

DonsProxy

Don Branson
October 21, 2008

A Brief History

- ▶ Original version created in 1998 for A. G. Edwards, Inc.
- ▶ Current version is a complete re-write to:
 - ▶ Be Open Source
 - ▶ Be more OO, less JOBOL
 - ▶ Be Faster
 - ▶ Be more easily extended
 - ▶ Include a GUI

What It's For

- ▶ Watching traffic between two networked applications
 - ▶ Typically a browser and a server
 - ▶ Often between two servers (say, Apache and Tomcat)
 - ▶ Could be any two apps that connect by tcp – user your imagination
- ▶ Collecting traffic to feed other tools
 - ▶ Traffic generation scripts
 - ▶ Traffic analysis
- ▶ More like Tcpmon than it is like Ethereal/Wireshark

Special Features

- ▶ Extensible
 - ▶ Connection behavior decorators
 - ▶ Subscribers
 - ▶ XmlHttp Subscriber output processing
- ▶ NIO-based for optimal performance
- ▶ Does a man-in-the-middle to expose HTTPS traffic

Problems I've Solved With Donsproxy

- ▶ Session loss
 - ▶ Tomcat and Hadoop – different ports on the same host
 - ▶ Tomcat session lost – had to keep signing in
 - ▶ DonsProxy showed JSESSIONID reset – browser using overwriting Tomcat's JSESSIONID with Hadoop/Jetty JSESSIONID
- ▶ Simulate satellite latency (Original version)
 - ▶ A. G. Edwards Broker Workstation
- ▶ Seeing the difference between F5 and ctrl-F5
- ▶ Watch traffic between an application server and its reverse proxy
 - ▶ What is the reverse proxy telling the app server?
- ▶ KP SOA Pilot – Are caching directives being set correctly?

Basic Example

- ▶ Breakout
 - ▶ Graphic - Basic Example
- ▶ Google traffic – google.com
 - ▶ Proxy tab shows connection activity – who closed what, etc.
 - ▶ HttpHeaders shows who sent what
- ▶ Google traffic with 20Kbps throttle
 - ▶ Won't see true picture – most sites link external for graphics
- ▶ Same for amazon.com

Advanced Example

- ▶ Use httpd to route requests, so we don't have to continuously change Donsproxy config
- ▶ Configuring Apache httpd
- ▶ Now we can see the true picture when throttling sites that link external for graphics
 - ▶ Sites link external:
 - ▶ To distribute load
 - ▶ To increase Google rank, which calculates speed only for content from the main site
 - ▶ Enable throttle and/or latency and show some sites
- ▶ Notice the number of connections that a browser opens

Configurations

- ▶ When you close then re-open DonsProxy, your settings are still there
- ▶ Use save/load to name and re-use common configurations

Encrypted Data

- ▶ No longer reliable enough to be considered usable – needs a complete re-work
 - ▶ Seems like sites are moving away from TLS 1.0 to SSLv3, which isn't supported
- ▶ Requires “advanced technique,” that is, using a proxy
 - ▶ Why? It triggers on the CONNECT handshake a browser sends to the proxy

JMX

- ▶ Run jconsole
- ▶ View MBeans
- ▶ View threads, filter with DP
- ▶ Useful for developers of DonsProxy to get visibility into buffer states, threads, etc.

HttpXml Subscriber

- ▶ Original (AGE) version
 - ▶ Had similar XML generation capability
 - ▶ Many XSLT transforms to convert XML to bash or perl scripts
 - ▶ Used scripts to monitor and load agedwards.com
- ▶ Would someone like to contribute some XSLT?
- ▶ Currently writes XML to a file and logs summary to console
- ▶ Analyzers
 - ▶ Extend `HttpXmlHandler`
 - ▶ `CachingSummary` example parses xml output and writes HTML summary of caching directives
- ▶ Show sample XML output

Write A Subscriber

- ▶ Here's how to approach it:
 - ▶ Set up LDAP server and browser, and check setup
 - ▶ Point Donsproxy at server's 389, and listen on an available port
 - ▶ Point the ldap browser at DonsProxy
 - ▶ Run with Print subscriber enabled – see the garbage?
 - ▶ Find LDAP protocol info – IETF RFC2251

Write A Subscriber - continued

- ▶ New LDAP subscriber, the quick & dirty way (i.e., not using Jigloo designer)
 - ▶ Copy HttpHeadersSubscriberView.properties (~4 seconds)
 - ▶ Copy/modify HttpHeadersSubscriberPresenter.java (~10 seconds)
 - ▶ Copy/modify HttpHeadersSubscriberView.java (~10 seconds)
 - ▶ Modify ApplicationView (~2 minutes)
- ▶ Create LdapHeadSubscriber that extends AbstractSubscriber (~5 seconds)
- ▶ Implement `processPacket()` (varies with complexity of protocol – but think hours for simple protocols to days for complex protocols)

Write A Subscriber - source

```
public class LdapHeadSubscriber extends AbstractSubscriber {

    @Override
    protected void processPacket(PacketHeader header, byte[] payload, OutputStream out)
        throws IOException
    {
        PrintStream printStream = new PrintStream(out);

        printStream.println("Sender is " + header.getSourceType() + " at "
            + header.getSenderIP() + ":" + header.getSenderPort());
        printStream.println("Packet ID:" + header.getPacketIdentifier());

        printStream.println("Decoding LDAP packet of " + payload.length + " bytes.");

        try {
            dumpBerEncodedBuffer("", printStream, payload);
        } catch (IOException e) {
            printStream.println("ERROR: Exception decoding packet: " + e.getMessage());
        }
        printStream.println();
    }

    private void dumpBerEncodedBuffer(String indent, PrintStream printStream, byte[] buffer)
        throws IOException
    {
        ...
    }
}
```

Write A Channel Decorator

- All of the Channel Decorators currently operate on the read side
- Look at ThrottleDecorator or ChannelLossDecorator as examples
- Tie your Decorator in via ByteReaderFactory or ByteWriterFactory
- Decorators are enabled via system properties, a vestige of the command-line version
- Add your decorator to the GUI in the DecoratorInputPanel
 - Real estate is an issue here; you'll have to use the Jigloo eclipse plugin to edit it
- Update ProxyView and IProxyView, adding your decorator

Programmatically Start a Proxy

- ▶ ProxyPresenter as an example
 - ▶ Enables Channel Decorators
 - ▶ Starts Proxy
- ▶ Review `startProxy()`

```
public void startProxy() {  
    if (proxy == null) {  
        ...  
        processDecorators();  
        proxy = new Proxy();  
        proxy.start(createproxySettings());  
        ...  
    }  
}  
  
private ProxySettings createproxySettings() {  
    return new ProxySettings(view.getListenPort(), view.getTargetHost(),  
        view.getTargetPort(), ProxySettings.DEFAULT_REQUEST_PORT,  
        ProxySettings.DEFAULT_RESPONSE_PORT, ProxySettings.DEFAULT_DUPLEX_PORT);  
}
```

Future

- ▶ Please contribute! Contribution style modeled after Apache – send a patch or whatever, and I'll merge it in. What's needed:
 - ▶ Fix SSL
 - ▶ XML processing: XSLTs, Java parsers, or whatever
 - ▶ More protocol-specific listeners – LDAP, whatever

Where To Get DonsProxy

<http://donsproxy.sourceforge.net>