Wireshark CA Plug-in
EPICS Channel Access Dissector

Kazuro Furukawa, KEK
Ron Rechenmacher, Fermilab
Anze Zagar, Cosylab
Klemen Zagar, Cosylab

Presented by
Masanori Satoh, KEK

Background

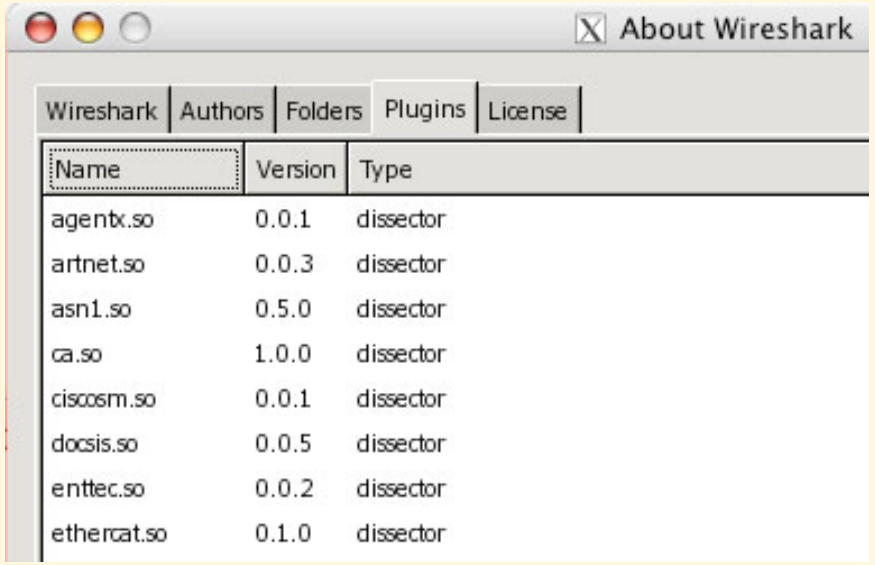
- Ideas and efforts from several groups in the past
  - Tech-talk proposal of CA Sniffer from Ned Arnold, APS
  - Implementation of primary CA Plugin for Ethereal by Ron Rechenmacher, Fermilab
  - (Managers love to have analyzers)
- KEK needed CA analyzer for efficient EPICS operation
  - Without knowing above efforts
  - Thought about Tcpdump extension for textual processing
  - Discussion at ICALEPCS with Bob Dalesio and Jeff Hill
  - Discussion with Ron Rechenmacher, Fermilab
  - Implementation by Klemen and Anze Zagar, Cosylab

CA Plug-in for Wireshark

- **Wireshark (formally Ethereal)**
  - Is the most famous network protocol analyzer and is open source
    - <http://www.wireshark.org/>
- **Wireshark Plugin architecture**
  - EPICS channel access protocol dissection in CA plugin
    - Development is well separated from main program
    - Plugin distribution is simpler
      - Only one file (shared/dinamic library file) for binary distribution
      - One plugin directory and a simple patch (Makefile, etc) in a tar file for source

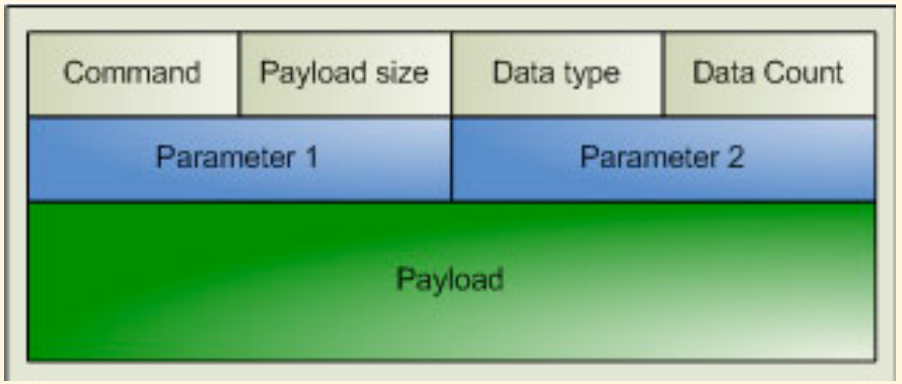
CA Plug-in for Wireshark

- Graphical or Textual user interface
  - Graphical interface for Online capture and Offline analysis
    - With flexible filters
  - Textual interface (tshark) for batch operation
    - Original intention at KEK was long-term rare event capturing and analysis
      - Background operation was preferable
      - Almost the same as tcpdump
      - Captured data can be analyzed later
        - »With Graphical user interface



CA Plugin

- Dissects all CA packet header
  - Commands/replies and parameters
    - In Channel Access Protocol specification
      - <http://epics.cosylab.com/cosyjava/JCA-Common/Documentation/CAproto.html>
- Also tracks PV/Channel names along virtual circuit
  - Each packet only contains ID (CID/SID/SubscriptionID)
    - Indispensable for human-readable analysis
- Does not dissect payload
  - Use other EPICS tools
    - For data contents



Installation

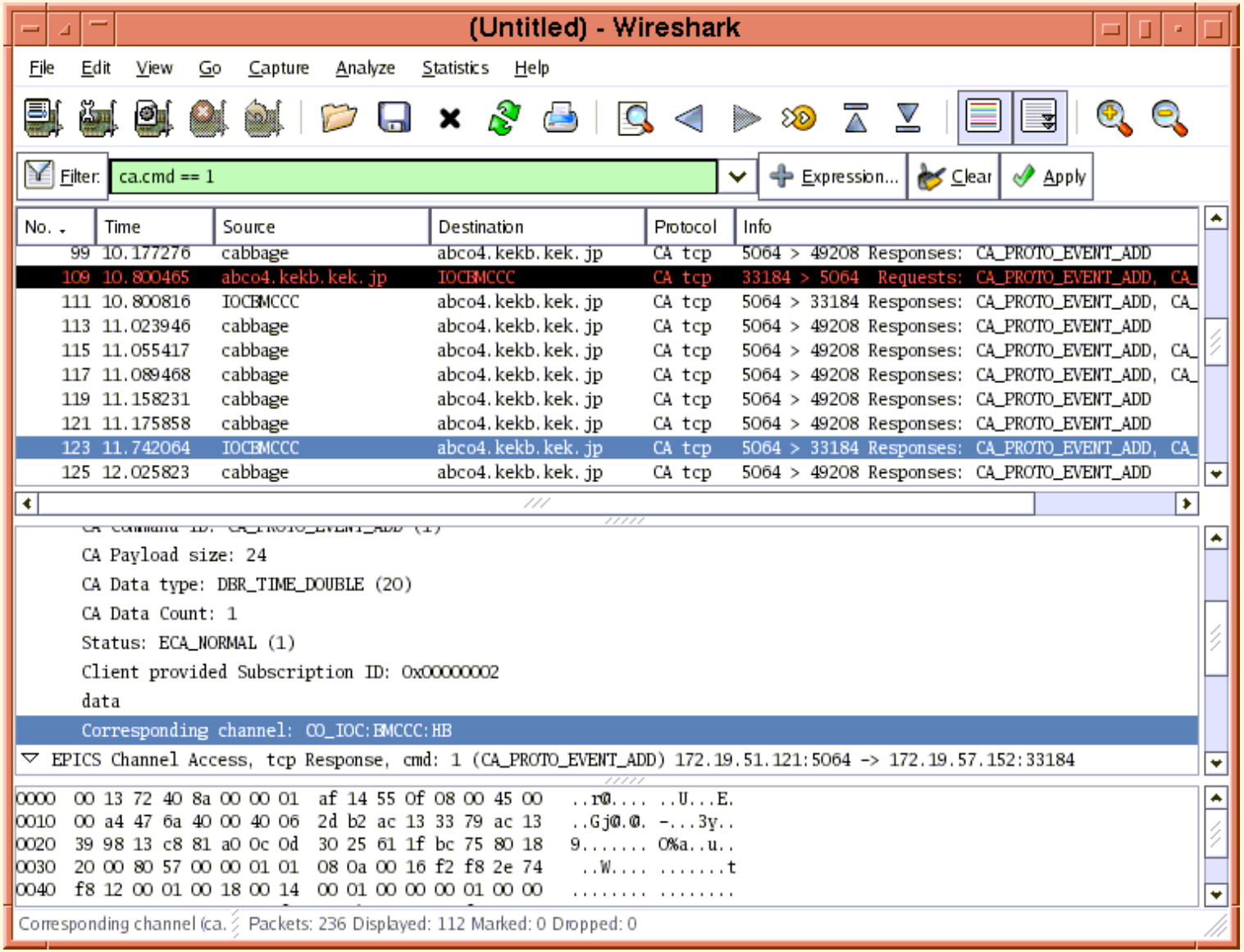
- **Binary installation**
  - Install normal Wireshark 0.99.8 or 0.99.7
  - Install CA plugin binary
    - From <http://www-linac.kek.jp/cont/epics/wireshark/>
    - Windows, Linux, MacOSX (x86/ppc) for now
- **Building from source**
  - Get Wireshark (0.99.8 or 0.99.7)
  - Expand CA plugin source
  - Apply patch
  - Normal building procedure
    - <http://www-linac.kek.jp/cont/epics/wireshark/> for details
    - Gtk+ and packet capture software are required

Simple Usage for EPICS

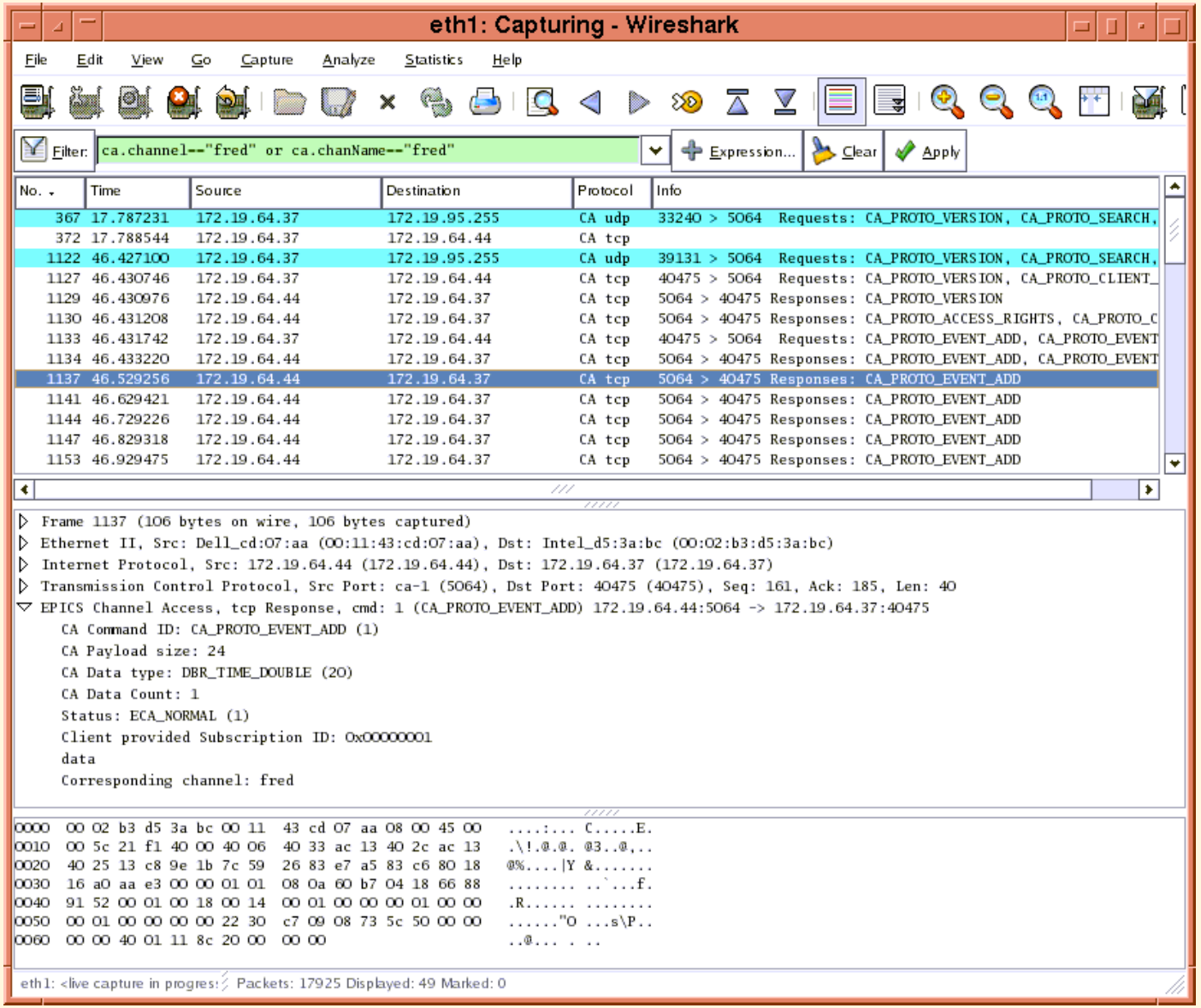
- Invoke Wireshark
- Capture options
  - Capture Filter: “port 5064 or port 5065”
- Start capture
- (Stop capture)
- Apply display/analysis filter
  - Filter examples
    - ca.cmd==1
      - Symbolic names like CA_PROTO_SEARCH in Helper
    - ca.chanName==“fred” or ca.channel==“fred”
      - Packets related to a PV named fred
    - ca.channel matches “^VAC:IP.*:Pressure”
    - ca.channel contains “VAC:IP”
      - PV name string or regular-expression matching

ca.cmd - CA Command ID
ca.paySz - CA Payload size
ca.type - CA Data type
ca.cnt - CA Data Count
ca.p1 - CA Parameter 1
ca.p2 - CA Parameter 2
ca.tcpPort - TCP port of responding server
ca.srvrId - Temporary SID
ca.chnId - Channel CID
ca.minorVer - Minor protocol version
ca.srvrVer - Server protocol version
ca.desiredPrio - Desired Priority
ca.userName - User name
ca.hostName - Host name
ca.chanName - Channel name
ca.accRghts - Access Rights
ca.ioid - Client provided IOID
ca.subscrptId - Client provided Subscription ID
ca.evLo - Low value
ca.evHi - High value
ca.evTo - To value
ca.evMonMsk - Monitor mask
ca.status - Status
ca.reply - Reply
ca.reserved - Reserved (Should be zero)
ca.unused - Unused
ca.clientip - Client IP address
ca.serverip - Server IP address
ca.repeaterip - Repeater IP address
ca.strDat - String data
ca.dblDat - Double prec.float data
ca.deprecated - Obsolete (Obsolete)
ca.data - data (formatted data)
ca.zero - zero (should be zero)
ca.undecoded - undecoded (Yet undecoded by dissector)
ca.channel - Corresponding channel

Selecting EVENT_ADD command/response



Selecting “fred” related packets



Hints

- Combination with CA Snooper may enhance network trouble-shooting
- Expression button helps filter expression construction
- tshark may be used to capture packets, and later Wireshark can be used to analyze them
- Data contents dissection necessary?

Summary

- Wireshark CA plugin was build with efforts by many people
- It may be used for the efficient operation of EPICS system and for the trouble-shooting
- Please send any comments to
  - < kazuro.furukawa @ kek.jp >

Thank you