Network Radar

Todd Heberlein

Net Squared, Inc.

todd@NetSQ.com

Watch the Watcher

- Everyone wants in the network security business
- Shell company on Wall Street
- Wants to put me in it; "I'm serious"
- Have his lawyer fly out ASAP
- P.O. Box 4000
  Springfield, MO 65808-4000

Supporting the Warfighter

- How does this support the warfighter?
- It doesn't, at least not directly.
- If we do our job right, the warfighter should never know we are there.
- The other I.T. technologies support the warfighter.
- We preserve that capability.
- Provide the warfighter a target.

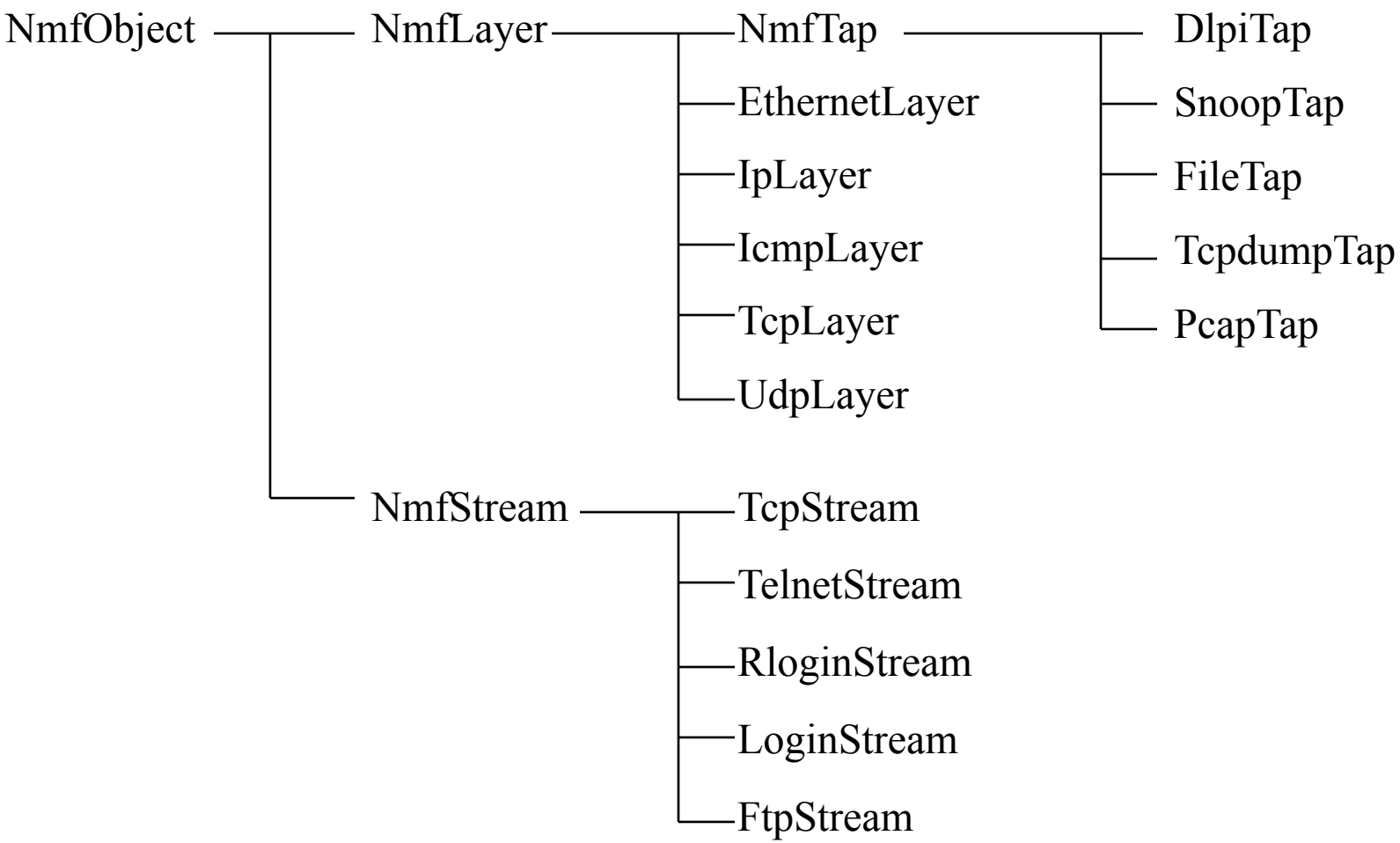
Outline

- Objectives
- Approach
- Expected Major Achievements
- Accomplishments, Results, and Lessons Learned
- Major Risks

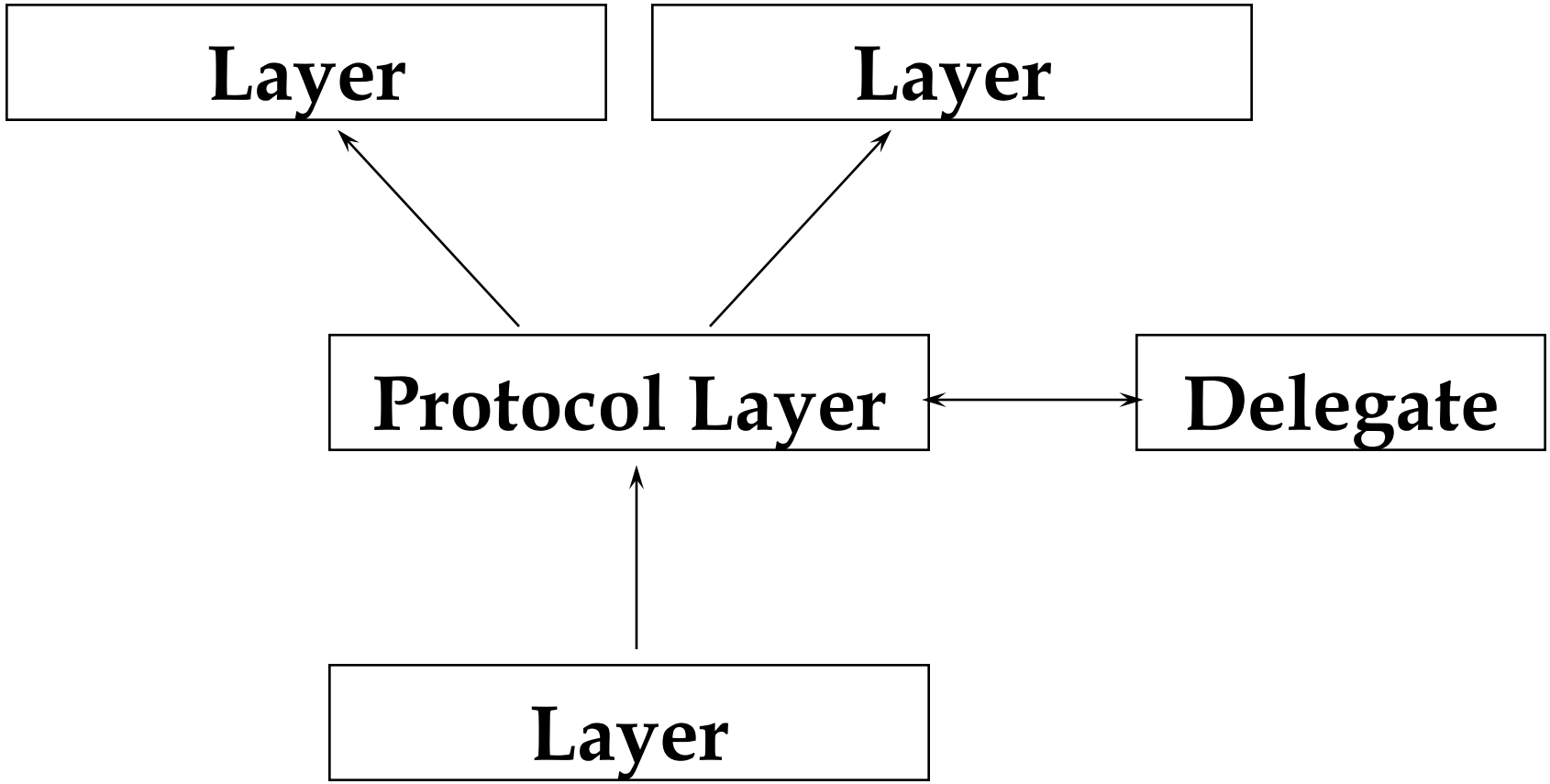
Objectives

- Develop a flexible framework to monitor network traffic - the Network Monitoring Framework (NMF).
- Develop a suite of NMF-based monitors, tailored to specific purposes, to enhance knowledge and control over the network.
- Create a vulnerability database to serve as a foundation for both operations and research.

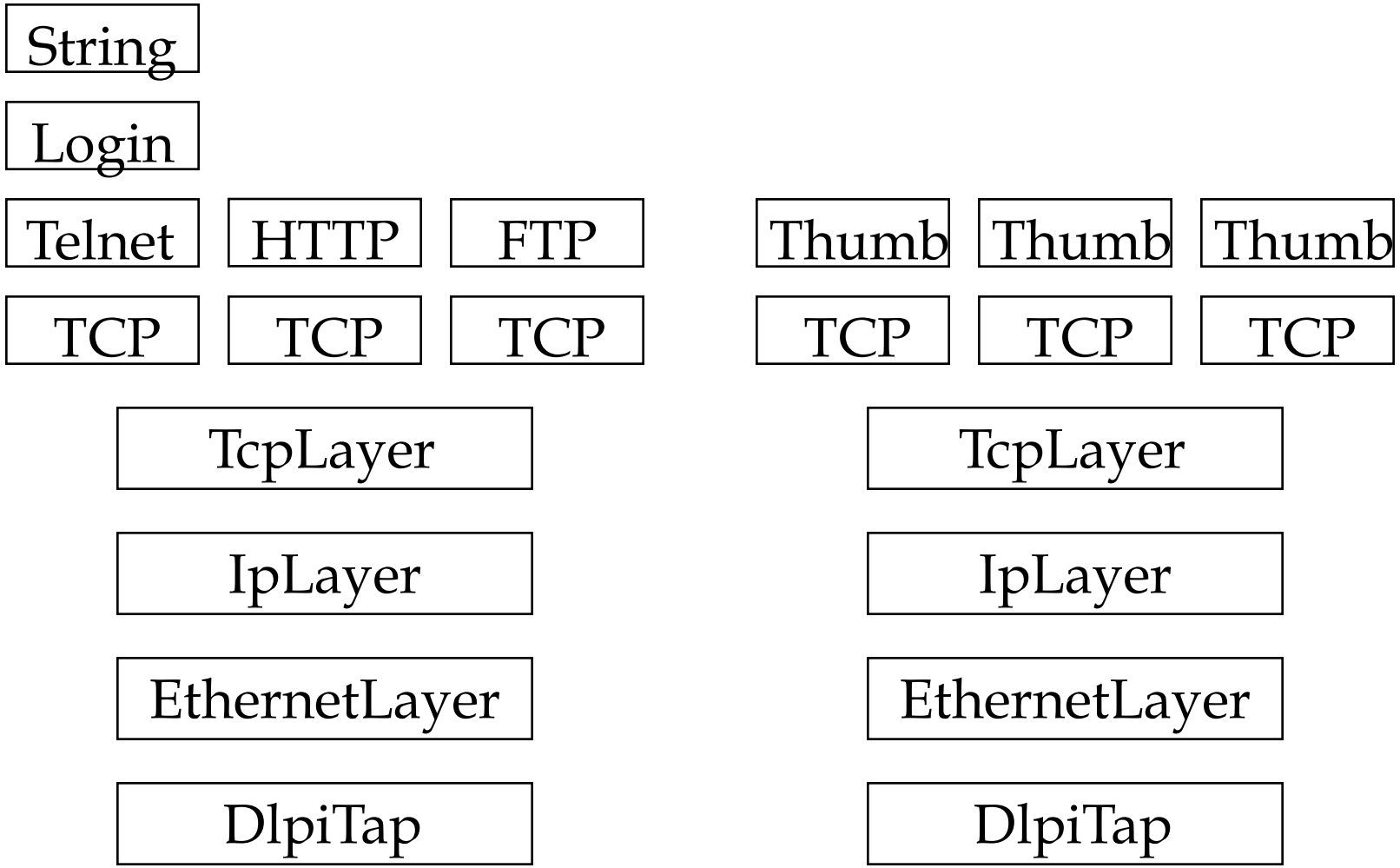
Sample Object Hierarchy



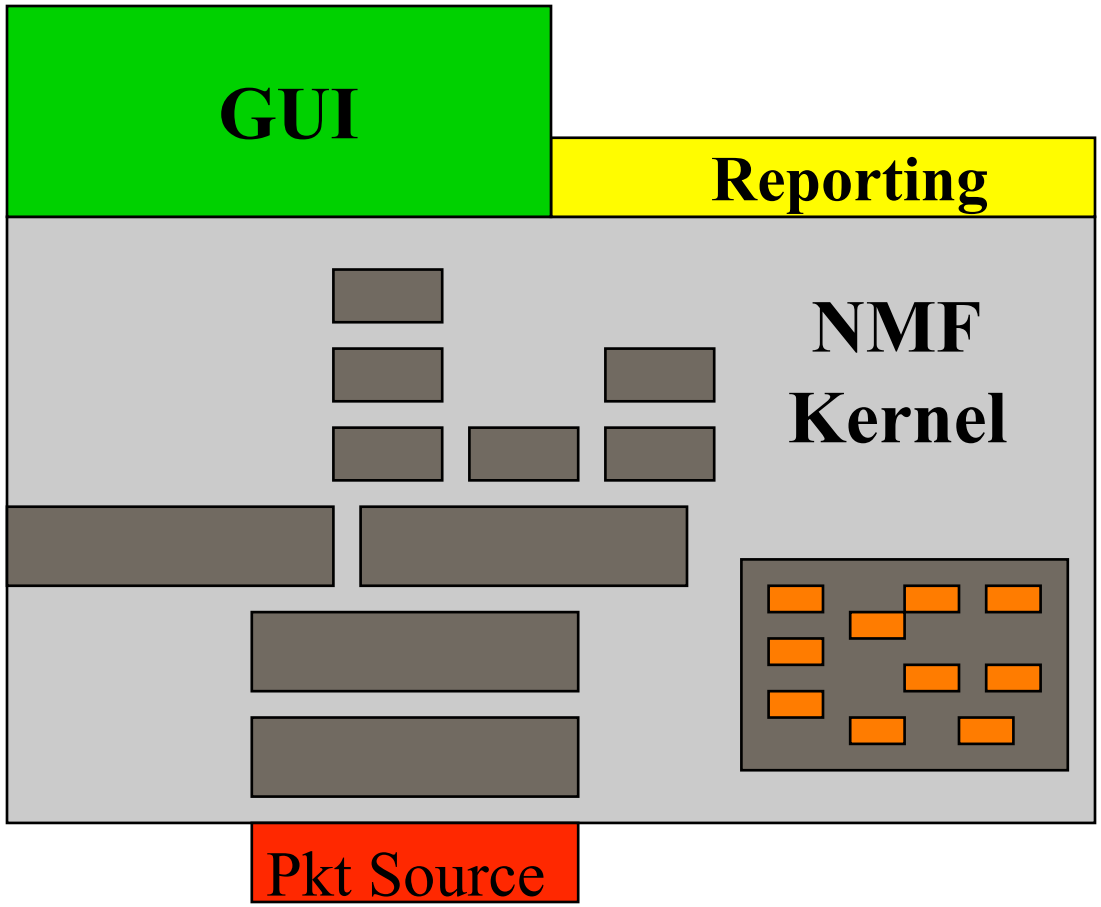
Abstractions & Delegation



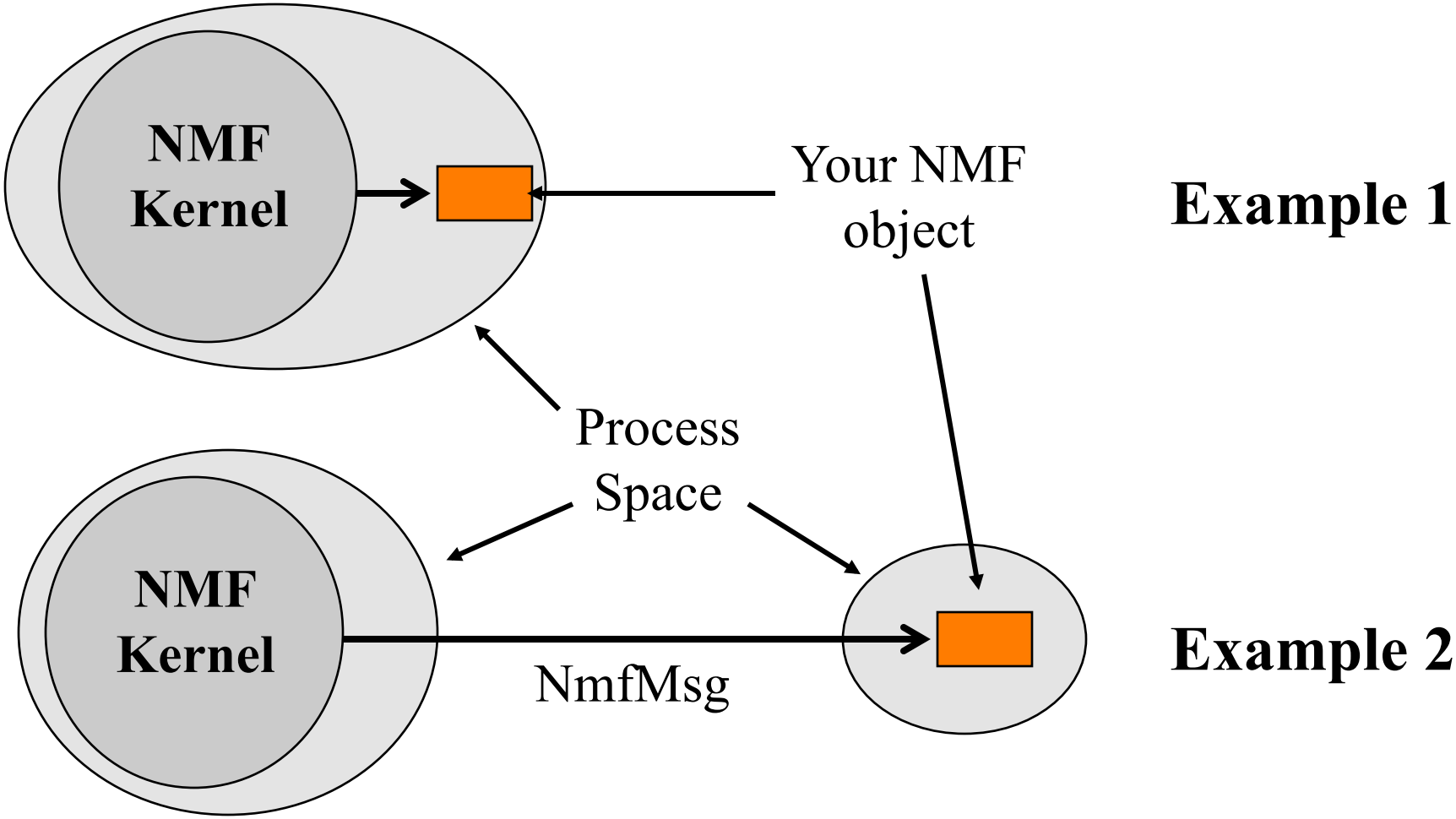
Example NMF Kernel



Functional Blocks



Distributed Object Architecture



Bridge Declaration

```
class GenericAttackBridge : public NmfObject {
public:
    GenericAttackBridge(NmfUByte4 my_class_val =
                        GenericAttackBridge_CLASS);
    virtual NmfByte4  inBox(NmfMsg* p_msg, NmfObject* sender);
};
```

Bridge Source Code

```
GenericAttackBridge::GenericAttackBridge(NmfUByte4 my_class_val)
        : NmfObject(my_class_val)
{
        /* initialize your local variables here */
}
```

```
NmfByte4 GenericAttackBridge::inBox(NmfMsg* p_msg, NmfObject* sender)
{
        NmfAttackMsg* p_attack = NULL;

        if ((p_msg == NULL) || (p_msg->classVal() != NmfAttackMsg_CLASS)) {
                return NMF_FAIL;
        }

        p_attack = (NmfAttackMsg*)p_msg;
        /* do something with p_attack fields */
}
```

Inserting Bridge Into Code

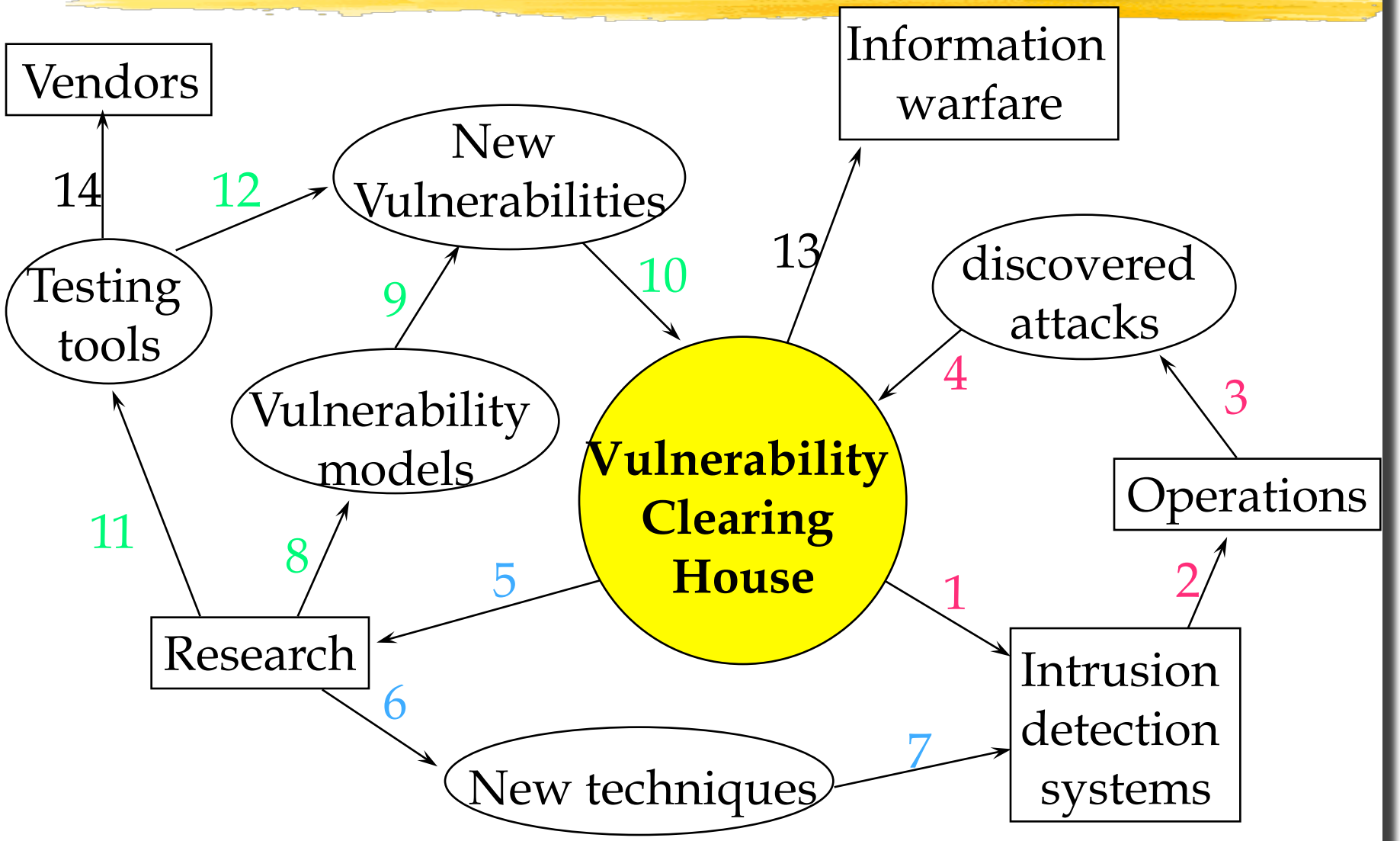
```
GenericAttackBridge  my_object;

NmfStream::groupSubscribe(NmfStream_ATTACK_MSGS,  ( NmfObject*)&my_object);
NmfLayer::groupSubecribe(NmfLayer_ATTACK,  ( NmfObject*)&my_object);
```

```
GenericAttackBridge  my_object;
ClientSideNmfKernelProxy  proxy;
SimpleIpV4Addr server_address = nmfGetAddress("yoda");
char* empty_str = "";

proxy.connectToServer( server_address);
proxy.infoSubscribe(NmfStreamAttack_MSG, empty_str, (NmfObject*)&my_object);
proxy.infoSubscribe(NmfLayerAttack_MSG, empty_str, (NmfObject*)&my_object);
```

Vulnerability Work Flow



High-Level Measures of Success

- Applications are “easy” to build
  - On third-generation of framework (oops)
- Others can extend the framework
  - Boeing will try to integrate IDIP (CIDF?)
  - Someone interested in Streams module for IIOP
- Find vulnerabilities before the bad guys do
- Others find vulnerabilities information useful in their own research

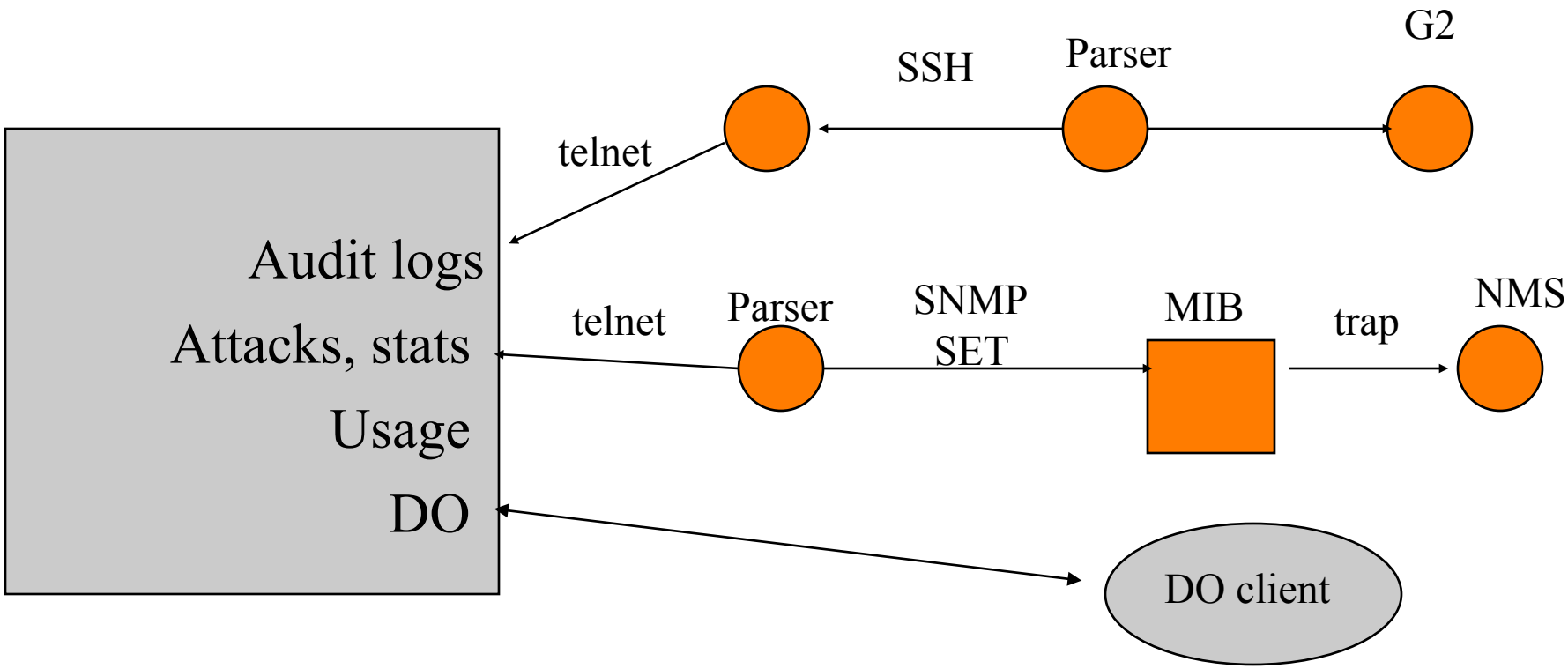
Expected Achievement

- Provide audit log of network activity (what should be done at the host)
- Enhance situational awareness by identifying active entities on the network
- Provide a view from 80,000 feet down to the bit level
- Get ahead of the “bad guys”
- Platform for new research & development

Network Radar Audit Trail NRAT

- Parse network traffic generating audit log
- Identify some network attacks
- Track statistical behavior of hosts
- Provide live access to
  - audit logs
  - attack, statistics reports
  - usage statistics
- Distributed Object Architecture

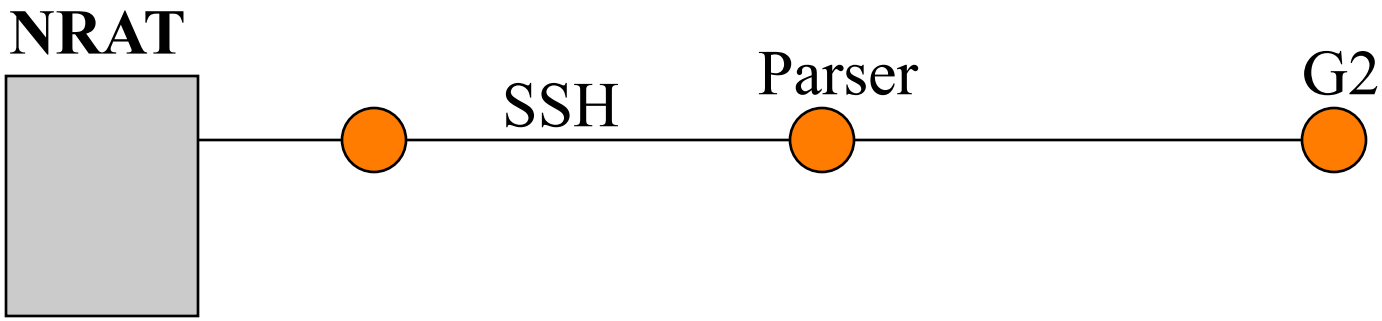
NRAT cont.

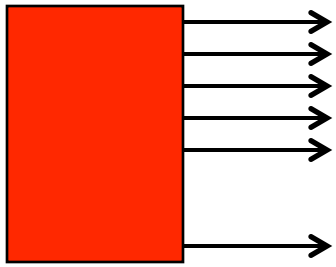

G2
SSH
Parser
telnet
Audit logs
Attacks, stats
Usage
DO
telnet
Parser
SNMP
SET
MIB
trap
NMS
DO client

Rome's G2 Project

- Integrate information from many data sources
- Move all network audit records to G2 live
- Centralized audit analysis is not uncommon
  - Centrax
  - Stalker (at least early versions)
  - Intrusion Detection, Inc. ??

G2 Under Attack Conditions





36 Parallel Sockets

Sweep of lower 1024 ports: 1 sec

Each attempt generate two reports

Adversary Running nmap port sweep on local LAN

All Packets Are NOT Created Equal

Operations for normal packet	Additional steps for SYN packet
Update session	Update client/server statistics
Find session in table	Find client/server
Parse each header	Report session
Copy to user space	Insert session in table
Read each packet	Create session record

Flexibility and Constraints Have a Cost

- Extensive use of “virtual” methods
  - late binding
  - native behavior of Objective C (Java, Smalltalk?)
- Generic container classes with potential help functions
- Use of “const” data type
  - Extra copies of data

Lessons Learned

- Models which work fine under normal conditions fail under extraordinary conditions
- Need to adjust level of information reported on the fly
- All packets are not created equal
- Software engineering and flexibility have a cost
- Don’t change code at the last minute

Looking for Warez

- Warez - FTP sites used to distributed pirated media
- Usually Anonymous FTP, but sometimes normal users
- Often use “hidden” file names
  - .*anything* except “.” and “..”
  - non-printable characters (less than SPACE, greater than tilde)

False +, Filename Error

```
-----------------------------------------------------
490483  130.158.85.241  -->  128.120.57.42  (1401 -> 21)
from: 00:30:38 ( 7/23/1998)  to: 00:32:56 ( 7/23/1998)
client flags: SAF     server_flags: SAF
        ---- FTP ----------------------------------------
        USER: ftp
        PASS: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
        RETR: ./UCD-SNMP-MIB.txt
        CWD:  /pub/snmp
        FAILURES: 0
```

False +, Innocent Store

```
-----------------------------------------------------------
829653   131.89.188.252  -->  128.120.143.125  (3648 -> 21)
from: 08:52:01 ( 7/23/1998)  to: 09:11:04 ( 7/23/1998)
client flags: SA      server_flags: SAF
        ---- FTP ----------------------------------------
        USER: anonymous
        PASS: xxxxxxxxxxxx
        STOR: trans60.e00
              trans.e00
        CWD:  incoming/larry
        FAILURES: 0
```

False +, Directory Names

```
-----------------------------------------------------------
2428813     128.120.57.1  -->  192.35.156.23(2659 -> 21)
from: 18:16:01 ( 7/23/1998)  to: 18:21:18 ( 7/23/1998)
client flags: SAF     server_flags: SAF
        ---- FTP ----------------------------------------
        USER: ftp
        PASS: xxxxxx
        RETR: qpopper2.53.tar.Z
        CWD:  eudora
              ../pub
              ../edora/servers
              ../eudora/servers
              ../eudora/servers
              unix
              popper
        FAILURES: 2
```

Hidden File Names?

```
-----------------------------------------------------------
2531844  169.237.61.27  --> 166.104.221.216 (1404 -> 21)
from: 19:02:47 ( 7/23/1998)  to: 19:17:38 ( 7/23/1998)
client flags: SA R    server_flags: SA
        ---- FTP ------------------------------------
        USER: anonymous
        PASS: xxxxxxxxx
        RETR: /!!!¹Ýµå½Ã ÀÐ¾î°¸¼Å¾ß ÇÕ´Ï´Ù!!!.txt
              /Mpeg-°¡¿ä/[ÀÌ½ÂÈ¯] ÃµÀÏµ¿¾È MV-by ego.MPG
        CWD:  /
              /Mpeg-°¡¿ä/
        FAILURES: 0
```

Audit Logs Show Web Site

```
-----------------------------------------------------------
2529427  169.237.61.27  -->  202.30.143.17  (1402 -> 80)
from: 19:01:36 ( 7/23/1998)  to: 19:01:38 ( 7/23/1998)
client flags: SAF     server_flags: SAF
-----------------------------------------------------------
```

http://www.shinbiro.com/home.html

Non-ASCII Character Sets

Lessons Learned

- Detects exactly what I asked it to detect - make sure I ask it to detect the right thing
- .cshrc is not a hidden file
- tokenize file and directory names and analyze tokens
- The world does not run on ASCII
  - Weapons inspector in Iraq had similar probem

Sunkill Example

- Sunkill exploits a feature in Solaris system designed to support a large number of logged in user
- Simple denial-of-service attack
- Place telnet protocol in unusual state
- Send a large number of ctrl-d (?) bytes to server

False +, Wrong Sunkill

```
          Thu Jul 23 17:35:07 1998

-----------------------------------------------------------
2306714  128.120.251.229  --> 128.120.8.190 (2852 -> 23)
from: 17:16:52 ( 7/23/1998)  to: 17:38:46 ( 7/23/1998)
client flags: SAF     server_flags: SAF
        ---- TELNET ------------------------------------
        Terminal type: DEC-VT100
        Terminal size: 30 rows X 89 columns
        ---- LOGIN ---------------------------------------
        login count: 1,  password count: 1
        Login: ez059422
        Password: xxxxxxxx
        ---- STRING MATCHES ------------------------------
        From Server: Last login: = 1
```

Lessons Learned

- Telnet client and servers get out of sync
- Need to refine “signature”
- Need better Telnet audit log file?
- What works in the lab doesn’t always work in the real world

Statistics Collected per IP Address

- TCP_SES_NEW_C
- TCP_SES_NEW_S
- UDP_SES_NEW_C
- TCP_SES_ABORT1_C
- TCP_SES_ABORT1_S
- TCP_SES_ABORT2_S
- TCP_SES_ABORT3_C
- TCP_SES_ABORT3_S
- TCP_RST_SND
- TCP_RST_RCV
- TRANSPORT_FRAG_HEAD_SND
- ICMP_PORT_UNREACH_SND
- ICMP_PORT_UNREACH_RCV
- ICMP_TIMXCEED_RCV
- ICMP_ECHO_REQUEST_SND
- ICMP_ECH_REQUEST_RCV

Measures Rate Events Are Occurring

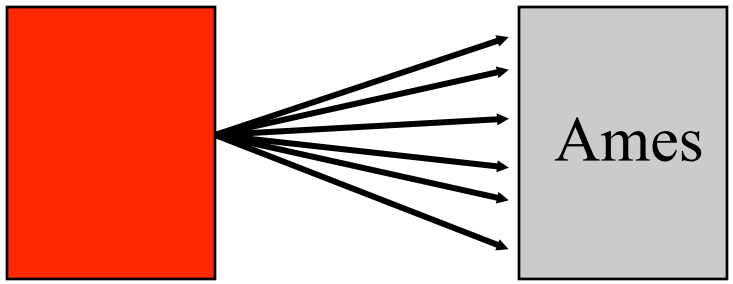
- Uses floating point values (expensive?)
- Uses [N]IDES statistics measure
- Define Zones where a statistic is
  - zone 0 is from 0 to 120 events/minute
  - zone 7 has at least 720 events/minute
  - zone 8 has at least 840 events/minute
- Attackers like to use brute force, even with new attacks

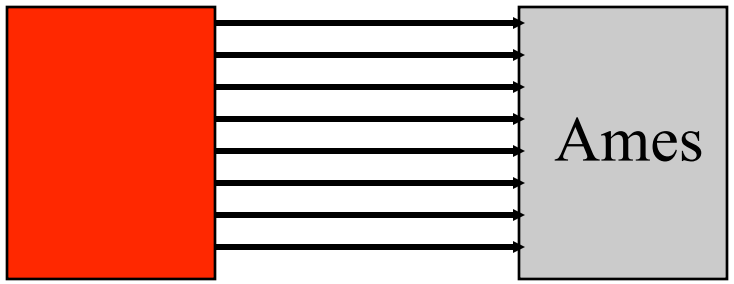
Signature Port Scan

C5300-old-1-58.hfa.netvision.net.il

ames.cipic.ucdavis.edu

```
   62.0.149.60  stat =    TCP_SES_NEW_C   old =  7  new =  8
128.120.67.159  stat =    TCP_SES_NEW_S   old =  7  new =  8
128.120.67.159  stat =      TCP_RST_SND   old =  7  new =  8
   62.0.149.60  stat =      TCP_RST_RCV   old =  7  new =  8
   62.0.149.60  stat = TCP_SES_ABORT1_C   old =  7  new =  8
128.120.67.159  stat = TCP_SES_ABORT1_S   old =  7  new =  8
```





Vertical Scan, Parallel Scan

```
1149081  62.0.149.60  -->  128.120.67.159  (29900 -> 1)
1149152  62.0.149.60  -->  128.120.67.159  (29901 -> 2)
1149170  62.0.149.60  -->  128.120.67.159  (29902 -> 3)
1149171  62.0.149.60  -->  128.120.67.159  (29903 -> 4)
1149215  62.0.149.60  -->  128.120.67.159  (29904 -> 5)
1149216  62.0.149.60  -->  128.120.67.159  (29905 -> 6)
1149218  62.0.149.60  -->  128.120.67.159  (29907 -> 8)
```

```
1149491  62.0.149.60  -->  128.120.67.159  (31003 -> 96)
1149217  62.0.149.60  -->  128.120.67.159  (29906 -> 7)
1149509  62.0.149.60  -->  128.120.67.159  (31004 -> 97)
1149510  62.0.149.60  -->  128.120.67.159  (31005 -> 98)
```

Applying Filter to Audit Trail: Actual Active Servers

```
TCP     client  addr    =       62.0.149.60
TCP     server  addr    =       128.120.67.159
TCP     server  flags   <       S

1149081  62.0.149.60  -->  128.120.67.159  (29900 -> 1)
1149275  62.0.149.60  -->  128.120.67.159  (29915 -> 13)
1149309  62.0.149.60  -->  128.120.67.159  (29927 -> 21)
1149217  62.0.149.60  -->  128.120.67.159  (29906 -> 7)
1149382  62.0.149.60  -->  128.120.67.159  (30036 -> 37)
1149306  62.0.149.60  -->  128.120.67.159  (29922 -> 19)
1149468  62.0.149.60  -->  128.120.67.159  (30579 -> 80)
1149467  62.0.149.60  -->  128.120.67.159  (30578 -> 79)
1149261  62.0.149.60  -->  128.120.67.159  (29908 -> 9)
1149342  62.0.149.60  -->  128.120.67.159  (29929 -> 23)
1150640  62.0.149.60  -->  128.120.67.159  (8330 -> 903)
```

Horizontal Scan Signature

hawk.phys.nd.edu (129.74.75.162)

```
129.74.75.162  stat =  TCP_SES_NEW_C  old =  7  new =  6
129.74.75.162  stat =  TCP_SES_NEW_C  old =  6  new =  7
129.74.75.162  stat =  TCP_SES_NEW_C  old =  7  new =  8
```

```
2374817 129.74.75.162  -->   169.237.1.250  (13940 -> 53)
2373553 129.74.75.162  -->  128.120.252.10  (13937 -> 53)
2373771 129.74.75.162  --> 169.237.250.250  (13939 -> 53)
2375101 129.74.75.162  -->   169.237.1.250  (13940 -> 53)
2381503 129.74.75.162  -->    128.120.55.3  (13942 -> 53)
2381963 129.74.75.162  -->   128.120.57.95  (14001 -> 53)
2382843 129.74.75.162  -->  128.120.17.102  (14002 -> 53)
```

From DNS to POP/IMAP

```
2392131 129.74.75.162  -->  128.120.54.45  (14362 -> 53)
2392158 129.74.75.162  -->  128.120.54.45  (14363 -> 53)
2392252 129.74.75.162  -->  128.120.54.45  (14364 -> 53)
2392289 129.74.75.162  -->  128.120.54.45  (14400 -> 53)
```

```
2388663 129.74.75.162  --> 128.120.59.216  (14348 -> 53)
2396782 129.74.75.162  -->  128.120.1.250  (19447 -> 110)
2396817 129.74.75.162  -->   128.120.2.19  (19616 -> 110)
2396822 129.74.75.162  -->   128.120.2.24  (19684 -> 110)
2396824 129.74.75.162  -->   128.120.2.26  (19686 -> 110)
2396839 129.74.75.162  -->   128.120.2.38  (19922 -> 110)
```

Horizontal with a Twist

```
address = linux.recochem.com (207.236.153.180)

207.236.153.180  stat = TCP_SES_NEW_C  old = 5 new = 6
207.236.153.180  stat = TCP_SES_NEW_C  old = 6 new = 7
207.236.153.180  stat = TCP_SES_NEW_C  old = 7 new = 6
207.236.153.180  stat = TCP_SES_NEW_C  old = 6 new = 7
207.236.153.180  stat = TCP_SES_NEW_C  old = 7 new = 8
```

Selected Service Scan

```
-----------------------------------------------
CLIENT ADDRESS 207.236.153.180: 20243 connections
    SERVER PORT    80: 3681 connections
    SERVER PORT    23: 3440 connections
    SERVER PORT    53: 3291 connections
    SERVER PORT    79: 3253 connections
    SERVER PORT   143: 3244 connections
    SERVER PORT   110: 3244 connections
    SERVER PORT   111: 68 connections
    SERVER PORT   113: 22 connections
```

Consistent Across Hosts

```
-----------------------------------------------
CLIENT ADDRESS 207.236.153.180: 20243 connections
    SERVER ADDRESS   128.120.56.38: 15 connections
    SERVER ADDRESS 169.237.137.130: 13 connections
    SERVER ADDRESS    128.120.22.4: 12 connections
    SERVER ADDRESS   128.120.16.10: 11 connections
    SERVER ADDRESS   128.120.8.122: 11 connections
    SERVER ADDRESS   128.120.8.178: 11 connections
    SERVER ADDRESS    128.120.15.7: 11 connections
    SERVER ADDRESS  128.120.15.154: 11 connections
    SERVER ADDRESS  128.120.15.158: 11 connections
    SERVER ADDRESS  128.120.15.161: 11 connections
```

Snapshot of Single Target

```
-----------------------------------------------------------
2657428  207.236.153.180  -->    128.120.16.10  (8460 -> 80)
from: 20:15:14 ( 7/23/1998)  to: 20:15:14 ( 7/23/1998)
client flags: S       server_flags:  A R
-----------------------------------------------------------
2657366  207.236.153.180  -->    128.120.16.10  (7568 -> 79)
from: 20:15:12 ( 7/23/1998)  to: 20:15:14 ( 7/23/1998)
client flags: SAF     server_flags: SAF
-----------------------------------------------------------
2657450  207.236.153.180  -->    128.120.16.10  (8640 -> 143)
from: 20:15:14 ( 7/23/1998)  to: 20:15:14 ( 7/23/1998)
client flags: S       server_flags:  A R
-----------------------------------------------------------
2657404  207.236.153.180  -->    128.120.16.10  (8261 -> 23)
from: 20:15:13 ( 7/23/1998)  to: 20:15:15 ( 7/23/1998)
client flags: SAFR    server_flags: SAF
```

Vertical UDP Port Scan?

```
cleopatra.rz.tu-clausthal.de (139.174.253.10)
skate.ece.ucdavis.edu (128.120.54.55)

 128.120.54.55  stat =  ICMP_PORT_UNREACH_SND  old = 5  new = 6
139.174.253.10  stat =  ICMP_PORT_UNREACH_RCV  old = 5  new = 6
```

Vertical TCP Scan?

```
Name:     philo.ucdavis.edu
Address:   128.120.237.201

Name:     durer.CS.Berkeley.EDU
Address:   128.32.42.135
```

```
128.120.237.201 stat =    TCP_SES_NEW_S   old = 7 new = 8
  128.32.42.135 stat =    TCP_SES_NEW_C   old = 7 new = 8
128.120.237.201 stat =      TCP_RST_SND   old = 7 new = 8
  128.32.42.135 stat =      TCP_RST_RCV   old = 7 new = 8
  128.32.42.135 stat = TCP_SES_ABORT1_C   old = 7 new = 8
128.120.237.201 stat = TCP_SES_ABORT1_S   old = 7 new = 8
```

Confused?

```
8188586    128.32.42.135  -->  128.120.237.201  (42548 -> 1701)
8188598    128.32.42.135  -->  128.120.237.201  (42549 -> 1701)
8188602    128.32.42.135  -->  128.120.237.201  (42550 -> 1701)
8188606    128.32.42.135  -->  128.120.237.201  (42551 -> 1701)
8188610    128.32.42.135  -->  128.120.237.201  (42552 -> 1701)
8188614    128.32.42.135  -->  128.120.237.201  (42553 -> 1701)
8188622    128.32.42.135  -->  128.120.237.201  (42554 -> 1701)
8188628    128.32.42.135  -->  128.120.237.201  (42555 -> 1701)
8188630    128.32.42.135  -->  128.120.237.201  (42556 -> 1701)
```

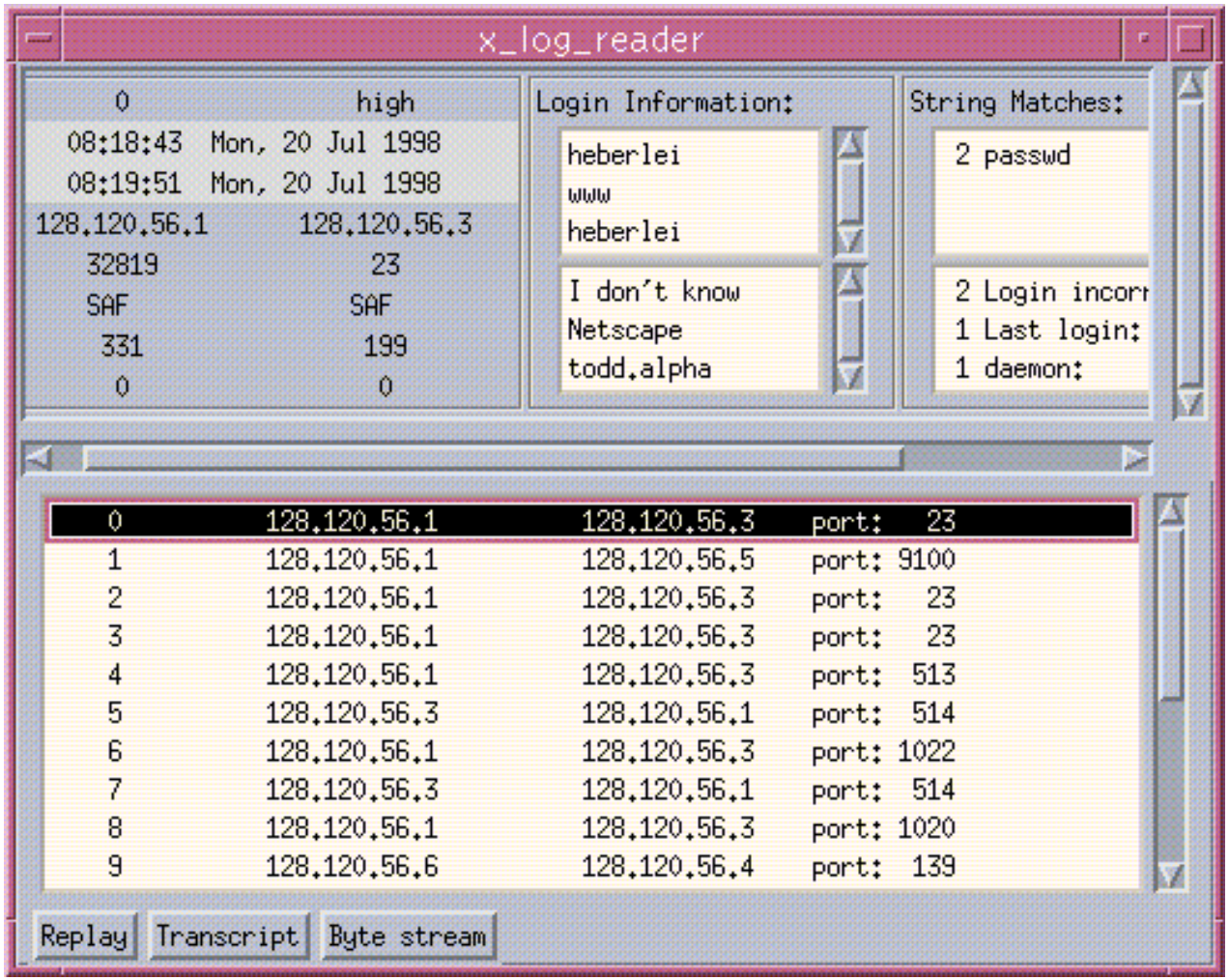
Lessons Learned

- Best way to get improvement is to get developers to eat their own dog food
- I don’t want to hear about an attempted attack
  - Summary of attack
  - Summary of successful attack
- Profile attack tools (and attackers?)
- Unusual stuff on the network

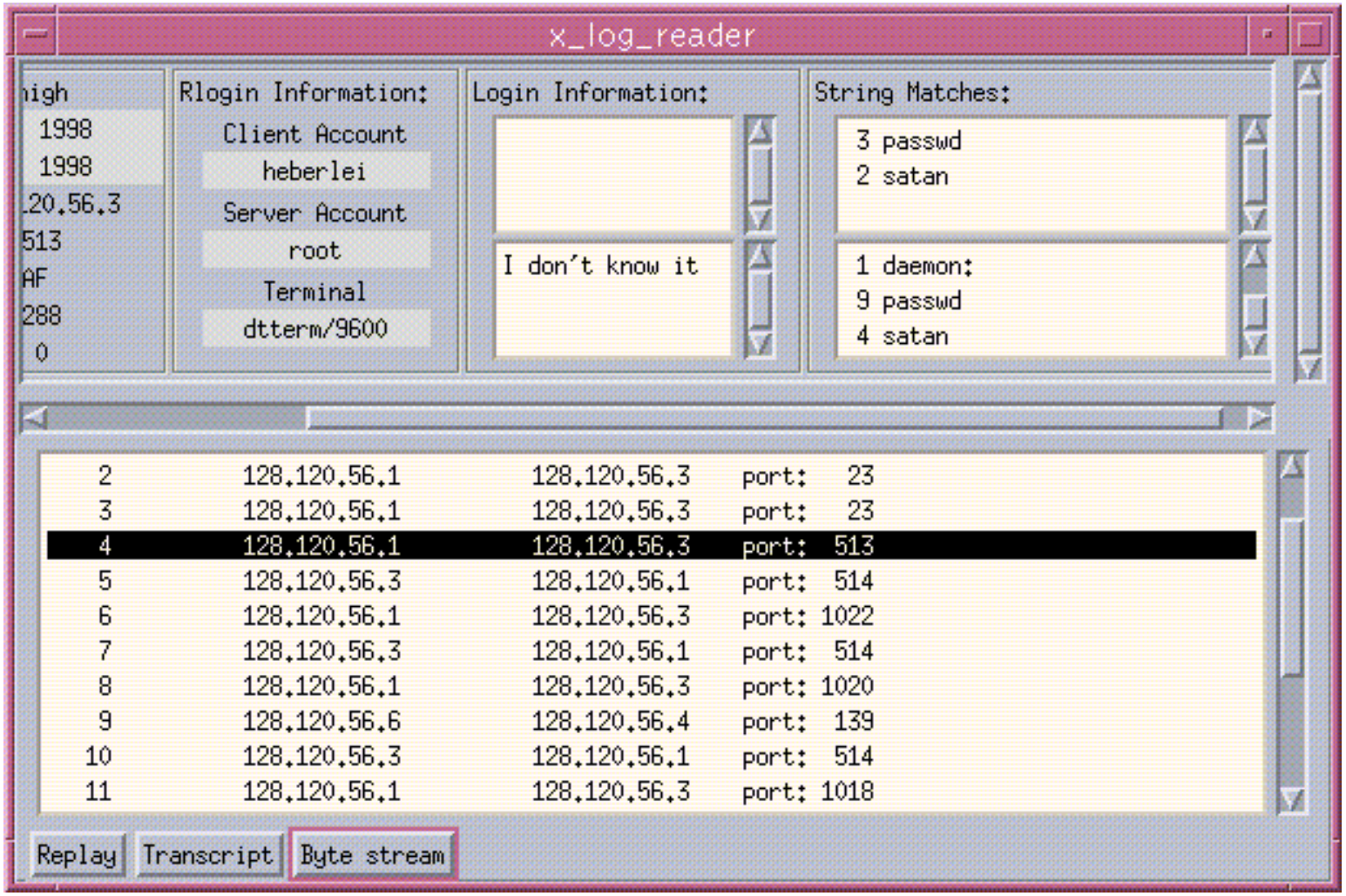
X windows Log reader

- Simple audit log viewer which wraps a number of X objects and applications
- Example Scenario:
  - Someone hands you Tcpdump files
  - Collect Tcpdump files on Sidewinder
  - Nrat or Tracker audit logs
  - Event-triggered reporting

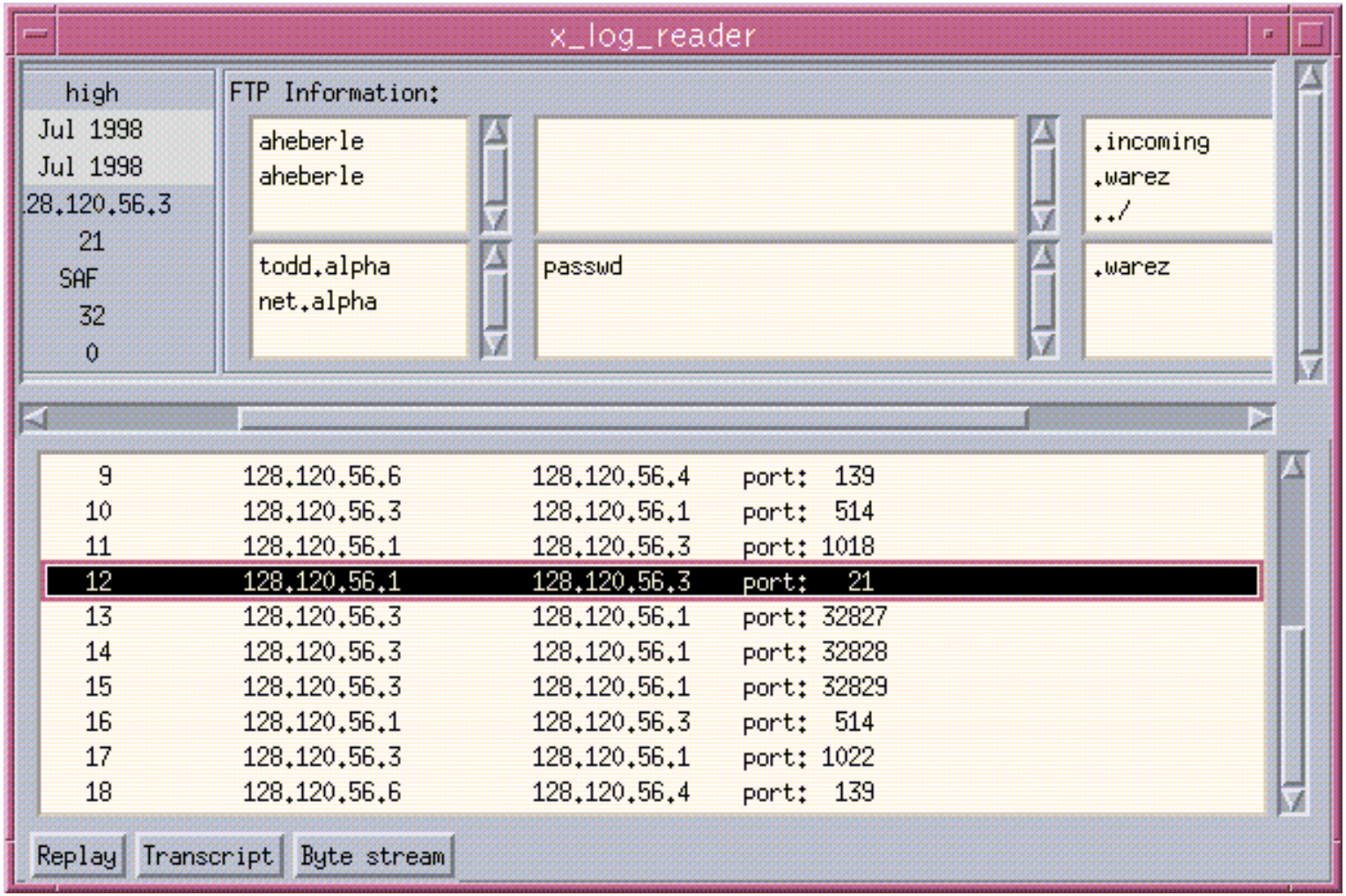
X Log Reader: Telnet



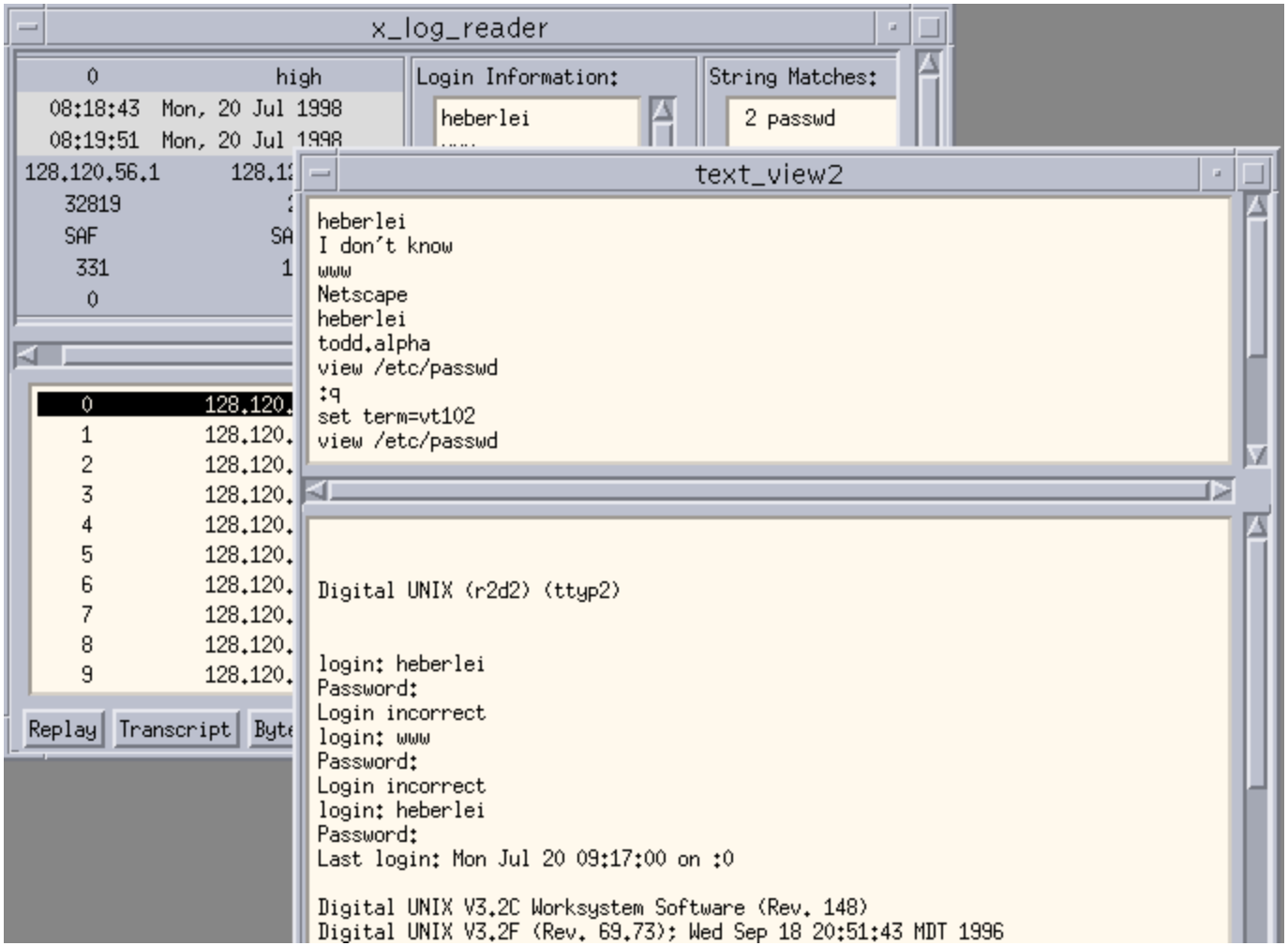
X Log Reader: Rlogin



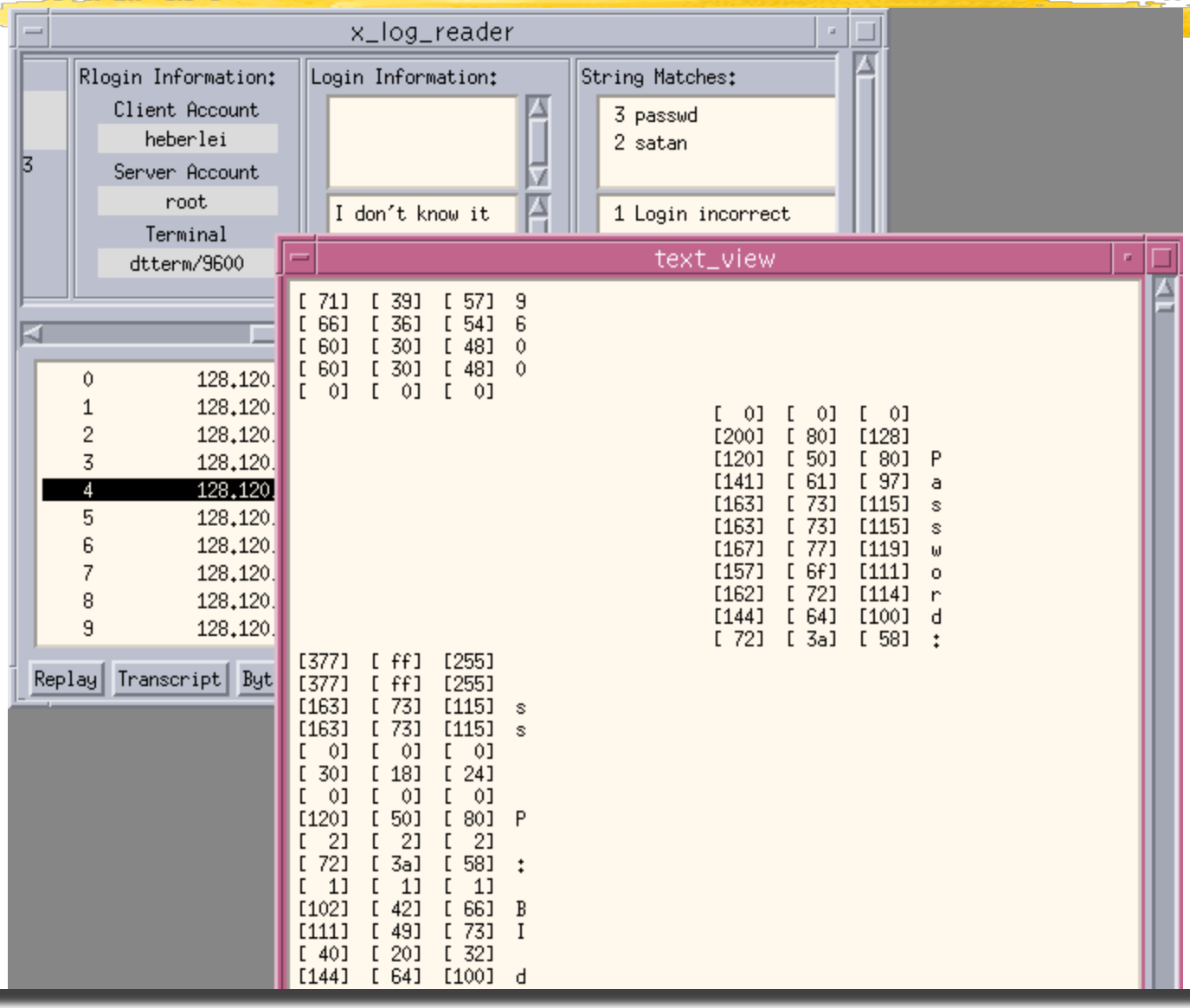
X Log Reader: FTP



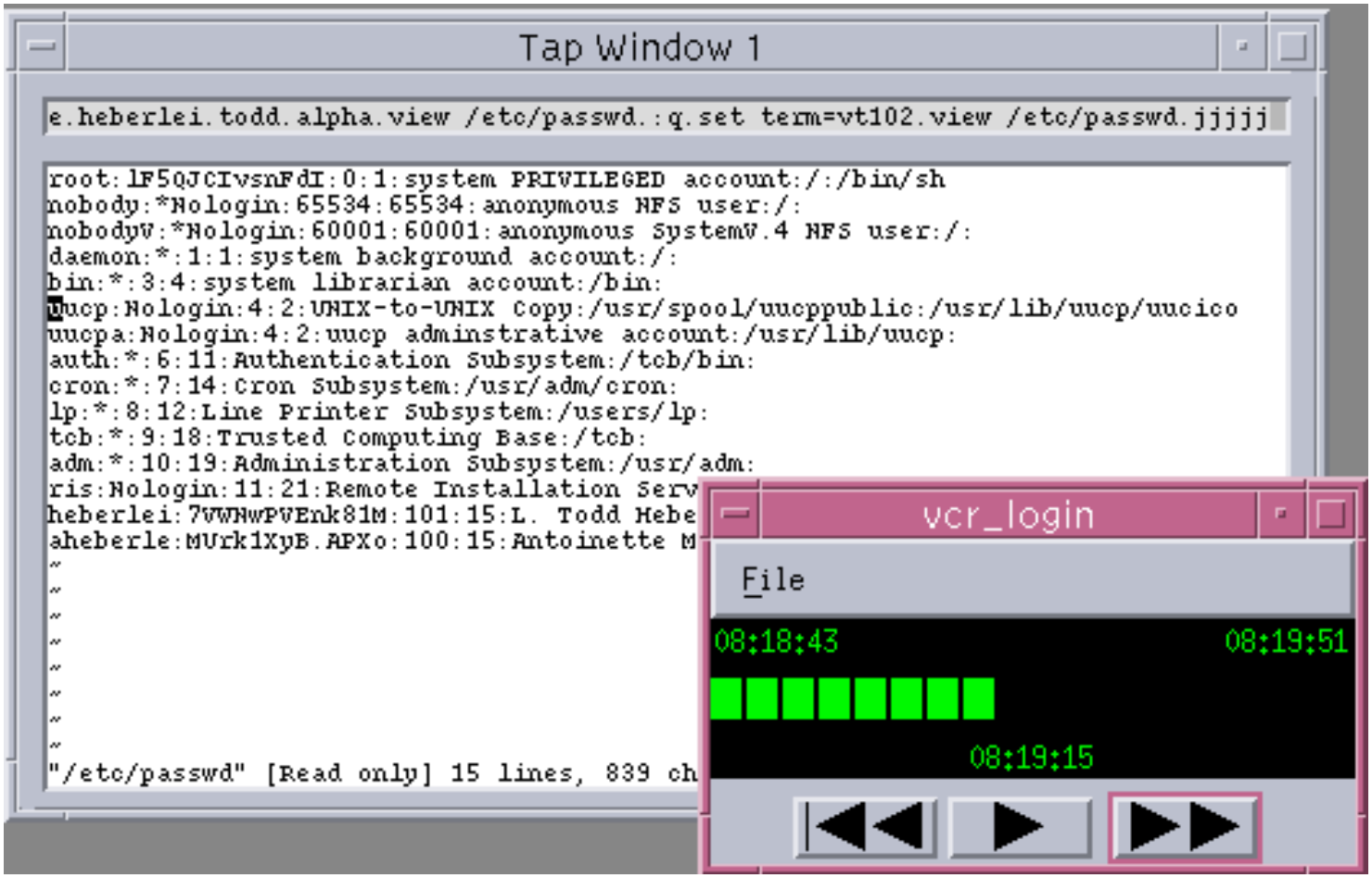
Transcript View



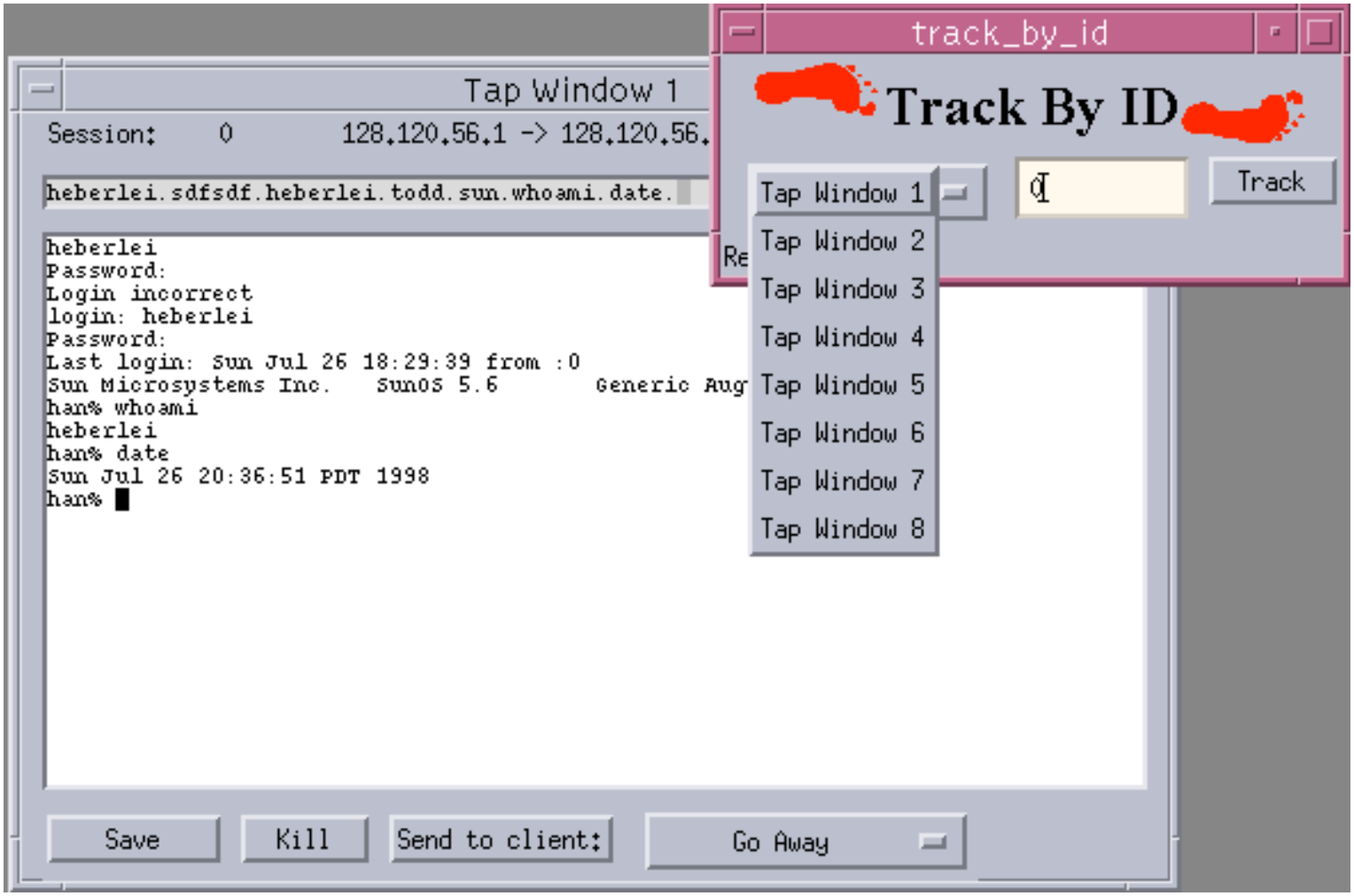
Byte Stream View



VCR Login Program

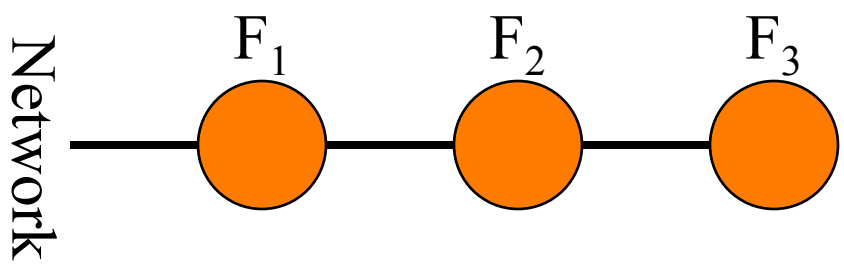


Track By ID: Epic App.



Risks

- New network technologies
  - switched Ethernet (Lawrence's question)
  - ATM door-to-door
    - UC Davis is moving to ATM to the building
    - OC-12 to campus, 4 OC-3, OC-3 to building
    - No IP packets outside the building
- Encryption
- Drop or Block



Encryption Risks