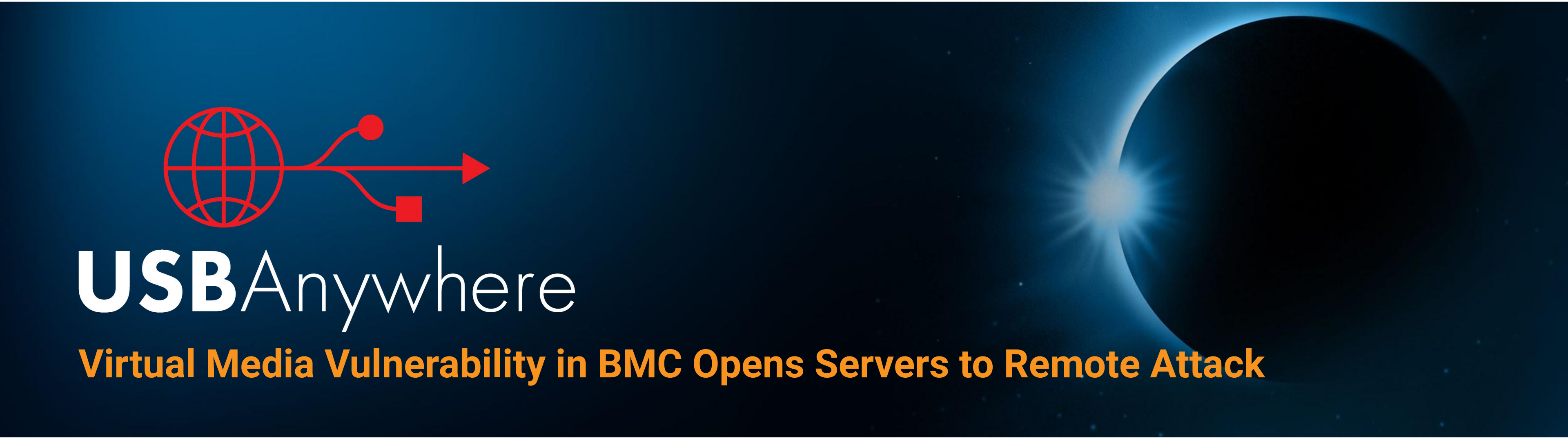


USBAnywhere

Virtual Media Vulnerability in BMC Opens Servers to Remote Attack

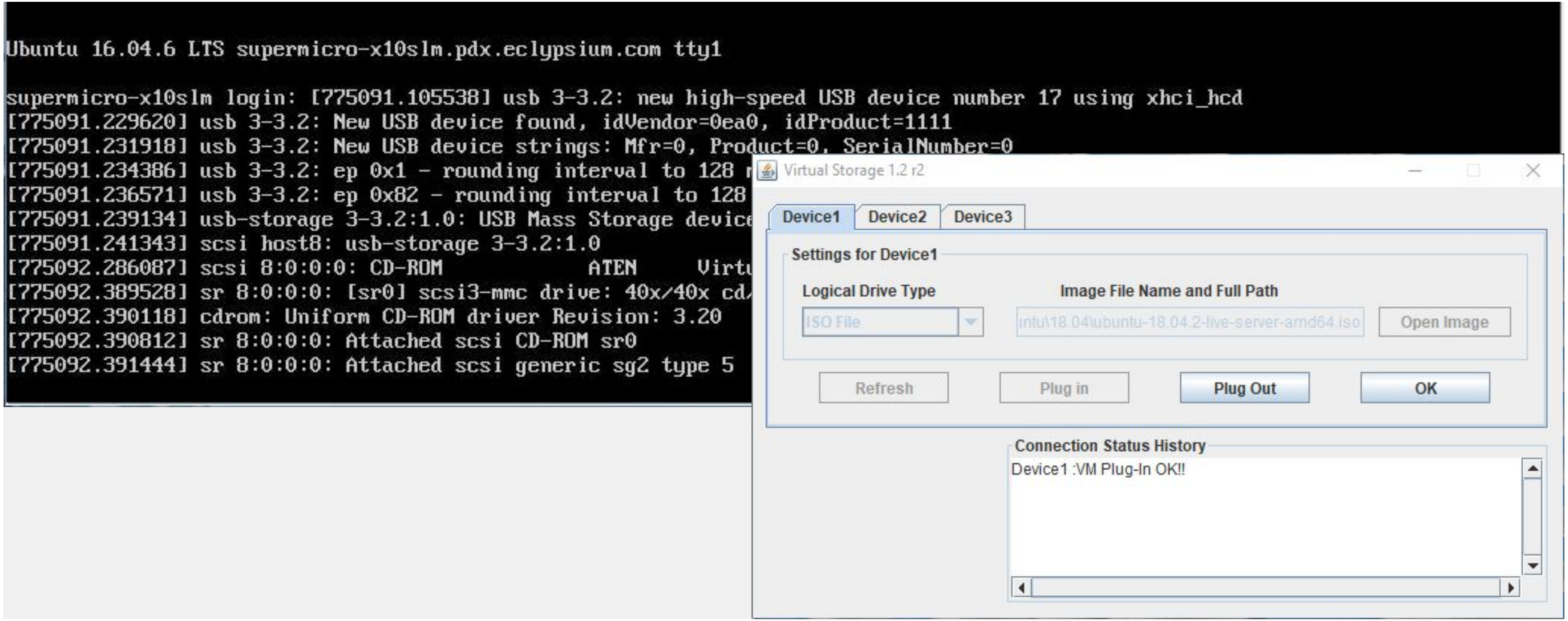
Responsible Disclosure Timeline

USBAnywhere

- **2019-06-19** - Vulnerability reported to Supermicro by Eclyspium
- **2019-07-09** - Additional findings reported to Supermicro
- **2019-07-29** - Supermicro acknowledges report and asks for verification of fixes in beta firmware
- **2019-07-30** - Eclypsium verified provided fixes resolve the vulnerability
- **2019-08-16** - Eclypsium notifies CERT/CC due to large number of public systems affected
- **2019-08-16** - Supermicro confirms intent to publicly release firmware before September 3rd
- **2019-08-23** - Eclypsium discovers that Supermicro X9 platforms are also affected.
- **2019-09-03** - Details published and presented at Open Source Firmware Conference.

How Does Virtual Media Work?

USBAnywhere



```
Ubuntu 16.04.6 LTS supermicro-x10slm.pdx.eclypsium.com tty1

supermicro-x10slm login: [775091.105538] usb 3-3.2: new high-speed USB device number 17 using xhci_hcd
[775091.229620] usb 3-3.2: New USB device found, idVendor=0ea0, idProduct=1111
[775091.231918] usb 3-3.2: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[775091.234386] usb 3-3.2: ep 0x1 - rounding interval to 128 r
[775091.236571] usb 3-3.2: ep 0x82 - rounding interval to 128
[775091.239134] usb-storage 3-3.2:1.0: USB Mass Storage devic
[775091.241343] scsi host8: usb-storage 3-3.2:1.0
[775092.286087] scsi 8:0:0:0: CD-ROM            ATEN     Virtu
[775092.389528] sr 8:0:0:0: [sr0] scsi3-mmc drive: 40x/40x cd/
[775092.390118] cdrom: Uniform CD-ROM driver Revision: 3.20
[775092.390812] sr 8:0:0:0: Attached scsi CD-ROM sr0
[775092.391444] sr 8:0:0:0: Attached scsi generic sg2 type 5
```

Settings for Device1

Logical Drive Type: ISO File

Image File Name and Full Path: intu\18.04\ubuntu-18.04.2-live-server-amd64.iso

Connection Status History

Device1 :VM Plug-In OK!!

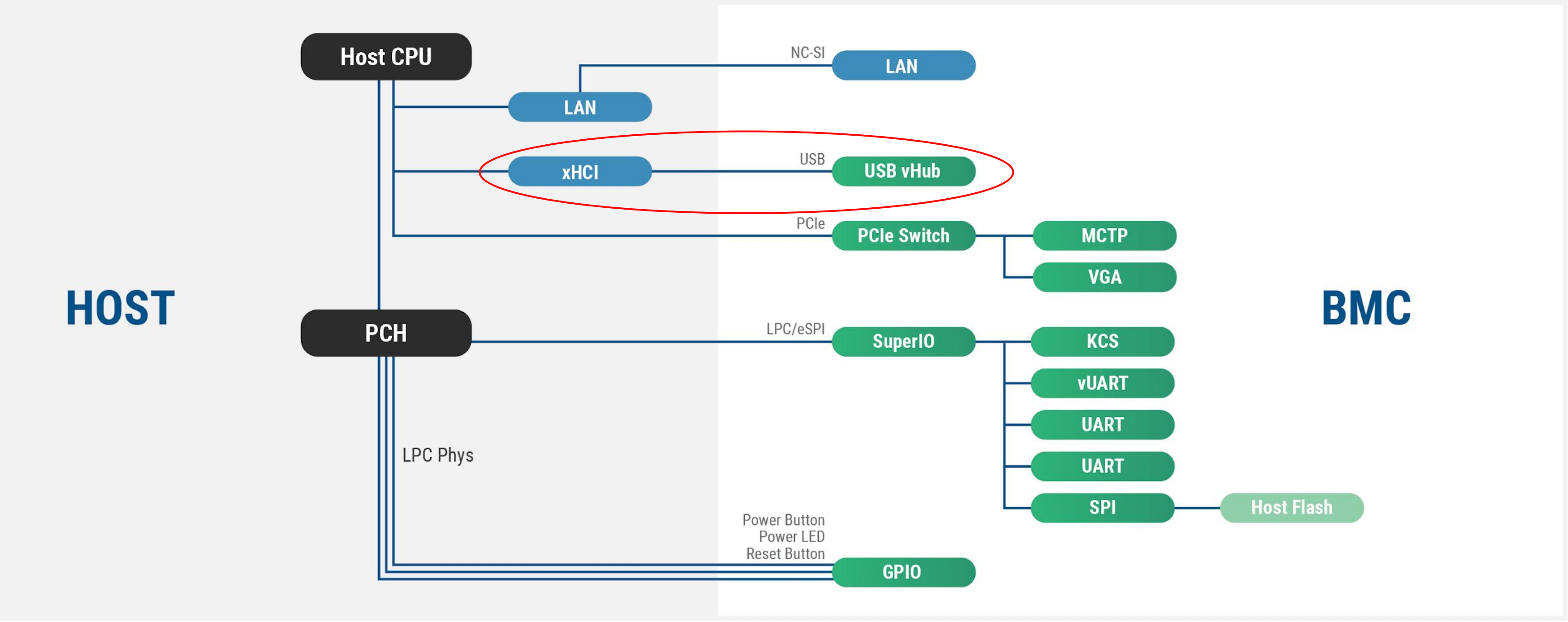
What We Know

USBAnywhere

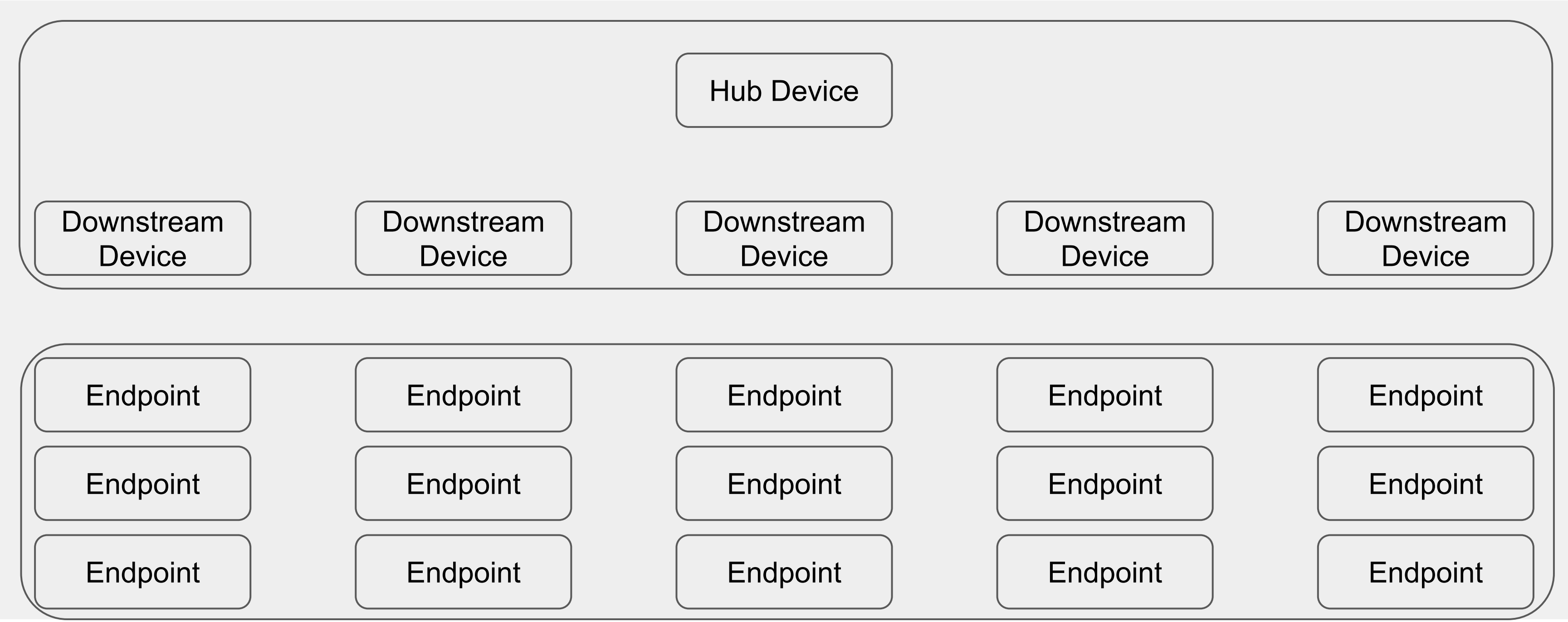
- Java applet launched via JNLP
- ISO located on system running Java applet
- “Plugging in” the ISO attaches a USB device on the remote host
  - **USB Class:** Mass Storage
  - **USB Subclass:** SCSI Transparent Command Set
  - **SCSI PDT[1]:** Multimedia Commands (MMC)
  - ATEN Virtual CDROM
- iKVM also uses USB for virtual keyboard and mouse

[1] SCSI Peripheral Device Type

Connections Between Host and BMC



Virtual USB Hub



Hub Device

Downstream Device

Downstream Device

Downstream Device

Downstream Device

Downstream Device

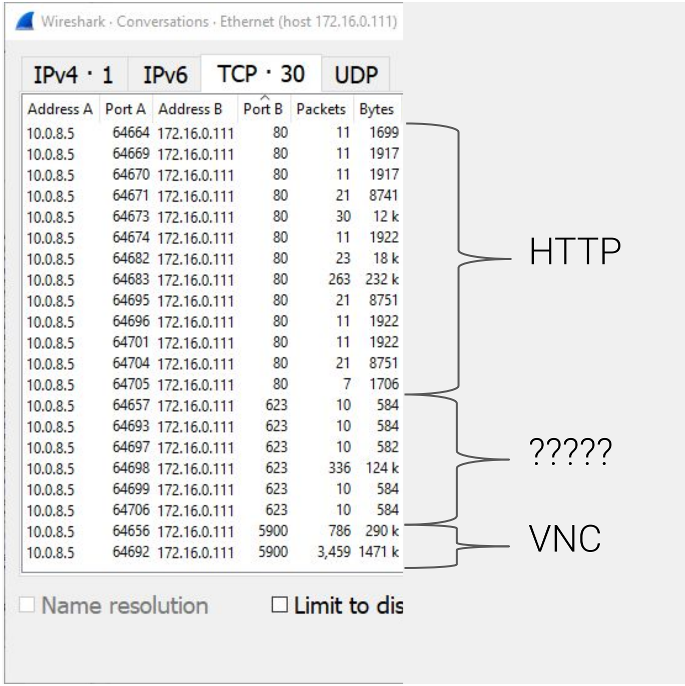
Endpoint Endpoint Endpoint Endpoint Endpoint

Endpoint Endpoint Endpoint Endpoint Endpoint

Endpoint Endpoint Endpoint Endpoint Endpoint

What’s Going Over the Network?

- HTTP
  - JNLP launcher
  - Java JARs
- VNC
  - iKVM
- TCP/623
  - Started when Virtual Media UI opened

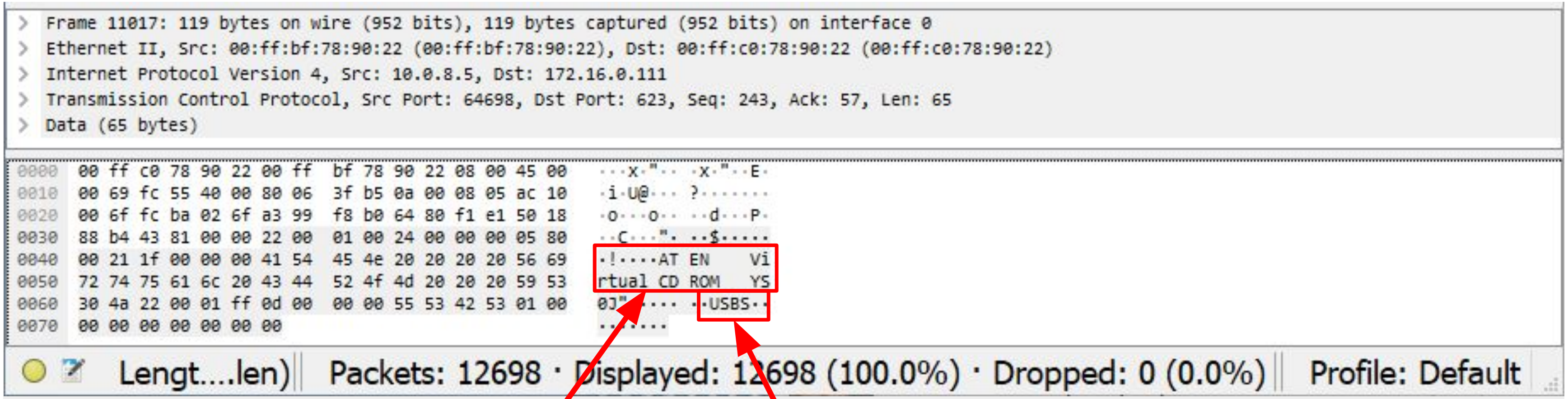


Wireshark · Conversations · Ethernet (host 172.16.0.111)

Address A	Port A	Address B	Port B	Packets	Bytes
10.0.8.5	64664	172.16.0.111	80	11	1699
10.0.8.5	64669	172.16.0.111	80	11	1917
10.0.8.5	64670	172.16.0.111	80	11	1917
10.0.8.5	64671	172.16.0.111	80	21	8741
10.0.8.5	64673	172.16.0.111	80	30	12 k
10.0.8.5	64674	172.16.0.111	80	11	1922
10.0.8.5	64682	172.16.0.111	80	23	18 k
10.0.8.5	64683	172.16.0.111	80	263	232 k
10.0.8.5	64695	172.16.0.111	80	21	8751
10.0.8.5	64696	172.16.0.111	80	11	1922
10.0.8.5	64701	172.16.0.111	80	11	1922
10.0.8.5	64704	172.16.0.111	80	21	8751
10.0.8.5	64705	172.16.0.111	80	7	1706
10.0.8.5	64657	172.16.0.111	623	10	584
10.0.8.5	64693	172.16.0.111	623	10	584
10.0.8.5	64697	172.16.0.111	623	10	582
10.0.8.5	64698	172.16.0.111	623	336	124 k
10.0.8.5	64699	172.16.0.111	623	10	584
10.0.8.5	64706	172.16.0.111	623	10	584
10.0.8.5	64656	172.16.0.111	5900	786	290 k
10.0.8.5	64692	172.16.0.111	5900	3,459	1471 k

□ Name resolution □ Limit to dis

Unencrypted USB over TCP?!?!



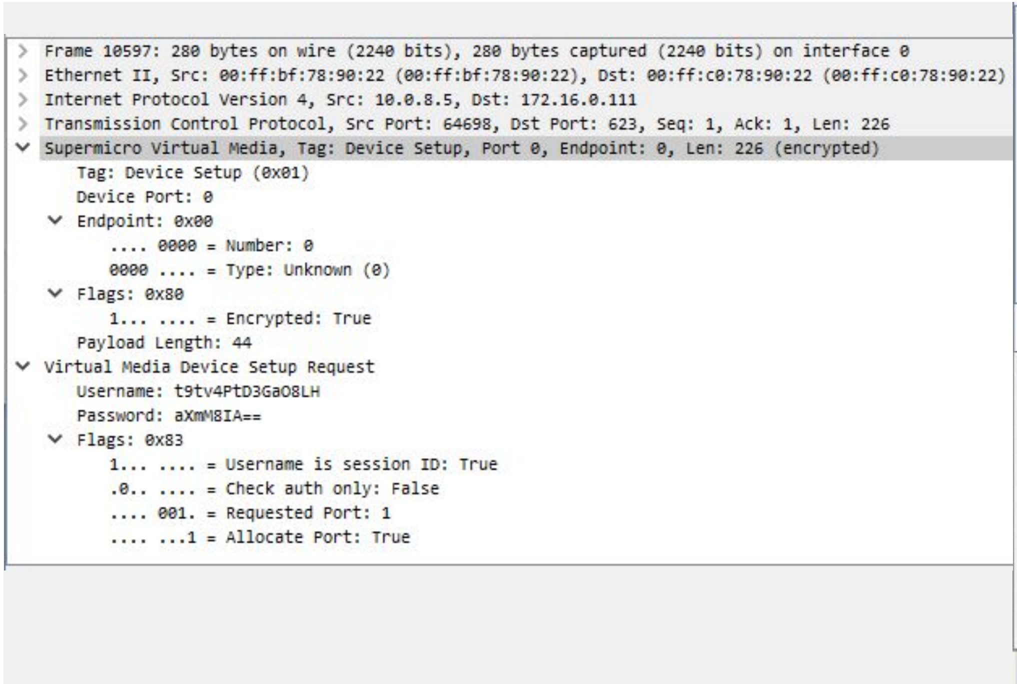
```
> Frame 11017: 119 bytes on wire (952 bits), 119 bytes captured (952 bits) on interface 0
> Ethernet II, Src: 00:ff:bf:78:90:22 (00:ff:bf:78:90:22), Dst: 00:ff:c0:78:90:22 (00:ff:c0:78:90:22)
> Internet Protocol Version 4, Src: 10.0.8.5, Dst: 172.16.0.111
> Transmission Control Protocol, Src Port: 64698, Dst Port: 623, Seq: 243, Ack: 57, Len: 65
> Data (65 bytes)

0000  00 ff c0 78 90 22 00 ff  bf 78 90 22 08 00 45 00   ···x·"·· ·x·"··E·
0010  00 69 fc 55 40 00 80 06  3f b5 0a 00 08 05 ac 10   ·i·U@··· ?·······
0020  00 6f fc ba 02 6f a3 99  f8 b0 64 80 f1 e1 50 18   ·o···o·· ··d···P·
0030  88 b4 43 81 00 00 22 00  01 00 24 00 00 00 05 80   ··C···"· ··$·····
0040  00 21 1f 00 00 00 41 54  45 4e 20 20 20 20 56 69   ·!····AT EN    Vi
0050  72 74 75 61 6c 20 43 44  52 4f 4d 20 20 20 59 53   rtual CD ROM   YS
0060  30 4a 22 00 01 ff 0d 00  00 00 55 53 42 53 01 00   0J"····· ··USBS··
0070  00 00 00 00 00 00 00                               ·······

Lengt....len) | Packets: 12698 · Displayed: 12698 (100.0%) · Dropped: 0 (0.0%) | Profile: Default
```

SCSI Vendor and Product IDs

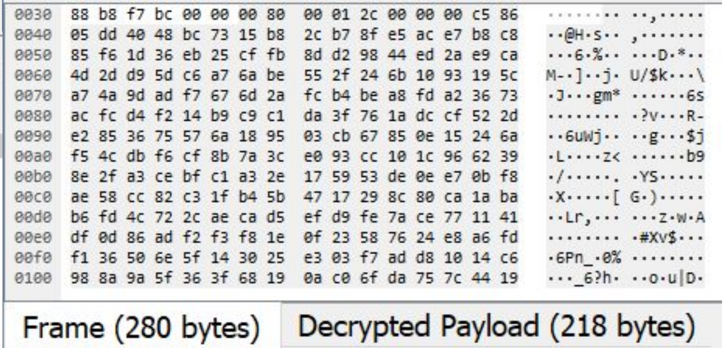
USB Mass Storage Class (MSC) Bulk-only Transport (BOT) Command Status Wrapper (CSW) Signature

Understanding the Protocol

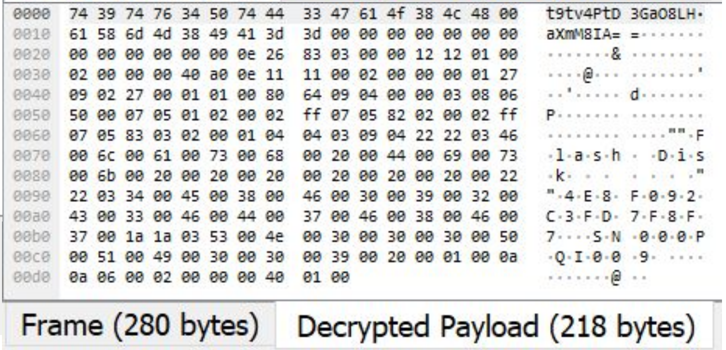
USBAnywhere



```
>  Frame 10597: 280 bytes on wire (2240 bits), 280 bytes captured (2240 bits) on interface 0
>  Ethernet II, Src: 00:ff:bf:78:90:22 (00:ff:bf:78:90:22), Dst: 00:ff:c0:78:90:22 (00:ff:c0:78:90:22)
>  Internet Protocol Version 4, Src: 10.0.8.5, Dst: 172.16.0.111
>  Transmission Control Protocol, Src Port: 64698, Dst Port: 623, Seq: 1, Ack: 1, Len: 226
⌄  Supermicro Virtual Media, Tag: Device Setup, Port 0, Endpoint: 0, Len: 226 (encrypted)
      Tag: Device Setup (0x01)
      Device Port: 0
   ⌄  Endpoint: 0x00
         .... 0000 = Number: 0
         0000 .... = Type: Unknown (0)
   ⌄  Flags: 0x80
         1... .... = Encrypted: True
      Payload Length: 44
⌄  Virtual Media Device Setup Request
      Username: t9tv4PtD3GaO8LH
      Password: aXmM8IA==
   ⌄  Flags: 0x83
         1... .... = Username is session ID: True
         .0.. .... = Check auth only: False
         .... 001. = Requested Port: 1
         .... ...1 = Allocate Port: True
```

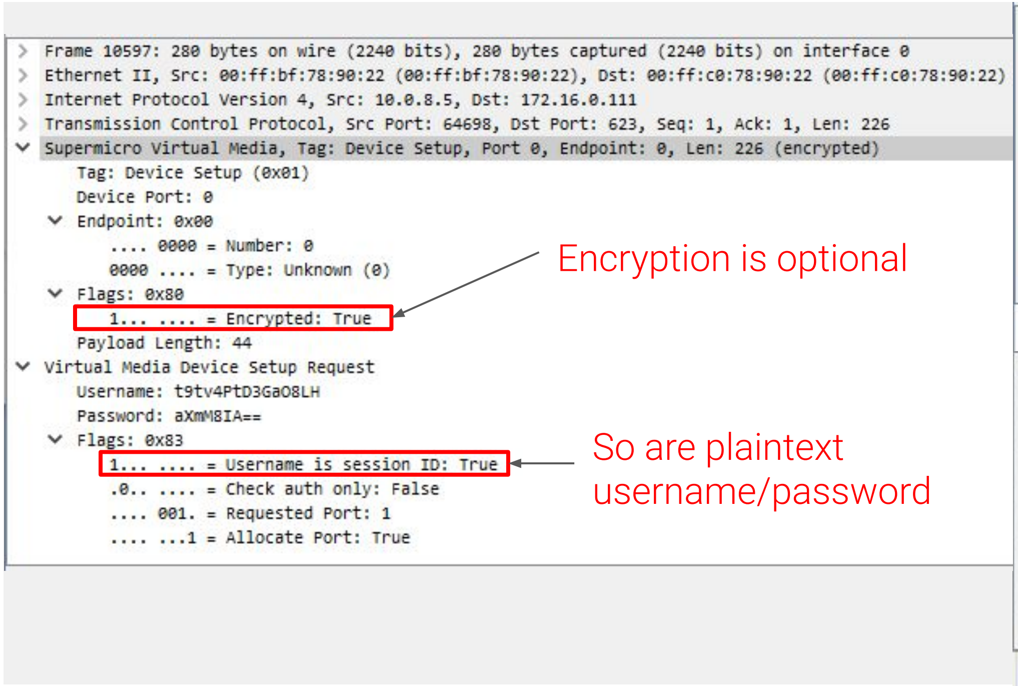


```
0030  88 b8 f7 bc 00 00 00 80  00 01 2c 00 00 00 c5 86   ········ ··,·····
0040  05 dd 40 48 bc 73 15 b8  2c b7 8f e5 ac e7 b8 c8   ··@H·s·· ,·······
0050  85 f6 1d 36 eb 25 cf fb  8d d2 98 44 ed 2a e9 ca   ···6·%·· ···D·*··
0060  4d 2d d9 5d c6 a7 6a be  55 2f 24 6b 10 93 19 5c   M-·]··j· U/$k···\
0070  a7 4a 9d ad f7 67 6d 2a  fc b4 be a8 fd a2 36 73   ·J···gm* ······6s
0080  ac fc d4 f2 14 b9 c9 c1  da 3f 76 1a dc cf 52 2d   ········ ·?v···R-
0090  e2 85 36 75 57 6a 18 95  03 cb 67 85 0e 15 24 6a   ··6uWj·· ··g···$j
00a0  f5 4c db f6 cf 8b 7a 3c  e0 93 cc 10 1c 96 62 39   ·L····z< ······b9
00b0  8e 2f a3 ce bf c1 a3 2e  17 59 53 de 0e e7 0b f8   ·/·····. ·YS·····
00c0  ae 58 cc 82 c3 1f b4 5b  47 17 29 8c 80 ca 1a ba   ·X·····[ G·)·····
00d0  b6 fd 4c 72 2c ae ca d5  ef d9 fe 7a ce 77 11 41   ··Lr,··· ···z·w·A
00e0  df 0d 86 ad f2 f3 f8 1e  0f 23 58 76 24 e8 a6 fd   ········ ·#Xv$···
00f0  f1 36 50 6e 5f 14 30 25  e3 03 f7 ad d8 10 14 c6   ·6Pn_·0% ········
0100  98 8a 9a 5f 36 3f 68 19  0a c0 6f da 75 7c 44 19   ···_6?h· ··o·u|D·
```



```
0000  74 39 74 76 34 50 74 44  33 47 61 4f 38 4c 48 00   t9tv4PtD 3GaO8LH·
0010  61 58 6d 4d 38 49 41 3d  3d 00 00 00 00 00 00 00   aXmM8IA= =·······
0020  00 00 00 00 00 00 0e 26  83 03 00 00 12 12 01 00   ·······& ········
0030  02 00 00 00 40 a0 0e 11  11 00 02 00 00 00 01 27   ····@··· ·······'
0040  09 02 27 00 01 01 00 80  64 09 04 00 00 03 08 06   ··'····· d·······
0050  50 00 07 05 01 02 00 02  ff 07 05 82 02 00 02 ff   P······· ········
0060  07 05 83 03 02 00 01 04  04 03 09 04 22 22 03 46   ········ ····""·F
0070  00 6c 00 61 00 73 00 68  00 20 00 44 00 69 00 73   ·l·a·s·h · ·D·i·s
0080  00 6b 00 20 00 20 00 20  00 20 00 20 00 20 00 22   ·k· · ·  · · · ·"
0090  22 03 34 00 45 00 38 00  46 00 30 00 39 00 32 00   "·4·E·8· F·0·9·2·
00a0  43 00 33 00 46 00 44 00  37 00 46 00 38 00 46 00   C·3·F·D· 7·F·8·F·
00b0  37 00 1a 1a 03 53 00 4e  00 30 00 30 00 30 00 50   7····S·N ·0·0·0·P
00c0  00 51 00 49 00 30 00 30  00 39 00 20 00 01 00 0a   ·Q·I·0·0 ·9· ····
00d0  0a 06 00 02 00 00 00 40  01 00                     ·······@ ··
```

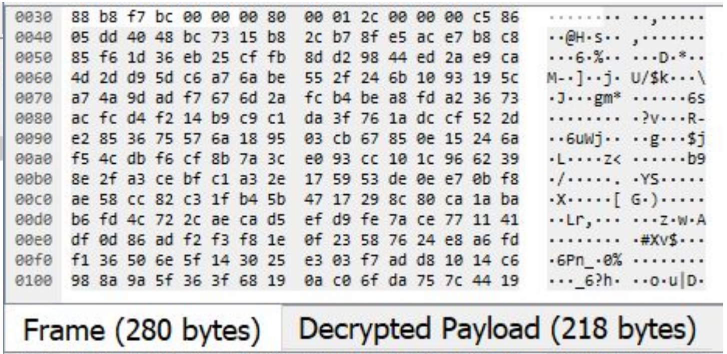
Understanding the Protocol



```
> Frame 10597: 280 bytes on wire (2240 bits), 280 bytes captured (2240 bits) on interface 0
> Ethernet II, Src: 00:ff:bf:78:90:22 (00:ff:bf:78:90:22), Dst: 00:ff:c0:78:90:22 (00:ff:c0:78:90:22)
> Internet Protocol Version 4, Src: 10.0.8.5, Dst: 172.16.0.111
> Transmission Control Protocol, Src Port: 64698, Dst Port: 623, Seq: 1, Ack: 1, Len: 226
v Supermicro Virtual Media, Tag: Device Setup, Port 0, Endpoint: 0, Len: 226 (encrypted)
    Tag: Device Setup (0x01)
    Device Port: 0
  v Endpoint: 0x00
      .... 0000 = Number: 0
      0000 .... = Type: Unknown (0)
  v Flags: 0x80
      1... .... = Encrypted: True
    Payload Length: 44
v Virtual Media Device Setup Request
    Username: t9tv4PtD3GaO8LH
    Password: aXmM8IA==
  v Flags: 0x83
      1... .... = Username is session ID: True
      .0.. .... = Check auth only: False
      .... 001. = Requested Port: 1
      .... ...1 = Allocate Port: True
```

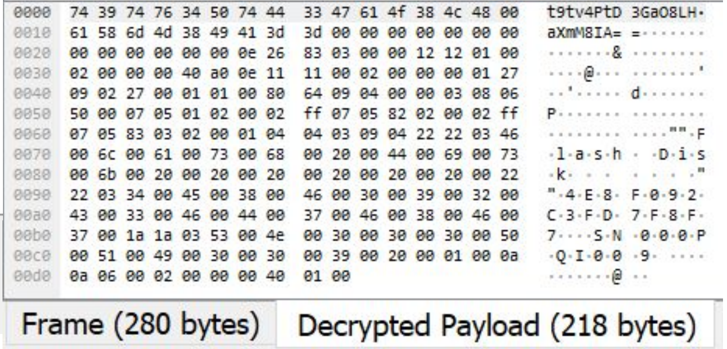
Encryption is optional

So are plaintext username/password



```
0030  88 b8 f7 bc 00 00 00 80  00 01 2c 00 00 00 c5 86   ········ ··,·····
0040  05 dd 40 48 bc 73 15 b8  2c b7 8f e5 ac e7 b8 c8   ··@H·s·· ,·······
0050  85 f6 1d 36 eb 25 cf fb  8d d2 98 44 ed 2a e9 ca   ···6·%·· ···D·*··
0060  4d 2d d9 5d c6 a7 6a be  55 2f 24 6b 10 93 19 5c   M-·]··j· U/$k···\
0070  a7 4a 9d ad f7 67 6d 2a  fc b4 be a8 fd a2 36 73   ·J···gm* ······6s
0080  ac fc d4 f2 14 b9 c9 c1  da 3f 76 1a dc cf 52 2d   ········ ·?v···R-
0090  e2 85 36 75 57 6a 18 95  03 cb 67 85 0e 15 24 6a   ··6uWj·· ··g···$j
00a0  f5 4c db f6 cf 8b 7a 3c  e0 93 cc 10 1c 96 62 39   ·L····z< ······b9
00b0  8e 2f a3 ce bf c1 a3 2e  17 59 53 de 0e e7 0b f8   ·/·····. ·YS·····
00c0  ae 58 cc 82 c3 1f b4 5b  47 17 29 8c 80 ca 1a ba   ·X·····[ G·)·····
00d0  b6 fd 4c 72 2c ae ca d5  ef d9 fe 7a ce 77 11 41   ··Lr,··· ···z·w·A
00e0  df 0d 86 ad f2 f3 f8 1e  0f 23 58 76 24 e8 a6 fd   ········ ·#Xv$···
00f0  f1 36 50 6e 5f 14 30 25  e3 03 f7 ad d8 10 14 c6   ·6Pn_·0% ········
0100  98 8a 9a 5f 36 3f 68 19  0a c0 6f da 75 7c 44 19   ···_6?h· ··o·u|D·
```

Frame (280 bytes) | Decrypted Payload (218 bytes)



```
0000  74 39 74 76 34 50 74 44  33 47 61 4f 38 4c 48 00   t9tv4PtD 3GaO8LH·
0010  61 58 6d 4d 38 49 41 3d  3d 00 00 00 00 00 00 00   aXmM8IA= =·······
0020  00 00 00 00 00 00 0e 26  83 03 00 00 12 12 01 00   ·······& ········
0030  02 00 00 00 40 a0 0e 11  11 00 02 00 00 00 01 27   ····@··· ·······'
0040  09 02 27 00 01 01 00 80  64 09 04 00 00 03 08 06   ··'····· d·······
0050  50 00 07 05 01 02 00 02  ff 07 05 82 02 00 02 ff   P······· ········
0060  07 05 83 03 02 00 01 04  04 03 09 04 22 22 03 46   ········ ····""·F
0070  00 6c 00 61 00 73 00 68  00 20 00 44 00 69 00 73   ·l·a·s·h · ·D·i·s
0080  00 6b 00 20 00 20 00 20  00 20 00 20 00 20 00 22   ·k· · ·  · · · ·"
0090  22 03 34 00 45 00 38 00  46 00 30 00 39 00 32 00   "·4·E·8· F·0·9·2·
00a0  43 00 33 00 46 00 44 00  37 00 46 00 38 00 46 00   C·3·F·D· 7·F·8·F·
00b0  37 00 1a 1a 03 53 00 4e  00 30 00 30 00 30 00 50   7····S·N ·0·0·0·P
00c0  00 51 00 49 00 30 00 30  00 39 00 20 00 01 00 0a   ·Q·I·0·0 ·9· ····
00d0  0a 06 00 02 00 00 00 40  01 00                     ·······@ ··
```

Frame (280 bytes) | Decrypted Payload (218 bytes)

It Gets Worse

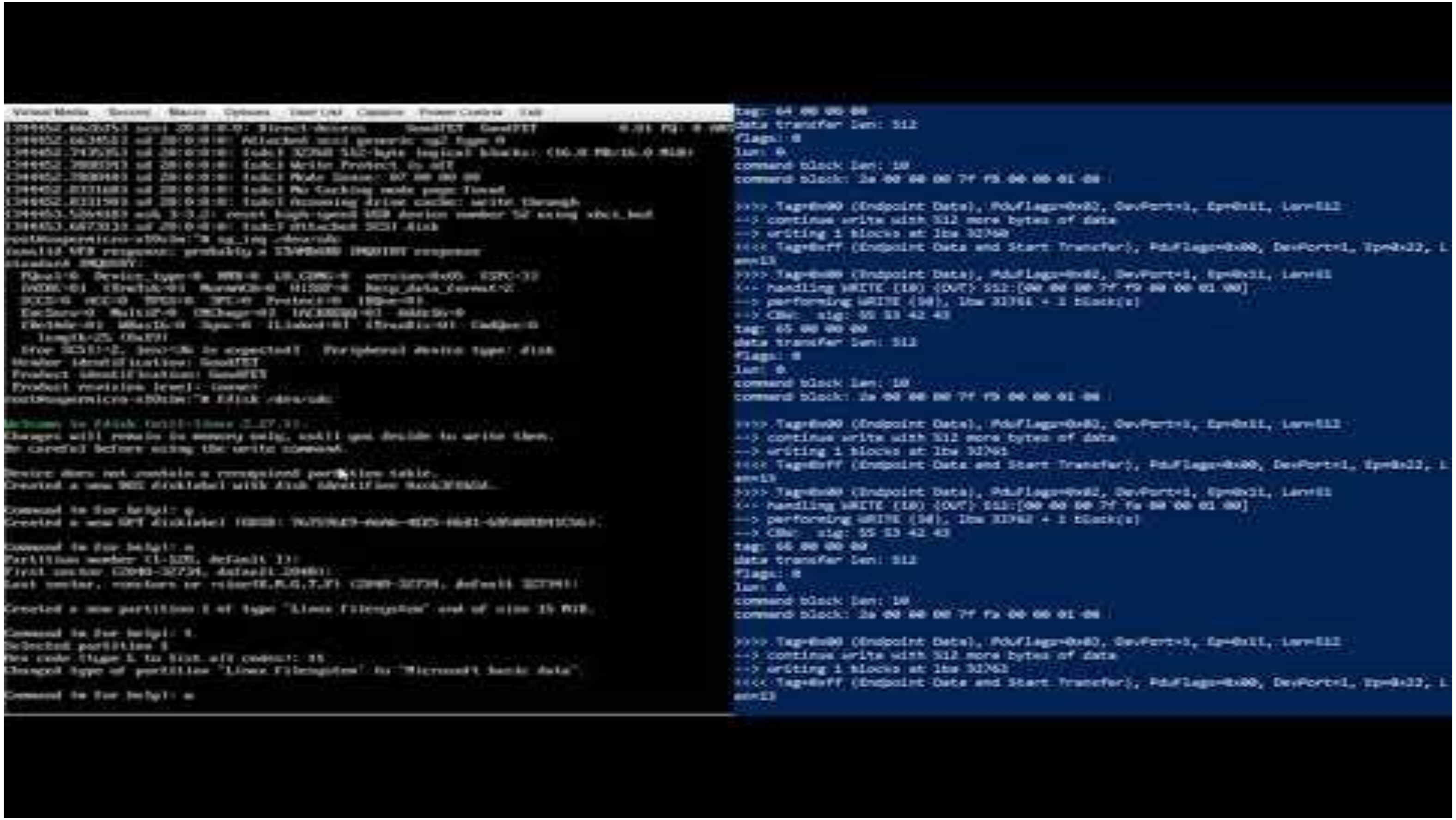
- Encryption
  - RC4 with same fixed key on all X9, X10, and X11 systems
  - Could have been used for every packet but wasn’t
- USB device is implemented entirely client-side
  - Server caches client-provided USB descriptors
  - Almost all endpoint traffic sent directly to client
- Authentication bypass on X10 and X11
  - Credentials cached by socket file descriptor
  - Client disconnect fails to invalidate cache
  - Very high chance of unintentional reuse by a new client

Making My Own Client

USBAnywhere

- Facedancer[1] is a Python framework for emulating USB devices
  - Originally designed for use with special-purpose hardware
  - Now has a plugin architecture for backends
- USBAnywhere backend
  - Opts to not use encryption
  - Uses plaintext username/password auth
  - PoC quality

[1] https://github.com/usb-tools/Facedancer



```
[344452.662573] scsi 20:0:0:0: Direct-Access     GoodFET  GoodFET          0.01 PQ: 0 ANS
[344452.663451] sd 20:0:0:0: Attached scsi generic sg2 type 0
[344452.743535] sd 20:0:0:0: [sdc] 32768 512-byte logical blocks: (16.8 MB/16.0 MiB)
[344452.788839] sd 20:0:0:0: [sdc] Write Protect is off
[344452.788848] sd 20:0:0:0: [sdc] Mode Sense: 07 00 00 00
[344452.833188] sd 20:0:0:0: [sdc] No Caching mode page found
[344452.833159] sd 20:0:0:0: [sdc] Assuming drive cache: write through
[344453.526418] usb 3-3.2: reset high-speed USB device number 52 using xhci_hcd
[344453.687313] sd 20:0:0:0: [sdc] Attached SCSI disk
root@supermicro-x10sbm:~# sg_inq /dev/sdc
invalid VPD response; probably a STANDARD INQUIRY response
standard INQUIRY:
  PQual=0  Device_type=0  RMB=0  LU_CONG=0  version=0x05  [SPC-3]
  [AERC=0]  [TrmTsk=0]  NormACA=0  HiSUP=0  Resp_data_format=2
  SCCS=0  ACC=0  TPGS=0  3PC=0  Protect=0  [BQue=0]
  EncServ=0  MultiP=0  [MChngr=0]  [ACKREQQ=0]  Addr16=0
  [RelAdr=0]  WBus16=0  Sync=0  [Linked=0]  [TranDis=0]  CmdQue=0
    length=25 (0x19)
  [illegible] SCSI=2, len=36 is expected!   Peripheral device type: disk
 Vendor identification: GoodFET
 Product identification: GoodFET
 Product revision level: [illegible]
root@supermicro-x10sbm:~# fdisk /dev/sdc

Welcome to fdisk (util-linux 2.27.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier [illegible].

Command (m for help): g
Created a new GPT disklabel (GUID: [illegible]).

Command (m for help): n
Partition number (1-128, default 1):
First sector (2048-32734, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-32734, default 32734):

Created a new partition 1 of type 'Linux filesystem' and of size 15 MiB.

Command (m for help): t
Selected partition 1
Hex code (type L to list all codes): 11
Changed type of partition 'Linux filesystem' to 'Microsoft basic data'.

Command (m for help): w
```

```
tag: 64 00 00 00
data transfer len: 512
flags: 0
lun: 0
command block len: 10
command block: 2a 00 00 00 7f f8 00 00 01 00

>>>> Tag=0x00 (Endpoint Data), Pduflags=0x02, DevPort=1, Ep=0x11, Len=512
--> continue write with 512 more bytes of data
--> writing 1 blocks at lba 32760
<<<< Tag=0xff (Endpoint Data and Start Transfer), PduFlags=0x00, DevPort=1, Ep=0x22, L
en=13
>>>> Tag=0x00 (Endpoint Data), PduFlags=0x02, DevPort=1, Ep=0x11, Len=31
<-- handling WRITE (10) (OUT) 512:[00 00 00 7f f9 00 00 01 00]
--> performing WRITE (10), lba 32761 + 1 block(s)
--> CBW:  sig: 55 53 42 43
tag: 65 00 00 00
data transfer len: 512
flags: 0
lun: 0
command block len: 10
command block: 2a 00 00 00 7f f9 00 00 01 00

>>>> Tag=0x00 (Endpoint Data), PduFlags=0x02, DevPort=1, Ep=0x11, Len=512
--> continue write with 512 more bytes of data
--> writing 1 blocks at lba 32761
<<<< Tag=0xff (Endpoint Data and Start Transfer), PduFlags=0x00, DevPort=1, Ep=0x22, L
en=13
>>>> Tag=0x00 (Endpoint Data), PduFlags=0x02, DevPort=1, Ep=0x11, Len=31
<-- handling WRITE (10) (OUT) 512:[00 00 00 7f fa 00 00 01 00]
--> performing WRITE (10), lba 32762 + 1 block(s)
--> CBW:  sig: 55 53 42 43
tag: 66 00 00 00
data transfer len: 512
flags: 0
lun: 0
command block len: 10
command block: 2a 00 00 00 7f fa 00 00 01 00

>>>> Tag=0x00 (Endpoint Data), PduFlags=0x02, DevPort=1, Ep=0x11, Len=512
--> continue write with 512 more bytes of data
--> writing 1 blocks at lba 32762
<<<< Tag=0xff (Endpoint Data and Start Transfer), PduFlags=0x00, DevPort=1, Ep=0x22, L
en=13
```

Impact

USBAnywhere

- **47,000** affected BMCs found on the Internet
  - 1905 Autonomous Systems (AS)
  - 90+ countries
- How many are on your enterprise network?
- Attack scenarios
  - Exfiltrate data over virtual USB mass storage device
  - Boot machine from attacker-provided ISO
  - Network-attached USB Rubber Ducky[1]
  - and the list goes on...

[1] https://shop.hak5.org/products/usb-rubber-ducky-deluxe

Resources

- Eclypsium Blog
  https://eclypsium.com/2019/09/03/usbanywhere-bmc-vulnerability-opens-servers-to-remote-attack/
- Proof-of-concept Demo Video
  https://youtu.be/8UI7oicMisY
- Tools, Packet Captures, etc
  https://github.com/eclypsium/USBAnywhere