NetAnalyzer 使用说明书

一．抓包与协议分析

墨云软件 ● 2020-07-25

目录

1. 初识 NetAnalyzer

NetAnalyzer 是一款集网络数据采集、报文协议分析、统计、网络流量监控于一体的网络工具软件，你可以可以通过该软件获取网络数据，并对相关的数据进行分析，对于网络管理人员或从事网络软件开发的人是一个不错的工具。系统提供多种辅助工具方便用户更加深入的对原始数据进行还原。目前该系统已经支持 80 多种协议，覆盖 TCP/IP、IPX 等协议模型各层，支持 EthernetII、PPP、Cisco HDLC、Linux SLL 等多种底层网络，并且提供 TCP、UDP 载荷数据还原与分析，为符合国内用户软件还提供了多种中文编码方式，方便查看中文数据。另外还提供了远程抓包功能，方便对远程机器进行监控，作为目前最新版的协议分析工具，使用了当下流行的 Ribbon 界面，并且为了软件兼容性和易用性，新版的协议分析工具做了大量处理工作；包括 NetAnalyzer 整体架构的改造，以适应不断增加的功能点，优化代码结构，简化使用方法，还有对于 Winpcap 驱动，则单独提取相关的文件出来，在安装时自动安装，这样就可以避免安装完软件之后还要安装驱动的尴尬情景。除此之外还包括功能扩展、稳定性增强等多种方面的改善。

配置要求

系统：WinXP/Win7/Win8/Win8.1/Win10 （x86 和 x64）

平台：NET Framework4.0

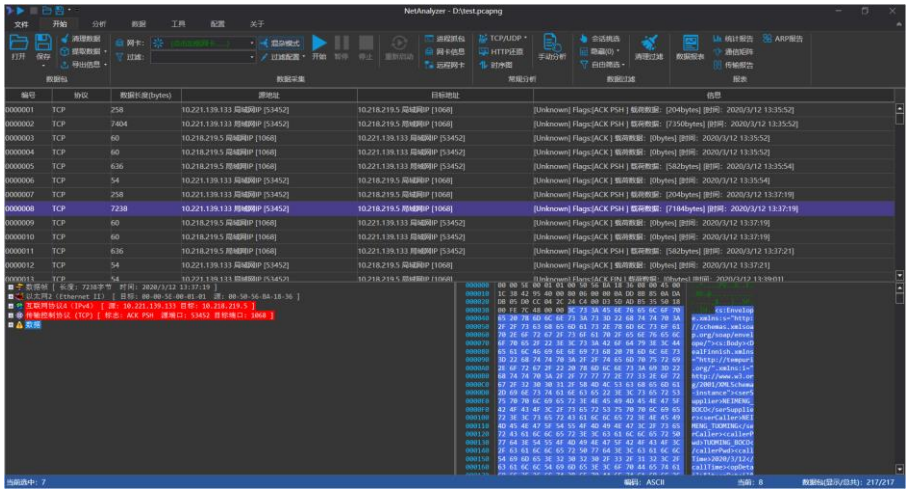
驱动：Winpcap4.1.3

浏览器：Internet Explorer7.0 以上版本

扩展开发：Visual Studio 2010 以及以上版本

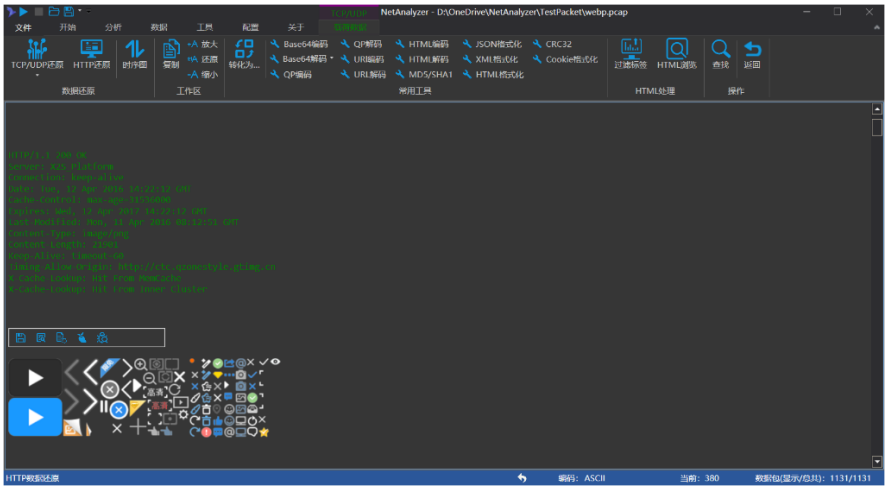
1.1.界面介绍

NetAnalyzer 工具栏部分使用最新的 Ribbon 界面，大大简化操作方式，而在工作区域继承原来的设计风格，在进行融合之后既具有主流软件的设计感，又具备操作上的易用性。



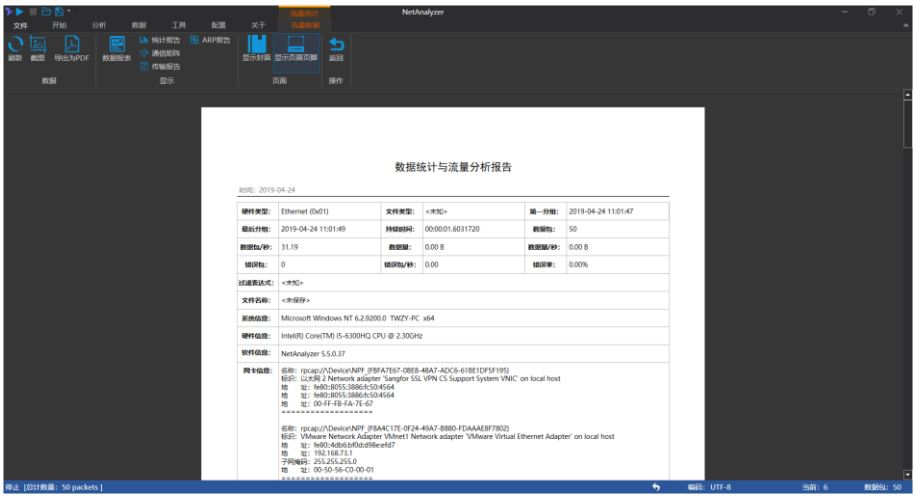
NetAnalyzer 主界面

在工具区域部分使用分页模式，即通过不同的模式进入不同的工作区域，如上面图片就显示了数据抓包的工作界面，在进行查看载荷数据的时候，我们又可以进入载荷数据查看模式，此时通过查看 tcp 载荷数据或是 http 数据，就可以进入载荷数据查看模式，在该模式下可以查看所选连接的载荷数据，并能对一些简单信息进行现场还原。



载荷数据查看界面

当在菜单栏中选择了统计或报表相关的内容的时候，则进入了报表统计模式，该模式下展示了当次数据采集或打开文件中包含的全部统计信息。



统计与报表界面

目前 NetAnalyzer 只支持三种模式的切换，对于载荷数据查看模式和数据统计模式，都可以通过对应下的**返回**功能，则模式切换为数据采集模式，通过状态栏

返回抓包模式

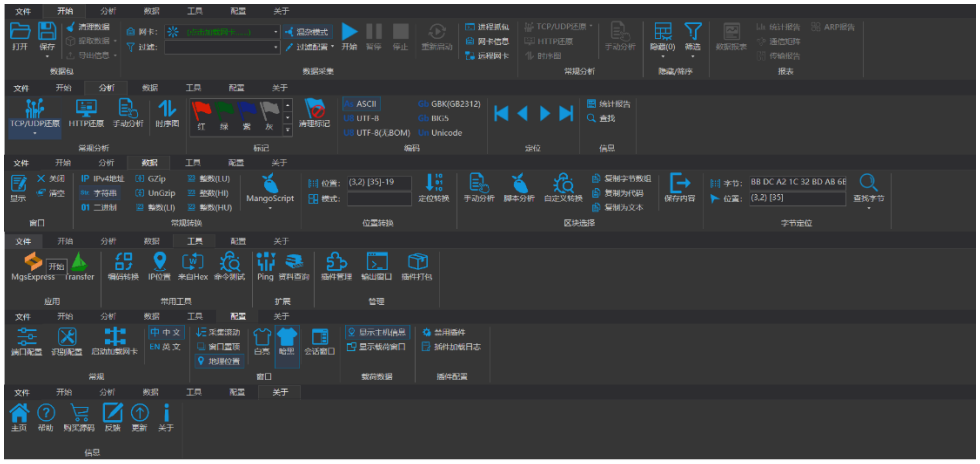
通过状态栏返回抓包模式

返回主界面也可以返回到采集模式。

对 TCP/UDP 载荷数据的分析，一直以来是 NetAnalyzer 的重要功能。对于载荷数据分析，尤其是对于应用层协议的内容还原一直是作为 NetAnalyzer 核心功能在开发。在该版本中载荷数据分析功能被极大的增强，不但加强了载荷数据分析的稳定、准确性，更提供了大量的分析功能，和数据转换工具。

1.2.功能预览

NetAnalyzer 主要分为 7 个标签页，涵盖从网络包抓取到数据分析统计、扩展工具、系统设置等各种功能点。



NetAnalyzer 全部菜单

关于信息

和常规的 Ribbon 界面一样，NetAnalyzer 也包含了文件标签，在该标签中除了常规的文件操作外提供了很多 NetAnalyzer 的信息，有兴趣的读者可以去自行了解。

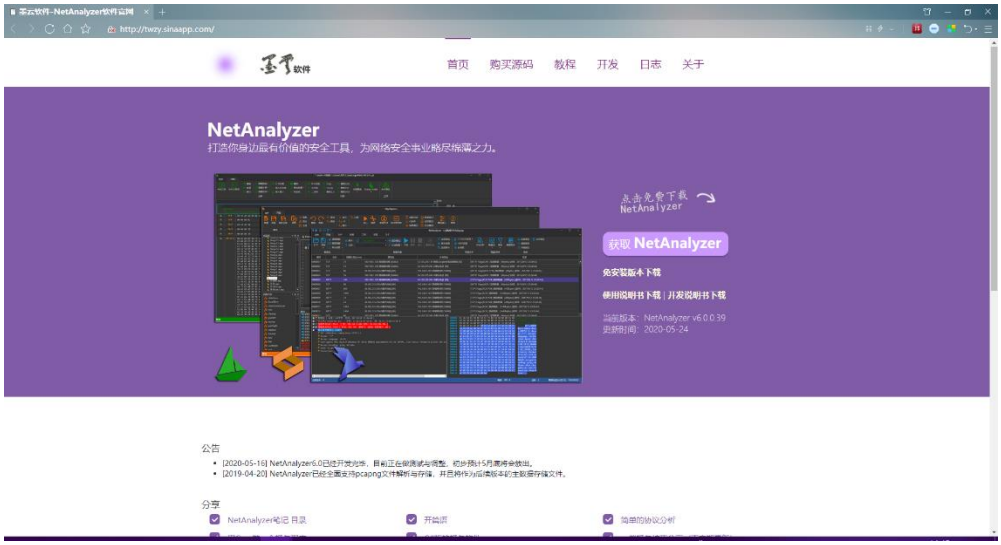
对于剩余的 6 个标签页的功能项将会在后面的内容中进行详细的说明。

2. NetAnalyzer 使用方法

2.1.快速开始

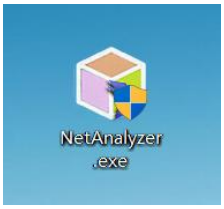
前面说了很多，那让我们快速的开始使用 NetAnalyzer 吧。

2.1.1. 首先从墨云软件官网（http://twzy.sinaapp.com/）下载 NetAnalyzer，点击**获取 NetAnalyzer** 就可以获取到 NetAnalyzer 的安装文件了，如果不想安装也可以直接下载免安装版本，解压完成后就可以直接使用了。



NetAnalyzer 官网页面

2.1.2 安装 NetAnalyzer，安装常规的 Windows 安装流程，完成对 NetAnalyzer 的安装即可，NetAnalyzer 依赖.Net 4.0 和 winpcap，该安装包已经集成了相关的组件，只要正常安装就可以使用了，当前的的 NetAnalyzer 已经支持不用加载 Winpcap 相关的驱动就可以完成除数据采集外的全部功能。



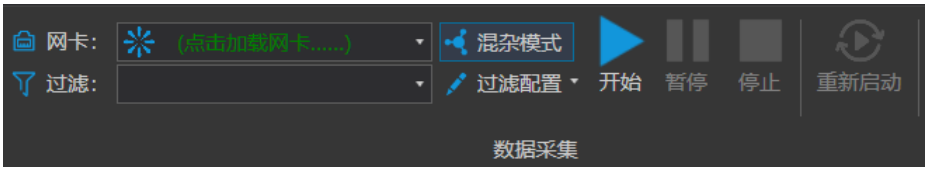
NetAnalyzer 安装包

2.1.3. 安装完成后，会在桌面自动创建 NetAnalyzer、MgsExpress、Transfer 三个程序

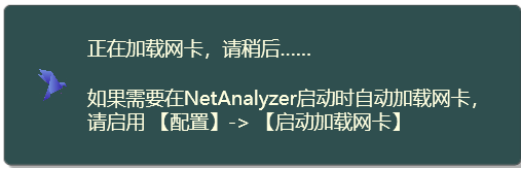
的图标，NetAnalyzer 主要负责网络数据抓包与常规分析、MgsExpress 则为 MangScript 的集成开发环境，Transfer 为自定义手动分析工具，这里优先说明 NetAnalyzer，双击启动。

NetAnalyzer 主程序菜单

2.1.4. 在【开始】标签中选择当前系统连接网络的网卡，然后点击开始，开始抓包。

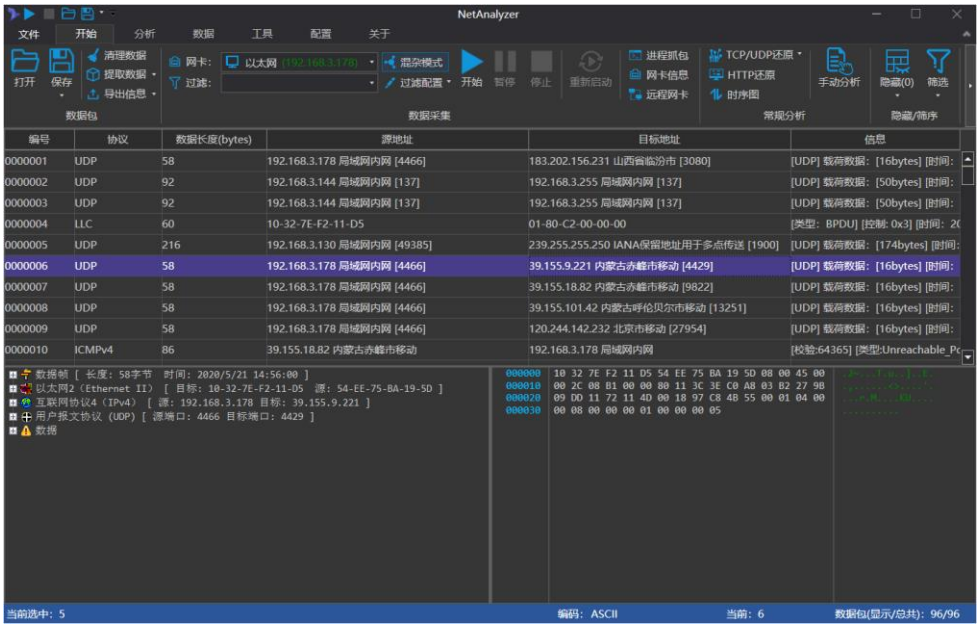


NetAnalyzer 抓包菜单



NetAnalyzer 延时加载网卡界面

在 NetAnalyzer5.5 之后的版本中为了优化启动速度，默认不在启动时候加载网卡(可以通过配置菜单修改为启动时加载)，当使用的时候点击网卡下拉列表或是点击开始才会开始加载网卡。



采集到的数据包信息

至此，完成 NetAnalyzer 最基本的使用方法。

2.2.数据获取

通过上面的说明，我们已经可以很快的获取到网络数据了，但是，仅仅知道上面内容却远远不够。

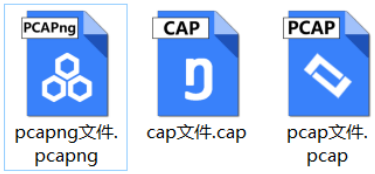
在 NetAnalyzer 中将数据加载到软件有三种方式。

- 数据文件
- 字节字符串录入
- 网卡采集

抓包开始菜单

在**开始**标签中数据包组合数据采集组包含了这三种功能点

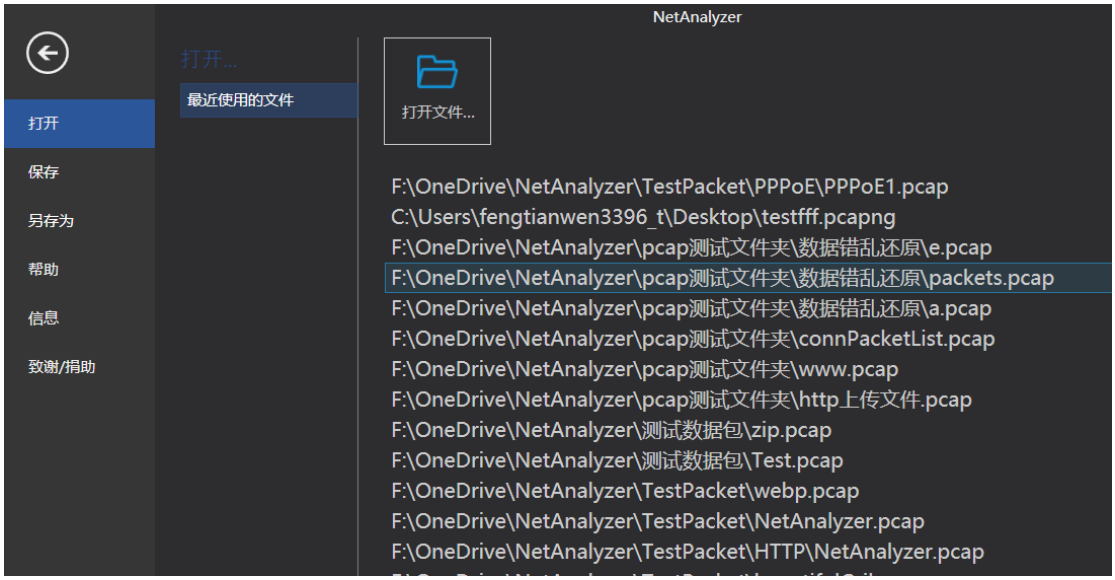
数据文件，很多的数据采集软件都具备将采集到的数据包汇总成为文件的功能，用于交流和存储，通常这些数据通过一定的方案进行存储，目前比较流行的方式是基于 tcpdump 方案的数据存储方式，NetAnalyzer 支持这类的存储方案，所以我们可以将这些数据文件读入到内存中进行分析。



NetAnalyzer 支持的文件以及图标

目前 NetAnalyzer 提供了兼容 tcpdump 的 pcapng、pcap、cap 等文件的支持，并且默认保存类型为 *.pcapng 文件，兼容市面上绝大部分的协议分析软件（如：wireshark 等软件）。

为了更加方便使用数据，NetAnalyzer 在文件便签中添加了**最近使用的文件**功能，双击即可打开文件。还可以通过传统的打开文件方式打开相关的文件。



最近使用的文件列表

读入后的文件，在标题栏将会显示该文件的路径。

当前加载的的文件信息

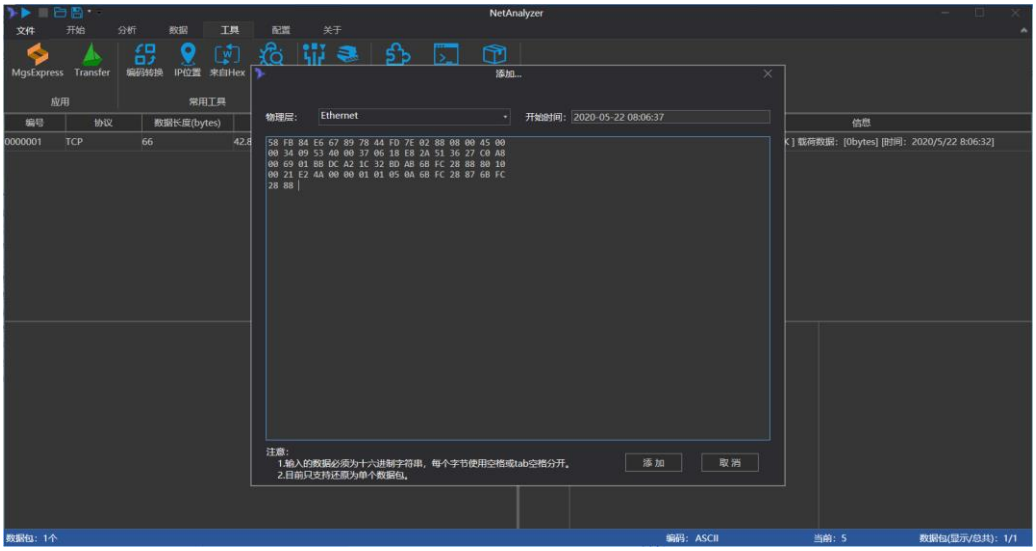
对于一些大文件在读取的时候，状态栏会显示文件载入进度，如果在读取过程中不想再读取了，可以点击后面的 X 按钮就可以停止文件读入，并且清理前期读入的内容。

文件载入进度

同样 NetAnalyzer 支持将内存中获取到的数据包保存为文件，软件默认存储格式为 pcapng，如果是通过文件读入的，则可以使用保存按钮下拉的另存为功能进行存储。

相对于 pcap，cap 等类型的文件，pcapng 文件更多的记录了数据采集时的系统信息，如 CPU、操作系统、网卡等各种信息。这对数据统计非常有用。

字节录入方式，该方式是通过输入一组十六进制数据的字符串，再通过配置硬件类型和时间生成一个数据包，目前该功能只支持生成单个数据包。



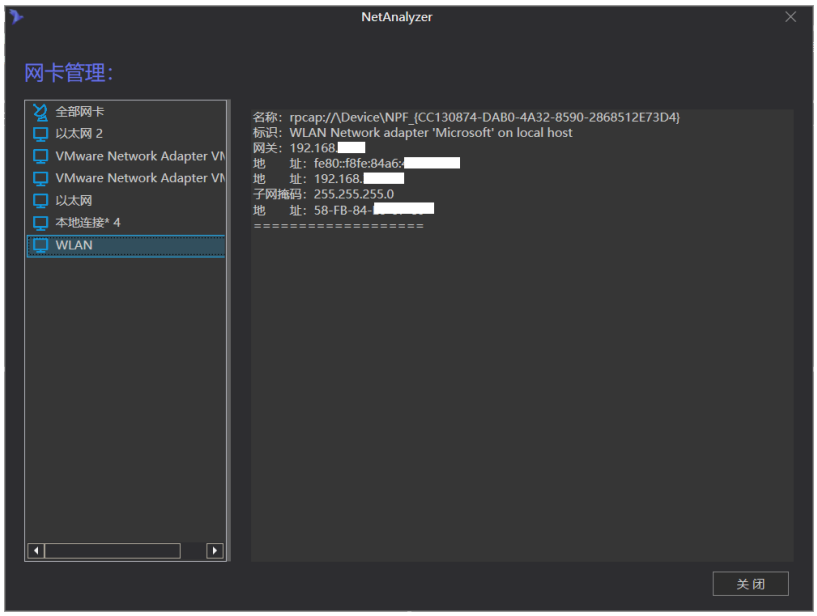
通过字符串导入数据包

网卡采集，点击【网卡】下拉列表，即可列出当前系统已启用的网卡(包含虚拟网卡)，在新版的 NetAnalyzer 中已经显示网卡的 IP 地址。

网卡列表

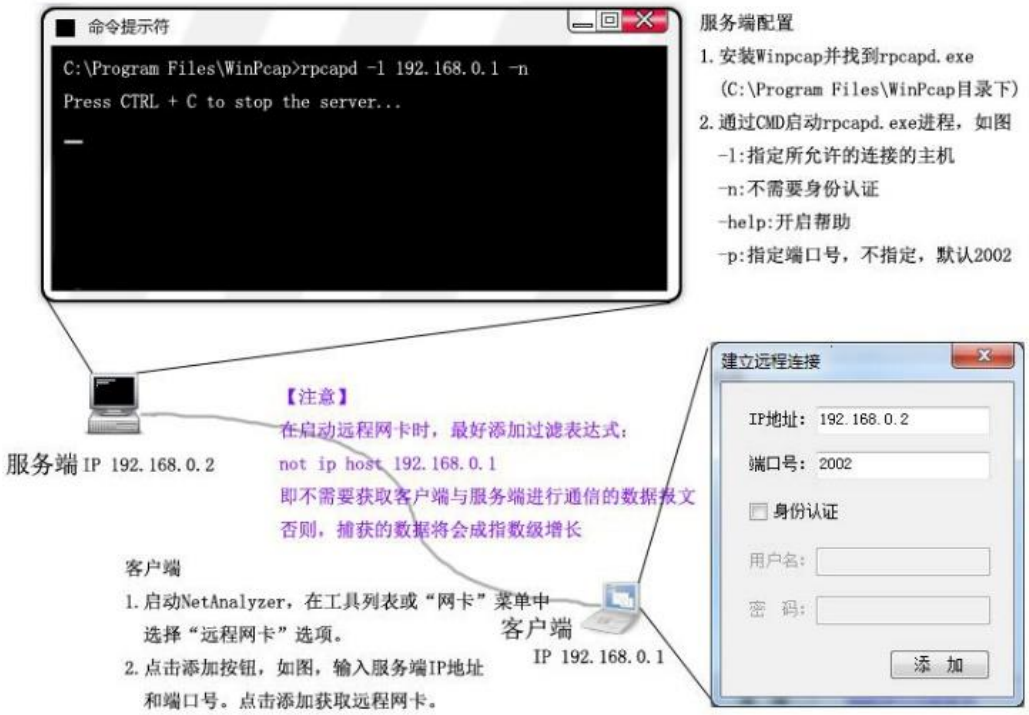
第一个为监听全部网卡选项，顾名思义，选该选项的是后，开启对后面多个网卡的共同监听。当我们选择了一个网卡，接下来就是对这个网卡进行操作，如开启抓包，停止抓包等等。在这里我们只看到了网卡的系统给定的名称，有时候我们想要看看这块网卡具体的的配置信息，如 IP 地址、MAC 地址之类的信息是，只需要点击 

便可以看到我们需要的内容。



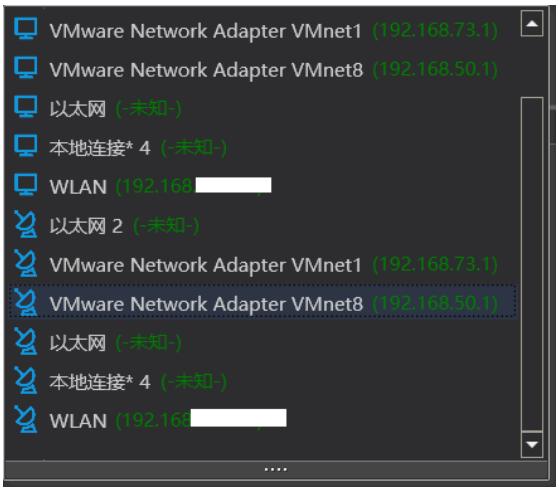
网卡信息

我们从这里只可以看到本机的一些网卡，但是有时候我们需要看看远程电脑的网卡那该怎么操作呢，首先在远程电脑上安装 Winpcap，然后运行 C:\Program Files (x86)\Winpcap\rpcapd.exe，具体使用方法如下：



配置 winpcap 远程监控方式

在 NetAnalyzer 中添加远程网卡

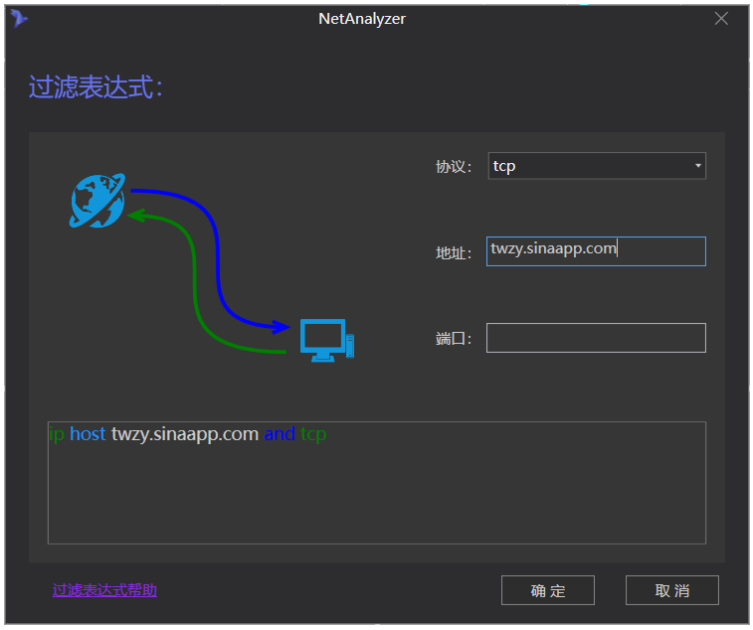


添加了远程网卡的网卡列表

具体操作步骤如下，最后点击确定按钮，完成对远程网卡的添加（本次模拟添加当前系统的网卡）

过滤表达式，对于过滤表达式，依托于 Winpcap，只要使用符合 Winpcap 内核的过滤表达即可，在此处不会过多的讨论表达式的编写，这里会介绍一些基本的使用方法和一些抓包的特殊技巧。点击 **开始** 标签页 

进行过滤表达式配置，当然也可以直接在过滤配置前面的输入框中设置过滤表达式。



配置过滤表达式

点击过滤表达式帮助可以看到一些简单的过滤表达式说明，如果想要完整的表达式说明，可以参考通过 winpcap 官网获取。

通过表达式示例

可以通过**主机**输入 IP 或域名地址，然后在**端口**输入端口地址，配置器会自动生成简单的过滤表达。同时，配置窗口还提供了一些示例，可以通过**示例与记录**查看。设置结果如图

配置完成的过滤表达式

接下来就是过滤表达式的一些使用技巧，该技巧同样适用于 Wireshark 软件

接下来就是开始进行通过，采集数据了，在这里网卡是数据源，过滤表达式为筛选条件，有了这两个准备我们就可以进行数据抓包了。

首先选择网卡：我们选择**以太网**，因为我现在使用的是网线连接，所以此处选在以太网作为我们将要监控的网卡。

接下来就是设置过滤表达式：我们设置了 port http 所以接下来我们只抓 http 协议的数据包，然后点击开始，就可以抓包了

配置了过滤表达式为：port http 的采集信息

状态栏会实时显示抓包信息

抓包状态

在抓包过程中我们可以随时暂停或停止抓包，当然也可以重新启动重新开始抓包。

开始抓包

暂停抓包：当点击**暂停**仅仅停止网卡监控，NetAnalyzer 中的分析线程等还在继续工作，如果点击开始，软件不会进行初始化操作，而是在此基础上继续进行抓包。

停止抓包：当点击**停止**时软件会完全停止所有的操作，包括数据采集和数据分析，并且取消界面中的功能限制。当在此点击开始时，会销毁当前采集到的数据。

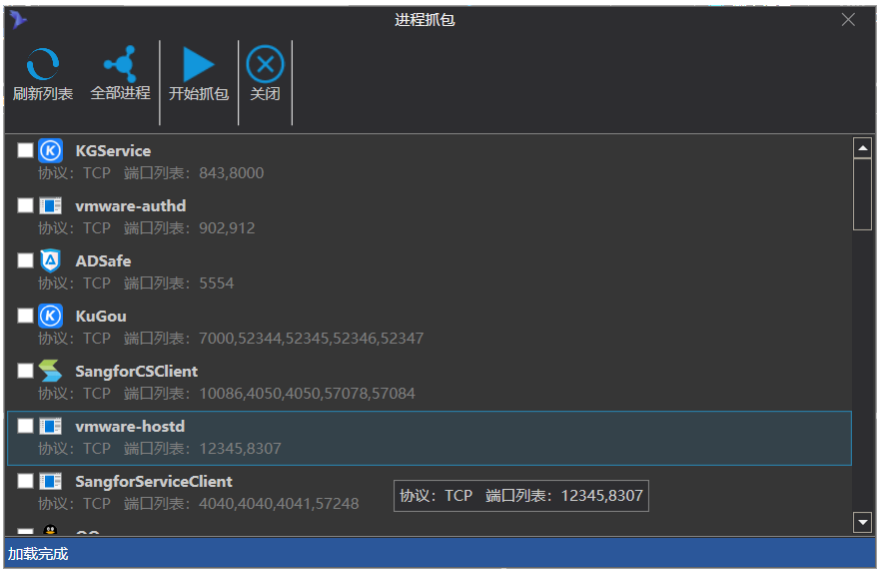
重新启动：当点击**重新启动**时，软件会自动终止当前的抓包分析等操作并且销毁当前抓到的数据，然后自动启动重新进行抓包。

（*注 在销毁抓取到的数据的时候会有是否保存的提示）

除了刚才常规的抓包方式，NetAnalyzer 还提供了基于进程的抓包方式。对基于进程的抓包本质上来说其实就是自动生成过滤表达式抓包，获取系统打开端口的所有进程，并获取进程所开启的端口、IP 地址以及使用的协议等信息，生成对应的过滤表达式，然后给 NetAnalyzer

设置该过滤表达式，最后开始抓包。

点击**开始**标签页中的 



进程抓包

勾选需要监控的进程，点击开始抓包，进行基于进程的抓包操作。

通过上面三种数据获取方式，我们就可以在 NetAnalyzer 中得到需要网络数据了，当然在使用过程中也会有一些小的使用技巧，在这里总结一下。

技巧 一 抓取环回地址(127.0.0.1)的数据包

通过 route add 添加本地 IP 地址跳转，是数据经过指定的网管然后再传输到本机，通过 route delete 移除跳转，以减少不必要的跳转，影响系统网络效率

示例：192.168.1.110 为本机 IP 地址 192.168.1.1 为网管地址

子网掩码视情况而定，若不清楚具体的 iP 地址，可以在 DOS 中 通过 ipconfig 查看

代码如下：

route add 192.168.1.110 mask 255.255.255.255 192.168.1.1 metric 1

route delete 192.168.1.110 mask 255.255.255.255 192.168.1.1 metric 1

技巧 二 抓取 ASDL 数据包

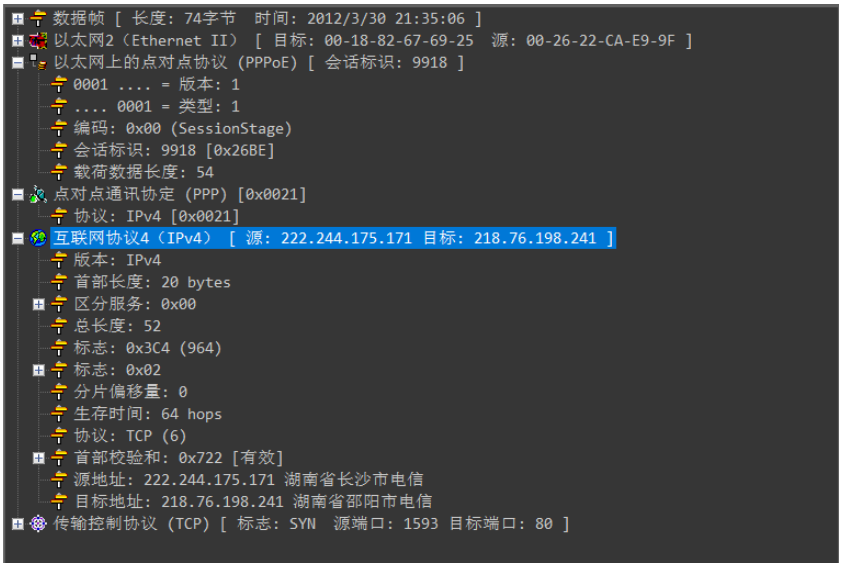
在一些个别地方还在使用拨号上网（ASDL）,我们在抓包时，设置了 TCP 或 UDP 端口的过滤

表达式，往往不起作用，事实上，因为拨号上网在 IP 层上封装了 PPP 协议，然后再通过 PPPoE 封装 PPP 协议，如下图所示

对于该种，协议 Winpcap 所使用的过滤表达式会与一般的方式不同，对于这部分抓包需要使用 pppoes and (XXX) 方式

示例：

pppoes and (ip host 192.168.0.1 and tcp port 80)



协议分析数据解析

技巧 三 抓取一段端口

有时候需要监控一段端口比如要监控目标地址的 8000~80099 端口之间的所有 Tcp 数据包，那么设置如下表达式如下表达式：

tcp dst portrange 8000-8009

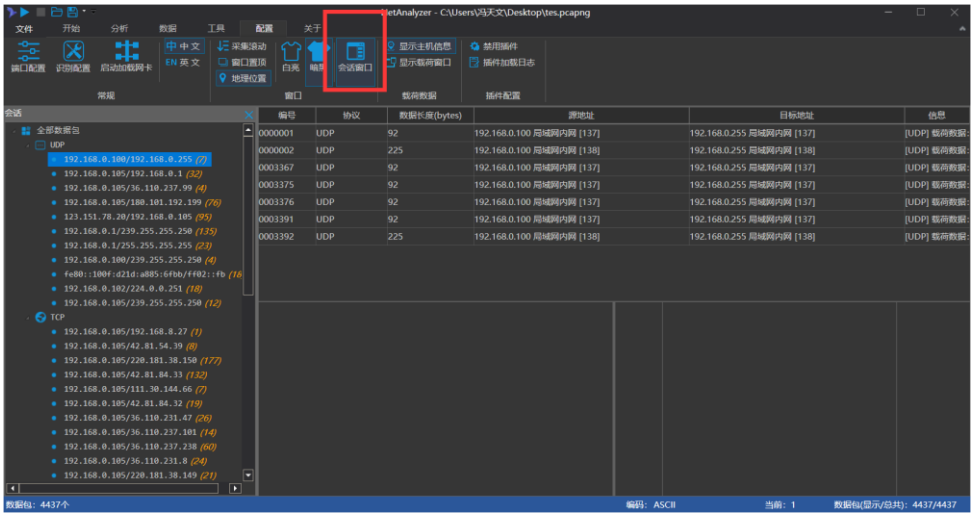
2.3.数据选择

NetAnalyzer 提供了多种筛序方式，包含了会话筛选，自定义筛选、会话挑选、隐藏、查找、标记等功能。

会话定义：相互做网络数据交换一次通信叫做会话，会话标志为该组 IP 地址与端口号(对于非传输协议使用 ip 地址或 mac 地址)

会话功能，通过配置菜单找的会话窗口就可以打开会话窗口功能，**需要注意的是，为了 NetAnalyzer 的数据包处理性能考虑，只有在打开会话窗口后，载入的数据才会进行会话分组**，

通过点击会话中的数据条目，可以快速选择当前的会话数据。

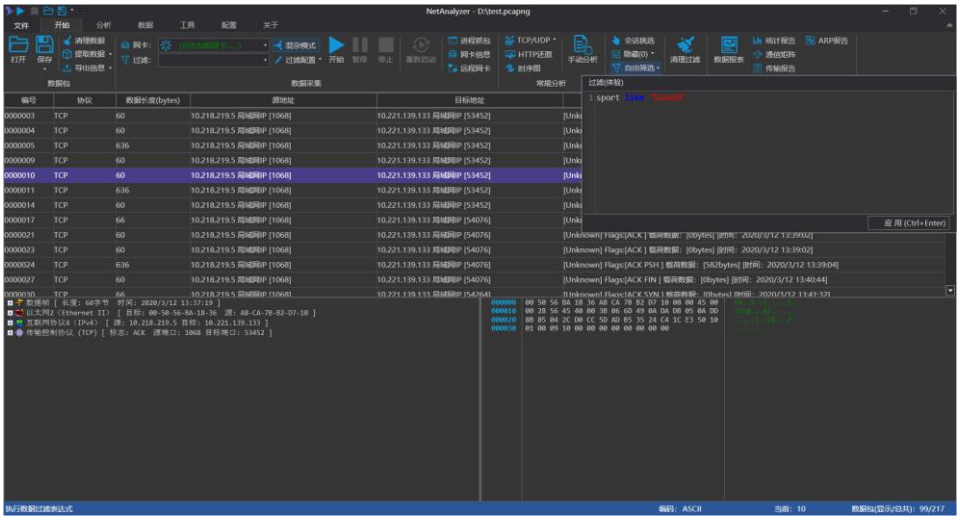


会话筛序窗口

筛选功能，NetAnalyzer 中使用 C#绑定的方式来处理数据包列表的呈现，所以提供了类似 RowFilter 的筛选功能。通过使用 RowFilter 表达式完成对数据包的精确查找。

字段	说明	示例
id	界面：编号	id>10
pro	界面：协议	pro='tcp'
len	界面：数据长度(bytes)	len> 200
src	界面：源地址	src like '%10%'
dst	界面：目标地址	dst like '%10%'
info	界面：信息	info like '%时间%'
sip	IP 地址	sip = '192.168.0.1'
dip	IP 地址	dip = '192.168.0.1'
sport	端口号	sport='80'
dport	端口号	dport='80'
prlst	协议,包含数据包中全部的协议，使用逗号分隔	prlst like '%arp%'

筛选字段表

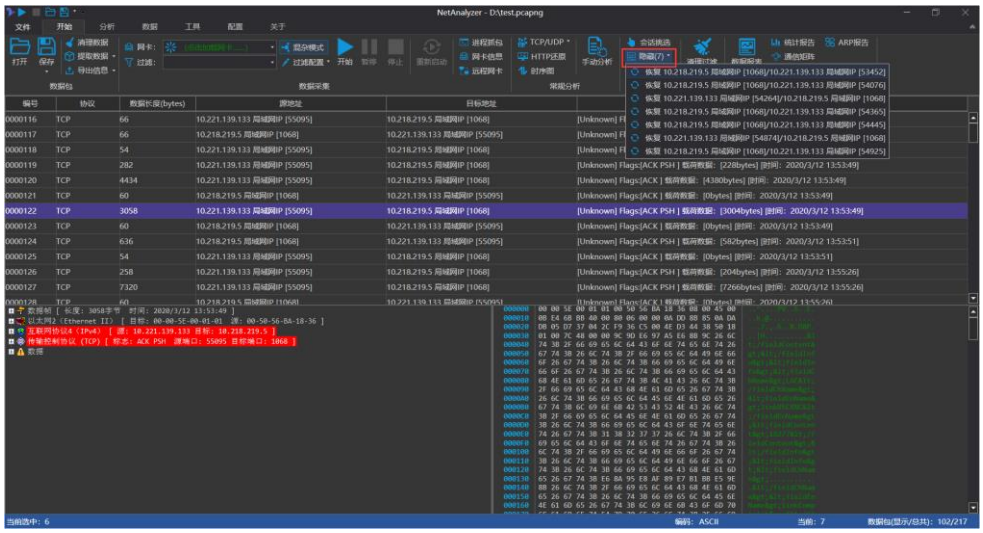


数据筛选功能

会话挑选 为了便于快速找到与当前点选的会话相关联的数据，而隐藏掉其他暂时不关心的数据此时就需要使用到会话挑选功能，使用非常方便，选中要进行挑选的行，然后点击

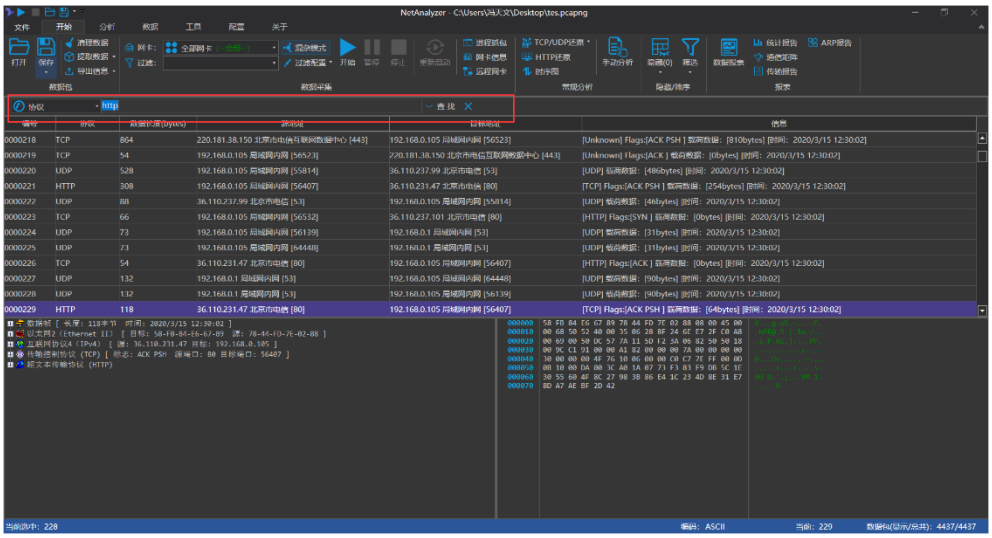
NetAnalyzer 就会只呈现当前会话的数据列表。

隐藏，对于无用的数据可以选择临时屏蔽，此时就可以使用到隐藏功能，选中数据包，点击隐藏，则可以隐藏当前会话，并在隐藏列表中显示该会话标志。



隐藏功能

查找 对于指定的数据包查询、NetAnalyzer 提供了查询功能，通过点击查找或是使用 Ctrl+F 快捷键，就可以打开查找功能。该功能提供了编号、地址、关键字、字节数组等多种查询方式。



查找功能

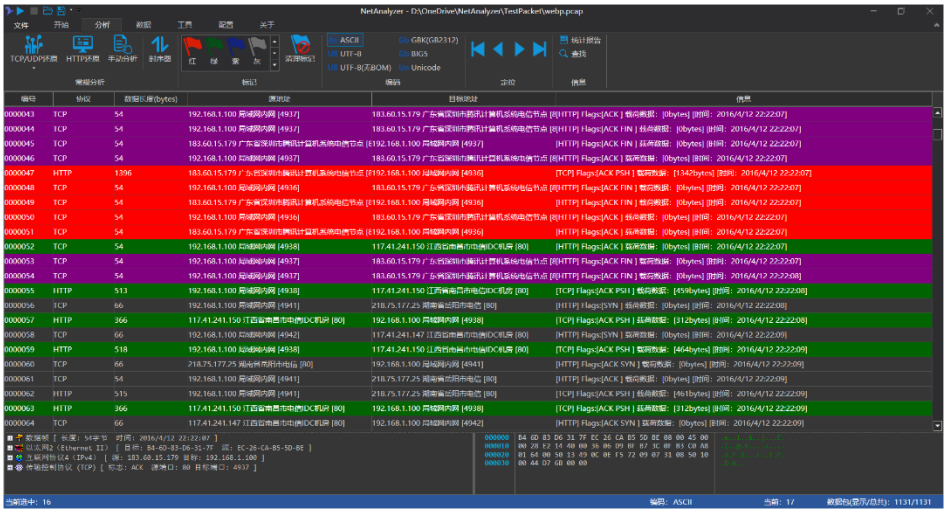
标记 为了实现多组会话数据的快速识别，NetAnalyzer 添加了标记功能，该功能会对指定的会话进行颜色标记处理，选中一行数据，在分析标签下面，有**标记**功能，实现对当前采集会话数据连接的进行快速识别。

执行标记

NetAnalyzer 提供了四中颜色对数据包链接进行区分。

如 TCP 数据包，就会通过源 IP 地址+源端口地址+目标 IP 地址+目标端口 作为一个特征来进行识别，此处的源和目标具有相对性。

注* Shift+鼠标左键 可以实现对数据会话的快速标记 颜色为红色



执行标记后的显示效果

通过点击清理标记，可还原数据。

至此对于数据抓包和筛序就讲述完毕了，下一节将对单个数据包进行分析。

2.4.数据分析

完成了数据的抓取，那么接下来就是 NetAnalyzer 的第二个重点部分了，协议分析作为整个软件的核心之一，在最新的 NetAnalyzer 中已经得到了巨大的提升。NetAnalyzer 中协议分析分为单数据包分析，和联合分析两种分析方式，对于联合分析会根据不同的协议特性进行形成不同的分析方案，目前支持传输协议（TCP/UDP）协议分析， HTTP 协议分析。在数据统计部分部分还增加了针对 ARP 协议的图形化分析。对于协议分析，需要了解相关的网络知识或是有相关专业背景支持。

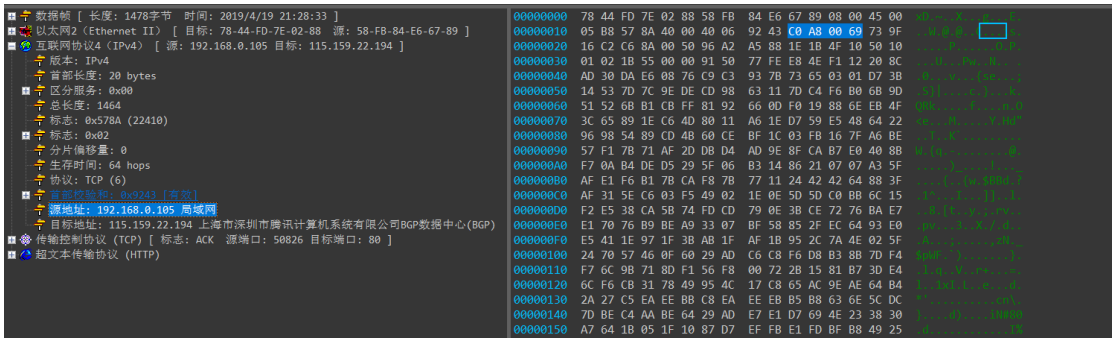
单数据包分析，在获取到数据包后，软件工作界面数据包列表框中会显示所获取的所用数据包，并且对这次数据做了一些简单的分析，我们可以凭借这些数据简单判断所对应的的数据包类型。

编 号	协 议	数据长度(byte)	源地址	目的地址	信息
0000001	TCP	54	115.159.22.194 上海市深圳市腾讯...	192.168.0.105 局域网 [50826]	[HTTP] Flags:[ACK] 载荷数据：[0bytes] [时间：2019/4/19 21:28:33]
0000002	HTTP	1478	192.168.0.105 局域网 [50826]	115.159.22.194 上海市深圳市腾讯...	[TCP] Flags:[ACK] 载荷数据：[1424bytes] [时间：2019/4/19 21:28:33]
0000003	HTTP	1478	192.168.0.105 局域网 [50826]	115.159.22.194 上海市深圳市腾讯...	[TCP] Flags:[ACK] 载荷数据：[1424bytes] [时间：2019/4/19 21:28:33]
0000004	TCP	54	115.159.22.194 上海市深圳市腾讯...	192.168.0.105 局域网 [50826]	[HTTP] Flags:[ACK] 载荷数据：[0bytes] [时间：2019/4/19 21:28:33]

数据包列表

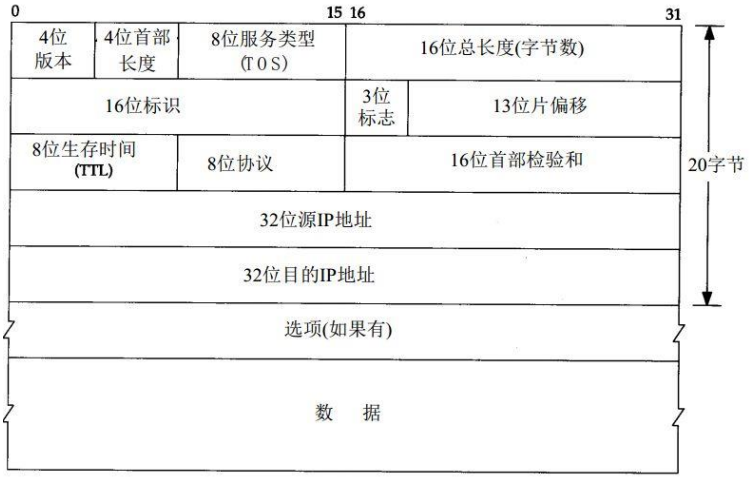
当我们选中一行，即选中一个数据包，我们可以看到对该数据包详细的数据分析信息，并一树状结构树呈现出来，并在右侧显示该数据包原始信息。当我们选中协议树中一个字段时，

右侧的数据就会定位到当前字端所分析数据的位置。



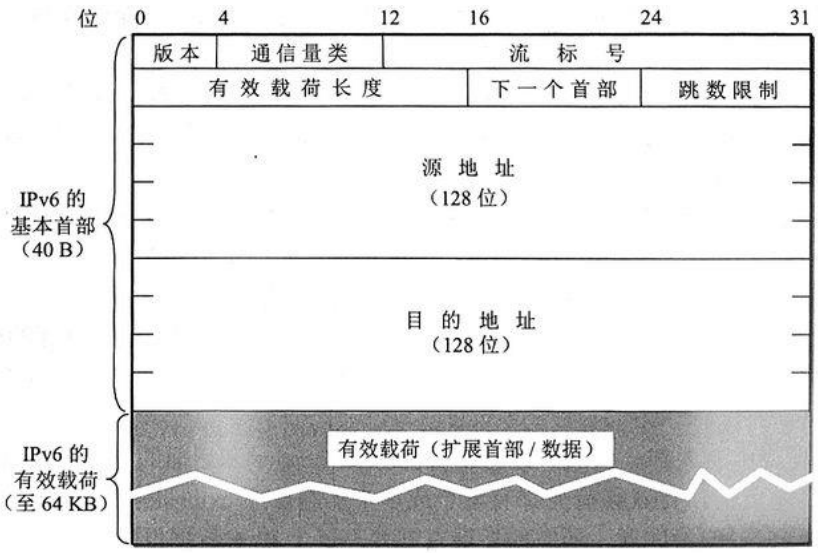
数据分析

然后通过对应的协议格式进行匹配与分析，如这部分的 IP 协议。



IPv4 协议格式

需要注意的是，NetAnalyzer 目前对于选中的字段只能精确到字节层次，对于一些协议，其中一个字节可能包含了多个字段，或是跨字节的字段，则会选择全部的字节数据，比如 IPv6 协议。

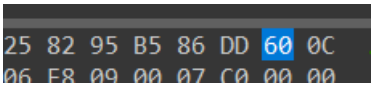


IPv6 协议格式

其中的版本字段只占用了 4bit（1 字节为 8bit），通信类型占了 8bit 也就是 1 字节，但是因为其中前面部分使用了版本字段所在字节后面的 4bit，所以改字段为一个典型的跨字节字段，同样流标签字段使用了 20bit，占用第二个字节的 4bit 加上后面自身的 2 个字节(16bit)。

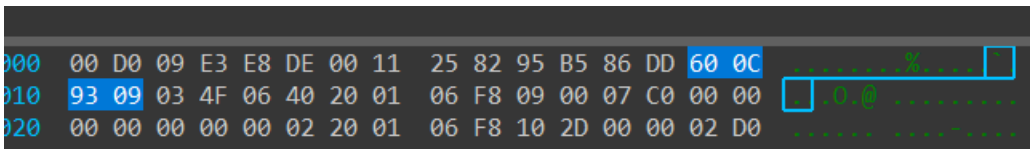
解析后的 IPv6 数据

对于类型的字段因为 NetAnalyzer 使用十六进制显示数据，并不能清晰表达 bit 层次的信息所以当选定字段后默认选中改字段所在的字节，如点击版本选中方式如下，



IPv6 版本信息

选中通信类型和流标签则呈现方式如下。



通信类型和流标签共用数据

数据分析标签

虽然 NetAnalyzer 尽可能多分析每个数据包所包含的信息，但是依旧存在很多数据需要我们手动去解析。所以软件增加了数据标签。

数据分析

在**数据**标签页点击 **显示** 按钮 就可以打开数据转换窗口，当然也可以在常规转换中点击任意功能可以打开转换窗口

转换窗口管理

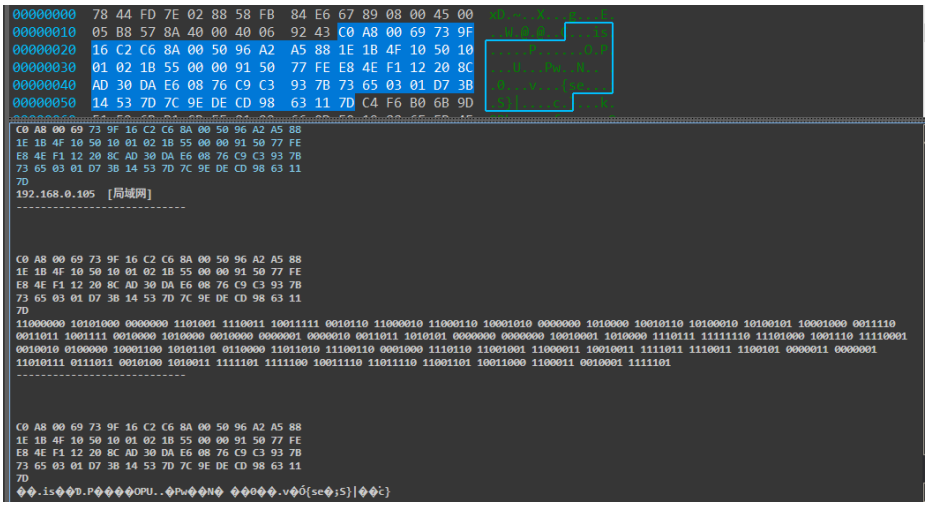
关闭按钮为关闭转换窗口，清空则是清空当前窗口内的数据。

点击清空按钮，则清空转换信息。

常规转换窗口

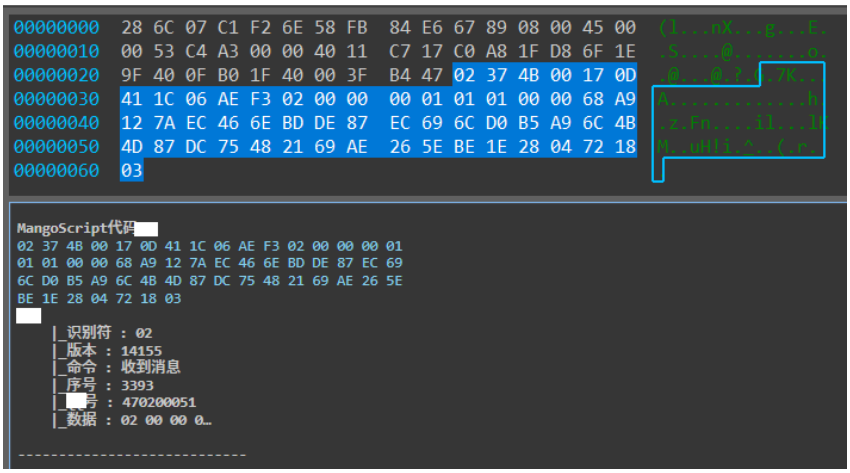
NetAnalyzer 中提供了一部分简单的转换功能，这些功能只有在载荷数据被选中的情况的才可以启用，

如点击二进制按钮，则对所选的数据转换为对应的二进制字符串。如下图所示。



常规数据转换窗口

除了一些简单的转换功能，还集成了 MangoScript 扩展方式和插件扩展方式(无可用插件的时候不显示)的转换。



扩展 MangoScript 的解析

如下面通过 MangoScript 针对某即时通信软件的数据分析。

针对于 MangoScript 和插件两种方式的转换，将会在在《NetAnalyzer 使用说明书 二 扩展与开发》中详细说明，此处不再赘述。

定位转换功能需要配合常规转换进行使用，有时候我们确定某个字节会在一个确定的位置出现，比如 IP 地址字段，我们选中该位置，位置字段就会出现一串代码 (10,1) [26]-4

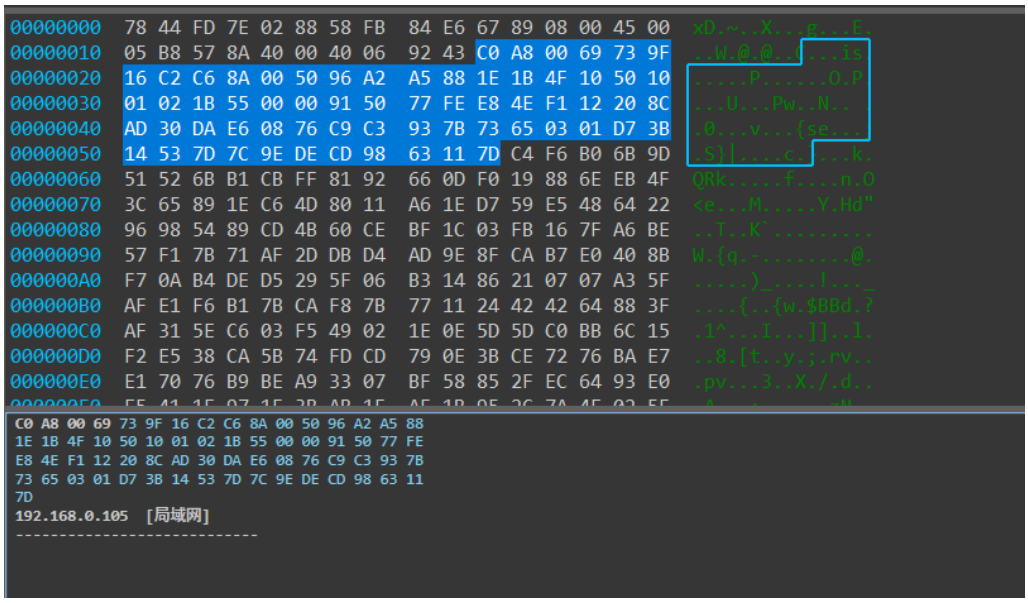
(x,y) [offset] – length

x: 十六进制编辑器水平方向的偏移量

y： 十六进制编辑器垂直方向的偏移量

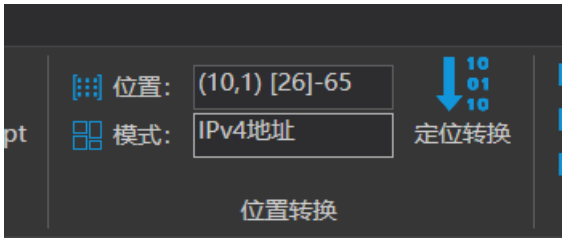
offset ： 字节偏移量，offset = y * 16 + x

length： 当前选择的数据长度

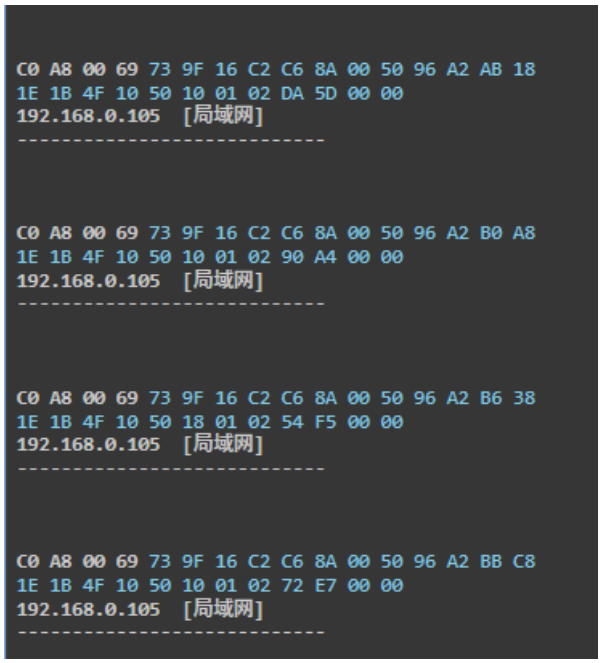


数据转换

所以代码 (10,1) [26]-4 确定了当前 IP 地址的位置，此时点击 **常规转换** -> **IPv4 地址** 则会在**模式**中记录当前的转换模式，然后点击**定位转换**，就会在当前数据包列表中针对每个数据包这个位置执行定位操作，这对于寻找所需要的数据非常重要。



选择了 IPv4 转换



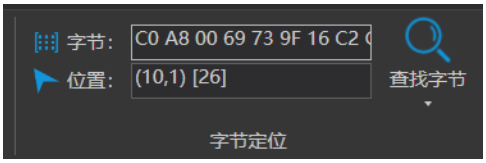
执行定位转换

对于 MangoScript 和插件扩展依然支持定位转换。

区块复制，主要是对一些已经选中的字节进行复制转为代码，字节数组，以及保存的功能，以及数据做手动分析，脚本分析以及自定义转换等，后续将会说明，此处不再详细介绍。

数据块操作

字节定位，与定位转换类似，但是字节定位主要是用来在数据包列表中查找相同位置出现相同字节序列的数据包。算作一个查找功能。



字节定位

分析标签

分析标签下个功能依托于数据包列表，分别有载荷数据提取，数据包标记，编码转换，数

据查找，统计等相关功能，是联合分析的主要功能，下面将会着重对一下功能进行说明。

数据分析标签

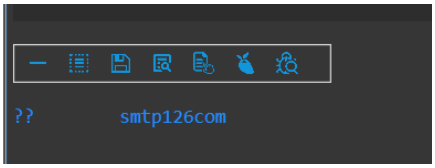
TCP/UDP 协议分析　前面介绍的都是基于单包的数据分析，而在协议分析中，我们大部分分析的数据都是依托于 TCP/UDP 的长连接数据，这部分数据的特点就是有多个数据包通过 tcp 或 udp 相关协议完成数据重组后才可以使用（基于 udp 的连接数据可能不是很严格）。

NetAnalyzer 除了提供基于单包的数据分析，更提供了基于连接数据的分析，而分析出来的数据不仅仅是在窗口上呈现一堆乱码，更可以通过 DocBar 将获取的数据提取出来进行使用。

在**开始** 标签最后一部分就是基于长连接的分析。点击 **TCP/UDP** 按钮

基于 TCP/UDP 载荷数据查看

此时 NetAnalyzer 便会切换到**载荷数据模式**(该过程可以通过配置，使用独立窗口打开)。在该模式下会打开专有的载荷数据菜单，数据区域也会变为对于载荷数据的分析，这里先介绍一个 NetAnalyzer 中的 DocBar 工具，如下图



DocBar

在文本模式下，分析载荷数据会显示该工具条，该工具条会提供针对当前数据块的各种操作，当然在不动情况下，显示的工具和数量，都有所不同，下面是对当前各个功能的说明。

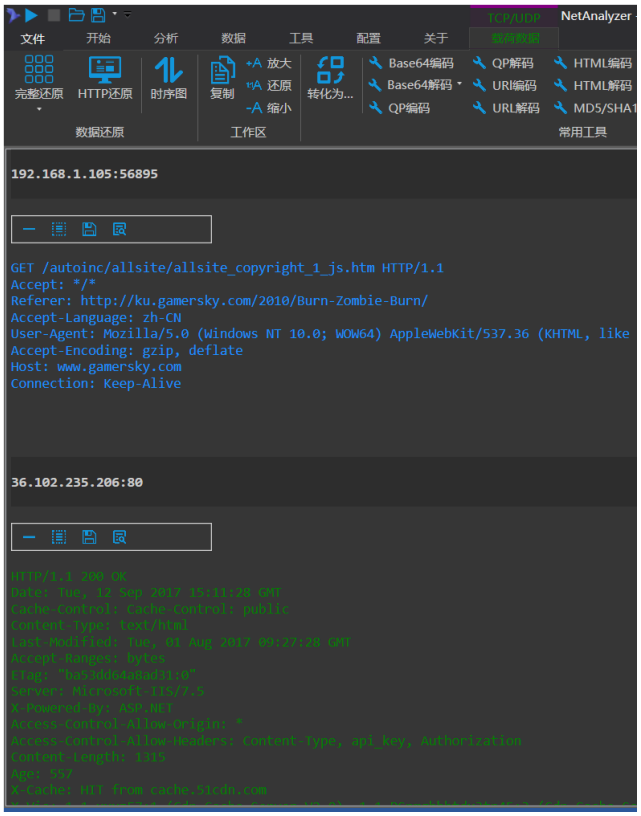
- 对当前数据块进行折叠
- 选中当前的分析数据
- 保存当前原始数据

- 查看原始数据(bytes 数据)
- MangScript 解析数据
- 手动测试数据

对于其他情况下的工具在这里不会一一介绍，但是碰到的时候会有说明，并且随着后续功能点的增加，DocBar 可能会有更多的功能添加进来。

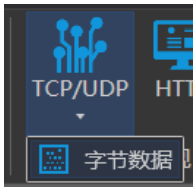
tcp/udp 的分析分为 **文本模式**和 **原始模式** ，文本模式主要是用于分析载荷数据为文本的数据，我们可以通过下面两种方式更改文本编码方式，分析数据。

文本模式下，呈现方式如下:

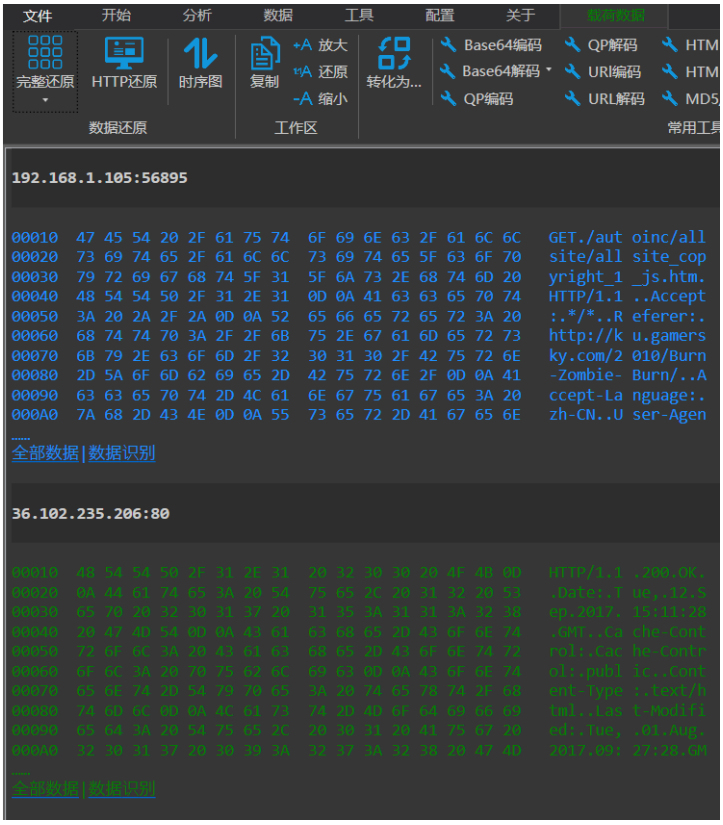


查看载荷数据

原始模式分析如下，可用通过 **TCP/UDP** 的下拉菜单命令 字节数据 切换为原始数据



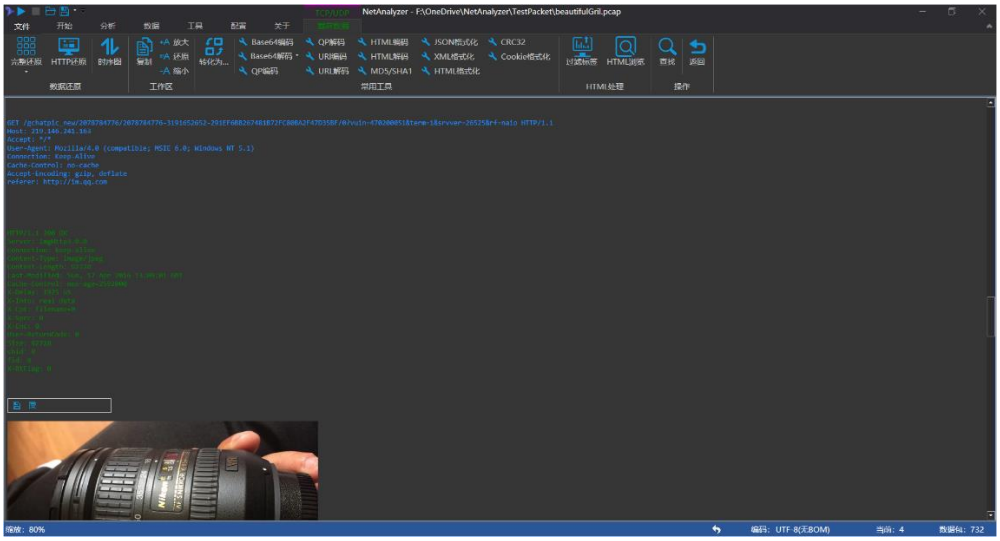
字节查询方式



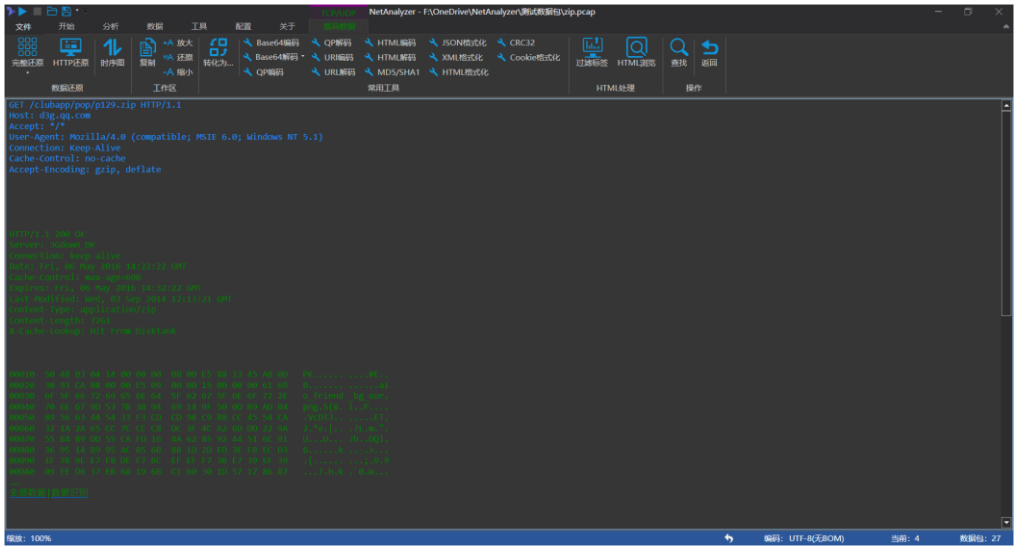
字节方式呈现

对于在该功能下针对 TCP 的所有数据都已经进行过 TCP 重组，所以最终分析完成的数据并不是按照数据包方式做简单呈现就可以的，都会做数据的筛查与整理。如果需要单包分析的使用者需要注意一下。

HTTP 数据分析 http 作为最有网络代表意义的协议，NetAnalyzer 提供了更加完善的分析，http 基于 tcp 协议，所以数据还原等都建立在 tcp 数据还原的基础之上。通过 http 分析，我们可以还原很多有意义的数据，如获取到 Http 所传输的的 html、js、css 数据文件，还可以获取到基于 http 协议分析得到的图片，文件等信息，如下图分别为还原后的图片和 zip 压缩包。

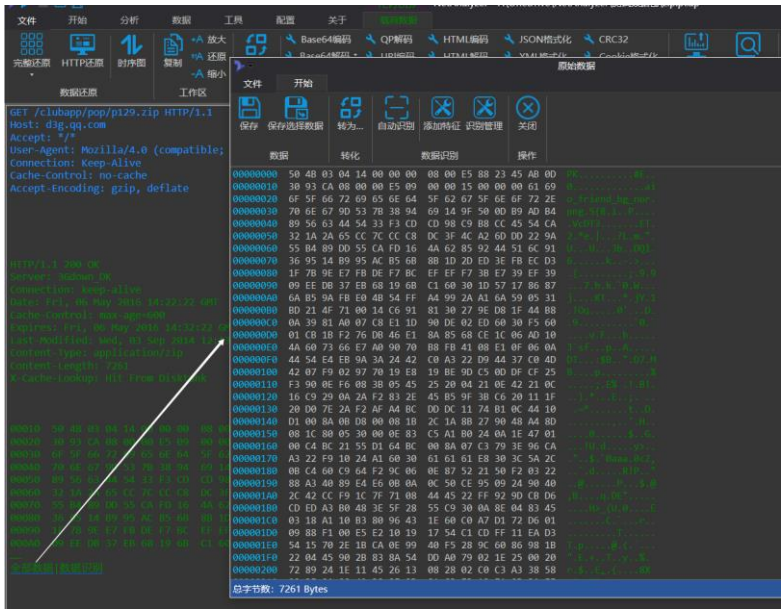


http 方式分析出的图片



http 方式分析出的文件

对于常规的字符串或图片可以直接在 NetAnalyzer 呈现，但是对于其他类型的文件，如视频、音乐、以及上面提到的 zip 压缩包文件，在在 NetAnalyzer 会简单显示为二进制数据，该数据如果过长，则会截断显示，但是在后面会加入【全部数据】下钻选项，当点击该数据后则会打开原始数据对话框，并且会完整显示当前的数据，如下图所示。



查看原始数据

原始数据对话框中，提供了简单的数据另存为和数据识别相关的功能。

原始数据保存

保存 保存当前窗口中的数据为一个文件。

保存选择数据 是当选择对话框中其中的一段数据保存为文件，有时候数据可能存在偏差，或者我们需要提取选定的数据保存为文件，可以通过下拉保存选定的数据进行保存。

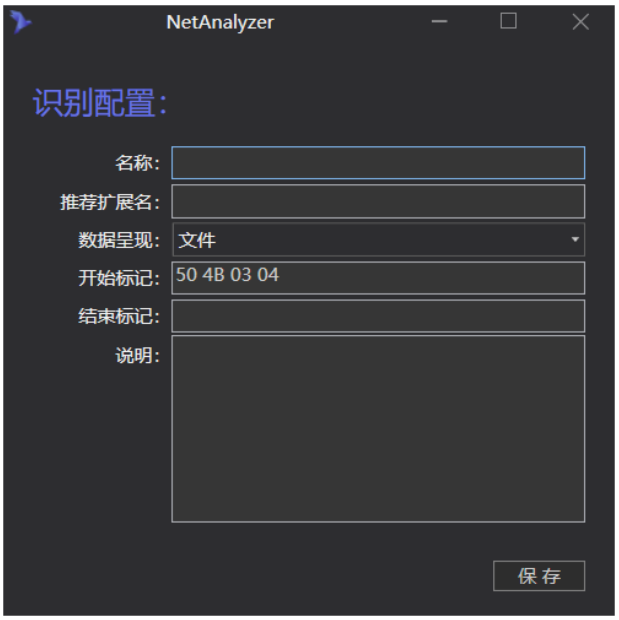
数据识别功能。

转为… 则是将当前的数据转到编码转换工具中进行进一步分析。

自动识别 为了更加快速的实现数据提取，NetAnalyzer 增加了数据识别模块，通过整理不同文件的头部或尾部字节形成数据识别特征，当进行自动识别的时候，可以快速定位字节。

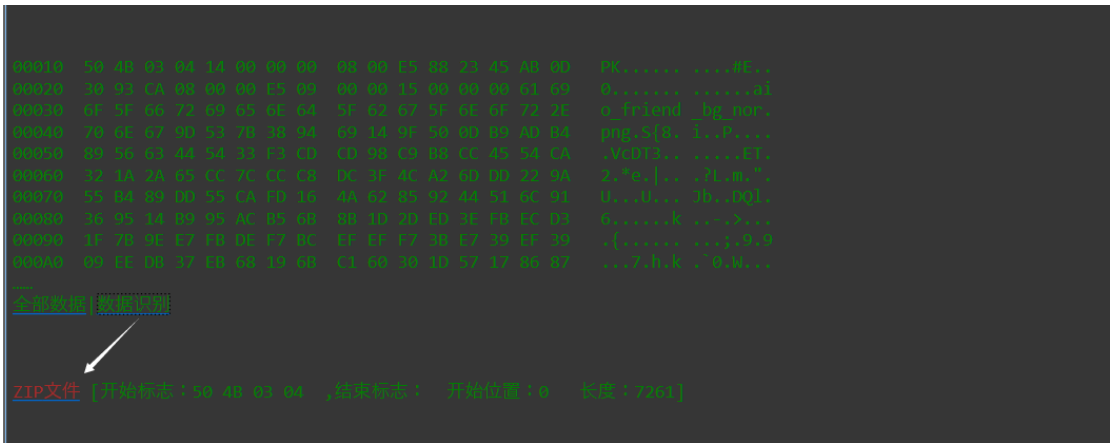
文件识别

添加特征 将选定的指定字节添加为文件识别头，并且添加相关信息，形成一个特征。



添加文件识别

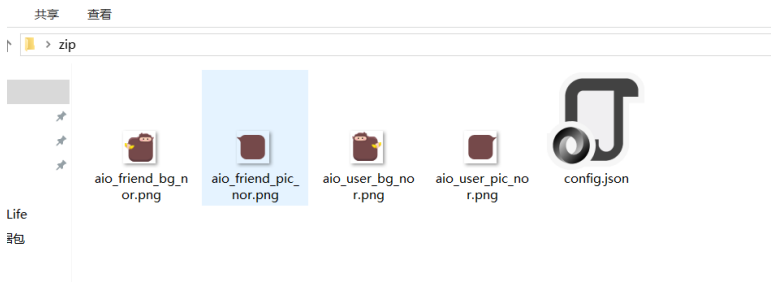
识别管理 管理特征库，在后续将详细介绍该功能点。



载荷数据分析出的文件

除了使用常规的识别方式，在载荷数据提取中也加入了数据识别功能。在使用的时候点击数据识别就可以在下方显示被识别到的数据类型，有时候可能会存在多个类型和误识别的情况，使用的时候请务必注意。

有时候通过 HTTP 协议还原部分二进制数据，如下面还原 ZIP 文件，文档会以二进制数据呈现，而我们可以通过 **0x50 0x4B(PK)**推断出该文件很有可能是 zip 文件 ，所以我们点击**全部数据** ，打开原始数据窗口，这部分数据正好是 zip 的全部数据。



保存的 zip 文件内容

此时点击将当前数据保存为 zip 文件。减压就可以看到对应的文件内容。

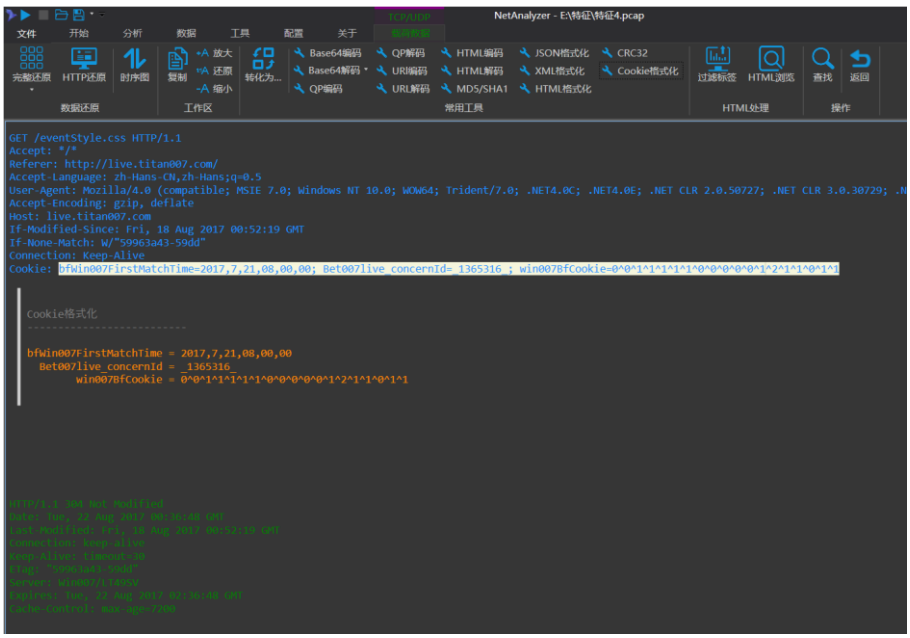
在载荷数据模式下，菜单会自动切换为，载荷模式菜单

载荷数据标签

该菜单下提供了很多常用的字符串转换工具

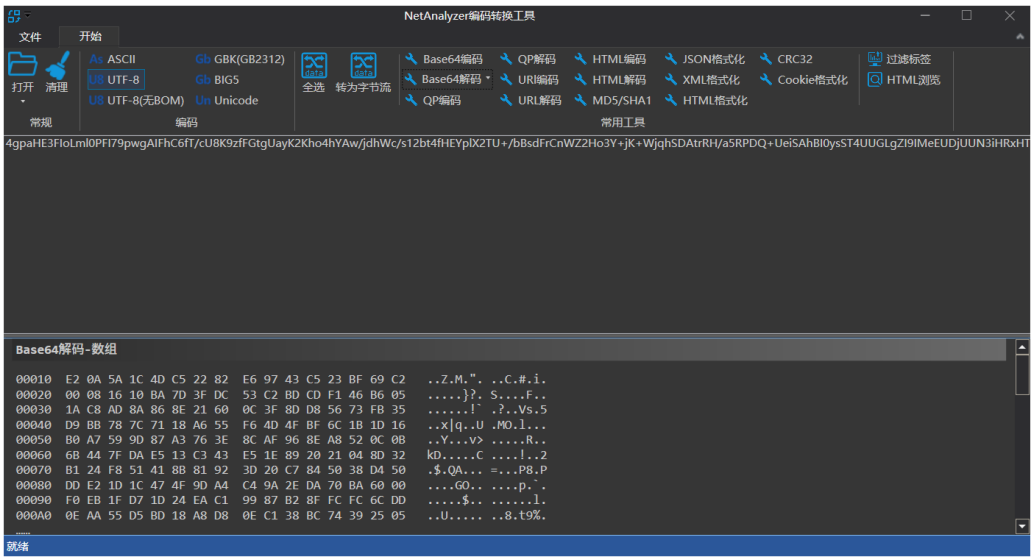
格式转换工具

如下面通过通过 Cookie 格式化，格式化了 http 头中的 cookie 字段



Cookie 格式化

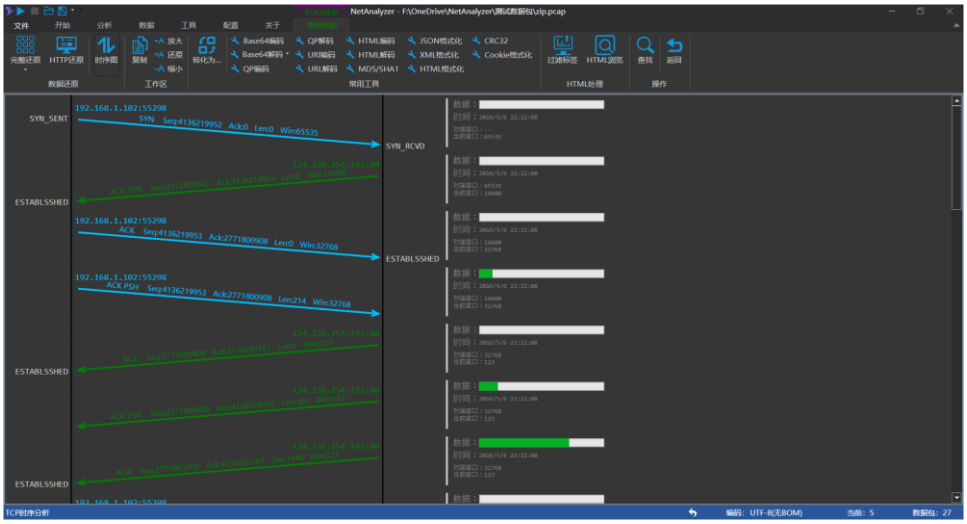
需要注意的是使用这些字段首先需要选中被转换的文本，然后点击需对应的功能项。其中如果点击**转换为…**，则启动 NetAnalyzer 附带的编码转换工具，进行集中处理。



编码转换工具

针对 html 字符串数据，还提供了过滤标签和 HTML 预览功能，因为该部分功能都很类型，且使用简单，用户自行尝试使用即可。

时序图 在数据分析中，除了对于数据本身的分析之外，有时候我们还要去评测一些数据质量等方面的内容。并且可以通过图像化的方式表现出来。



TCP 时序图分析

时序图模拟 TCP/UDP 在数据网络中的数据传输过程，还原网络通信场景，如该图可以完整的反映 TCP 三次握手以及断开连接四次挥手的情景。可以作为对当前分析数据从另外一个

方面的反馈，更具有参考意义，点击

时序图选项

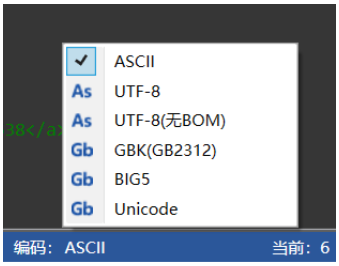
就可以看到针对于当前 tcp/udp 数据交互的情况。

编码方式

在通过 TCP/UDP 或 HTTP 功能还原数据的时候，有时候会出现乱码，尤其是对非英文字符。在 HTTP 协议中通常都会在头部信息中携带编码方法，通过提取就可以获取到编码方式，但是仍然后部分服务并不提供编码字段，这时候就需要我们通过手动切换，来尝试还原相关信息。

通过菜单栏或者是状态栏都可以对编码方案进行切换

字符编码



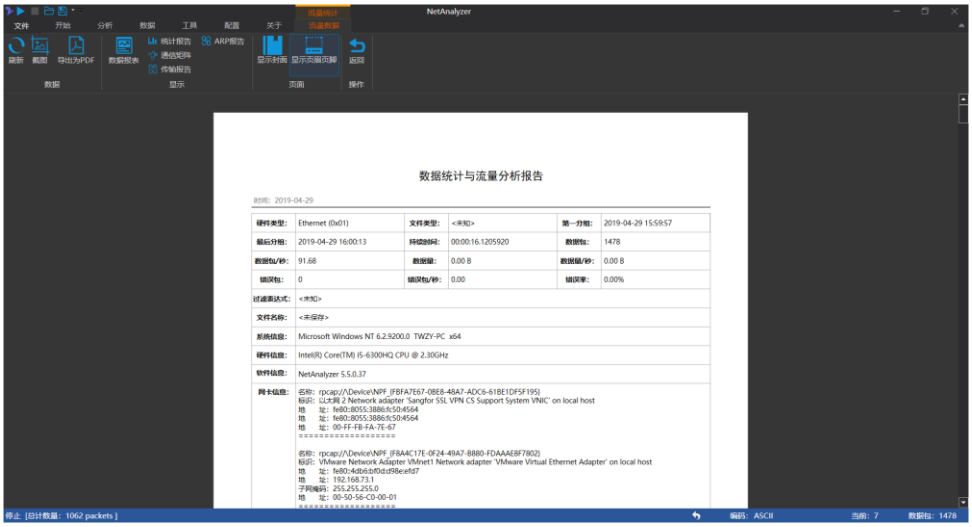
状态栏字符编码

这里需要注意的是如果 http 头部包含了编码方式，则使用头部提供的编码方式。

数据统计

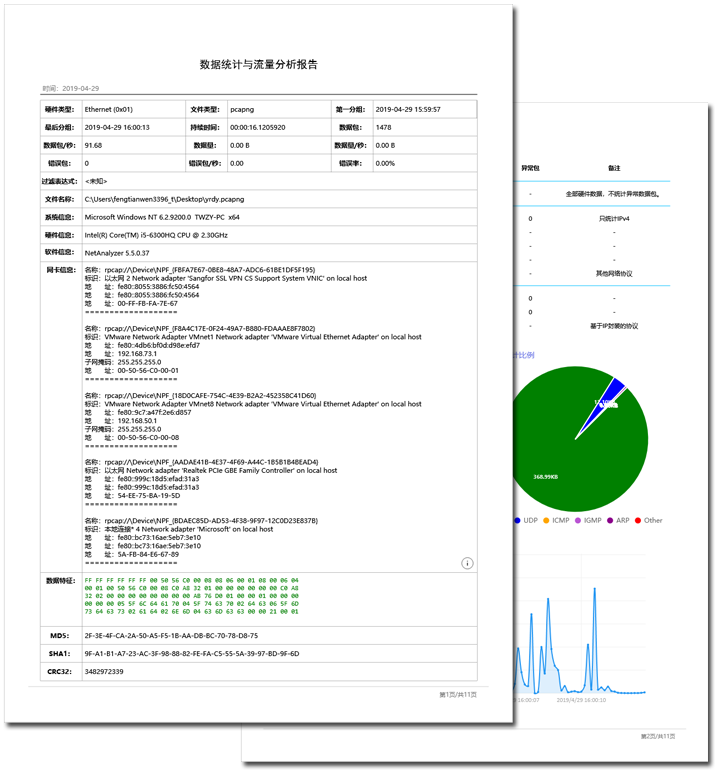
目前 NetAnalyzer 显示了大量的统计方式，涵盖了数据报表、流量分析、主机通信矩，传输报告、ARP 报告等多种统计方式。

数据报表

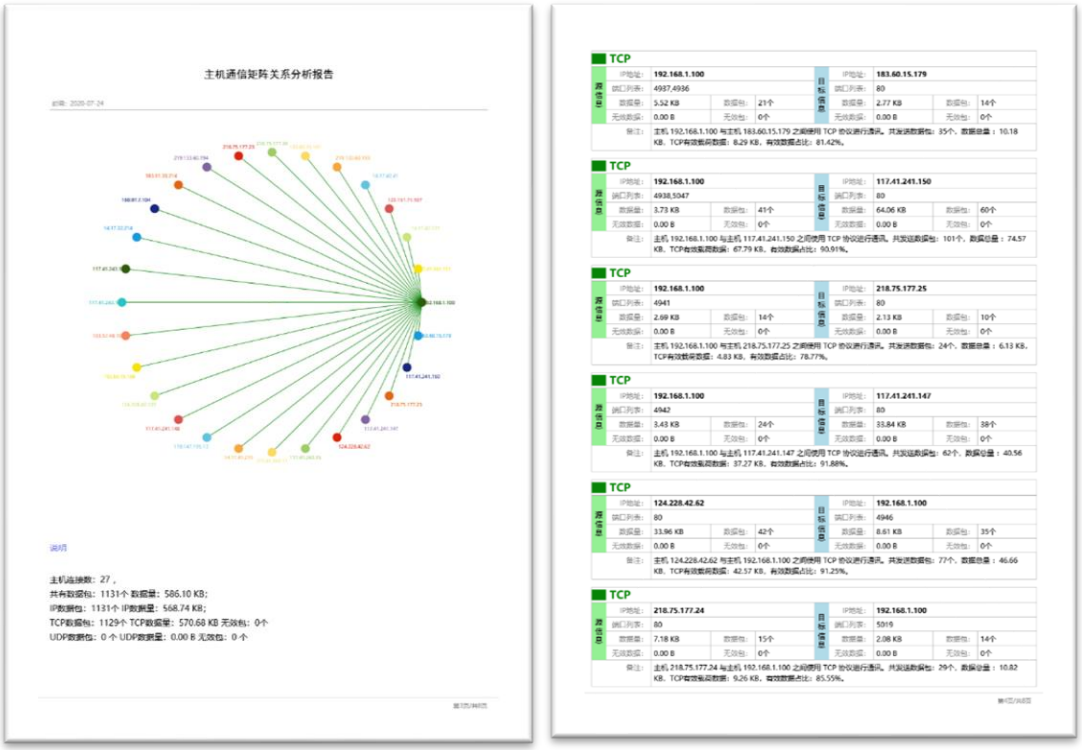


报表信息

对当前捕获的数据表中的数据进行统计与归类。呈现方式如有图所示。



报表内容



主机通信图和协议数据图

包含一些基本信息，数据量与时间直线图，数据量占比，关系图等信息。

3. 辅助功能

为了更加方便的分析与处理数据 NetAnalyzer 提供了大量的辅助分析工具，并且 NetAnalyzer 使用了插件模式，这样可以让具备开发能力的使用者可以更加合理的定制自己的个性化工具。

工具菜单

工具标签提供了很多辅助工具，和相关的扩展功能，因为版本不一样，该部分功能截图可能与实际情况有所出入。

该标签下面的工具分为四组。

应用：NetAnalyzer 系统基本工具，NetAnalyzer 未来将以工具集的方式提供，届时 NetAnalyzer、MgsExpress、DataTransfer 将分别作为独立的应用解决不同的问题。

常用工具 ： NetAnalyzer 内置的工具，

扩展：插件常规方式引入的功能模块

管理：对插件的管理功能。

下面将对这部分内容进行详细说明。

3.1.应用

NetAnalyzer 将从 6.0 以后目标定位为打造一款针对非公共协议的分析工具，借助 NetAnalyzer 数据包网络数据采集以及完善的 TCP 数据重组机制，在此上面进行自定义的协议分析，所以建立了两套方案：

- 插件项目
- MangScript 项目

插件机制在 NetAnalyzer3.0 就开始在做了，但是因为其要求使用者必须具备 C#相关的开发经验，加上没有更加明确的目标，不知道应该将重心放在对 NetAnalyzer 本身的功能性增强，

还是放在让使用者针对数据进行处理，再加上对现有 UI 扩展度有限，一直以来进展缓慢。但是我们对 NetAnalyzer 要求分析更多协议的呼声越来越高，所以建立了 MangScript 项目，目前两个项目在并行开发中，当然偏重于 MangScript 的开发。

MangoScript 的初衷就是为了快速构建协议分析方案，如一些私有协议是一个公司或一个组织定义的一套专属于内部的数据交流方案、这些协议可能因为涉密或是团体影响力过小并不能被外部人员获取到。而想要分析这些数据，借助协议分析工具进行分析是不可能的，而手动从各种二进制数据中获取信息，效率又极其低下。MangoScript 的思想就是通过将数据方案转换为对应的脚本代码，将代码绑定到 NetAnalyzer，通过 NetAnalyzer 实现与解析公共协议无差别的数据分析。

MangoScript 作为 NetAnalyzer 扩展协议分析的专职语言，设计的更像一种配置文件，可以通过不同的配置方式，实现对数据流的解析。脚本使用协议分析树的逻辑方法，脚本编辑方式就是协议树的呈现方式，即是没有接触过编程的人也可以轻松进行代码编写。

当然，因为 MangoScript 正处于测试开发阶段，所提供的功能也不近完善，这需要读者的体谅，也很希望读者可以提供一些好的建议与意见。目前脚本采取宽泛执行的方式，即对于一些语法错误会自动忽略，以保证尽可能的完成数据分析。

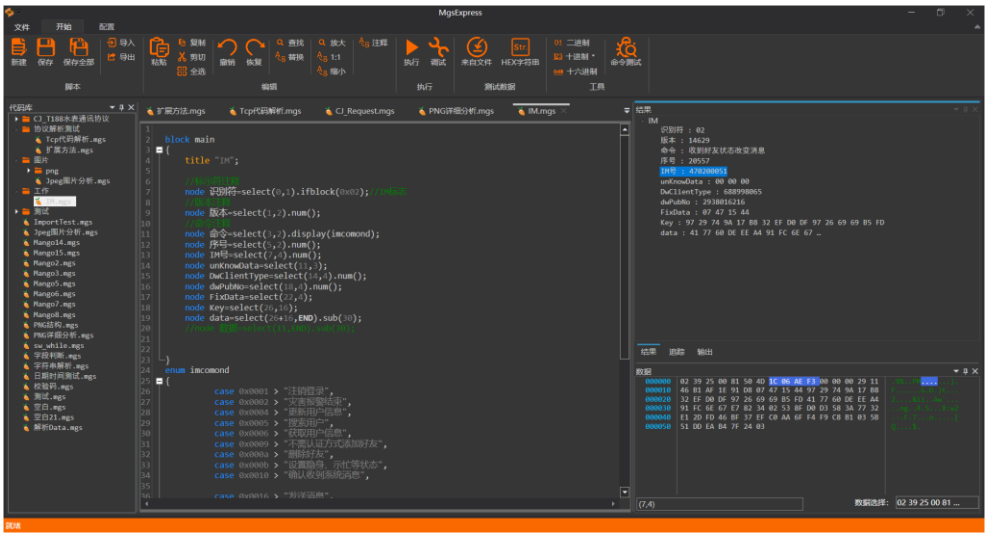
对于 MangScript 限于篇幅此处只做简单介绍和使用方式，NetAnalyzer 中某些功能点也或多或少也用到了相关的内容，在这里将会进行说明。对于具体的语法规则和使用方法请见《NetAnalyzer 使用说明书 二 扩展与开发》

目前 MangScript 可以使用两种方式执行，脚本模式和命令行模式。

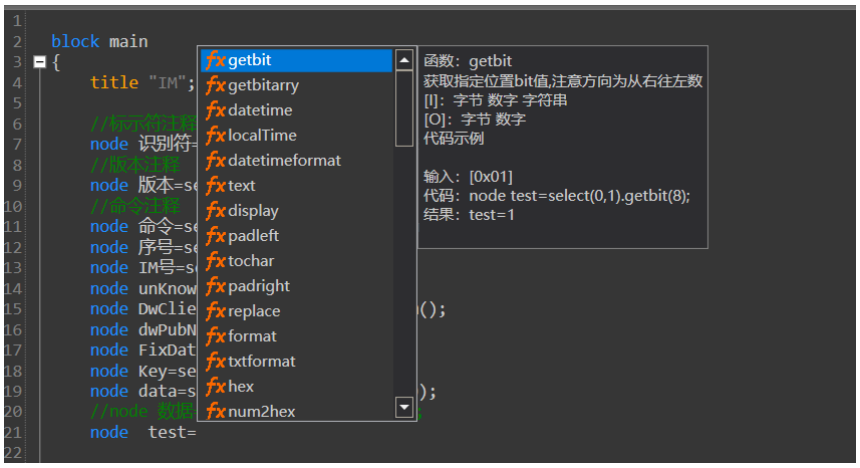
脚本模式可以将完成的脚本直接挂载到 NetAnalyzer 上面和端口绑定实现与现有协议无差别的数据分析；命令模式则可以实现灵活的数据转换与处理。

MgsExpress 就是专门为 MangScript 开发的集成开发环境，该工具包含了脚本管理、代码编辑、执行、调试、监视等各种丰富的功能。

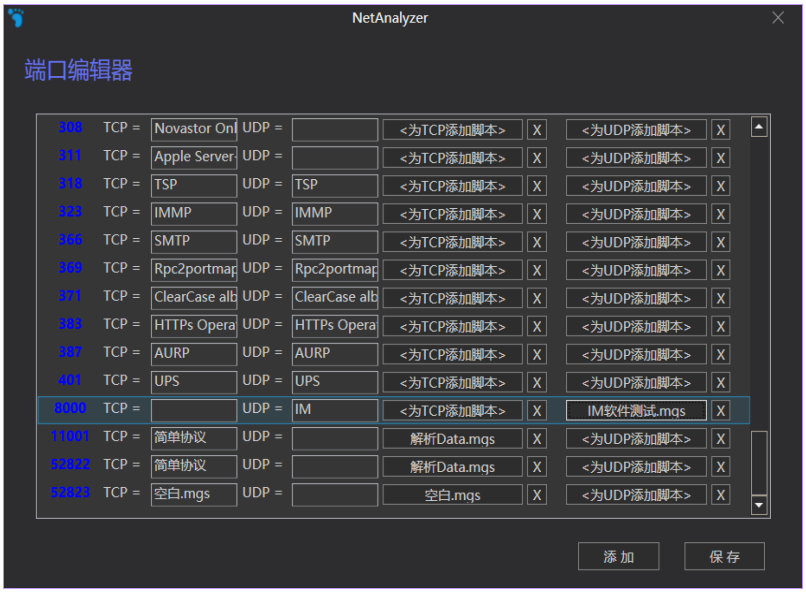
并且代码编辑器，提供了自动完成、函数说明、示例等多种辅助手段。



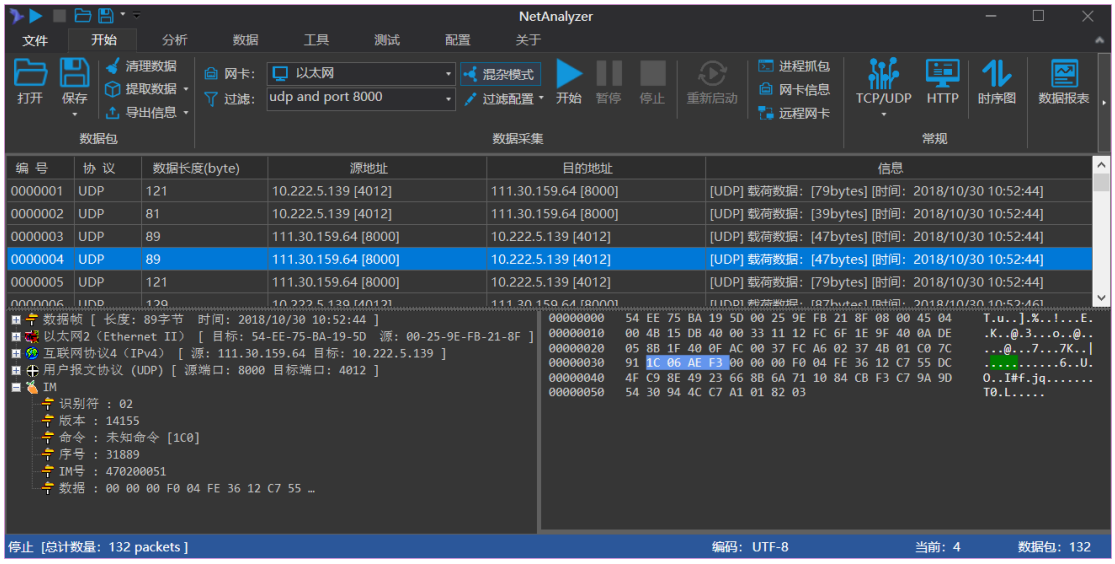
MgsExpress



自动完成功能



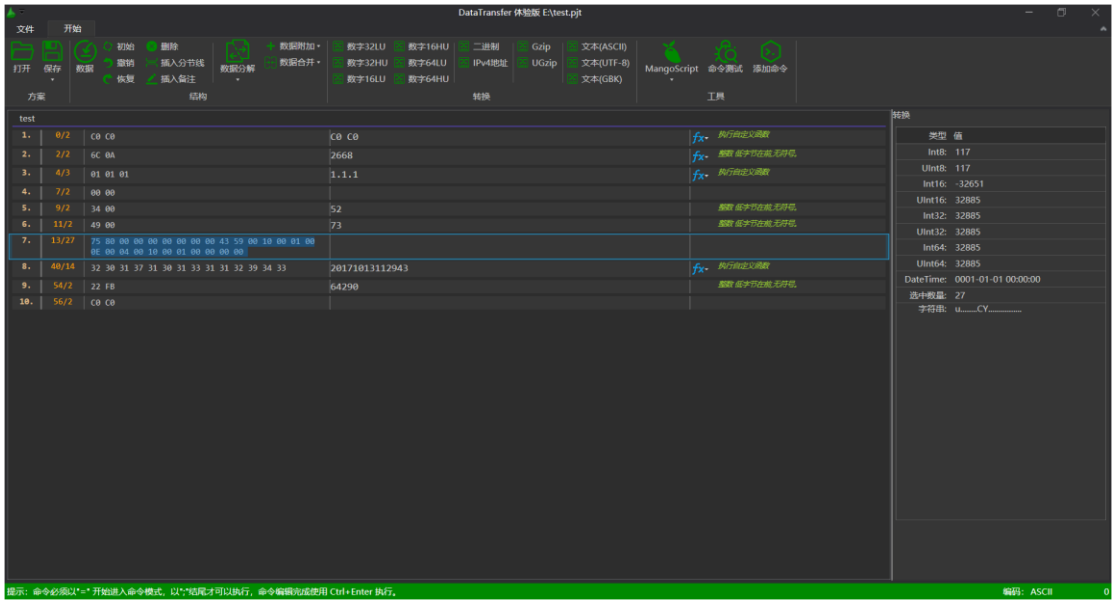
绑定到指定的端口



NetAnalyzer 中使用 MangScript 分析数据

更详细的 MgsExpress 使用方法和 MangScript 的语法规则请见《NetAnalyzer 使用说明书 二扩展与开发》。

DataTransfer 数据手动分析工具，MangScript 的命令行模式应用程序之一，该工具主要功能就是对一段数据进行分析，借助该工具对字节数组的格式化操作，选择转换功能，以及借助 MangScript 命令行的便利性，也可以更加快速的完成数据的分析。



DataTransfer 分析数据

打开或保存分析方案文件

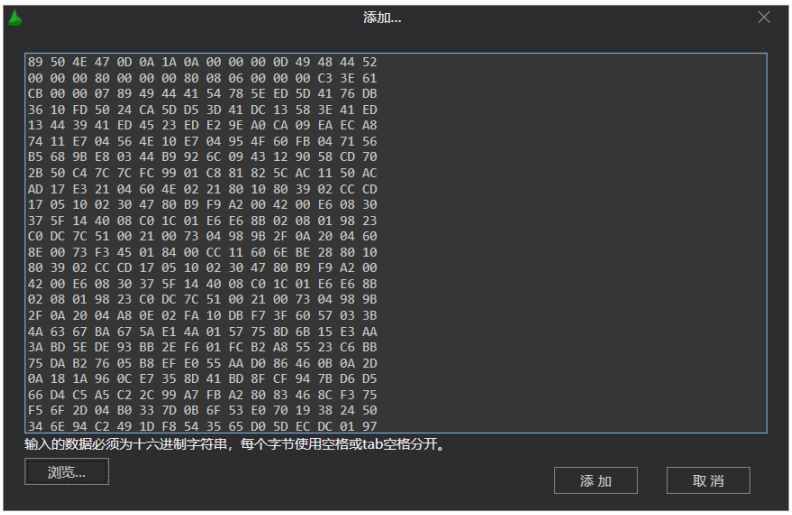
分析方案文件就是将现在分析的内容进行完整的保存或打开继续分析。

DataTransfer 的功能相对来说较为简单，基本可以分为对数据结构的分析和对数据值的分析

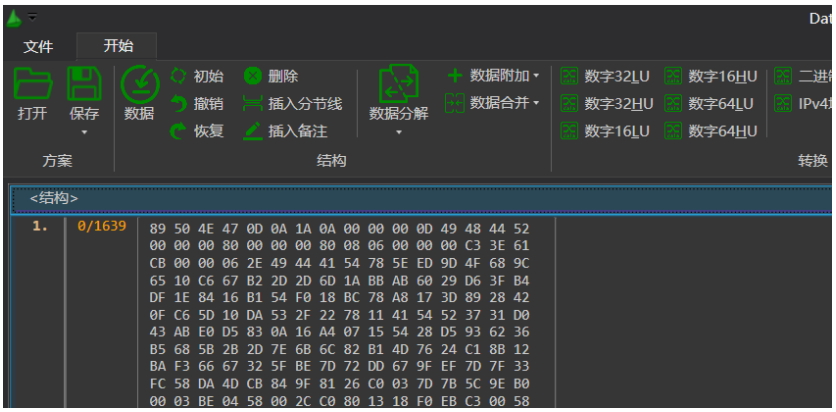
结构和转换菜单

结构部分：大概分为数据载入、数据分解和数据合并三部分

点击**数据** 可以通过文件载入数据或是通过 HEX 字符串载入数据



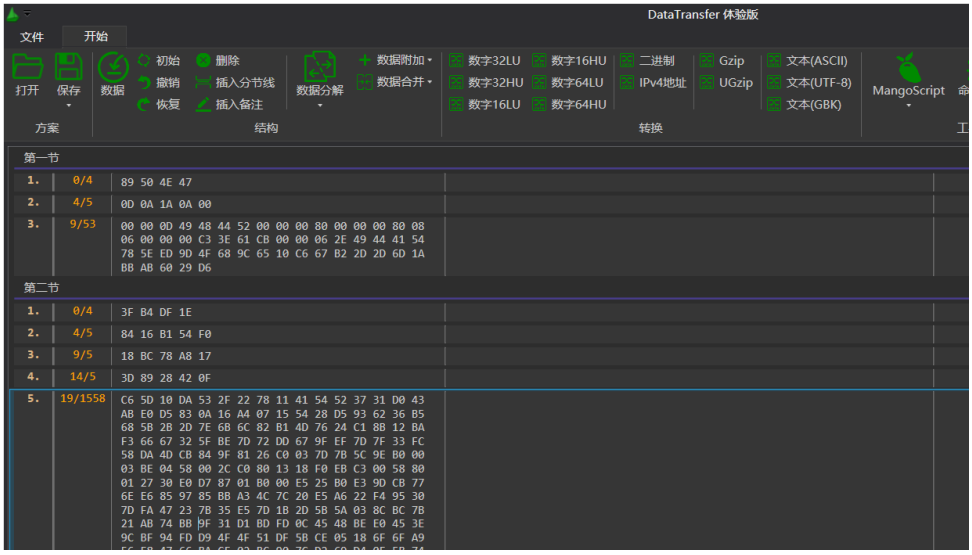
添加数据



完成数据载入

载入数据后会自动添加一个节名称为<结构>，双击可以修改名称。

节，在分析过程中可能会遇到多组数据的分析，或是对分析的数据可以分成过个块进行独立定义，这里就引入了节的概念。需要注意的是每个节的独立索引值独立进行计数。



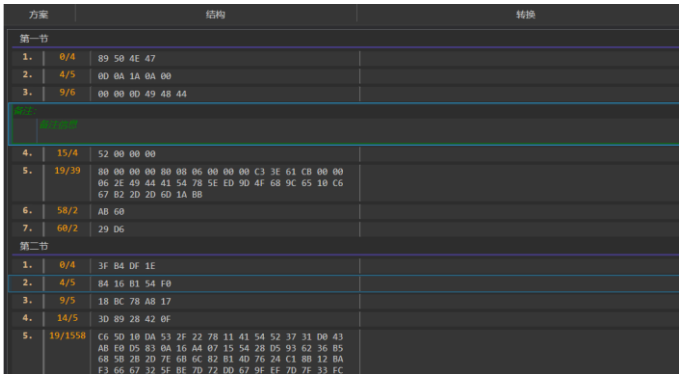
引入两个节的数据分析

数据分解：数据分解提供了多种方式，DataTransfer 中的十六进制编辑器对十六进制数据选择进行了特殊处理，选中的数据必然为一组完整的十六进制数组。所以当用光标定位到一个字节上面会自动做位置修正。当稳定后按回车键，当前数据就会被分解为两个组；还可以选择一部分数据然后点击菜单栏上面的**数据分解**或右击选择**数据分解**，软件会根据你选择数据的位置自动进行适配计算，是要分解为两组或是三组，如果选择的是前段或是后端，则分解为两组，如果选择的是中间的数据，则会分为三组。

数据合并：当分为两个组的数据需要合并的时候，此时就可以使用数据合并功能，该功能提供两个选项与前面的数据合并和后面的数据合并，合并的数据必须是同一个节内的两个相邻的数据组，否则会给出异常提示，无法合并。

数据附加：该功能会改变当前分析的数据内容，在选定的数组前面或后面添加数据。

插入备注：该功能显示与节类似，但是不会影响节内数据的分解与合并，而且不会打断数组之间的索引值。

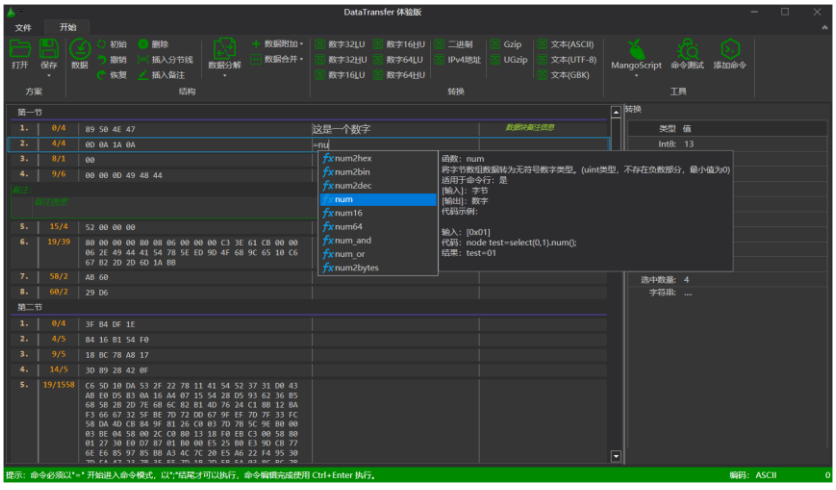


添加备注

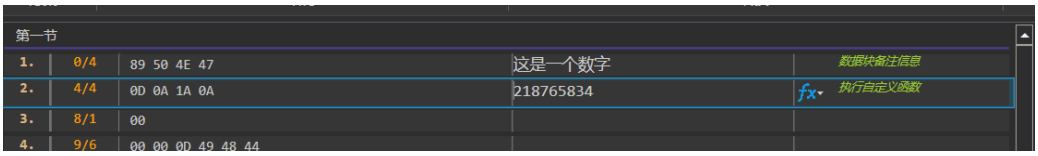
转换，处理完了结构部分，接下来就是做数据的转换，在数据框后面的文本框内可以直接编写转换的内容。最后则是备注

常规值输入与备注信息

但是值文本框不止于简单的输入，为了使用习惯，这里借用了某著名表格处理软件的经验，通过输入“=”进入 MangScript 模式，当正确输入完成命令后，一定要在结尾添加“;”，然后执行 Ctrl+Enter，就可以执行框内代码。



基于 MangScript 命令模式



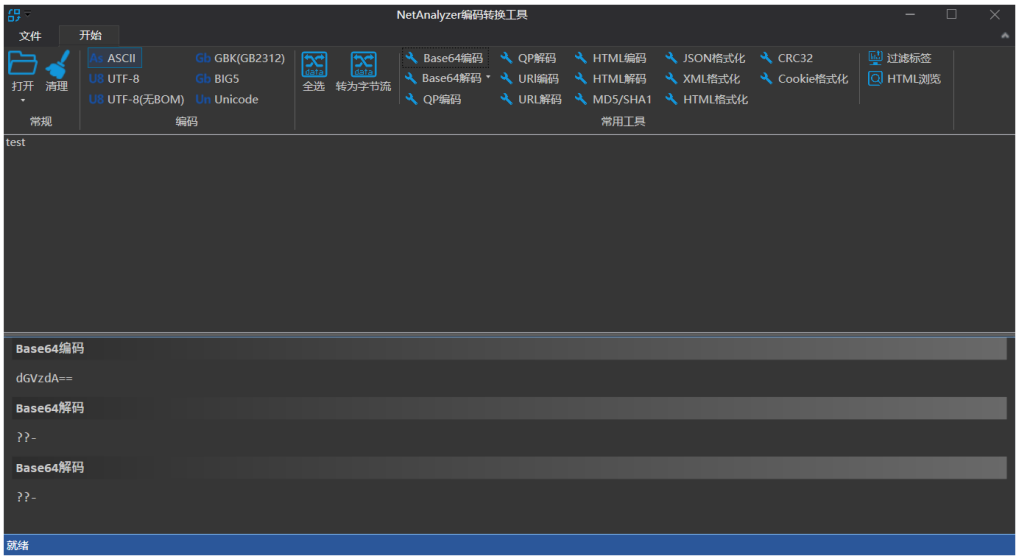
MangScript 命令执行完成的结果

此时在备注框内会显示一个 作为使用过函数的标志。如果要对值进行重算或维护，双击值窗口会呈现代码，进行修改即可，重新执行就可以。

为了快速的实行转换，DataTransfer 在转换菜单增加了一些基本的功能按钮，而这些按钮就是携带 MangScript 命令代码的一组控件，本质和输入命令一致。

3.2.常用工具

在前面做文本分析，我们发现有个**转换为…**的功能就是我们将要说明的编码转换功能。该工具文本分析工具，可以独立使用，提供了大部分的编码转换功能。



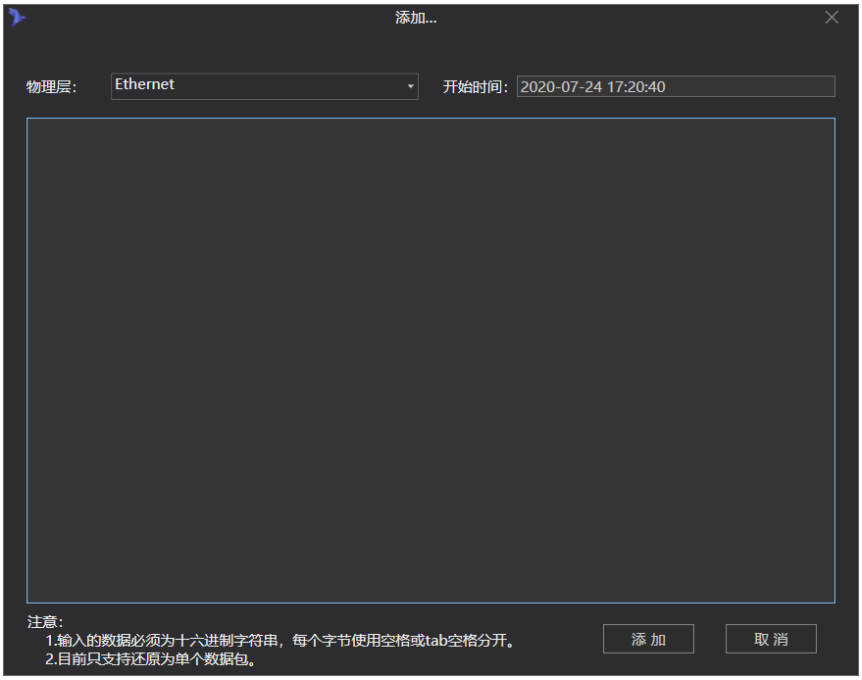
转码工具

编码转换工具菜单栏为转换功能，工作区域主要包括输入窗口和输出窗口两部分，在输入窗口输入想要处理的内容，点击对应的功能菜单，就可以在输出窗口输出对应的转换内容。

IP 位置，提供通过 IP 地址查询地理位置的功能

地址查询

来自 Hex，该功能可以将一段 hex 字符串转为一个单独的数据包，但是转换的数据必须是严格协议格式的数据否则可能会产生数据分析的错误。

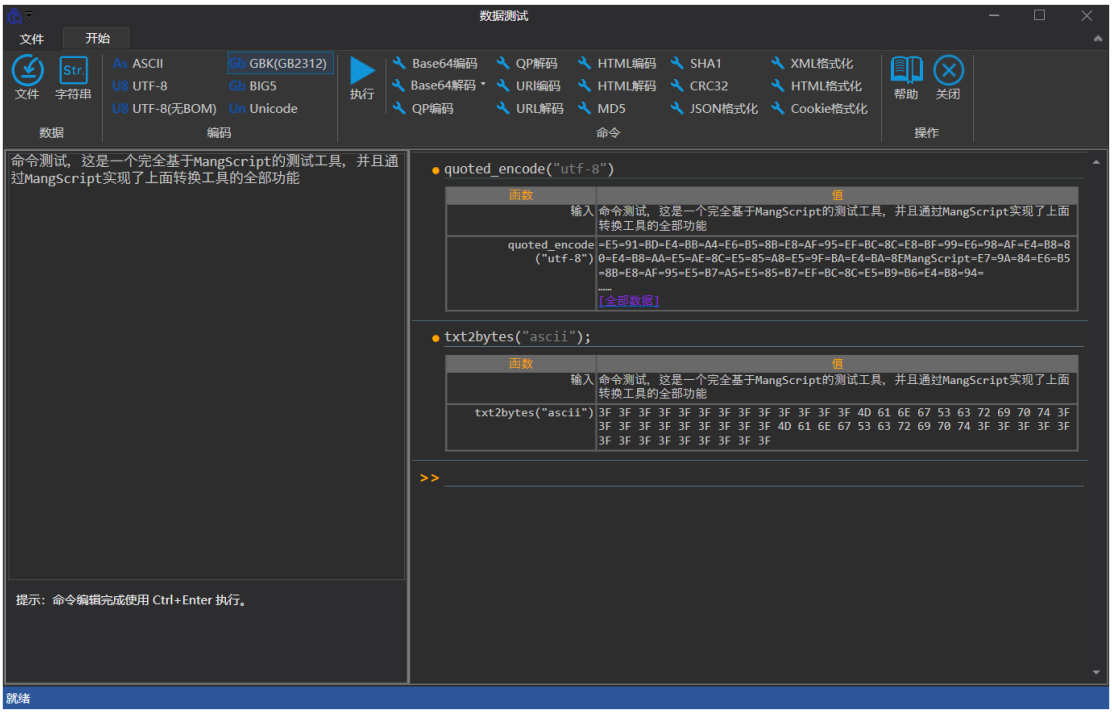


HEX 转数据包工具

命令测试，这是一个完全基于 MangScript 的测试工具，并且通过 MangScript 实现了上面**转换工具**的全部功能，但是改工具功能更加强大，它可以根据调用函数自动判断输入的数据，将其转为对应的数据类型，如：输入一段十六进制字符串，在执行函数的时候，如果对应函数要求输入是字节数组，那这段字符串自动转为对应的字节数据。

对于字节数组输入，这里需要说明一下：默认全部识别为十六进制 ，如果当前数据组为十

进制前面添加“*”，如果是二进制方式使用“@”，如果使用十六进制使用“#”，对于十六进制还可以在每个字节前加“0x”，二进制使用“0b”。



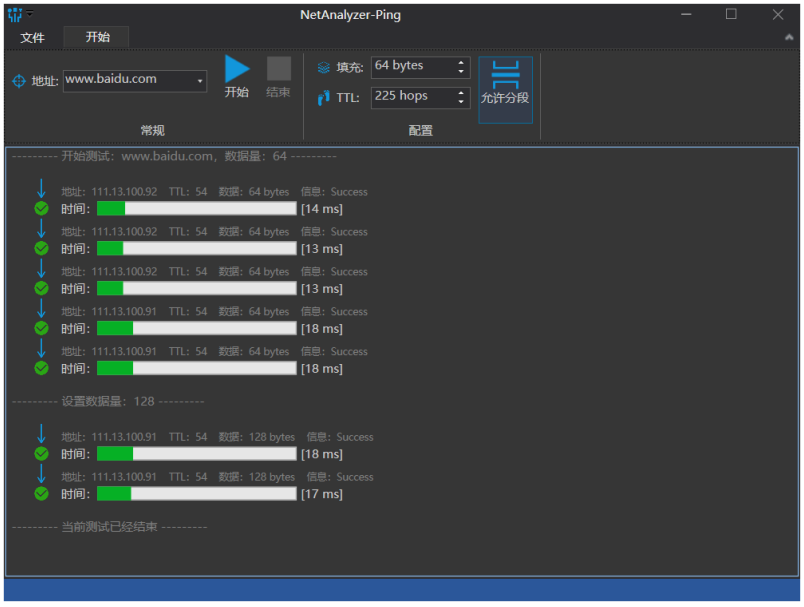
数据测试工具

为了使用更加便捷，数据测试工具在回显数据较长的时候会做一些截断处理，通过点击**[全部数据]**就可以查看全部内容

3.3.扩展

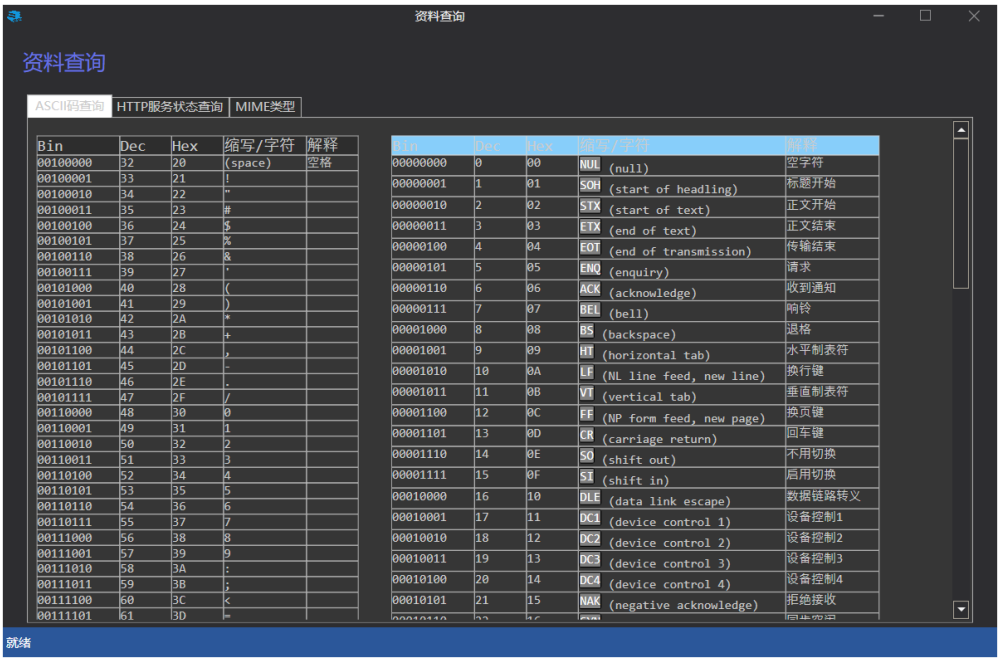
接下来介绍扩展部分的两个工具

Ping ping 工具区别与传统工具，可以通过自动增加数据量对目前主机进行网络性能测试。



Ping 工具

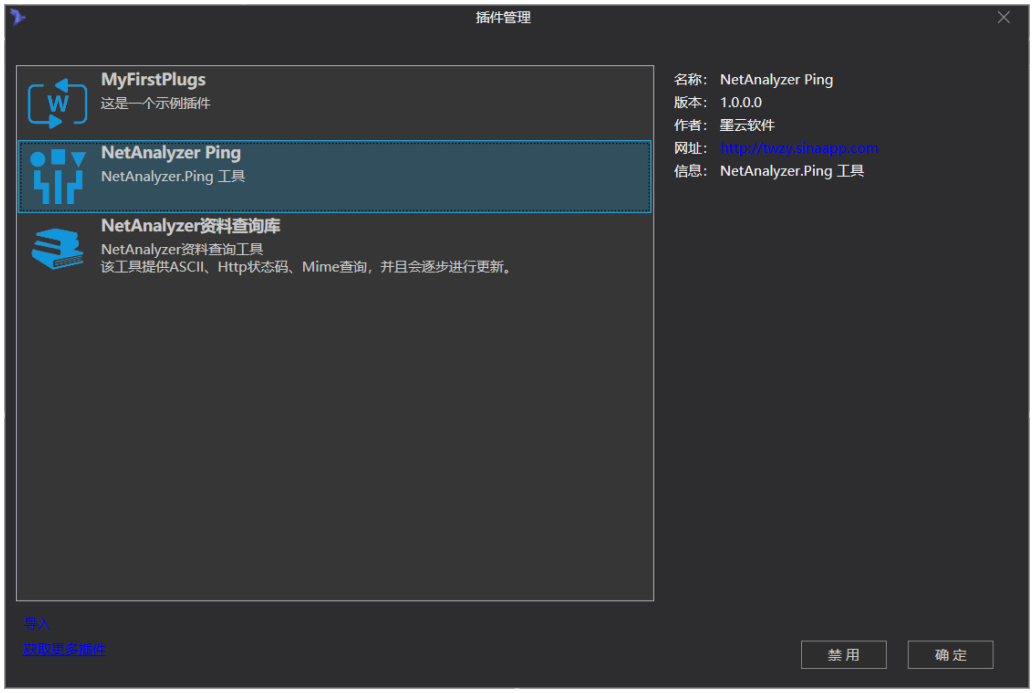
资料查询 提供 ASCII，Http 状态，Mime 相关内容查询



信息查询

3.4.插件管理

插件管理，NetAnalyzer 提供了插件机制，通过插件可以对 NetAnalyzer 进行扩展。



插件管理

插件位置：**<安装目录>\Mods**

该目录下面，会针对每个插件有一个独立的文件夹

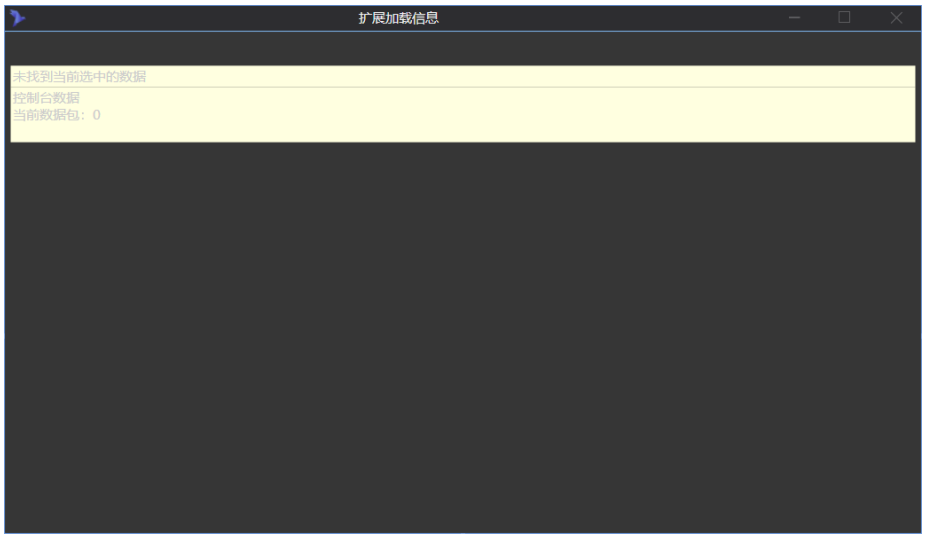
插件类型：NetAnalyzer 使用两种插件方式，生成的 DLL 方式，脚本方式

该两种方式开发方式将会在下个章节中详细说明

我们可以通过手动加入和通过 ntpk 安装包，安装插件中方法使用插件。墨云会不定期在官网增加新的插件，大家可以通过 访问官网获取对应的插件包。

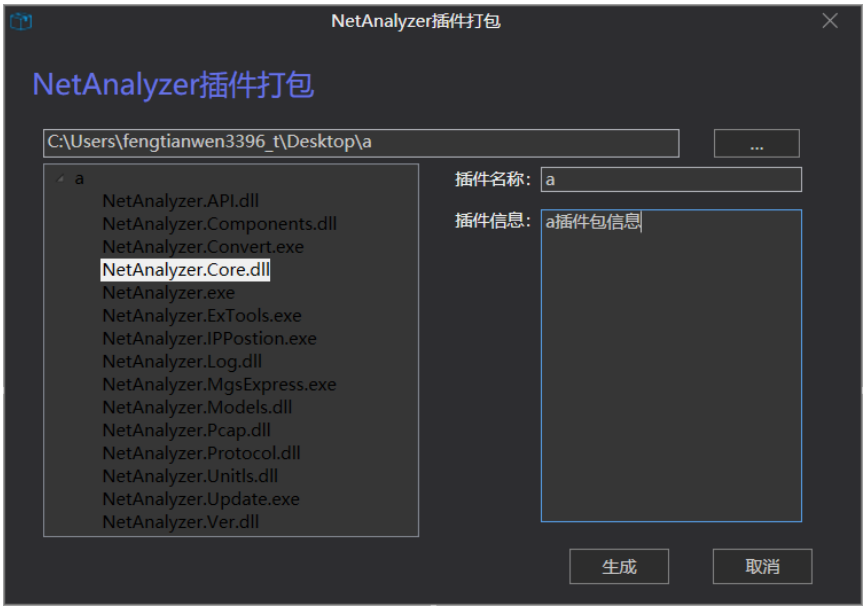
插件控制：选定一个插件后，可以通过禁用和启用插件既可以实现对插件功能呢的控制。

输出窗口，该部分主要为调试插件而用，通过调用相关接口，就可以数据对应的数据。

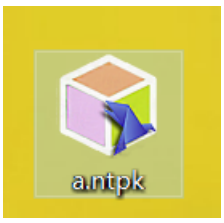


控制台信息

插件打包，该功能主要是为打包*.ntpk 插件包用，首先选择需要打包的目录，然后输入插件名称和插件信息，需要注意的是，打包的目录将会成为 NetAnalyzer 中 Mods 文件夹下的目录。

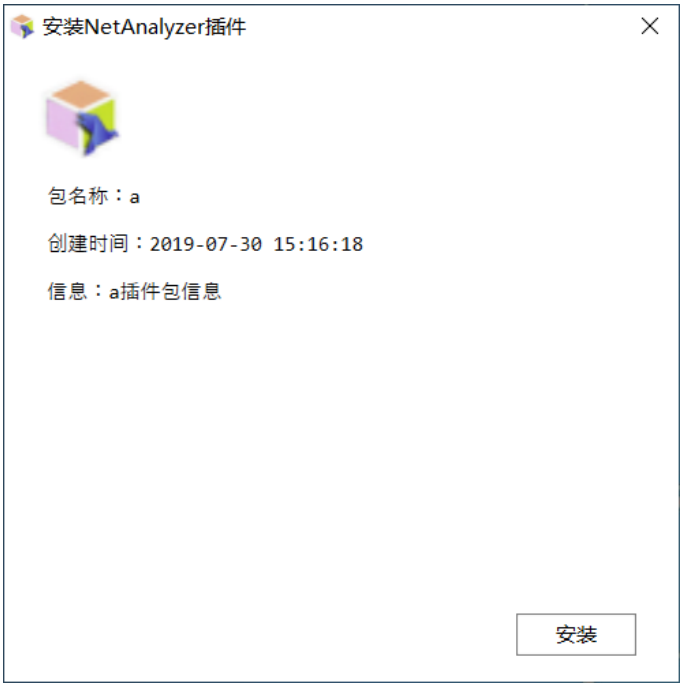


插件打包



Ntpk 插件安装包文件

双击该文件夹就可以打开安装文件。



插件安装界面

点击**安装**就可以将对应的插件安装到 NetAnalyzer 的 Mods 文件夹下，然后会提示重启 NetAnalyzer，然后重新启动就可以完成插件的安装。

3.5.配置管理

配置标签集成了针对 NetAnalyzer 部分配置，包含对于传输协议的端口配置、数据识别、加载选项、UI 呈现等各个方面的配置。

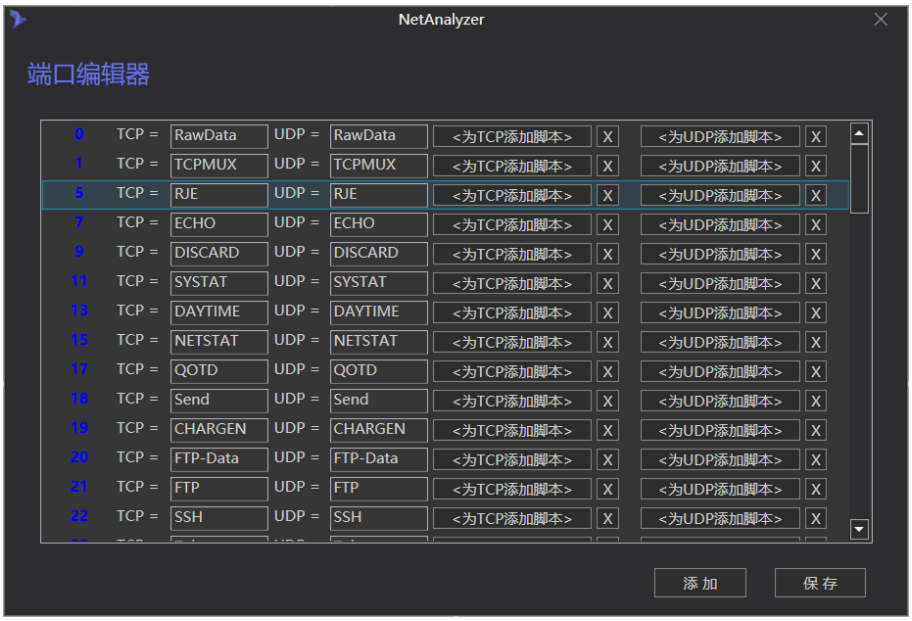
配置

端口配置

对于传输层协议 UDP/TCP，如何识别应用层提供的协议，端口号是一个很有用的标志。在计算机行业中对于一部分端口有统一的要求，如：80 为 http 服务协议，21 为 FTP 控制协议等等，但是对于一些其他协议尤其是超过 2000 以后的协议并没有严格的说明，通常意义上

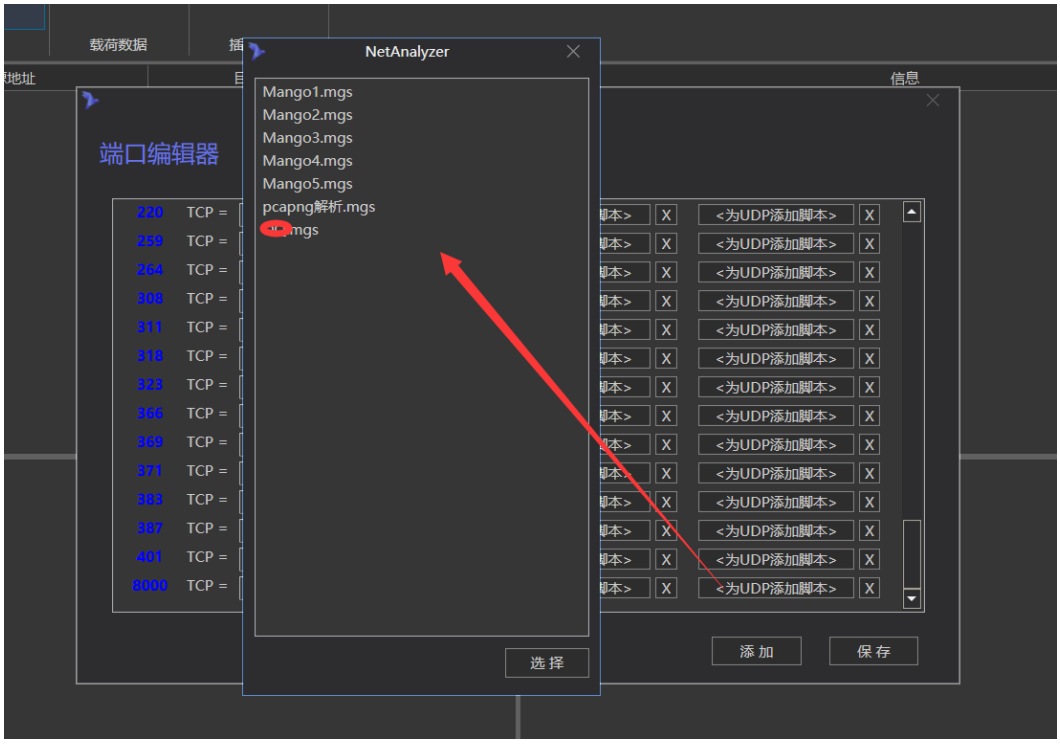
NetAnalyzer 对于这部分端口只是识别为一般数据，呈现为对应的原始字节数据。

对于端口配置就是为了满足对这部分是数据的识别，结合 MangoScript 更可以对端口下的应用层数据进行自定义解析(对于 MangoScript 脚本的开发，请参考《NetAnalyzer 使用说明 2~扩展与开发》相关的说明)。



端口配置

点击对应的端口号和协议添加脚本



配置脚本

就可以看到配置的脚本，对应的 tcp/udp 名称下会显示脚本配置的名称。

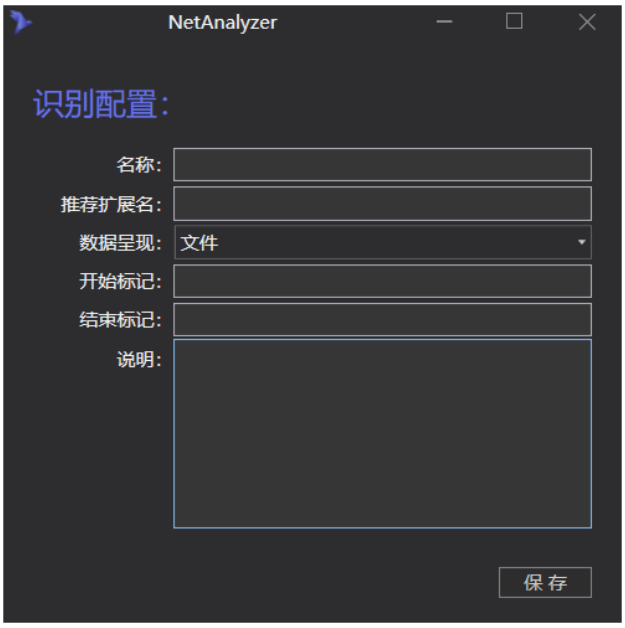
完成对 8000 端口脚本配置

数据识别

在 NetAnalyzer 进行数据分析的时候，为了快速识别一些文件或传输数据，所以配置增加的可以自定义维护的数据识别功能，该功能只是作为一项辅助功能，其原理是通过判断数据开始字节序列和结束字节序列来定位文件在采集到的数据中的位置，当前选中的数据就可以当做需要识别的内容。该功能在讲解处理载荷数据的时候有所提及，而此处就是配置数据功能。

数据识别配置界面

单击添加或修改就可以打开特征数据添加/修改数据特征界面。



添加/修改数据特征

名称：数据将要被识别的名称

推荐扩展名：如果是文件则需要填写扩展名以便于 NetAnalyzer 在识别到数据后保存时候自己添加对应的文件扩展名。

数据呈现：即为数据类型，可以选择为文件、图片、视频、音乐等等。

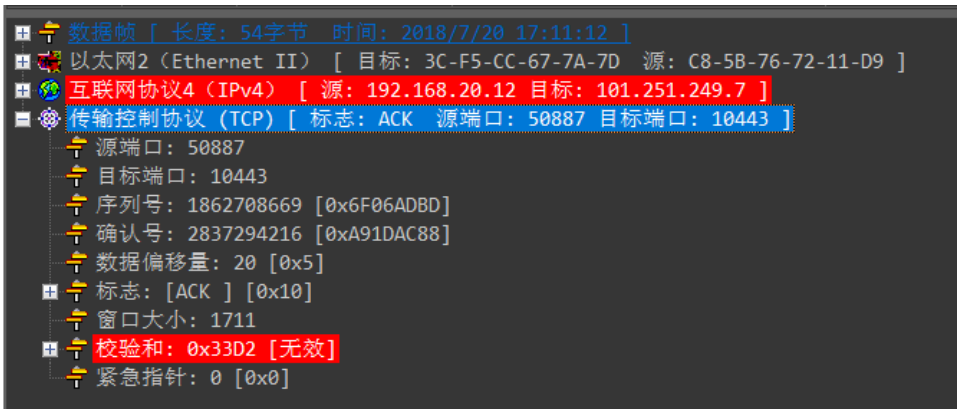
开始/结束标记：数据识别的关键部分，请填写十六进制数据字符串。

说明：备注信息描述。

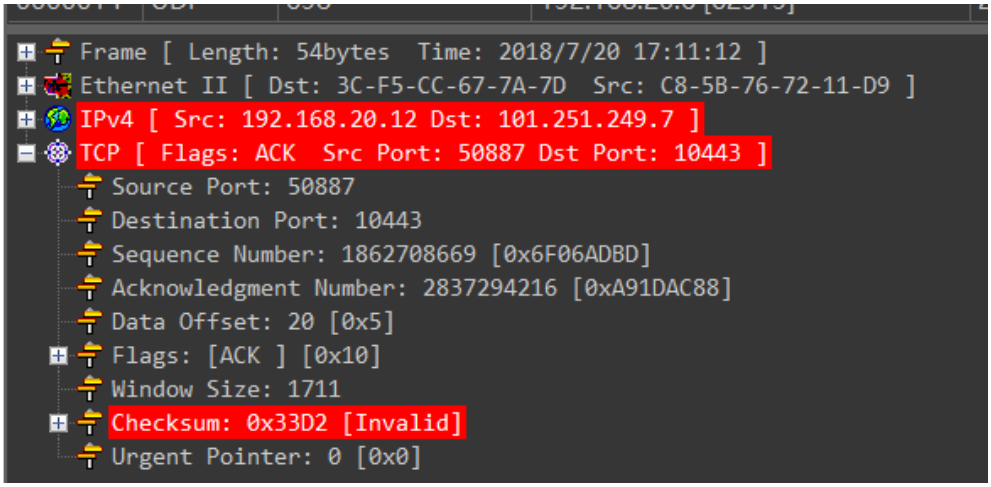
分析语言配置：

NetAnalyzer 提供了对解析的协议信息中英文切换，默认为中文显示，通过配置界面选择中文或英文可以实现解析协议描述的实时是切换

中英文切换选项



协议中文描述呈现



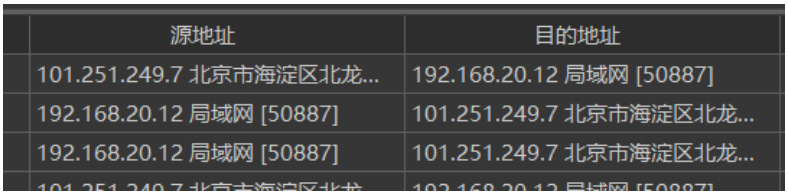
协议英文描述呈现

采集滚动 当选中后在进行抓包的时候数据列表会自动滚动到最后的位置。

窗口置顶 选中后 NetAnalyzer 整体界面置顶。

地理位置 该选项用于控制在获取数据的时候是否自动查询 IP 地址的物理位置

采集滚动/窗口置顶/地理位置选项



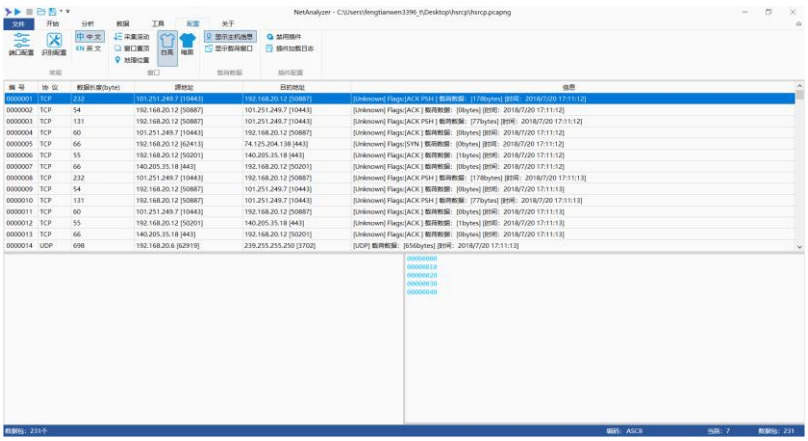
包含地理位置的数据包列表

源地址	目的地址	
101.251.249.7 [10443]	192.168.20.12 [50887]	[Un
192.168.20.12 [50887]	101.251.249.7 [10443]	[Un
192.168.20.12 [50887]	101.251.249.7 [10443]	[Un
101.251.249.7 [10443]	192.168.20.12 [50887]	[Un

不包含地理位置的数据包列表

主题方式呈现

更变主题



呈现白亮主题的 NetAnalyzer

4. 声明与资料

4.1.声明信息

1. 请使用本软件的用户自觉遵守国家相关法律，不得利用本软件从事任何违法犯罪活动，因使用本软件而造成的一切违反法律法规的责任与软件作者无关。
2. 非经作者许可不得对软件中的任何内容进行修改、删除、进行反向工程等操作。
3. NetAnalyzer 使用了部分开源代码、组件、图表、图标以及图片，该部分开源代码、组件、图表、图标以及图片版权归对应的作者和机构所有。
4. 使用者可以对本软件提供的非第三方的代码组件进行非商业用途的调用或二次开发，对于开源代码、组件、图表、图标以及图片请在遵循对应授权条件下使用。
5. 该软件可免费用于学习研究，但非经作者同意不得用于商业用途。
6. 使用者可以在非商业用途的情况下，自由复制、传播、分发本软件；从事商业用途必须经过作者同意。
7. 作者对本软件不提供任何保证，不对任何用户因本软件所遭遇到的任何理论上的或实际上的损失承担责任，不对用户使用本软件造成的任何后果承担责任。
8. 本软件相关资料来源于互联网，并不会涉及使用者隐私，因此不会侵害使用者的隐私。
9. 软件中使用了纯真 IP 地址数据库，纯真 IP 地址数据库归纯真网络所有。
10. 软件中使用了 ECharts.js 图表库，版权归百度 ECharts 团队所有。
11. 除第 3、第 8、第 9 条所涉及的开源代码、组件、图表、图标以及图片外，本软件作者保留该软件的所有代码权利。
12. NetAnalyzer 使用了插件技术，对于所使用的插件安全性当由用户自行甄别，对于用户因为使用恶意插件而造成的任何后果与软件作者无关。
13. 作者保留对 NetAnalyzer 程序、《NetAnalyzer 协议分析软件声明》的最终解释权利。

4.2.资料引用

这里推荐一些常用的网络协议分析网站，最后三个可以下载数据包

Wikipedia http://en.wikipedia.org/wiki/Communications_protocol

RFCSourcebook http://www.networksorcery.com/enp/default1101.htm

中国协议分析网 http://www.cnpaf.net/

Packet Captures - Packet Life http://packetlife.net/captures/

SampleCaptures - The Wireshark Wiki http://wiki.wireshark.org/SampleCaptures

Protocols | pcapr http://www.pcapr.net/browse/protos

4.3.其他信息

这里面的协议是 NetAnalyzer 可以完整分析的协议，共 80 多个协议，并且正在不断更新中。

NetAnalyzer 可分析协议列表
应用层
HTTP，DNS，DHCPv4，FTP，Gopher，NNTP，POP2，POP3，SMTP，Telnet，SSDP，BGP，RIPv1，RIPv2，RIPng，Echo，IPP，AODV，ESMTP，COPS，DCAP，QQ，IMAP，HSRP，MGCP，RTSP，GDOI，SIP，Kismet，MSNMS ，RTP，TRIP，Stun，Tacacs，TUNNEL
传输层
TCP，UDP，OSPFv2，OSPFv3，GREv0，GREv1，UDP-Lite，AH，ESP，CBT，DCCP，SCTP，EGP，GGP，DSR，IFMP，PIM，PGM
网络层
IPv4，IPv6，ARP，PARP，ICMPv4，ICMPv6，IGMPv1，IGMPv2，RGMP， PPPoES，PPPoED，IPCP，IPv6CP，CCP，BVCP，IPIP，IPX，AARP
数据链路层
Ethernet II， PPP，Cisco HDLC，Linux SLL，LCP，Cisco SLARP，EAP，CHAP，LLDP，WOL，ECP，IEEE802.3/802.2，Novell Ethernet，LLC，SNAP，Null/Loopback

共：85 个协议