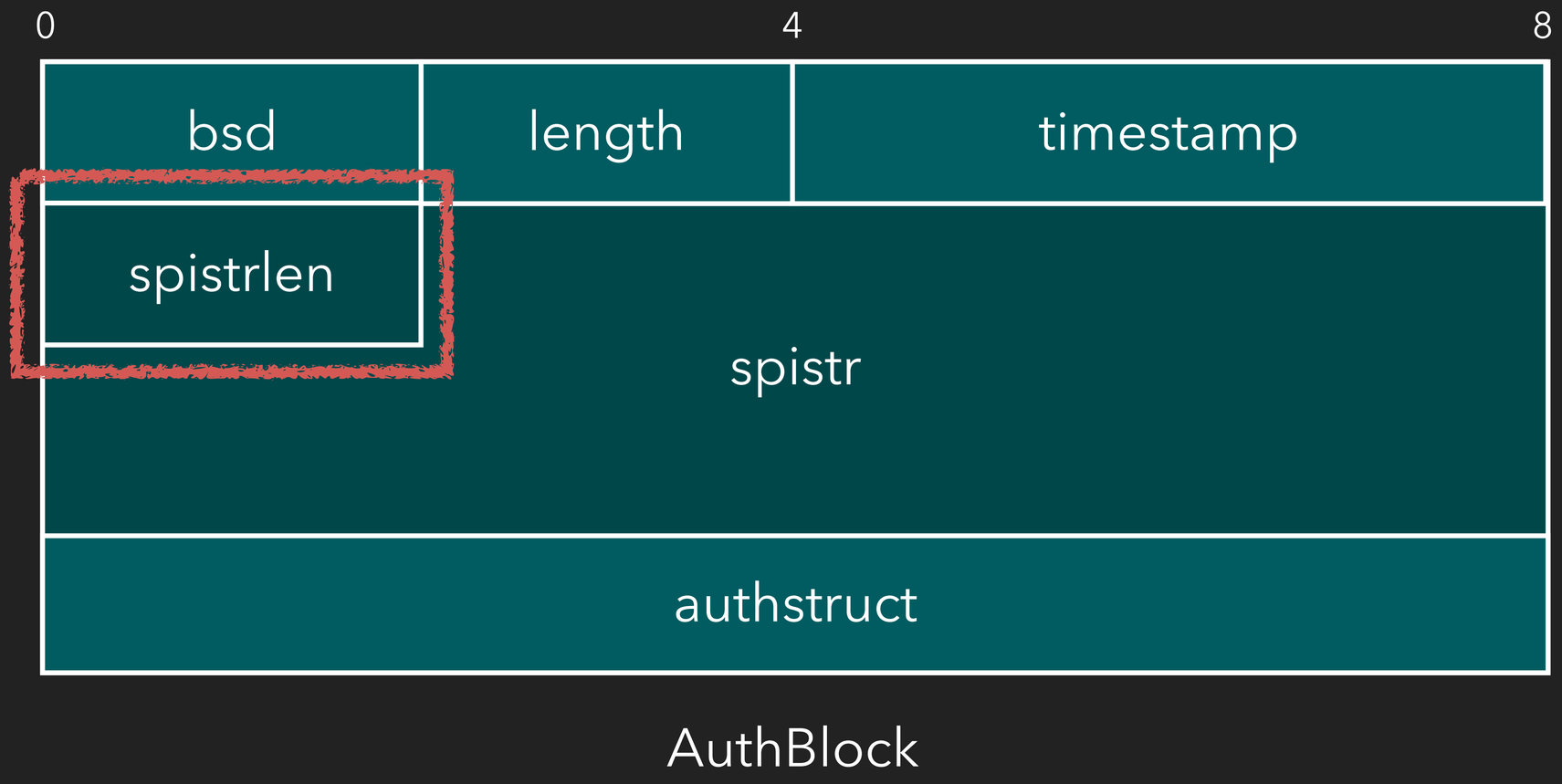
openSLP 

# Parsing SLP Message

- ParseSrvReg -> ParseUrlEntry -> ParseAuthBlock



# ParseAuthBlock



# Parsing spistr

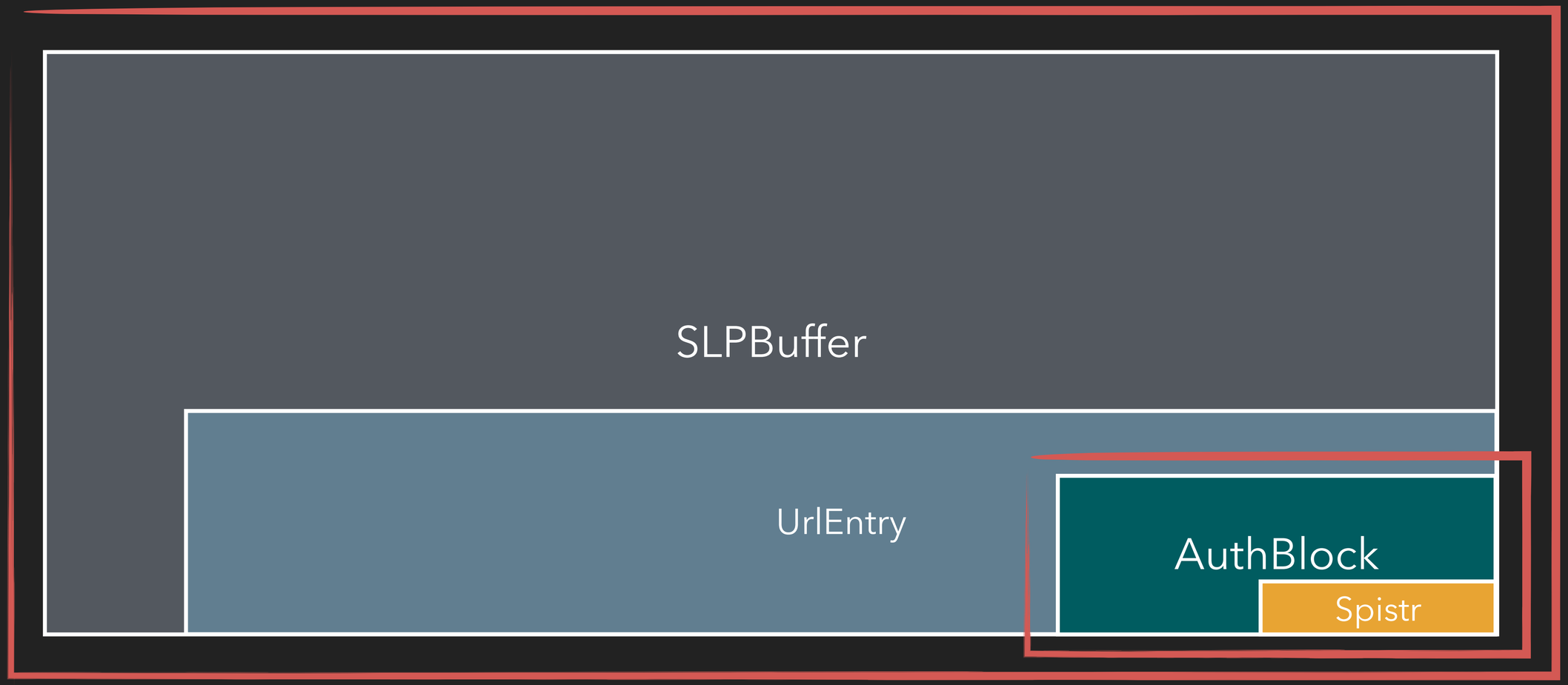
```
authblock->spistrlen    = AsUINT16(buffer->curpos + 8);
authblock->spistr       = buffer->curpos + 10;

if(authblock->spistrlen > buffer->end - buffer->curpos + 10)
{
    return SLP_ERROR_PARSE_ERROR;
}
```

**Check spistr is in SLPBuffer**

# Parsing SLP Message

- ParseSrvReg -> ParseUrlEntry -> ParseAuthBlock



# Vulnerability in SLPAuthVerifyDAAdvert

**authblock** length **spistr** length

```
signaturelen = autharray[i].length - (autharray[i].spistrlen +
10);
```

**Bug if spistrlen +10 > authblock length**

```
if(SLPCryptoDSAVerify(key, digest, sizeof(digest),
                      autharray[I].authstruct,
                      signaturelen))
{
   break;
}
```

# How to trigger

- SLP v2 security feature

```
#ifdef ENABLE_SLPv2_SECURITY
  errorcode = SLPAuthVerifyDAAdvert(G_SlpdSpiHandle, 0,
                                    &(message->body.daadvert));
#endif
```

# VMWare 沒開 🙃

看 source code 偷吃步失敗 QQ

# Conclusion

決定目標前先問問自己：我有多少時間？想玩到什麼深度？想學到什麼？

- 剛好的挑戰正是進化的關鍵！

# Conclusion

研究過程中撞洞、被 patch、fake bug 都是常有的事

- 這些都是過程，重要的是過程中學到什麼

DEVCORE
SECURITY
CONSULTING

# Thank you for listening!

Q&A

**To be continued …**