Security Analytics: having fun with Splunk and a packet capture file

GIAC (GCIA) Gold Certification

Author: Alexandre Teixeira, forensick@ymail.com
Advisor: Dominicus Adriyanto

Accepted: 18 May 2014

Abstract

Machine data is one of the most important artifacts when it comes to monitoring and detecting computer security threats. However, while having more data increases chances to spot suspicious patterns, handling and processing it without making use of specialized tools is quite difficult. Network packet analyzers do not offer a broad view on what is happening within a network. Splunk can help addressing that challenge by indexing packet capture (pcap) data, benefiting from its data analytics capabilities.

Table of Contents

1. Introduction

Security Analytics is one of the most discussed topics within the Information Security (IS) industry, especially when combined with another buzzword such as Big Data. With a myriad of tools and platforms available in the market, often freely available for full evaluation, Splunk is among the ones standing out by promoting such trend, fostering data analytics practice within the IS community and professionals.

Data analytics is defined as the application of computer systems to the analysis of large data sets for the support of decisions. The analysis is often supported by additional scientific disciplines, such as statistics, pattern recognition and machine learning (Runkler, 2012). Therefore, Security Analytics is about making use of such approach for enabling actions and driving decisions towards securing data. Nevertheless, when defining or assessing a data set, its size is not the only attribute taken into account since it might relate to a lot of different data types and it might change in a fast pace.

In 2001, analyst Doug Laney defined data growth challenges and opportunities as being three-dimensional, with increasing volume, velocity and variety. Those are known as the 3Vs used to describe the term Big Data (Krishnan, 2013).

According to researchers at Securosis, Big Data is not really about data, it's about tools that manage and derive value from data[1]. Thus, Big Data Security Analytics is not limited to lots of data processing, nor about using a specific security tooling, but about building up a scalable platform for enabling skilled professionals to rapidly mine the data they are looking for, as a core component of an enterprise's Security Program.

Splunk is well known as a platform used to collect and query on machine data. Machine data is data produced all the time by nearly every piece of software or electronic device. It can be generated by both machine-to-machine (M2M) as well as human-to-machine (H2M) interactions (Mohanty et al, 2013). Throughout this document, Splunk refers to Splunk Enterprise which is explained in more details later in this document (Lab setup).

[1] Security Analytics with Big Data, https://securosis.com/assets/library/reports/SecurityAnalytics_BigData_V2.pdf

1.1. What knowledge is shared here?

This paper outlines some of Splunk's major analytics features by leveraging one digital artifact widely known by the computer security community: a network packet capture (pcap) file. Here are some of objectives covered by this document:

- Prepare a Linux operating system for hosting the exercises, covered in details later on section “Lab's questionnaire”;
- Prepare data to be consumed by Splunk by using *tshark*, one of the command-line components of Wireshark package;
- Install and configure Splunk Enterprise (version 6.0.2) for indexing and enabling querying on pcap file's content referred in this document;
- Use Splunk major features to generate stats and charts, which can be seamlessly applied to any other data sets, enabling readers to evaluate Splunk as a data mining or Business Intelligence (BI) platform.

Note that event correlation is not covered here since that feature is often found and performed by the Security Information and Event Management (SIEM) engines. This paper aims at highlighting Splunk's features around data mining and reporting from a didactic point of view.

Knowledge about Open Source Software (OSS) as well as network concepts will help understanding and applying the experiments carried on here, which are fully repeatable. Additional information is found from the references section available at the end of this paper.

2. Lab setup

2.1. Description

Splunk actually supports several operating systems, including FreeBSD and Linux. For this lab, Fedora Linux is used as the hosting operating system (OS) for a sort of all-in-one Splunk setup described later. Even though some components are covered in more details, please refer to specific documents indicated on the appendix section for a comprehensive documentation, including *Fedora Linux installation guide* (2013) and *Wireshark User's Guide* (Lamping, U., Sharpe, R., & Warnicke, E., 2013).

Splunk recommends the following hardware (HW) requirements in order to run Splunk Enterprise software on Linux: 2x six-core, 2+ GHz CPU, 12 GB RAM, Redundant Array of Independent Disks (RAID) 0 or 1+0, with a 64 bit OS installed. However, for a simple lab evaluation, it should run well with low-end HW, as well as on virtualized environment with reasonable performance degradation. More details can be found from the following link:

http://docs.splunk.com/Documentation/Splunk/latest/Installation/Systemrequirements

The following describes the main components used to perform the exercises presented throughout this paper:

- **Hardware**
  - Intel(R) Core(TM)2 Duo CPU T6600 @2.20GHz (64-bit)
  - 4GB RAM
  - 150GB disk space with standard EXT4 partitioning
- **Software**
  - Linux distribution: Fedora release 20 (Heisenbug), Desktop Edition2
  - Splunk version: splunk-6.0.2-196940.x86_64 (RPM package)
  - Timezone: UTC (to change, execute: timedatectl set-timezone UTC)

Full details about the HW and OS specs, including a full *rpm -qa* output, can be found at the appendix section "Lab specs".

[2] Fedora Linux project's website – Download area, http://fedoraproject.org/en/get-fedora

2.2. Lab configuration

From this point on, it's assumed that Linux is properly installed and requirements mentioned before are met.

Note: for this section, all commands need to be executed as root user or by using sudo functionality, unless mentioned otherwise. Internet connectivity is mandatory, unless a local Linux repository is available.

The following command will automatically download and update the latest Fedora software available from internet (by default) package repositories:

```
yum -y update
```

Aside from default installed packages, the following ones need to be installed by executing the commands below:

```
yum -y install wireshark
yum -y install wireshark-gnome
```

To download Splunk, an account must be used in order to get access to the Download area at Splunk portal. New users can sign in via the following link:

https://www.splunk.com/page/sign_up

Once a user is logged in to the portal, the following link provides different installation packages, according to the target OS and packaging system:

http://www.splunk.com/download

After downloading the RPM package for 2.6+ kernel Linux distributions (64-bit), the following command will install Splunk package (output included):

```
rpm -ivh splunk-6.0.2-196940-linux-2.6-x86_64.rpm
Preparing...                          ################################# [100%]
Updating / installing...
   1:splunk-6.0.2-196940              ################################# [100%]
complete
```

Note that the latest package version might be different, depending on the time this procedure is followed. For this lab, version 6.0.2 release 196940 is used. The installer already creates a dedicated user (*splunk*) for running and owning Splunk related processes and file, rather than using *root* for that. It also applies files permissions and ownership accordingly to Splunk's home directory, */opt/splunk*. Splunk can be deployed across multiple servers by using components such as forwarders, indexers, and search heads. More information is available at the Distributed Deployment Manual[3].

For this lab, the simplest setup is used, thus, no event forwarding is needed since the pcap file is imported directly from Splunk server's local disk. Also, the same standalone server performs indexing and searching functions.

3. Data preparation

The subject of the experiments presented here is a pcap file, a binary file accommodating network packet data, typically gathered via a network sniffer or a network packet inspection tool such as *tcpdump*.

Since there are tons of public available pcap files on the web, it turned out to be a good option making use of one of them. Netresec team put together an extensive list of files available for download at their website[4]. To narrow down the list of potential file candidates for using here, a few constraints are set:

1. The file should host more than 1 million IPv4 connections (UDP and TCP traffic);
2. It should contain over 10 different application protocols.

Having all those attributes within a single file makes it easier since no pcap file merging operation is needed. After examining some files from Netresec, the data set from DARPA shown to be a good one as it fulfills the requirements above. Even though the data set is considered old (dating from 2000), it still fits for the purpose of this paper.

The file used here is listed under “Windows NT Attack Data Set” and referenced as “Outside Tcpdump Data”. It can be downloaded from the following location:

http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/2000/NT_dataset/outside.tcpdump.gz

The extracted file (~273MB) should match the following MD5 hash:

```
md5sum outside.tcpdump
88c827386f30802784cc8f2ef1ceea76  outside.tcpdump
```

[3] Distributed Deployment Manual, http://docs.splunk.com/Documentation/Splunk/6.0.2/Deploy/Distributedoverview
[4] Publicly available PCAP files, http://www.netresec.com/?page=PcapFiles

3.1. Data conversion and selection

Since the packet capture file is binary file, it has to be prepared in such way that Splunk is able to parse its content. Here's where Wireshark tools come in very handy.

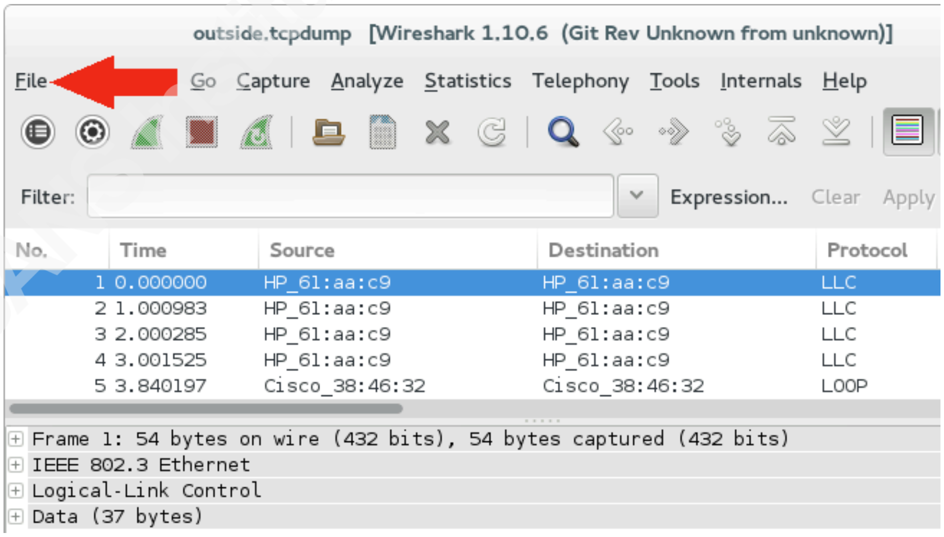
Note: for this section, all commands need to be executed as splunk user, unless mentioned otherwise. Internet connectivity is mandatory, unless the pcap file is already available locally. Default working directory should be /opt/splunk, automatically created by the RPM installer script.

Wireshark package offers two main interfaces for processing pcap files: command-line and graphical; *tshark* and *wireshark*, respectively. The extracted file (*outside.tcpdump*) indicates the capture was performed using *tcpdump* command, which is one of the options to gather network data. Although it doesn't have the typical .pcap file extension, it's easy to verify its file type by executing the following command:

```
file outside.tcpdump

outside.tcpdump: tcpdump capture file (big-endian) - version 2.4 (Ethernet, capture
length 66000)
```

For opening the pcap file on Wireshark use the standard method (File->Open), locating and selecting the file *outside.tcpdump*, as shown on Figure 1 below:

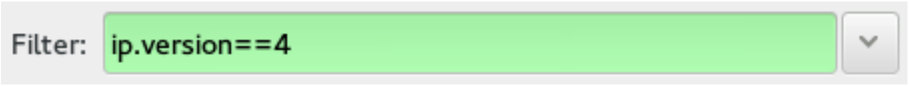


No.	Time	Source	Destination	Protocol
1	0.000000	HP_61:aa:c9	HP_61:aa:c9	LLC
2	1.000983	HP_61:aa:c9	HP_61:aa:c9	LLC
3	2.000285	HP_61:aa:c9	HP_61:aa:c9	LLC
4	3.001525	HP_61:aa:c9	HP_61:aa:c9	LLC
5	3.840197	Cisco_38:46:32	Cisco_38:46:32	LOOP

```
⊞ Frame 1: 54 bytes on wire (432 bits), 54 bytes captured (432 bits)
⊞ IEEE 802.3 Ethernet
⊞ Logical-Link Control
⊞ Data (37 bytes)
```

Figure 1

For instructing Wireshark to display only traffic related to IPv4 protocol, the following filter should be applied:



Filter: ip.version==4

Figure 2

In order to verify how many packets and connections (Displayed) are matching the filter, simply check the status bar, as shown below:

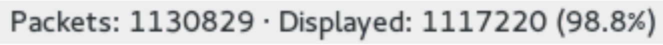

Packets: 1130829 · Displayed: 1117220 (98.8%)

Figure 3

After file integrity and contents are verified, next step is preparing the data to be consumed by Splunk's data intake process. Splunk supports a lot of different data and file types (referenced as data inputs), ranging from syslog to XML files. One of the simplest types is known as CSV, which stands for comma separated values. In order to turn a pcap file, which is basically binary packed data, *tshark* command comes into play by allowing specific fields to be dumped from the network capture file.

Before providing the *tshark* command to extract the data needed for the exercises, below is the list of all parameters used along with their description[5]:

```
-2  Perform a two-pass analysis. This causes tshark to buffer output until the entire
first pass is done, but allows it to fill in fields that require future knowledge, such
as 'response in frame #' fields. Also permits reassembly frame dependencies to be
calculated correctly.

-r  <infile>

           Read packet data from infile, can be any supported capture file format
(including gzipped files).  It's not possible to use named pipes or stdin here!

-T  pdml|psml|ps|text|fields

           Set the format of the output when viewing decoded packet data.  The options
are one of:

           pdml Packet Details Markup Language, an XML-based format for the details of a
decoded packet.  This information is equivalent to the packet details printed with the -V
flag.

           psml Packet Summary Markup Language, an XML-based format for the summary
information of a decoded packet.  This information is equivalent to the information shown
in the one-line summary printed by default.

           ps PostScript for a human-readable one-line summary of each of the packets, or
a multi-line view of the details of each of the packets, depending on whether the -V flag
was specified.

           text Text of a human-readable one-line summary of each of the packets, or a
multi-line view of the details of each of the packets, depending on whether the -V flag
was specified.  This is the default.

           fields The values of fields specified with the -e option, in a form specified
by the -E option.  For example,

             -T fields -E separator=, -E quote=d

           would generate comma-separated values (CSV) output suitable for importing into
your favorite spreadsheet program.

-E  <field print option>

           Set an option controlling the printing of fields when -T fields is selected.
Options are:
```

[5] Part of tshark's manual page, http://www.wireshark.org/docs/man-pages/tshark.html

```
           header=y|n If y, print a list of the field names given using -e as the first
line of the output; the field name will be separated using the same character as the
field values.  Defaults to n.

           separator=/t|/s|<character> Set the separator character to use for fields.  If
/t tab will be used (this is the default), if /s, a single space will be used.  Otherwise
any character that can be accepted by the command line as part of the option may be used.

           occurrence=f|l|a Select which occurrence to use for fields that have multiple
occurrences.  If f the first occurrence will be used, if l the last occurrence will be
used and if a all occurrences will be used (this is the default).

           aggregator=,|/s|<character> Set the aggregator character to use for fields
that have multiple occurrences.  If , a comma will be used (this is the default), if /s,
a single space will be used.  Otherwise any character that can be accepted by the command
line as part of the option may be used.

           quote=d|s|n Set the quote character to use to surround fields.  d uses double-
quotes, s single-quotes, n no quotes (the default).

-e  <field>

           Add a field to the list of fields to display if -T fields is selected.  This
option can be used multiple times on the command line.  At least one field must be
provided if the -T fields option is selected. Column names

           may be used prefixed with "col."

           Example: -e frame.number -e ip.addr -e udp -e col.info
```

Additionally, a filter can be applied inline, following the same syntax already showed above in Wireshark. Thus, as a quick example, for filtering only IPv4 traffic from the pcap file and retrieving a few IP header fields, the following command is executed:

```
tshark -r outside.tcpdump -T fields -E header=y -E separator=, -E occurrence=a -E quote=d
-e ip.proto -e ip.src -e ip.dst "ip.version==4"
```

Above command outputs the following content (first 5 lines only):

```
ip.proto,ip.src,ip.dst

"17","172.16.112.10","192.168.1.10"

"17","192.168.1.10","172.16.112.10"

"17","192.168.1.1","224.0.0.9"

"17","192.168.1.1","224.0.0.9"
```

However, additional fields are extracted for the purpose of this lab, so Splunk is able to process and correlate more data, generating more insightful results. The full list of fields is provided below, just as a reference:

- frame.time
- ip.version, ip.id, ip.len, ip.proto, ip.ttl, ip.flags, ip.src, ip.dst
- icmp.code, icmp.type, icmp.resptime
- udp.srcport, udp.dstport
- dns.id, dns.qry.type, dns.resp.type, dns.qry.name, dns.resp.addr
- tcp.stream, tcp.seq, tcp.flags, tcp.srcport, tcp.dstport

- http.request.method, http.host, http.request.version, http.user_agent, http.server, http.response.code, http.response.phrase

Field names already suggest what kind of information each field is related to. The first of the list is needed so that Splunk can treat it as a timestamp, especially when dealing with stats relying on chronological data, also referred to as time-series.

Without further do, here's the *tshark* command used to extract all those fields from the DARPA's pcap file downloaded before:

```
tshark -2 -r outside.tcpdump -T fields -E header=y -E separator=, -E occurrence=a  -E
quote=d -e frame.time -e ip.version -e ip.id -e ip.len -e ip.proto -e ip.ttl -e ip.flags
-e ip.src -e ip.dst -e icmp.code -e icmp.type -e icmp.resptime -e udp.srcport -e
udp.dstport -e dns.id -e dns.qry.type -e dns.resp.type -e dns.qry.name -e  dns.resp.addr
-e tcp.stream -e tcp.seq -e tcp.flags -e tcp.srcport -e tcp.dstport -e
http.request.method -e http.host -e http.request.version -e http.user_agent -e
http.server -e http.response.code -e http.response.phrase "ip.version==4" > out.csv
```

Below are the first lines from the generated file (out.csv):

```
frame.time,ip.version,ip.id,ip.len,ip.proto,ip.ttl,ip.flags,ip.src,ip.dst,icmp.code,icmp.
type,icmp.resptime,udp.srcport,udp.dstport,dns.id,dns.qry.type,dns.resp.type,dns.qry.name
,dns.resp.addr,tcp.stream,tcp.seq,tcp.flags,tcp.srcport,tcp.dstport,http.request.method,h
ttp.host,http.request.version,http.user_agent,http.server,http.response.code,http.respons
e.phrase

"Aug  7, 2000
12:00:16.440566000","4","0xb82f","76","17","254","0x02","172.16.112.10","192.168.1.10",,,
,"123","123",,,,,,,,,,,,,,,,,,

"Aug  7, 2000
12:00:16.441809000","4","0x01f0","76","17","64","0x00","192.168.1.10","172.16.112.10",,,,
"123","123",,,,,,,,,,,,,,,,,,
```

Before importing the CSV file into Splunk, some data sanitization is needed so that chances of having parsing issues are minimized. Some points to note:

1. When writing Splunk queries, the dot (".") symbol is used for concatenating strings; likewise, tshark dumps data from packets using the same symbol (dot) for compound field names (e.g., ip.src), which might cause issues when building up search queries;
2. Header fields (1st output line) need to be double-quoted as well so that the same pattern is kept along the whole file;
3. All values are delimited by comma and double-quoted, so checking for any unexpected double-quotes makes sense at this point.

For tackling #1 and #2 issues above, the following *sed* command does the job:

```
sed '1s/\./_/g;1s/\([^,\n]\{1,\}\)/"\1"/g' out.csv > out-sanitized.csv
```

Above command creates a new, final file called *out-sanitized.csv*, here is the *diff* output between both files:

```
$ diff out.csv out-sanitized.csv
1c1
<
frame.time,ip.version,ip.id,ip.len,ip.proto,ip.ttl,ip.flags,ip.src,ip.dst,icmp.code,icmp.
type,icmp.resptime,udp.srcport,udp.dstport,dns.id,dns.qry.type,dns.resp.type,dns.qry.name
,dns.resp.addr,tcp.stream,tcp.seq,tcp.flags,tcp.srcport,tcp.dstport,http.request.method,h
ttp.host,http.request.version,http.user_agent,http.server,http.response.code,http.respons
e.phrase
---
>
"frame_time","ip_version","ip_id","ip_len","ip_proto","ip_ttl","ip_flags","ip_src","ip_ds
t","icmp_code","icmp_type","icmp_resptime","udp_srcport","udp_dstport","dns_id","dns_qry_
type","dns_resp_type","dns_qry_name","dns_resp_addr","tcp_stream","tcp_seq","tcp_flags","
tcp_srcport","tcp_dstport","http_request_method","http_host","http_request_version","http
_user_agent","http_server","http_response_code","http_response_phrase"
```

In order to check for any double-quote right beside anything but a comma character within the final file, here's one of the several ways to make it:

```
grep -q "[^,]\"[^,]" out-sanitized.csv || echo file is OK
file is OK
```

As a final check, the file should contain exactly 1117220 + 1 lines, accounting for the number of IPv4 connections exported plus the file header, respectively. So here's the command to verify that:

```
$ wc -l out-sanitized.csv
1117221 out-sanitized.csv
```

The count shown from 2nd line, 1117221, comprises the sum of 1117220 (Figure 3) with 1, meaning that all IPv4 connections are correctly stored in the file along with an extra line (header).

Finally, the data is ready to be consumed by Splunk as a standard CSV file. Adding or removing fields is quite simple, as shown before, which provides great flexibility, depending on how much data needs to be processed.

4. Data intake

In this section, the process of importing the CSV file (*out-sanitized.csv*) is covered in details. Splunk offers capabilities to accomplish that over CLI or web frontends, however, didactic wise, the latter will is used here.

As stated before, */opt/splunk* is Splunk's default installation and home folder, hence the variable *$SPLUNK_HOME* pointing to it. That's important to note since that variable is often referenced within documentation and scripts.

Note: for this section, all commands need to be executed as splunk user, unless mentioned otherwise. Internet connectivity is not mandatory.

To start Splunk services, execute the following command:

```
/opt/splunk/bin/splunk start
```

If starting Splunk for the first time, a "y" should be provided as an answer for the license agreement question. Below is the standard output of such process:

```
Splunk> The IT Search Engine.
Checking prerequisites...
        Checking http port [8000]: open
        Checking mgmt port [8089]: open
        Checking configuration...  Done.
                Creating: /opt/splunk/var/lib/splunk
                Creating: /opt/splunk/var/run/splunk
                Creating: /opt/splunk/var/run/splunk/appserver/i18n
                Creating: /opt/splunk/var/run/splunk/appserver/modules/static/css
                Creating: /opt/splunk/var/run/splunk/upload
                Creating: /opt/splunk/var/spool/splunk
                Creating: /opt/splunk/var/spool/dirmoncache
                Creating: /opt/splunk/var/lib/splunk/authDb
                Creating: /opt/splunk/var/lib/splunk/hashDb
        Checking critical directories...     Done
        Checking indexes...
                Validated: _audit _blocksignature _internal _thefishbucket history main
summary
        Done
New certs have been generated in '/opt/splunk/etc/auth'.
        Checking filesystem compatibility...  Done
        Checking conf files for typos...     Done
All preliminary checks passed.
Starting splunk server daemon (splunkd)...
Done
 [  OK  ]
Starting splunkweb...  Generating certs for splunkweb server
Generating a 1024 bit RSA private key
...........................................++++++
...........++++++
writing new private key to 'privKeySecure.pem'
-----
Signature ok
subject=/CN=gcia/O=SplunkUser
Getting CA Private Key
writing RSA key
 [  OK  ]
Done
If you get stuck, we're here to help.
Look for answers here: http://docs.splunk.com
The Splunk web interface is at http://gcia:8000
```

As seen above, the lab server has its hostname set to *gcia* and no errors were found during Splunk initialization process.

Once that step is clear, a couple of modifications need to be done as the traffic comprising DARPA's data set is quite old, dating back from year 2000. By default, Splunk does not recognize date/time patterns older than 2000 days, hence a simple modification is needed. In case Splunk is already in production or default values must be reset back to defaults, it's recommended to make a backup of each target file listed below before proceeding with the changes.

By using any text editor, the following changes are needed:

Target file: */opt/splunk/etc/system/default/props.conf*

Default, old value:	MAX_DAYS_AGO=2000
New, suggested value:	MAX_DAYS_AGO=20000

So, basically, the parameter above needs its value to be changed from 2000 to a higher value (e.g., 20000), which will enable Splunk to recognize very old date/time patterns within the CSV file that is about to be indexed.

Additionally, Splunk has a scheduled job dedicated to handle archiving of old events. More details about the archiving options can be found at the online documentation[6].

[6] Managing Indexers and Clusters, http://docs.splunk.com/Documentation/Splunk/6.0.2/Indexer/Setaretirementandarchivingpolicy

Whenever an event date is found older than 188697600 seconds (6 years), Splunk's housekeeping jobs set that record as frozen. That in turn triggers the execution of a procedure, which, by default, deletes the event contents. By using any text editor, the following changes are needed:

Target file: */opt/splunk/etc/system/default/indexes.conf*

Default, old value:	frozenTimePeriodInSecs = 188697600
New, suggested value:	frozenTimePeriodInSecs = 630720000

That will set event frozen time to 20 years, enough for this lab. For more details about *indexes.conf* file customization, refer to the documentation indicated previously.

After changes are done, restart Splunk so that changes take effect:

```
/opt/splunk/bin/splunk restart
```

4.1. Indexes and Event processing

Splunk is able to index any type of time-series data, hence the need for a time based field when importing files into its database. When data is indexed, Splunk breaks it into events, based on timestamps. Splunk manages the job of storing data and make it searchable via indexes. By default, data is stored in the *main* index.

An index is basically a set files and directories for storing data, as it applies to any NoSQL based technology, without relying on standard Relational Database Management Systems (RDBMS). These are located under *$SPLUNK_HOME/var/lib/splunk*. For detailed information on index storage, refer to the specific section at online documentation[7].

4.1.1. Creating an index

For the purpose of this lab, a new index will be created to store the data related to the previously crafted CSV. Simply follow the steps below to get it done:

1. Log in to Splunk web interface as admin user (default password: *changeme*);
2. Click on “Settings”, then “Indexes” from right-hand side, top menu:

[7] How Splunk Enterprise stores indexes, http://docs.splunk.com/Documentation/Splunk/6.0.2/Indexer/HowSplunkstoresindexes

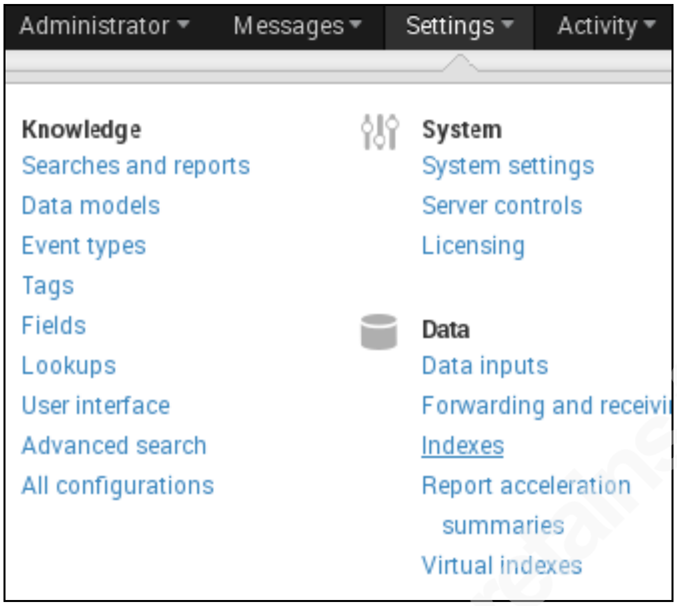


Figure 4

3. Click on "New" button to proceed;

4. On the form, type "darpa" into Index name field and then click on "Save" to finish the operation.

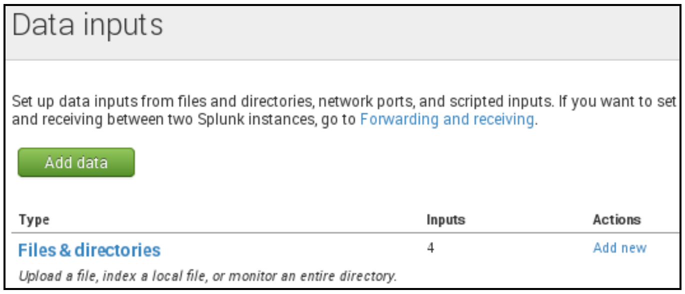
4.1.2. Indexing data

For finally importing the CSV data, the following steps should be followed:

1. Log in to Splunk web interface as admin user;

2. Click on "Settings", then "Data inputs" from right-hand side, top menu;

3. Click on "Add new" link from Actions column, on the right-hand side, from *Files & directories* as shown on Figure 5;



Data inputs

Set up data inputs from files and directories, network ports, and scripted inputs. If you want to set and receiving between two Splunk instances, go to Forwarding and receiving.

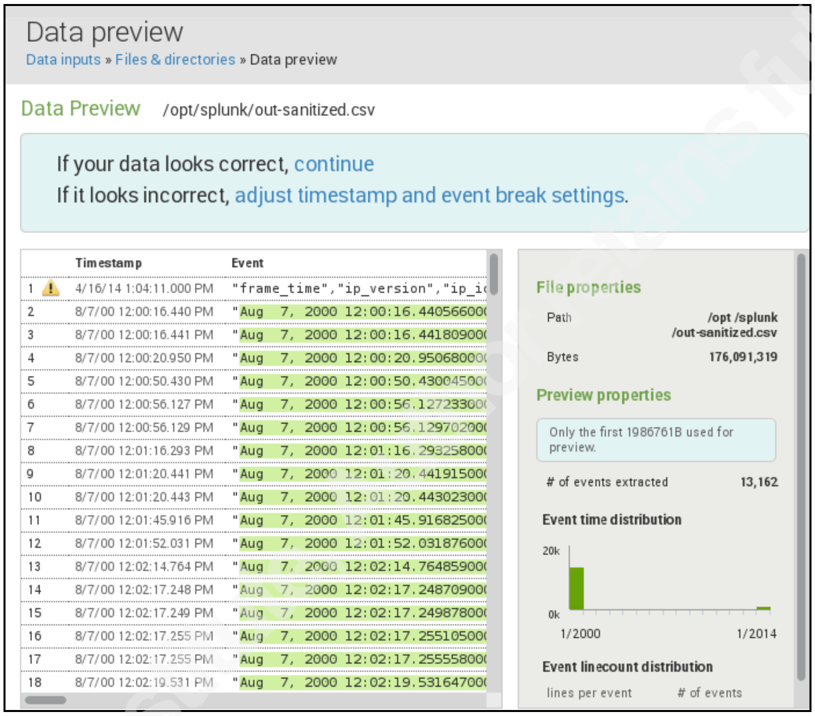
Type	Inputs	Actions
Files & directories *Upload a file, index a local file, or monitor an entire directory.*	4	Add new

Figure 5

4. Keep "Preview data before indexing" option selected, then select the CSV file named *out-sanitized.csv* (at */opt/splunk* directory). Click on "Continue" button;

5. On next window dialog, keep “Use auto-detected source type: csv” option selected and click on “Continue” button;

6. Next, from the Data Preview window, date/time values should be highlighted in green as shown below. The first line warning (header) can be ignored. Click “Continue” button to proceed;



Data preview

Data inputs » Files & directories » Data preview

Data Preview /opt/splunk/out-sanitized.csv

If your data looks correct, continue
If it looks incorrect, adjust timestamp and event break settings.

	Timestamp	Event
1 ⚠	4/16/14 1:04:11.000 PM	"frame_time","ip_version","ip_i
2	8/7/00 12:00:16.440 PM	"Aug 7, 2000 12:00:16.44056600
3	8/7/00 12:00:16.441 PM	"Aug 7, 2000 12:00:16.44180900
4	8/7/00 12:00:20.950 PM	"Aug 7, 2000 12:00:20.95068000
5	8/7/00 12:00:50.430 PM	"Aug 7, 2000 12:00:50.43004500
6	8/7/00 12:00:56.127 PM	"Aug 7, 2000 12:00:56.12723300
7	8/7/00 12:00:56.129 PM	"Aug 7, 2000 12:00:56.12970200
8	8/7/00 12:01:16.293 PM	"Aug 7, 2000 12:01:16.29325800
9	8/7/00 12:01:20.441 PM	"Aug 7, 2000 12:01:20.44191500
10	8/7/00 12:01:20.443 PM	"Aug 7, 2000 12:01:20.44302300
11	8/7/00 12:01:45.916 PM	"Aug 7, 2000 12:01:45.91682500
12	8/7/00 12:01:52.031 PM	"Aug 7, 2000 12:01:52.03187600
13	8/7/00 12:02:14.764 PM	"Aug 7, 2000 12:02:14.76485900
14	8/7/00 12:02:17.248 PM	"Aug 7, 2000 12:02:17.24870900
15	8/7/00 12:02:17.249 PM	"Aug 7, 2000 12:02:17.24987800
16	8/7/00 12:02:17.255 PM	"Aug 7, 2000 12:02:17.25510500
17	8/7/00 12:02:17.255 PM	"Aug 7, 2000 12:02:17.25555800
18	8/7/00 12:02:19.531 PM	"Aug 7, 2000 12:02:19.53164700

File properties

Path	/opt /splunk /out-sanitized.csv
Bytes	176,091,319

Preview properties

Only the first 1986761B used for preview.

# of events extracted	13,162

Event time distribution

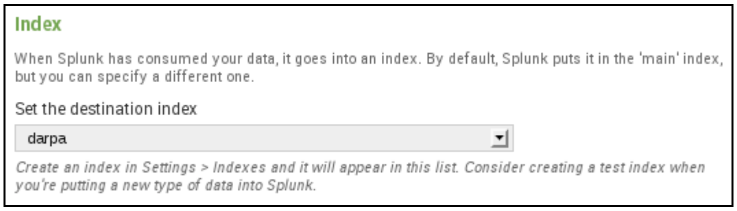
Event linecount distribution

lines per event	# of events

Figure 6

7. The last step is about indicating under which index the CSV data should be stored. Proceed as indicated below:

Select the option “Index a file once from this Splunk server” at the top, and “darpa” index at the bottom of the page.



Index

When Splunk has consumed your data, it goes into an index. By default, Splunk puts it in the 'main' index, but you can specify a different one.

Set the destination index: darpa

Create an index in Settings > Indexes and it will appear in this list. Consider creating a test index when you're putting a new type of data into Splunk.

Figure 7

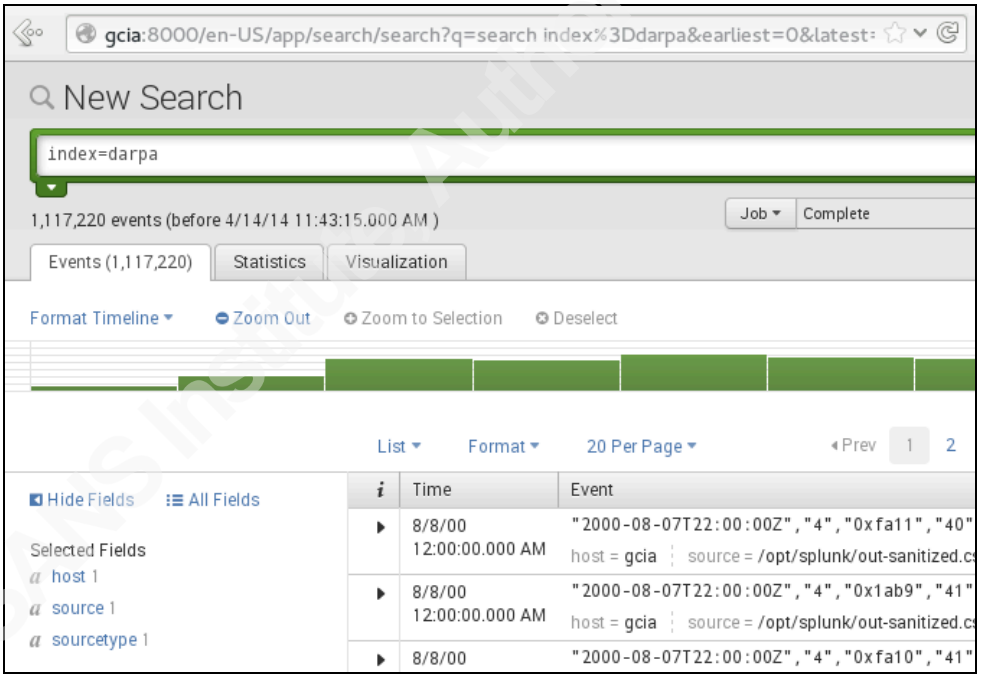
Click “Save” button to finish the operation. Next page should show the following message: *Successfully saved "/opt/splunk/out-sanitized.csv"*.

4.1.3. The first query

To verify the data was indexed correctly, a first search query is performed by following the steps below:

1. Go to the home search app location: http://gcia:8000/en-US/app/search/;
2. In the search field, type: *index=darpa*;
3. Keep "All time" drop-down menu value selected on the right-hand side;
4. Hit ENTER or click on the magnifier button.

The following results should be part of the output, indicating 1117220 events were successfully indexed by Splunk under *darpa* index, matching the number observed previously via *Wireshark* and CSV file.



```
gcia:8000/en-US/app/search/search?q=search index%3Ddarpa&earliest=0&latest:
```

New Search

```
index=darpa
```

1,117,220 events (before 4/14/14 11:43:15.000 AM)

Job ▾ Complete

Events (1,117,220) | Statistics | Visualization

i	Time	Event
▸	8/8/00 12:00:00.000 AM	"2000-08-07T22:00:00Z","4","0xfa11","40" host = gcia source = /opt/splunk/out-sanitized.cs
▸	8/8/00 12:00:00.000 AM	"2000-08-07T22:00:00Z","4","0x1ab9","41" host = gcia source = /opt/splunk/out-sanitized.cs
▸	8/8/00	"2000-08-07T22:00:00Z","4","0xfa10","41"

Figure 8

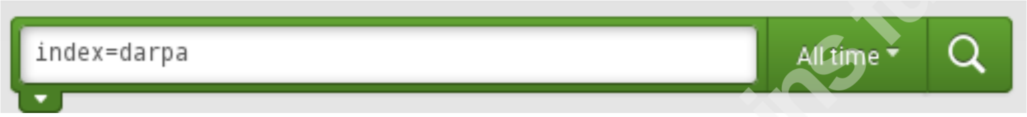
5. Data mining

Splunk offers the possibility of having distinct web accessible locations (URLs) for each application (apps), enabling specific content to be bound to a dedicated area rather than making use of the default search app. Creating distinct apps is a good approach in case multiple teams are using the system so that role based access control

(RBAC) is leveraged with more flexibility as well as keeping the content well organized. For this lab, since no searches need to be saved, makes no sense to create a different app, so default search app is used instead.

5.1. Query language basics

Splunk uses its own search processing language (SPL). Every successfully parsed field is searchable by simply providing *key=value* pairs at the search text field:



```
index=darpa
```

Figure 9

As an example, the following search query will display all events related to UDP protocol under *darpa* index:

```
index=darpa ip_proto=17
```

Similarly, for considering only UDP events originating from 192.168.1.1 IP address, the following query would accomplish the task:

```
index=darpa ip_proto=17 ip_src=192.168.1.1
```

Therefore, there is an implicit logical *AND* for evaluating search parameters, on the other hand, if a logical *OR* is the intended goal, this one must be provided explicitly as shown below:

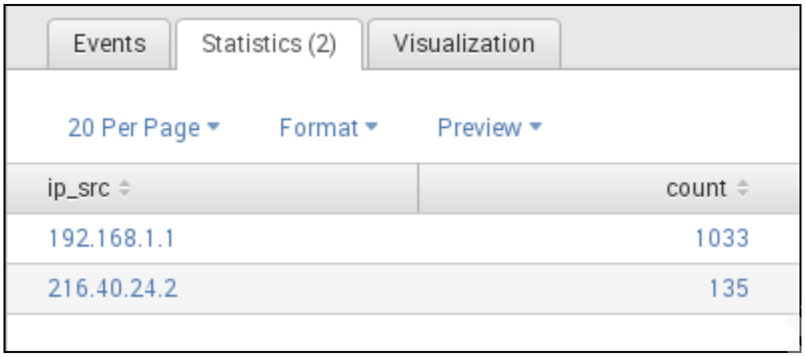
```
index=darpa (ip_proto=17 OR ip_proto=6) ip_src=192.168.1.1
```

Above query will only show events from 192.168.1.1 IP address as source, matching either UDP or TCP traffic from that host.

Moreover, query results can be later piped into additional commands. For example, the search query below will apply an additional filter, count the number of events and then render only IP source and count fields at the end:

```
index=darpa (ip_proto=17 OR ip_proto=6) | stats count by ip_src | where
(ip_src="192.168.1.1" OR ip_src="216.40.24.2") | table ip_src count
```

The result is shown on Figure 10:



ip_src	count
192.168.1.1	1033
216.40.24.2	135

Figure 10

All commands used within this document will be explained in more details later on “Lab questionnaire” section. To visualize data in different chart formats, simply click on “Visualization” tab, under the search field.

Similarly, a sort of table view is available from “Statistics” tab. Default tab “Events” shows all fields, including the raw event. Eventually, the view will automatically switch from the “Events” to “Statistics” tab according to the search criteria.

A very handy reference guide, sort of search cheat sheet, is found from Splunk website, presenting all key search commands syntax for basic querying:

http://www.splunk.com/web_assets/pdfs/secure/Splunk_Quick_Reference_Guide.pdf

5.2. Lab questionnaire

Throughout this section, some questions will be presented, considering the main goal is to highlight major Splunk data mining capabilities, with a didactic approach in mind. The questions are just high level ideas for basing a data query and introducing Splunk's search functions. More specific questions are made, as issues are raised along the exercises.

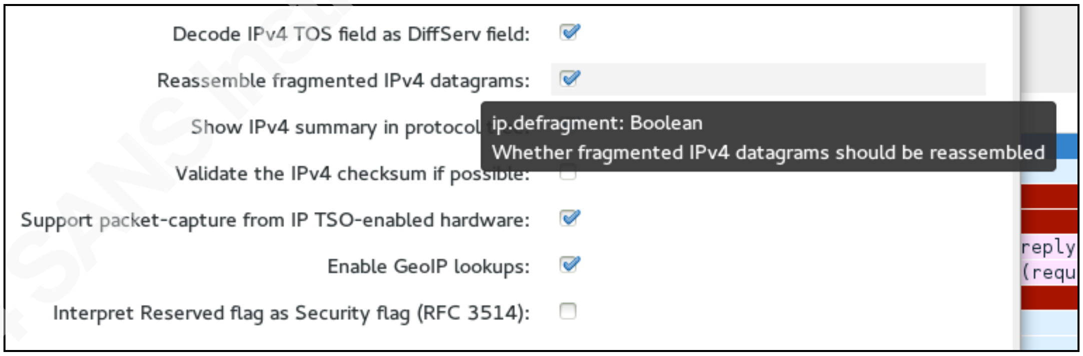
Important notes about the dataset

Understanding how an event relates to others is as important as the results interpretation. Here's a brief explanation about how the packet capture data is handled by *Wireshark*, so that event data mining is easily understood.

Unlike standard firewall logs, a pcap file usually provides packet traces regardless of the connection state or traffic direction, whether it is about an initialization packet (e.g., with SYN bit set) or about a blocked connection. Since it's out of the scope here to provide comprehensive explanation about how TCP/IP works, simply put, a TCP session is defined as a bidirectional exchange of TCP/IP packets between a pair of hosts (Sinha, 2007).

Although a TCP stream might simply refer to a sequence of bytes over a TCP session, within this document, a TCP stream corresponds to a single, unique TCP session between two hosts, similarly to how Wireshark acknowledges the same technical term. In order to reassembly TCP data streams, a lot of work is needed, starting with a protocol decoder development. Fortunately, Wireshark developers managed to have a *tcp.stream* field available from its pcap dumps, allowing easy TCP streams identification. That means whenever a TCP connection is seen, even if it's about one single packet, there should be a unique TCP stream identifier (number) for it. Also note that there is no fragmented packet data within *tshark's* dumps, hence no need to deal with fragmentation reassembly when processing data dumps.

The same applies, by default, when analyzing a pcap file with *Wireshark*, once a file is opened, no IP fragments are shown since the packets are already reassembled. Figure 12 shows the specific option from which such behavior can be changed from its defaults (enabled), available from Wireshark preferences menu:



Decode IPv4 TOS field as DiffServ field: ☑

Reassemble fragmented IPv4 datagrams: ☑

Show IPv4 summary in protocol

ip.defragment: Boolean
Whether fragmented IPv4 datagrams should be reassembled

Validate the IPv4 checksum if possible: □

Support packet-capture from IP TSO-enabled hardware: ☑

Enable GeoIP lookups: ☑

Interpret Reserved flag as Security flag (RFC 3514): □

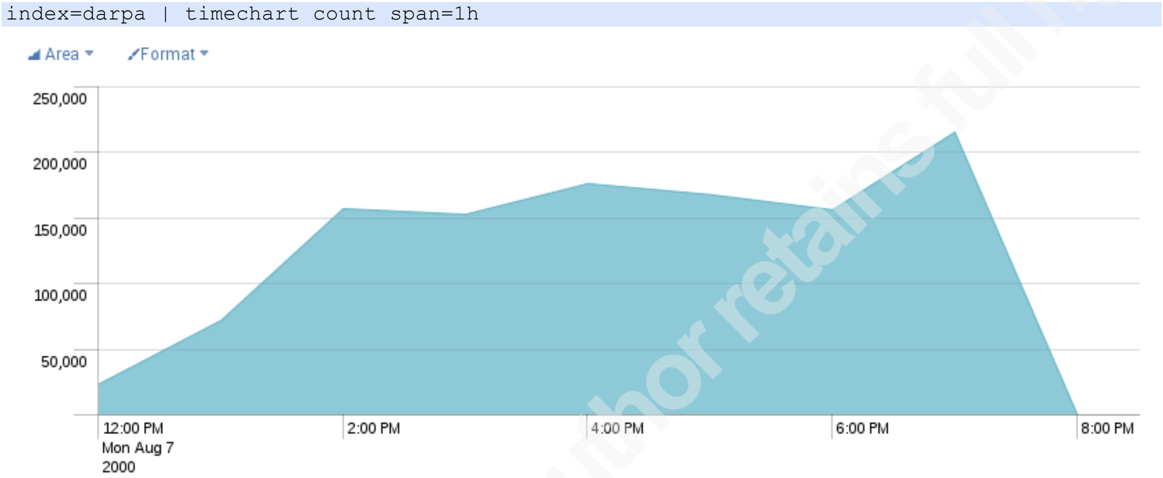
Figure 11

More specific notes and caveats are described in details along the answers below.

Note: all search queries presented within this document are considering the whole data sample, that is, the queries take all events into account. Thus, the time-picker drop-down menu located at the right-hand side of the search box is always set to "All time".

5.2.1. How many events are stored under *darpa* index? How are they distributed by protocol?

The first question was pretty much answered on the previous section, so in order to demonstrate how Splunk can leverage event timestamps, below is a simple graphical representation about how those events lie on a timeline:



```
index=darpa | timechart count span=1h
```

Figure 12 - Note that "Area" chart type is selected on top left corner.

The *timechart*[8] command enables aggregation functions to be performed on event data, similarly to standard SQL's *count*, *max* and *avg* functions. The stats are then accumulated over a time span value (e.g., 1h), as seen under X-axis.

For the second question "How are they distributed by protocol?", a pie chart is more suitable. The *top*[9] command generates two aggregated fields, by default: *count* and *percent*; providing the appropriate output structure for rendering the intended chart.

Since TCP and UDP account for most traffic volume indexed, ICMP is quite unnoticeable. Also, to make numbers available within the graph, simply make use of the *eval*[10] function for concatenating field values. Additionally, rather than showing protocol numbers, a *case* statement is added to the query for changing *ip_proto* numeric value (e.g., 17) to the corresponding string value (e.g., UDP). If the list is extensive, Splunk provides for lookup tables, which is covered in detail below. After combining all the commands mentioned, here's the query along with its output:

[8] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Timechart
[9] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Top
[10] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Eval

```
index=darpa | top ip_proto | eval ip_proto=case(ip_proto=17, "UDP", ip_proto=6, "TCP",
ip_proto=1, "ICMP") | eval ip_proto=ip_proto." (".count." events, ".round(percent,2)."%)"
```

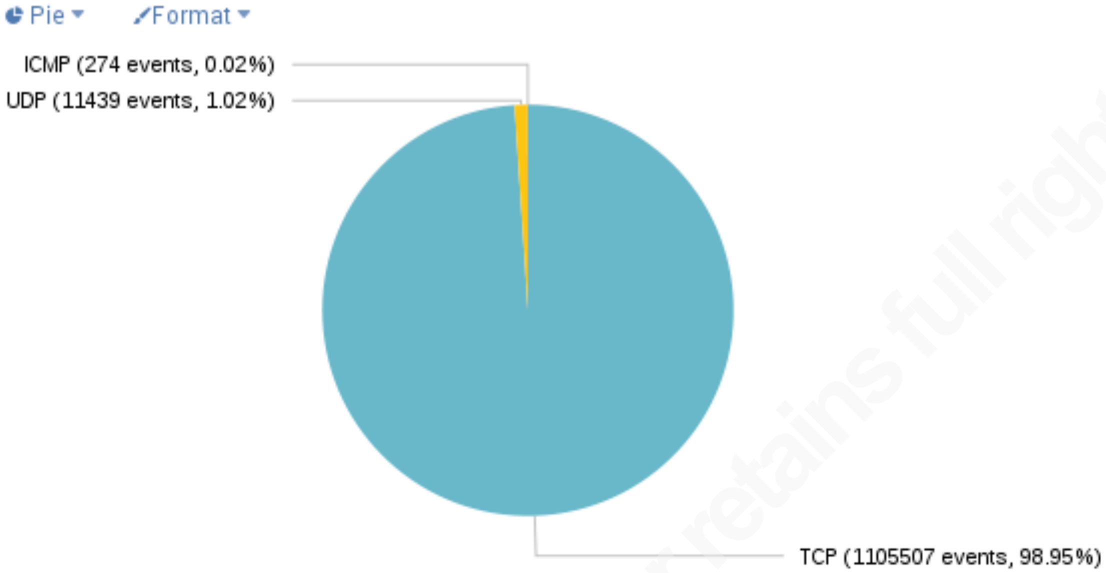


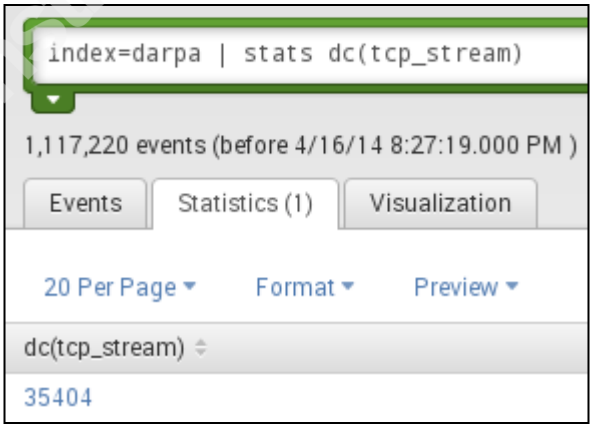
Figure 13

The *round* is part of *eval* command's functions, and is self-explanatory, in this case, rounding the value to 2 decimal places.

5.2.2. How many TCP streams are seen? How many unique hosts pairs?

To answer that, below search query is used. Output is shown on Figure 14.

```
index=darpa | stats dc(tcp_stream)
```



```
index=darpa | stats dc(tcp_stream)
```

1,117,220 events (before 4/16/14 8:27:19.000 PM)

dc(tcp_stream)
35404

Figure 14

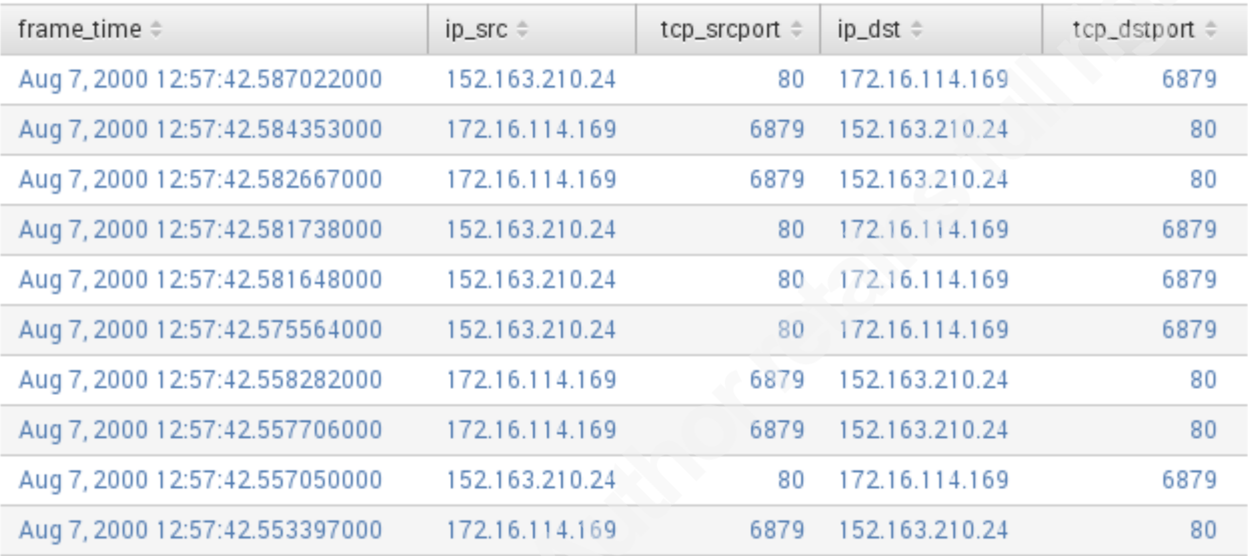
The *stats*[11] is basically used to generate statistics, one of its function is dc, which stands for distinct count. However, how to list only the very first host pair seen from each TCP stream indicating the client and server roles within the connection?

[11] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Stats

Before providing the answer, here's another quick exercise: the following query lists the events matching a specific TCP stream (666):

```
index=darpa tcp_stream=666 | table frame_time ip_src tcp_srcport ip_dst tcp_dstport
```

Output:



frame_time	ip_src	tcp_srcport	ip_dst	tcp_dstport
Aug 7, 2000 12:57:42.587022000	152.163.210.24	80	172.16.114.169	6879
Aug 7, 2000 12:57:42.584353000	172.16.114.169	6879	152.163.210.24	80
Aug 7, 2000 12:57:42.582667000	172.16.114.169	6879	152.163.210.24	80
Aug 7, 2000 12:57:42.581738000	152.163.210.24	80	172.16.114.169	6879
Aug 7, 2000 12:57:42.581648000	152.163.210.24	80	172.16.114.169	6879
Aug 7, 2000 12:57:42.575564000	152.163.210.24	80	172.16.114.169	6879
Aug 7, 2000 12:57:42.558282000	172.16.114.169	6879	152.163.210.24	80
Aug 7, 2000 12:57:42.557706000	172.16.114.169	6879	152.163.210.24	80
Aug 7, 2000 12:57:42.557050000	152.163.210.24	80	172.16.114.169	6879
Aug 7, 2000 12:57:42.553397000	172.16.114.169	6879	152.163.210.24	80

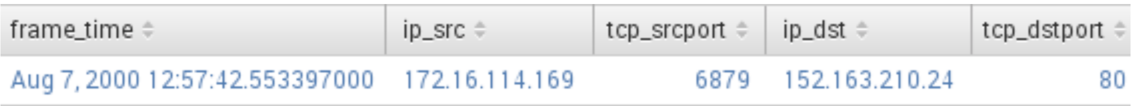
Figure 15

The *table* command enables a selection of fields to be kept in the results. It also changes the default view from “Events” to “Statistics”, which renders the results as a table. As seen on Figure 15, the first connection is made on “Aug 7, 2000 12:57:42.553397000”. Thus, the ideal query for listing all unique host pairs, considering the direction of the stream (*client* > *server*), needs to filter in only that very first connection.

In Splunk, the *dedup*[12] command provides the possibility of removing duplicate events from the output, based on its parameters. This way, the following query yields only the very first event seen within that TCP stream (666):

```
index=darpa tcp_stream=666 | dedup tcp_stream sortby frame_time | table frame_time ip_src
tcp_srcport ip_dst tcp_dstport
```

Output:



frame_time	ip_src	tcp_srcport	ip_dst	tcp_dstport
Aug 7, 2000 12:57:42.553397000	172.16.114.169	6879	152.163.210.24	80

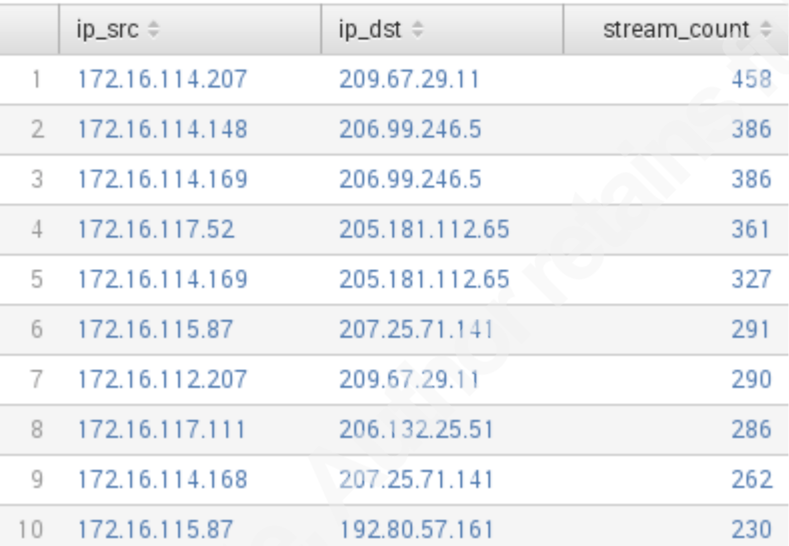
Figure 16

[12] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Dedup

In order to list all unique host pairs along with a sorted TCP streams count, assuming source IP as the client actor, i.e., the one who initiated the connection towards the destination host (server), the following query display such results:

```
index=darpa tcp_stream!="" | dedup tcp_stream sortby frame_time | stats dc(tcp_stream) AS
stream_count by ip_src, ip_dst | sort - stream_count
```

Output (first 10 rows):



	ip_src	ip_dst	stream_count
1	172.16.114.207	209.67.29.11	458
2	172.16.114.148	206.99.246.5	386
3	172.16.114.169	206.99.246.5	386
4	172.16.117.52	205.181.112.65	361
5	172.16.114.169	205.181.112.65	327
6	172.16.115.87	207.25.71.141	291
7	172.16.112.207	209.67.29.11	290
8	172.16.117.111	206.132.25.51	286
9	172.16.114.168	207.25.71.141	262
10	172.16.115.87	192.80.57.161	230

Figure 17

A full list is available by clicking at the "Export" button highlighted below:

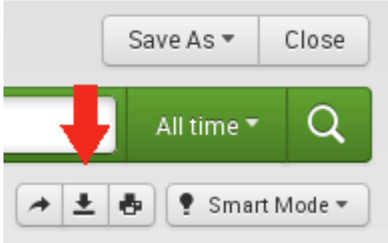
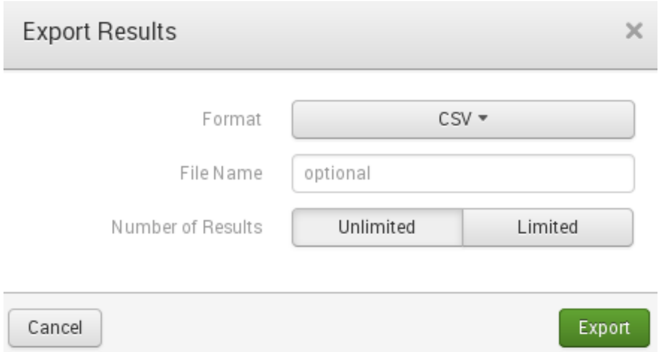


Figure 18

By selecting "Unlimited", all rows as exported, as seen below:



Export Results ×

Format: CSV ▾

File Name: optional

Number of Results: Unlimited | Limited

Cancel | Export

Figure 19

Alexandre Teixeira, forensick@ymail.com

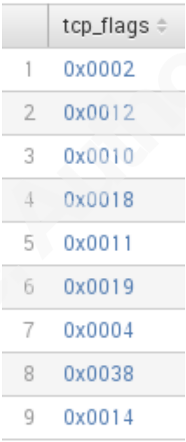
5.2.3. Which flags are set on initial connections seen from TCP streams?

A standard TCP connection establishment, known as the “three-way handshake”, is expected to be performed before actual data (payload) is transferred between hosts. Assuming scenario above, how can Splunk analytics capabilities help on verifying that? Do all initial connections carry a *SYN* bit?

Since the *tcp.flags* field was exported from the pcap via *tshark*, a *tcp_flags* corresponding field should also be available for querying at Splunk. For listing all unique flags combination seen, execute the following query:

```
index=darpa tcp_flags!="" | dedup tcp_flags | table tcp_flags
```

Output:



	tcp_flags
1	0x0002
2	0x0012
3	0x0010
4	0x0018
5	0x0011
6	0x0019
7	0x0004
8	0x0038
9	0x0014

Figure 20

These values are stored in hexadecimal, which makes it hard to understand. So after observing relevant traffic matching those on *Wireshark*, the following table displays the hex code along with a friendly value:

tcp_flags	tcp_flags_readable
0x0002	SYN
0x0012	SYN+ACK
0x0010	ACK
0x0018	PSH+ACK
0x0011	FIN+ACK
0x0019	FIN+PSH+ACK
0x0004	RST
0x0038	PSH+URG+ACK
0x0014	RST+ACK

That poses a good opportunity to introduce another Splunk feature called *Lookup tables*. The simplest way to build one is by configuring Splunk to query on a given CSV file whenever a *lookup*[13] command is executed from a search query.

Simply follow the steps below to have it set for this scenario:

1. Create a local file containing the data presented above (table) and save it *tcpflags.csv*. The contents of that file is available on appendix section;
2. Click on "Settings -> Lookups -> Lookup table files" link from top right menu;
3. Click on "New" button;
4. Keep "search" app selected, select tcpflags.csv file from "Browse..." button;
5. Fill in "Destination filename" with tcpflags.csv;
6. Click "Save" button to finish.

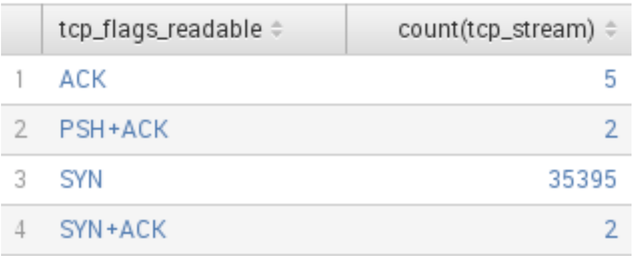
Next, a lookup definition needs to be created by following the steps below:

1. Click on "Settings -> Lookups -> Lookup definitions" link from top right menu;
2. Click on "New" button;
3. Keep pre-selected values, double-checking if "Lookup file" drop-down menu's value is *tcpflags.csv*;
4. Fill in "Name" with *tcpflags*;
5. Click "Save" button to finish.

Now back to the question: From TCP streams, do all initial connections carry a SYN flag only (SYN bit set to 1, others to 0)? The following query gives some insight about that:

```
index=darpa tcp_stream!="" | dedup tcp_stream sortby frame_time | lookup tcpflags
tcp_flags | stats count(tcp_stream) by tcp_flags_readable
```

Output:



	tcp_flags_readable	count(tcp_stream)
1	ACK	5
2	PSH+ACK	2
3	SYN	35395
4	SYN+ACK	2

Figure 21

[13] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Lookup

As seen from the first column, a new field is introduced after the CSV/lookup file is queried, performing sort of join operation on the *tcp_flags* field value, available from both the query's output and the lookup table's content.

Possible reasons for having those flag combinations (lines 1, 2 and 4 from figure above), different from a single SYN, are vast and beyond the scope of this document. But just to mention a few possible scenarios:

- The sniffer session was most likely started after the stream was actually established, thus not storing the initial packets;
- Some TCP stack implementations might behave differently from the standard protocol implementation (non RFC-compliant, etc.);
- Application code running on client/server handling the streams might behave differently or just incorrectly due to any other reason;
- Packet reassembly might get corrupted at some point, or even during *tshark's* reassembly processing.

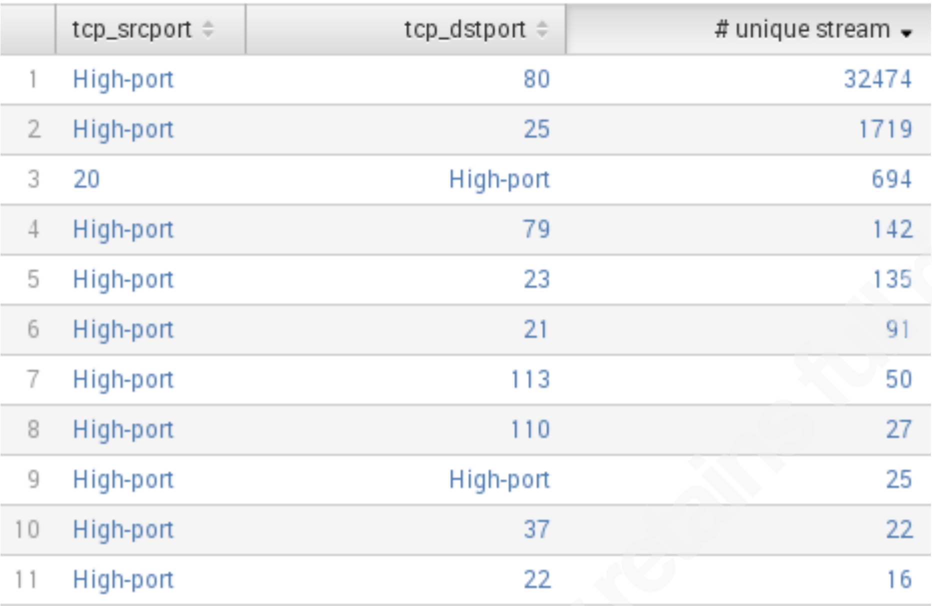
After all, that might also be the actual expected behavior, so further analysis beyond the scope would be needed to come to any conclusions here.

5.2.4. How many app-layer protocols are seen from packets containing only the SYN bit set?

A simple approach for helping determine the answer is to evaluate the port numbers values. In case the port number is within the high-port (greater than 1023) range, the port number is discarded and the number of occurrences (streams) is aggregated in a count field. This is done with the following query:

```
index=darpa tcp_stream!=""
| dedup tcp_stream sortby frame_time
| where tcp_flags="0x0002"
| eval tcp_srcport=if(tcp_srcport>1023, "High-port", tcp_srcport)
| eval tcp_dstport=if(tcp_dstport>1023, "High-port", tcp_dstport)
| stats count(tcp_stream) AS "# unique stream" by tcp_srcport, tcp_dstport
```

Output on Figure 22:



	tcp_srcport	tcp_dstport	# unique stream
1	High-port	80	32474
2	High-port	25	1719
3	20	High-port	694
4	High-port	79	142
5	High-port	23	135
6	High-port	21	91
7	High-port	113	50
8	High-port	110	27
9	High-port	High-port	25
10	High-port	37	22
11	High-port	22	16

Figure 22

Basically, from the first packet seen in a TCP stream, where only the SYN bit is set (*where tcp_flags="0x0002"*), two evaluations are performed, turning source and destination port values into the string “High-port” only if the evaluation is true.

Events are later aggregated by port pairs (source/destination). From *tcp_dstport* column values it's easy to spot well-known protocols:

- 80 = HTTP
- 25 = SMTP
- 79 = Finger
- 23 = Telnet
- 21 = FTP
- 113 = Ident
- 110 = POP3
- 37 = Time
- 22 = SSH

So, obvious next question is: what is streamed over the remaining port pairs?

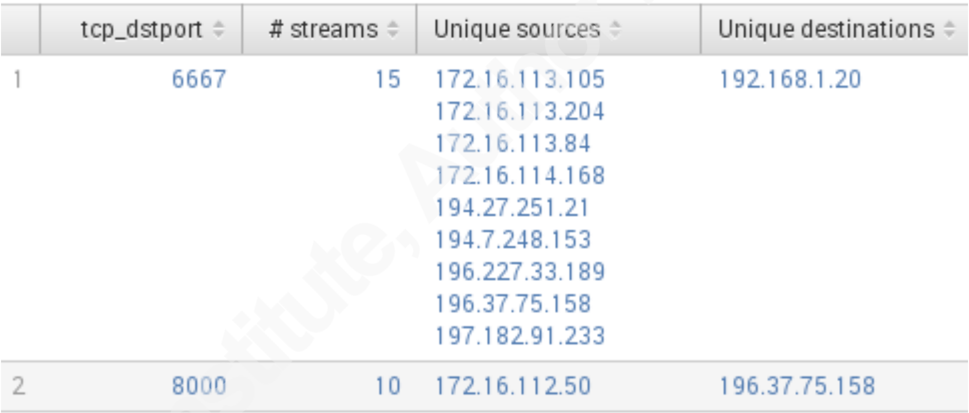
- 20 > High-port (694 unique streams)
- High-port > High-port (25 unique streams)

Firewall administrators familiarized with FTP connection issues will easily guess the former case (*tcp_srcport* = 20), since that's the standard source port related to a FTP data transfer method known as “Active FTP”[14].

To list hosts and ports associated with those 25 unique streams having both source and destination port values greater than 1023, the following query provides a better picture:

```
index=darpa tcp_stream!=""
| dedup tcp_stream sortby frame_time
| where tcp_flags="0x0002" AND tcp_srcport>1023 AND tcp_dstport>1023
| stats count(tcp_stream) AS "# streams", values(ip_src) AS "Unique sources",
values(ip_dst) AS "Unique destinations" by tcp_dstport
```

Output:



	tcp_dstport	# streams	Unique sources	Unique destinations
1	6667	15	172.16.113.105 172.16.113.204 172.16.113.84 172.16.114.168 194.27.251.21 194.7.248.153 196.227.33.189 196.37.75.158 197.182.91.233	192.168.1.20
2	8000	10	172.16.112.50	196.37.75.158

Figure 23

It turns out there are 15 streams matching a standard port (6667) related to Internet Relay Chat (IRC) traffic, and 10 streams matching a well know alternate HTTP (or web-proxy) port (8000).

The following query displays a pie chart, with values distributed by protocol:

```
index=darpa tcp_stream!=""
| dedup tcp_stream sortby frame_time
| where tcp_flags="0x0002"
| eval Type=case(
  tcp_dstport=80, "HTTP",
  tcp_dstport=25, "SMTP",
  tcp_dstport=79, "Finger",
  tcp_dstport=23, "Telnet",
```

[14] Active FTP vs. Passive FTP, a Definitive Explanation: http://slacksite.com/other/ftp.html

Alexandre Teixeira, forensick@ymail.com

```
  tcp_dstport=21, "FTP",
  tcp_dstport=113, "Ident",
  tcp_dstport=110, "POP3",
  tcp_dstport=37, "Time",
  tcp_dstport=22, "SSH",
  tcp_srcport=20 AND tcp_dstport>1023, "FTP-DATA",
  tcp_dstport=6667, "IRC",
  tcp_dstport=8000, "HTTP-8000",
  1=1, "Unknown")
| top Type limit=0 useother=f
| eval Type=Type." (".count." streams, ".round(percent,2)."%)"
```

Output:

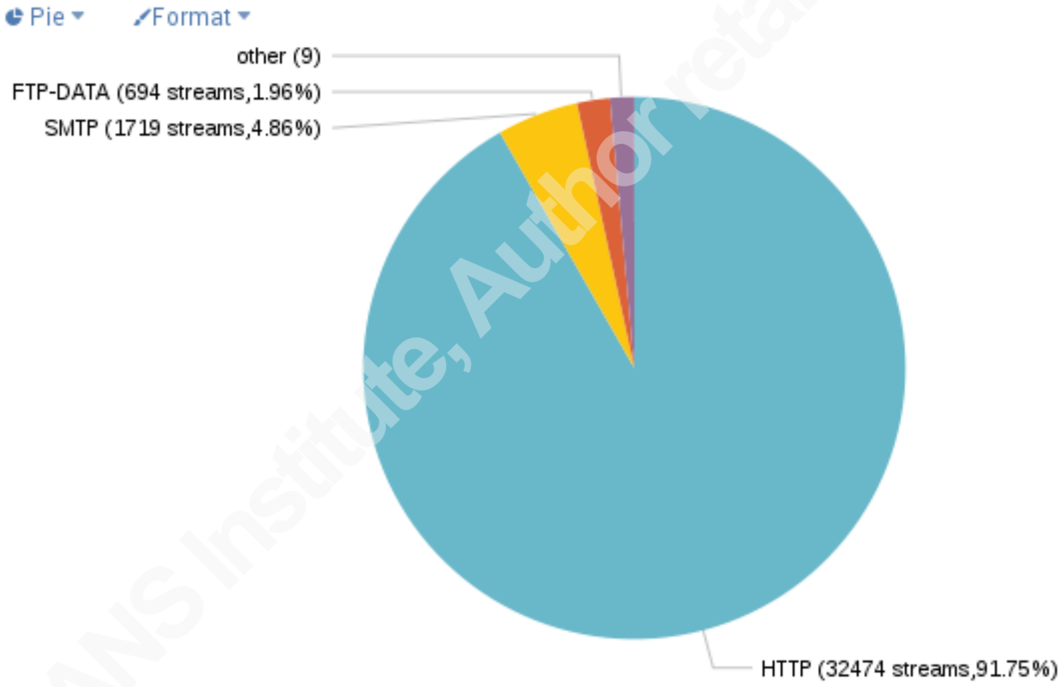


Figure 24

The long case evaluation can also be substituted by a lookup table, enabling more application protocols to be added later, without having to change search queries. Alternatively, just to provide another means to visualize data, here's a query based on a *timechart* command for displaying non-HTTP data over a 30 minutes span:

```
index=darpa tcp_stream!=""
| dedup tcp_stream sortby frame_time
| where tcp_flags="0x0002"
| eval Type=case(
  tcp_dstport=80, "HTTP",
  tcp_dstport=25, "SMTP",
  tcp_dstport=79, "Finger",
  tcp_dstport=23, "Telnet",
  tcp_dstport=21, "FTP",
  tcp_dstport=113, "Ident",
  tcp_dstport=110, "POP3",
  tcp_dstport=37, "Time",
  tcp_dstport=22, "SSH",
  tcp_srcport=20 AND tcp_dstport>1023, "FTP-DATA",
  tcp_dstport=6667, "IRC",
  tcp_dstport=8000, "HTTP-8000",
  1=1, "Unknown")
| where Type!="HTTP"
| timechart count span=30min useother=f by Type
```

Output:

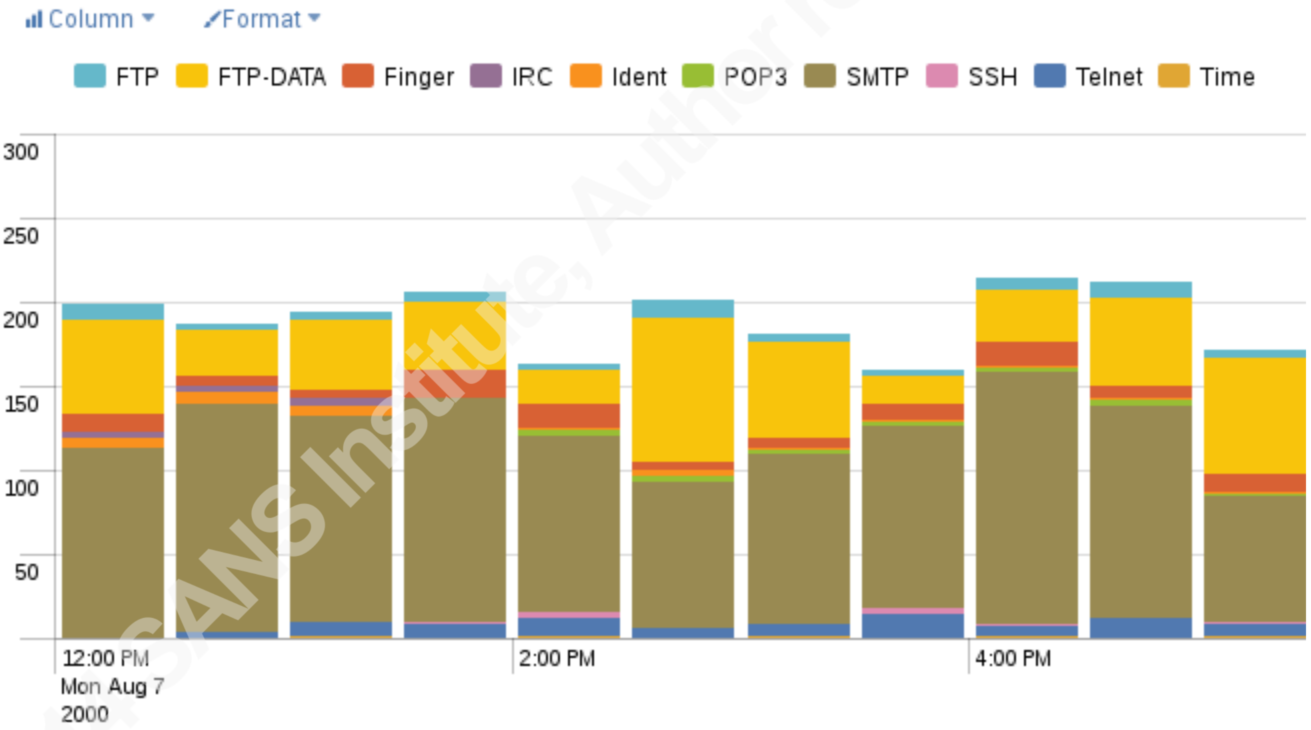


Figure 25

5.2.5. Who are the top talkers seen from the captured traffic?

Since this type of question might have multiple interpretations, for the purpose of this paper, enabling another interesting Splunk function to be demonstrated, the following objective will be approached here:

From available TCP streams (sessions), list the top 5 hosts (source/destination) accounting for most connections. However, only sessions with a verified three-way handshake established should be considered.

A standard, normal TCP connection establishment (Kozierok, 2005) is performed by following the procedure below (assuming SYN, ACK as TCP flags): the client sends a SYN message; the server sends a message that combines a ACK for the client's SYN and also contains the server's SYN; finally, the client sends an ACK for the server's SYN.

Thus, at least 3 events must exist and correlate for approaching the objective here. Thus, those events should carry a different TCP flag sequence (SYN, SYN+ACK, ACK). Splunk provides a command called *transaction*[15], which enables multiple events to be combined into one transaction, hence the name. In order to do that, a common field (or fields) with exactly the same value among the events should be specified, which becomes the transaction ID.

The following query verify the conditions above and displays the top 5 host pairs accounting for most streams established from the pcap file:

```
index=darpa tcp_stream!=""
| dedup 3 tcp_stream sortby frame_time
| lookup tcpflags tcp_flags
| transaction tcp_stream maxevents=3 mvlist=t startswith=(tcp_flags_readable="ACK")
endswith=(tcp_flags_readable="SYN")
| where mvcount(tcp_flags_readable)=3 AND mvindex(tcp_flags_readable,-2)="SYN+ACK"
| eval client=mvindex(ip_src,-1)
| eval server=mvindex(ip_dst,-1)
| eval stream=mvindex(tcp_stream,-1)
| stats dc(stream) AS count by client, server
| sort 5 - count
```

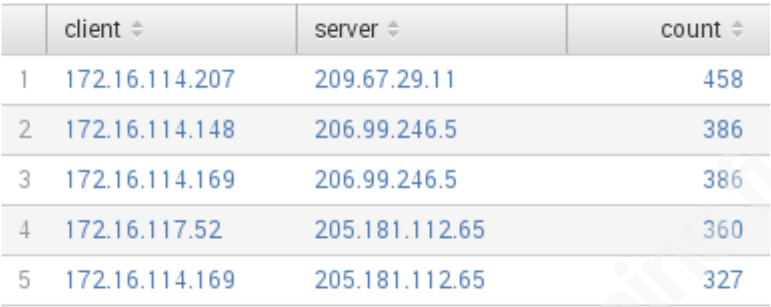
The *transaction* command parses events in reverse time order, hence the *startswith* and *endswith* parameters used in such way.

Also note that the *dedup* command will filter in only 3 connections seen with unique streams identification values, sorted by timestamp (ascending). The *transaction* command then group those in a single set, based on its criteria (parameters).

[15] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Transaction

Since each field member of a transaction is related with a multiple value, the values need to be extracted by using the *mvindex* command.

The output of above query is displayed below:



	client	server	count
1	172.16.114.207	209.67.29.11	458
2	172.16.114.148	206.99.246.5	386
3	172.16.114.169	206.99.246.5	386
4	172.16.117.52	205.181.112.65	360
5	172.16.114.169	205.181.112.65	327

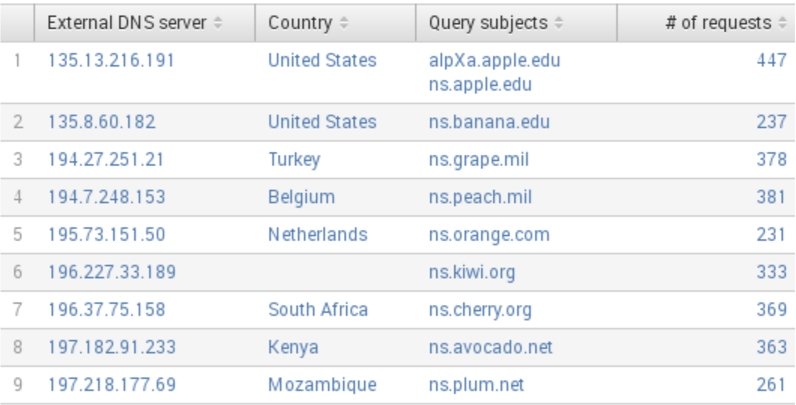
Figure 26

5.2.6. From which countries are the DNS servers' responses coming from?

To answer this question, two additional constraints are added to add a bit more of a didactic opportunity to leverage the query capabilities: only DNS requests with type A should be taken into account; only clients following private addressing scheme should be considered.

Since the *dns_qry_type* field is available from the dataset, the query should filter based on that field. By observing *Wireshark*, it's easy to spot that all type *A* requests have a value of *0x0001*. Thus, the following query should provide the expected result:

```
index=darpa dns_qry_type="0x0001" NOT (ip_src=10.0.0.0/8 OR ip_dst=172.16.0.0/12 OR
ip_dst="192.168.0.0/16")
| stats count AS "# of requests", values(dns_qry_name) AS "Query subjects" by ip_dst
| iplocation ip_dst
| table ip_dst Country "Query subjects" "# of requests"
| rename ip_dst AS "External DNS server"
```



	External DNS server	Country	Query subjects	# of requests
1	135.13.216.191	United States	alpXa.apple.edu ns.apple.edu	447
2	135.8.60.182	United States	ns.banana.edu	237
3	194.27.251.21	Turkey	ns.grape.mil	378
4	194.7.248.153	Belgium	ns.peach.mil	381
5	195.73.151.50	Netherlands	ns.orange.com	231
6	196.227.33.189		ns.kiwi.org	333
7	196.37.75.158	South Africa	ns.cherry.org	369
8	197.182.91.233	Kenya	ns.avocado.net	363
9	197.218.177.69	Mozambique	ns.plum.net	261

Figure 27

As seen from Figure 28, Splunk automatically detects if the filter value is related to a network/bit-mask combination, also known as CIDR value. Thus, IP addresses not matching those conditions are filtered out.

For quickly generating a graph about DNS requests per country with a simple modification to the previous query, simply add an additional stats command at the end of the query:

```
index=darpa dns_qry_type="0x0001" NOT (ip_src=10.0.0.0/8 OR ip_dst=172.16.0.0/12 OR
ip_dst="192.168.0.0/16")
| stats count AS "# of requests", values(dns_qry_name) AS "Query subjects" by ip_dst
| iplocation ip_dst
| table ip_dst Country "Query subjects" "# of requests"
| rename ip_dst AS "External DNS server"
| stats sum("# of requests") as count by Country
```

That should render the following chart in case a pie chart type is chosen:

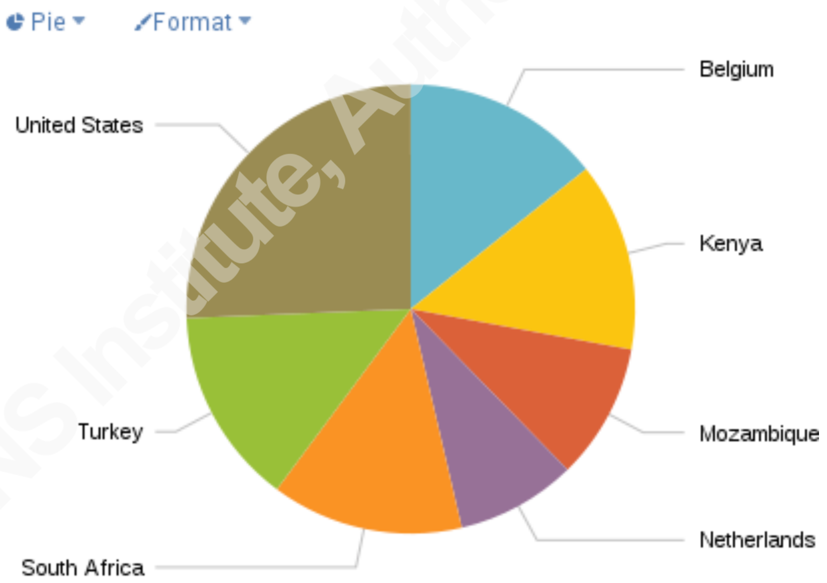


Figure 28

5.2.7. How many HTTP resources are successfully fetched along the time?

Besides *stats* and *timechart* commands, Splunk also provides the *trendline*[16] command, which enables a moving average line rendering across a timeline.

Since the *http_response_code* field is also available, the following query displays a trendline for all HTTP responses carrying the “200 OK” code:

[16] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Trendline

```
index=darpa http_response_code="200"
| timechart span=1min count AS "HTTP 200 OK"
| trendline sma60("HTTP 200 OK") AS "Simple moving avg (1h)"
| trendline ema60("HTTP 200 OK") AS "Exponential moving avg (1h)"
```

Output:

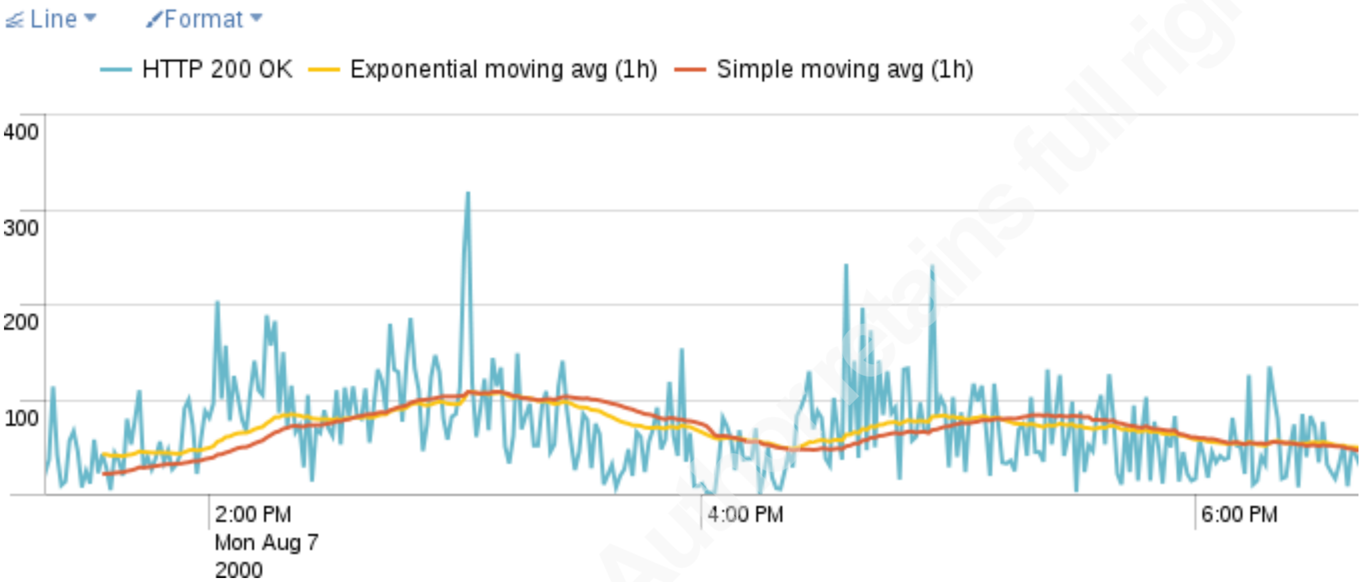


Figure 29

5.2.8. How user-agents and websites correlate with the number of HTTP requests made?

Finally, *chart*[17] is another command available from Splunk, enabling several field values to be combined into the same result set or graph, similarly to how stats works, but with more flexibility.

Here's a query example showing the top 5 websites requested over GET HTTP method along with *user-agents* values (browser string):

```
index=darpa http_host!=""
| chart count(eval(http_request_method="GET")) over http_user_agent by http_host limit=5
| addtotals
| sort - Total
| fields - Total
```

The output is shown on Figure 30.

[17] http://docs.splunk.com/Documentation/Splunk/6.0.2/SearchReference/Chart

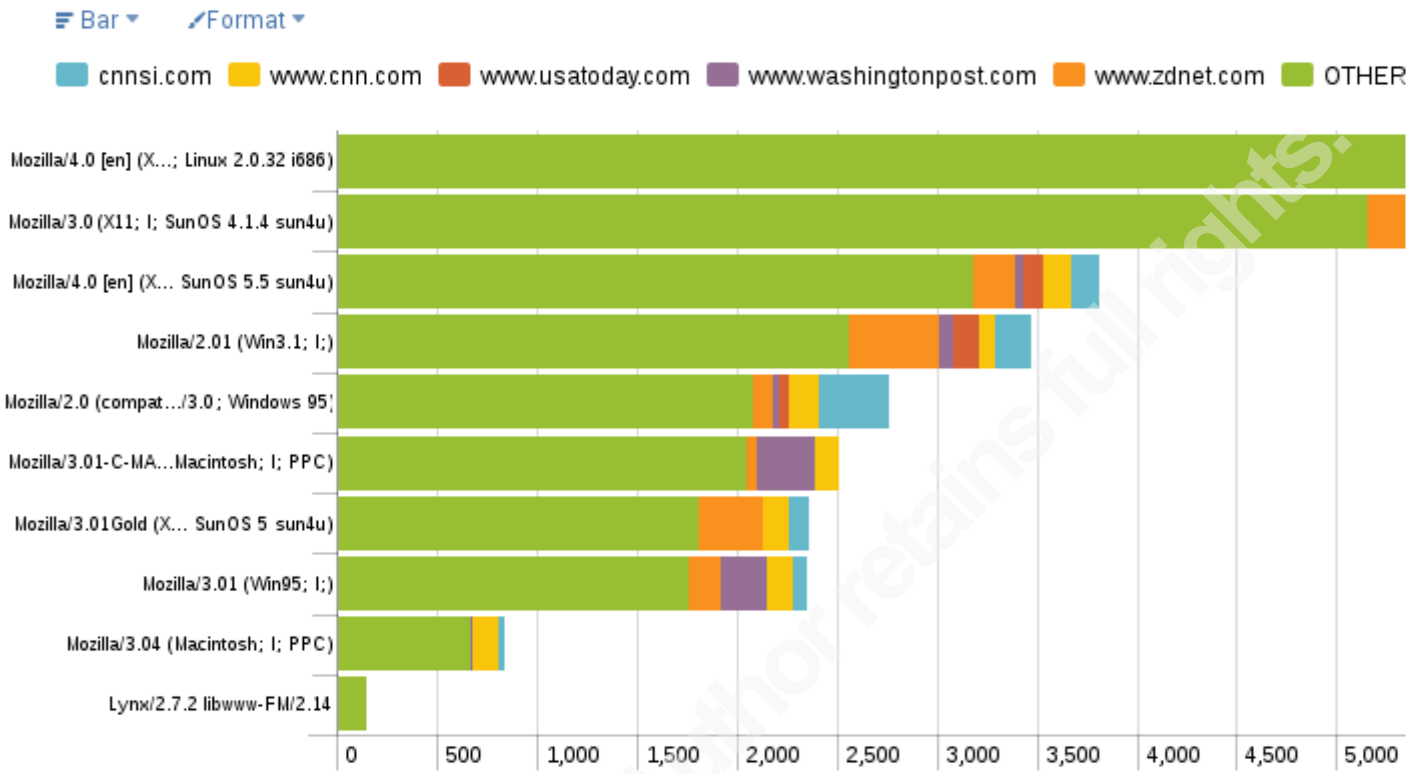


Figure 30

In this case, only events having a non-empty *http_host* (website) value are considered. Also, an evaluation (*http_request_method="GET"*) is made inside the aggregation count function so that only that specific request method is considered.

6. Appendix

Note that all commands listed below are executed as *root* user.

6.1. References

Sanders, C. (2011). Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems. : No Starch Press.

Kozierok, . M. (2005). The TCP/IP Guide: A Comprehensive, Illustrated Internet Protocols Reference. : No Starch Press.

Runkler, T. A. (2012). 1. Data Analytics: Models and Algorithms for Intelligent Data Analysis . : Vieweg+Teubner Verlag.

Mohanty, S., Jagadeesh, M., & Srivatsa, H. (2013). Big Data Imperatives: Enterprise 'Big Data' Warehouse, 'BI' Implementations and Analytics. : Apress.

Krishnan, K. (2013). Data Warehousing in the Age of Big Data. : Morgan Kaufmann.

Sinha, A. K. (2007). Methodologies and Analyses of Broadband Access Network Traffic. : ProQuest.

Security Analytics with Big Data (2014, January 19). Retrieved May 6, 2014, from https://securosis.com/assets/library/reports/SecurityAnalytics_BigData_V2.pdf

Wikipedia's Big Data entry. Retrieved April 15, 2014, from http://en.wikipedia.org/wiki/Big_data

Netresec, Publicly available PCAP files. Retrieved April 15, 2014, from http://www.netresec.com/?page=PcapFiles

DARPA Intrusion Detection Data Sets. Retrieved May 26, 2014, from http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/

Fedora Linux installation guide. (2013, January 1). . Retrieved May 15, 2014, from http://docs.fedoraproject.org/en-US/Fedora/20/html/Installation_Guide/index.html

Lamping, U., Sharpe, R., & Warnicke, E. (2013, January 1). Wireshark User's Guide. . Retrieved May 20, 2014, from http://www.wireshark.org/docs/wsug_html_chunked/

Wireshark Display Filter Reference. Retrieved May 20, 2014, from http://www.wireshark.org/docs/dfref/

Splunk Enterprise 6 documentation index. Retrieved March 18, 2014, from

http://docs.splunk.com/Documentation/Splunk/6.0.2

6.2. File: tcpflags.csv (Lookup file)

```
tcp_flags,tcp_flags_readable
0x0002,SYN
0x0012,SYN+ACK
0x0010,ACK
0x0018,PSH+ACK
0x0011,FIN+ACK
0x0019,FIN+PSH+ACK
0x0004,RST
0x0038,PSH+URG+ACK
0x0014,RST+ACK
```

6.3. Lab specs

```
[root@gcia ~]# timedatectl
      Local time: Wed 2014-04-16 08:08:21 UTC
  Universal time: Wed 2014-04-16 08:08:21 UTC
        RTC time: Wed 2014-04-16 08:08:21
        Timezone: UTC (UTC, +0000)
     NTP enabled: yes
NTP synchronized: no
 RTC in local TZ: yes
      DST active: n/a

[root@gcia ~]# cat /proc/cpuinfo
processor      : 0
vendor_id      : GenuineIntel
cpu family     : 6
model          : 23
model name     : Intel(R) Core(TM)2 Duo CPU     T6600  @ 2.20GHz
stepping       : 10
microcode      : 0xa0b
cpu MHz : 1200.000
cache size     : 2048 KB
physical id    : 0
siblings       : 2
core id : 0
cpu cores      : 2
apicid  : 0
initial apicid : 0
fpu            : yes
fpu_exception  : yes
```

```
cpuid level     : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36
clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon
pebs bts rep_good nopl aperfmperf pni dtes64 monitor ds_cpl est tm2 ssse3 cx16 xtpr pdcm
sse4_1 xsave lahf_lm dtherm
bogomips        : 4385.63
clflush size    : 64
cache_alignment        : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:

processor       : 1
vendor_id       : GenuineIntel
cpu family      : 6
model           : 23
model name      : Intel(R) Core(TM)2 Duo CPU     T6600  @ 2.20GHz
stepping        : 10
microcode       : 0xa0b
cpu MHz         : 1200.000
cache size      : 2048 KB
physical id     : 0
siblings        : 2
core id         : 1
cpu cores       : 2
apicid          : 1
initial apicid  : 1
fpu             : yes
fpu_exception   : yes
cpuid level     : 13
wp              : yes
flags           : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36
clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall nx lm constant_tsc arch_perfmon
pebs bts rep_good nopl aperfmperf pni dtes64 monitor ds_cpl est tm2 ssse3 cx16 xtpr pdcm
sse4_1 xsave lahf_lm dtherm
bogomips        : 4385.63
clflush size    : 64
cache_alignment        : 64
address sizes   : 36 bits physical, 48 bits virtual
power management:
```

```
[root@gcia ~]# free
             total       used       free     shared    buffers     cached
Mem:       3886536    3643272     243264     386400      43368    1153232
-/+ buffers/cache:    2446672    1439864
```

```
Swap:       4829180     214804    4614376
```

```
[root@gcia ~]# df -h
Filesystem                    Size  Used Avail Use% Mounted on
/dev/mapper/fedora_gcia-root  145G  6.7G  131G   5% /
devtmpfs                      1.9G     0  1.9G   0% /dev
tmpfs                         1.9G  4.5M  1.9G   1% /dev/shm
tmpfs                         1.9G  980K  1.9G   1% /run
tmpfs                         1.9G     0  1.9G   0% /sys/fs/cgroup
tmpfs                         1.9G  1.3M  1.9G   1% /tmp
/dev/sda3                     190M  133M   44M  76% /boot
```

```
[root@gcia ~]# rpm -qa | sort
abattis-cantarell-fonts-0.0.15-1.fc20.noarch
abrt-2.2.0-1.fc20.x86_64
abrt-addon-ccpp-2.2.0-1.fc20.x86_64
abrt-addon-kerneloops-2.2.0-1.fc20.x86_64
abrt-addon-pstoreoops-2.2.0-1.fc20.x86_64
abrt-addon-python-2.2.0-1.fc20.x86_64
abrt-addon-python3-2.2.0-1.fc20.x86_64
abrt-addon-vmcore-2.2.0-1.fc20.x86_64
abrt-addon-xorg-2.2.0-1.fc20.x86_64
abrt-dbus-2.2.0-1.fc20.x86_64
abrt-desktop-2.2.0-1.fc20.x86_64
abrt-gui-2.2.0-1.fc20.x86_64
abrt-gui-libs-2.2.0-1.fc20.x86_64
abrt-java-connector-1.0.9-1.fc20.x86_64
abrt-libs-2.2.0-1.fc20.x86_64
abrt-plugin-bodhi-2.2.0-1.fc20.x86_64
abrt-python-2.2.0-1.fc20.x86_64
abrt-python3-2.2.0-1.fc20.x86_64
abrt-retrace-client-2.2.0-1.fc20.x86_64
accountsservice-0.6.35-4.fc20.x86_64
accountsservice-libs-0.6.35-4.fc20.x86_64
acl-2.2.52-4.fc20.x86_64
adwaita-cursor-theme-3.10.0-1.fc20.noarch
adwaita-gtk2-theme-3.10.0-1.fc20.x86_64
adwaita-gtk3-theme-3.10.0-1.fc20.x86_64
aic94xx-firmware-30-6.fc20.noarch
alsa-firmware-1.0.27-2.fc20.noarch
alsa-lib-1.0.27.2-2.fc20.x86_64
```

```
alsa-plugins-pulseaudio-1.0.27-2.fc20.x86_64
alsa-tools-firmware-1.0.27-3.fc20.x86_64
alsa-utils-1.0.27.2-4.fc20.x86_64
anaconda-20.25.16-1.fc20.x86_64
anaconda-widgets-20.25.16-1.fc20.x86_64
anaconda-yum-plugins-1.0-10.fc20.noarch
argyllcms-1.6.3-1.fc20.x86_64
at-3.1.13-14.fc20.x86_64
atk-2.10.0-1.fc20.x86_64
atkmm-2.22.7-2.fc20.x86_64
atmel-firmware-1.3-12.fc20.noarch
at-spi2-atk-2.10.2-1.fc20.x86_64
at-spi2-core-2.10.2-1.fc20.x86_64
attr-2.4.47-3.fc20.x86_64
audit-2.3.5-1.fc20.x86_64
audit-libs-2.3.5-1.fc20.x86_64
audit-libs-python-2.3.5-1.fc20.x86_64
augeas-libs-1.2.0-2.fc20.x86_64
authconfig-6.2.6-4.fc20.x86_64
autocorr-en-4.2.3.2-3.fc20.noarch
avahi-0.6.31-21.fc20.x86_64
avahi-autoipd-0.6.31-21.fc20.x86_64
avahi-glib-0.6.31-21.fc20.x86_64
avahi-gobject-0.6.31-21.fc20.x86_64
avahi-libs-0.6.31-21.fc20.x86_64
avahi-ui-gtk3-0.6.31-21.fc20.x86_64
b43-fwcutter-017-3.fc20.x86_64
b43-openfwwf-5.2-10.fc20.noarch
baobab-3.10.1-1.fc20.x86_64
basesystem-10.0-9.fc20.noarch
bash-4.2.46-4.fc20.x86_64
bash-completion-2.1-3.fc20.noarch
bc-1.06.95-10.fc20.x86_64
bind-libs-9.9.4-12.P2.fc20.x86_64
bind-libs-lite-9.9.4-12.P2.fc20.x86_64
bind-license-9.9.4-12.P2.fc20.noarch
bind-utils-9.9.4-12.P2.fc20.x86_64
binutils-2.23.88.0.1-13.fc20.x86_64
biosdevname-0.5.0-2.fc20.x86_64
bluez-5.17-1.fc20.x86_64
bluez-cups-5.17-1.fc20.x86_64
boost-date-time-1.54.0-9.fc20.x86_64
boost-system-1.54.0-9.fc20.x86_64
```

```
boost-thread-1.54.0-9.fc20.x86_64
brasero-libs-3.10.0-1.fc20.x86_64
bridge-utils-1.5-8.fc20.x86_64
brlapi-0.6.0-9.fc20.x86_64
brltty-4.5-9.fc20.x86_64
btrfs-progs-3.12-1.fc20.x86_64
bzip2-1.0.6-9.fc20.x86_64
bzip2-libs-1.0.6-9.fc20.x86_64
ca-certificates-2013.1.97-1.fc20.noarch
cairo-1.13.1-0.1.git337ab1f.fc20.x86_64
cairo-gobject-1.13.1-0.1.git337ab1f.fc20.x86_64
cairomm-1.10.0-7.fc20.x86_64
c-ares-1.10.0-2.fc20.x86_64
caribou-0.4.13-1.fc20.x86_64
caribou-gtk2-module-0.4.13-1.fc20.x86_64
caribou-gtk3-module-0.4.13-1.fc20.x86_64
cdparanoia-libs-10.2-14.fc20.x86_64
celt-0.11.3-1.fc20.x86_64
celt051-0.5.1.3-7.fc20.x86_64
ceph-libs-0.67.3-2.fc20.x86_64
checkpolicy-2.1.12-5.fc20.x86_64
cheese-3.10.2-1.fc20.x86_64
cheese-libs-3.10.2-1.fc20.x86_64
chkconfig-1.3.60-4.fc20.x86_64
chrony-1.29.1-1.fc20.x86_64
cifs-utils-6.3-1.fc20.x86_64
cjkuni-uming-fonts-0.2.20080216.1-53.fc20.noarch
clucene-contribs-lib-2.3.3.4-10.fc20.x86_64
clucene-core-2.3.3.4-10.fc20.x86_64
clutter-1.16.2-4.fc20.x86_64
clutter-gst2-2.0.10-1.fc20.x86_64
clutter-gtk-1.4.4-3.fc20.x86_64
cogl-1.16.0-2.fc20.x86_64
colord-1.1.5-1.fc20.x86_64
colord-gtk-0.1.25-2.fc20.x86_64
colord-libs-1.1.5-1.fc20.x86_64
color-filesystem-1-13.fc20.noarch
comps-extras-23-1.fc20.noarch
control-center-3.10.3-1.fc20.x86_64
control-center-filesystem-3.10.3-1.fc20.x86_64
coreutils-8.21-21.fc20.x86_64
cpio-2.11-24.fc20.x86_64
cracklib-2.9.0-5.fc20.x86_64
```

```
cracklib-dicts-2.9.0-5.fc20.x86_64
crash-7.0.3-1.fc20.x86_64
crda-1.1.3_2013.11.27-4.fc20.x86_64
cronie-1.4.11-4.fc20.x86_64
cronie-anacron-1.4.11-4.fc20.x86_64
crontabs-1.11-7.20130830git.fc20.noarch
cryptopp-5.6.2-3.fc20.x86_64
cryptsetup-1.6.4-1.fc20.x86_64
cryptsetup-libs-1.6.4-1.fc20.x86_64
cryptsetup-python-1.6.4-1.fc20.x86_64
cups-1.7.1-8.fc20.x86_64
cups-filesystem-1.7.1-8.fc20.noarch
cups-filters-1.0.41-5.fc20.x86_64
cups-filters-libs-1.0.41-5.fc20.x86_64
cups-libs-1.7.1-8.fc20.x86_64
cups-pk-helper-0.2.5-2.fc20.x86_64
curl-7.32.0-8.fc20.x86_64
cyrus-sasl-gssapi-2.1.26-14.fc20.x86_64
cyrus-sasl-lib-2.1.26-14.fc20.x86_64
cyrus-sasl-md5-2.1.26-14.fc20.x86_64
cyrus-sasl-plain-2.1.26-14.fc20.x86_64
cyrus-sasl-scram-2.1.26-14.fc20.x86_64
dbus-1.6.12-8.fc20.x86_64
dbus-glib-0.100.2-2.fc20.x86_64
dbus-libs-1.6.12-8.fc20.x86_64
dbus-python-1.2.0-1.fc20.x86_64
dbus-x11-1.6.12-8.fc20.x86_64
dconf-0.18.0-2.fc20.x86_64
dejavu-fonts-common-2.34-1.fc20.noarch
dejavu-sans-fonts-2.34-1.fc20.noarch
dejavu-sans-mono-fonts-2.34-1.fc20.noarch
dejavu-serif-fonts-2.34-1.fc20.noarch
deltarpm-3.6-3.fc20.x86_64
desktop-backgrounds-gnome-20.0.0-1.fc20.noarch
desktop-file-utils-0.22-1.fc20.x86_64
device-mapper-1.02.82-5.fc20.x86_64
device-mapper-event-1.02.82-5.fc20.x86_64
device-mapper-event-libs-1.02.82-5.fc20.x86_64
device-mapper-libs-1.02.82-5.fc20.x86_64
device-mapper-multipath-0.4.9-56.fc20.x86_64
device-mapper-multipath-libs-0.4.9-56.fc20.x86_64
device-mapper-persistent-data-0.2.8-1.fc20.x86_64
dhclient-4.2.6-1.fc20.x86_64
```

```
dhcp-common-4.2.6-1.fc20.x86_64
dhcp-libs-4.2.6-1.fc20.x86_64
diffutils-3.3-4.fc20.x86_64
dmidecode-2.12-4.fc20.x86_64
dmraid-1.0.0.rc16-23.fc20.x86_64
dmraid-events-1.0.0.rc16-23.fc20.x86_64
dnf-0.4.20-1.fc20.noarch
dnsmasq-2.68-1.fc20.x86_64
dosfstools-3.0.26-1.fc20.x86_64
dotconf-1.3-7.fc20.x86_64
dracut-037-10.git20140402.fc20.x86_64
dracut-config-rescue-037-10.git20140402.fc20.x86_64
dracut-network-037-10.git20140402.fc20.x86_64
e2fsprogs-1.42.8-3.fc20.x86_64
e2fsprogs-libs-1.42.8-3.fc20.x86_64
ebtables-2.0.10-12.fc20.x86_64
ed-1.10-1.fc20.x86_64
efibootmgr-0.5.4-16.fc20.x86_64
elfutils-0.158-2.fc20.x86_64
elfutils-libelf-0.158-2.fc20.x86_64
elfutils-libs-0.158-2.fc20.x86_64
emacs-filesystem-24.3-13.fc20.noarch
empathy-3.10.3-1.fc20.x86_64
enca-1.14-2.fc20.x86_64
enchant-1.6.0-7.fc20.x86_64
espeak-1.47.11-3.fc20.x86_64
ethtool-3.10-2.fc20.x86_64
evince-3.10.3-1.fc20.x86_64
evince-libs-3.10.3-1.fc20.x86_64
evince-nautilus-3.10.3-1.fc20.x86_64
evolution-3.10.4-2.fc20.x86_64
evolution-data-server-3.10.4-3.fc20.x86_64
evolution-ews-3.10.4-1.fc20.x86_64
exempi-2.2.1-3.fc20.x86_64
exiv2-libs-0.23-5.fc20.x86_64
expat-2.1.0-7.fc20.x86_64
farstream-0.1.2-6.fc20.x86_64
farstream02-0.2.3-3.fc20.x86_64
fcoe-utils-1.0.29-1.fc20.x86_64
fedora-bookmarks-15-4.fc20.noarch
fedora-logos-21.0.1-1.fc20.x86_64
fedora-release-20-3.noarch
fedora-release-notes-20-0.9.noarch
```

```
festival-1.96-26.fc20.x86_64
festival-freebsoft-utils-0.10-7.fc20.noarch
festival-lib-1.96-26.fc20.x86_64
festival-speechtools-libs-1.2.96-26.fc20.x86_64
festvox-slt-arctic-hts-0.20061229-26.fc20.noarch
fftw-libs-double-3.3.4-1.fc20.x86_64
file-5.14-20.fc20.x86_64
file-libs-5.14-20.fc20.x86_64
file-roller-3.10.2.1-1.fc20.x86_64
file-roller-nautilus-3.10.2.1-1.fc20.x86_64
filesystem-3.2-19.fc20.x86_64
findutils-4.5.11-4.fc20.x86_64
fipscheck-1.4.1-2.fc20.x86_64
fipscheck-lib-1.4.1-2.fc20.x86_64
firebird-libfbembed-2.5.2.26539.0-8.fc20.x86_64
firefox-28.0-3.fc20.x86_64
firewall-config-0.3.9.3-1.fc20.noarch
firewalld-0.3.9.3-1.fc20.noarch
flac-libs-1.3.0-3.fc20.x86_64
flite-1.3-21.fc20.x86_64
folks-0.9.6-2.fc20.x86_64
fontconfig-2.11.0-1.fc20.x86_64
fontpackages-filesystem-1.44-9.fc20.noarch
foomatic-filters-4.0.9-6.fc20.x86_64
fpaste-0.3.7.1-12.fc20.noarch
fprintd-0.5.1-1.fc20.x86_64
fprintd-pam-0.5.1-1.fc20.x86_64
freerdp-1.0.2-6.fc20.x86_64
freerdp-libs-1.0.2-6.fc20.x86_64
freerdp-plugins-1.0.2-6.fc20.x86_64
freetype-2.5.0-5.fc20.x86_64
frei0r-plugins-1.4-1.fc20.x86_64
fros-1.0-4.fc20.noarch
fuse-2.9.3-2.fc20.x86_64
fuse-libs-2.9.3-2.fc20.x86_64
fxload-2002_04_11-14.fc20.x86_64
gavl-1.4.0-4.fc20.x86_64
gawk-4.1.0-2.fc20.x86_64
GConf2-3.2.6-7.fc20.x86_64
gcr-3.10.1-1.fc20.x86_64
gd-2.1.0-3.fc20.x86_64
gdb-7.6.50.20130731-19.fc20.x86_64
gdbm-1.10-7.fc20.x86_64
```

```
gdisk-0.8.10-2.fc20.x86_64
gdk-pixbuf2-2.30.3-1.fc20.x86_64
gdm-3.10.0.1-1.fc20.x86_64
gdm-libs-3.10.0.1-1.fc20.x86_64
gedit-3.10.3-1.fc20.x86_64
geoclue-0.12.99-5.fc20.x86_64
geoclue2-2.0.0-1.fc20.x86_64
geocode-glib-3.10.0-1.fc20.x86_64
GeoIP-1.5.1-3.fc20.x86_64
gettext-0.18.3.1-1.fc20.x86_64
gettext-libs-0.18.3.1-1.fc20.x86_64
gfs2-utils-3.1.6-4.fc20.x86_64
ghostscript-9.10-5.fc20.x86_64
ghostscript-fonts-5.50-32.fc20.noarch
giflib-4.1.6-9.fc20.x86_64
gjs-1.38.1-1.fc20.x86_64
glade-libs-3.16.1-1.fc20.x86_64
glib2-2.38.2-2.fc20.x86_64
glibc-2.18-12.fc20.x86_64
glibc-common-2.18-12.fc20.x86_64
glibmm24-2.38.1-1.fc20.x86_64
glib-networking-2.38.2-1.fc20.x86_64
glusterfs-3.4.2-1.fc20.x86_64
glusterfs-api-3.4.2-1.fc20.x86_64
glusterfs-libs-3.4.2-1.fc20.x86_64
glx-utils-8.1.0-4.fc20.x86_64
gmime-2.6.19-1.fc20.x86_64
gmp-5.1.2-2.fc20.x86_64
gnome-abrt-0.3.6-1.fc20.x86_64
gnome-backgrounds-3.10.1-1.fc20.noarch
gnome-bluetooth-3.10.0-1.fc20.x86_64
gnome-bluetooth-libs-3.10.0-1.fc20.x86_64
gnome-calculator-3.10.2-1.fc20.x86_64
gnome-clocks-3.10.1-1.fc20.x86_64
gnome-color-manager-3.10.1-2.fc20.x86_64
gnome-contacts-3.10.1-1.fc20.x86_64
gnome-desktop3-3.10.2-2.fc20.x86_64
gnome-disk-utility-3.10.0-2.fc20.x86_64
gnome-documents-3.10.2-1.fc20.x86_64
gnome-epub-thumbnailer-1.3-4.fc20.x86_64
gnome-font-viewer-3.10.0-1.fc20.x86_64
gnome-getting-started-docs-3.10.2-1.fc20.noarch
gnome-icon-theme-3.10.0-1.fc20.noarch
```

```
gnome-icon-theme-extras-3.6.2-3.fc20.noarch
gnome-icon-theme-legacy-3.10.0-1.fc20.noarch
gnome-icon-theme-symbolic-3.10.1-1.fc20.noarch
gnome-initial-setup-3.10.1.1-4.fc20.x86_64
gnome-js-common-0.1.2-8.fc20.noarch
gnome-keyring-3.10.1-1.fc20.x86_64
gnome-keyring-pam-3.10.1-1.fc20.x86_64
gnome-menus-3.10.1-1.fc20.x86_64
gnome-online-accounts-3.10.3-1.fc20.x86_64
gnome-online-miners-3.10.3-1.fc20.x86_64
gnome-packagekit-3.10.1-1.fc20.x86_64
gnome-screenshot-3.10.1-1.fc20.x86_64
gnome-session-3.10.1-1.fc20.x86_64
gnome-session-xsession-3.10.1-1.fc20.x86_64
gnome-settings-daemon-3.10.2-3.fc20.x86_64
gnome-settings-daemon-updates-3.10.2-3.fc20.x86_64
gnome-shell-3.10.4-2.fc20.x86_64
gnome-software-3.10.4-2.fc20.x86_64
gnome-system-monitor-3.10.2-1.fc20.x86_64
gnome-terminal-3.10.2-1.fc20.x86_64
gnome-themes-standard-3.10.0-1.fc20.x86_64
gnome-user-docs-3.10.2-1.fc20.noarch
gnome-video-effects-0.4.0-6.fc20.noarch
gnupg2-2.0.22-1.fc20.x86_64
gnutls-3.1.20-4.fc20.x86_64
gobject-introspection-1.38.0-1.fc20.x86_64
google-crosextra-caladea-fonts-1.002-0.3.20130214.fc20.noarch
google-crosextra-carlito-fonts-1.103-0.1.20130920.fc20.noarch
google-noto-sans-lisu-fonts-20130807-1.fc20.noarch
google-noto-sans-mandaic-fonts-20130807-1.fc20.noarch
google-noto-sans-meeteimayek-fonts-20130807-1.fc20.noarch
google-noto-sans-tagalog-fonts-20130807-1.fc20.noarch
google-noto-sans-tai-tham-fonts-20130807-1.fc20.noarch
google-noto-sans-tai-viet-fonts-20130807-1.fc20.noarch
gpgme-1.3.2-4.fc20.x86_64
gpg-pubkey-246110c1-51954fca
gpm-libs-1.20.7-3.fc20.x86_64
graphite2-1.2.2-4.fc20.x86_64
grep-2.18-1.fc20.x86_64
grilo-0.2.9-2.fc20.x86_64
grilo-plugins-0.2.9-2.fc20.x86_64
groff-base-1.22.2-8.fc20.x86_64
grub2-2.00-25.fc20.x86_64
```

```
grub2-efi-2.00-25.fc20.x86_64
grub2-tools-2.00-25.fc20.x86_64
grubby-8.28-1.fc20.x86_64
gsettings-desktop-schemas-3.10.1-1.fc20.x86_64
gsm-1.0.13-10.fc20.x86_64
gssdp-0.14.7-1.fc20.x86_64
gstreamer-0.10.36-6.fc20.x86_64
gstreamer1-1.2.3-1.fc20.x86_64
gstreamer1-plugins-bad-free-1.2.3-1.fc20.x86_64
gstreamer1-plugins-base-1.2.3-1.fc20.x86_64
gstreamer1-plugins-good-1.2.3-2.fc20.x86_64
gstreamer-plugins-bad-free-0.10.23-20.fc20.x86_64
gstreamer-plugins-base-0.10.36-6.fc20.x86_64
gstreamer-plugins-espeak-0.4.0-2.fc19.x86_64
gstreamer-plugins-good-0.10.31-10.fc20.x86_64
gstreamer-tools-0.10.36-6.fc20.x86_64
gtk2-2.24.22-2.fc20.x86_64
gtk2-immodule-xim-2.24.22-2.fc20.x86_64
gtk3-3.10.7-1.fc20.x86_64
gtk3-immodule-xim-3.10.7-1.fc20.x86_64
gtkhtml3-4.6.6-2.fc20.x86_64
gtkmm24-2.24.4-2.fc20.x86_64
gtkmm30-3.10.1-1.fc20.x86_64
gtksourceview3-3.10.2-1.fc20.x86_64
gtk-vnc2-0.5.3-1.fc20.x86_64
gucharmap-3.10.1-1.fc20.x86_64
gupnp-0.20.10-1.fc20.x86_64
gupnp-av-0.12.5-1.fc20.x86_64
gupnp-dlna-0.10.2-2.fc20.x86_64
gupnp-igd-0.2.3-1.fc20.x86_64
gutenprint-5.2.9-14.fc20.x86_64
gutenprint-cups-5.2.9-14.fc20.x86_64
gvfs-1.18.3-2.fc20.x86_64
gvfs-afc-1.18.3-2.fc20.x86_64
gvfs-afp-1.18.3-2.fc20.x86_64
gvfs-archive-1.18.3-2.fc20.x86_64
gvfs-fuse-1.18.3-2.fc20.x86_64
gvfs-goa-1.18.3-2.fc20.x86_64
gvfs-gphoto2-1.18.3-2.fc20.x86_64
gvfs-mtp-1.18.3-2.fc20.x86_64
gvfs-smb-1.18.3-2.fc20.x86_64
gvnc-0.5.3-1.fc20.x86_64
gzip-1.6-2.fc20.x86_64
```

```
hardlink-1.0-18.fc20.x86_64
harfbuzz-0.9.26-1.fc20.x86_64
harfbuzz-icu-0.9.26-1.fc20.x86_64
hawkey-0.4.12-1.fc20.x86_64
heisenbug-backgrounds-base-20.0.0-1.fc20.noarch
heisenbug-backgrounds-gnome-20.0.0-1.fc20.noarch
hfsplus-tools-540.1.linux3-4.fc20.x86_64
hicolor-icon-theme-0.13-1.fc20.noarch
highlight-3.16.1-2.fc20.x86_64
hostname-3.13-2.fc20.x86_64
hplip-common-3.13.11-4.fc20.x86_64
hplip-libs-3.13.11-4.fc20.x86_64
hunspell-1.3.2-14.fc20.x86_64
hunspell-en-US-0.20121024-6.fc20.noarch
hwdata-0.261-1.fc20.noarch
hyphen-2.8.6-4.fc20.x86_64
hyphen-en-2.8.6-4.fc20.noarch
ibus-1.5.6-2.fc20.x86_64
ibus-chewing-1.4.10.1-1.fc20.x86_64
ibus-gtk2-1.5.6-2.fc20.x86_64
ibus-gtk3-1.5.6-2.fc20.x86_64
ibus-hangul-1.4.2-7.fc20.x86_64
ibus-kkc-1.5.20-1.fc20.x86_64
ibus-libpinyin-1.6.92-2.fc20.x86_64
ibus-libs-1.5.6-2.fc20.x86_64
ibus-m17n-1.3.4-12.fc20.x86_64
ibus-rawcode-1.3.2-3.fc20.x86_64
ibus-setup-1.5.6-2.fc20.noarch
ibus-wayland-1.5.6-2.fc20.x86_64
info-5.1-4.fc20.x86_64
initscripts-9.51-2.fc20.x86_64
iproute-3.12.0-2.fc20.x86_64
iptables-1.4.19.1-1.fc20.x86_64
iptstate-2.2.5-3.fc20.x86_64
iputils-20121221-5.fc20.x86_64
ipw2100-firmware-1.3-16.fc20.noarch
ipw2200-firmware-3.1-9.fc20.noarch
irda-utils-0.9.18-19.fc20.x86_64
irqbalance-1.0.7-1.fc20.x86_64
iscsi-initiator-utils-6.2.0.873-17.fc20.x86_64
iso-codes-3.49-1.fc20.noarch
isomd5sum-1.0.11-2.fc20.x86_64
ivtv-firmware-20080701-25.noarch
```

```
iw-3.11-1.fc20.x86_64
iwl1000-firmware-39.31.5.1-37.fc20.noarch
iwl100-firmware-39.31.5.1-37.fc20.noarch
iwl105-firmware-18.168.6.1-37.fc20.noarch
iwl135-firmware-18.168.6.1-37.fc20.noarch
iwl2000-firmware-18.168.6.1-37.fc20.noarch
iwl2030-firmware-18.168.6.1-37.fc20.noarch
iwl3160-firmware-22.24.8.0-37.fc20.noarch
iwl3945-firmware-15.32.2.9-37.fc20.noarch
iwl4965-firmware-228.61.2.24-37.fc20.noarch
iwl5000-firmware-8.83.5.1_1-37.fc20.noarch
iwl5150-firmware-8.24.2.2-37.fc20.noarch
iwl6000-firmware-9.221.4.1-37.fc20.noarch
iwl6000g2a-firmware-17.168.5.3-37.fc20.noarch
iwl6000g2b-firmware-17.168.5.2-37.fc20.noarch
iwl6050-firmware-41.28.5.1-37.fc20.noarch
iwl7260-firmware-22.24.8.0-37.fc20.noarch
jack-audio-connection-kit-1.9.9.5-3.fc20.x86_64
jansson-2.6-1.fc20.x86_64
jasper-libs-1.900.1-25.fc20.x86_64
java-1.7.0-openjdk-1.7.0.60-2.4.5.1.fc20.x86_64
java-1.7.0-openjdk-headless-1.7.0.60-2.4.5.1.fc20.x86_64
javapackages-tools-3.4.1-1.fc20.noarch
jbigkit-libs-2.0-9.fc20.x86_64
jline-1.0-5.fc20.noarch
jomolhari-fonts-0.003-17.fc20.noarch
json-c-0.11-3.fc20.x86_64
json-glib-0.16.2-1.fc20.x86_64
kbd-1.15.5-12.fc20.x86_64
kbd-legacy-1.15.5-12.fc20.noarch
kbd-misc-1.15.5-12.fc20.noarch
kernel-3.11.10-301.fc20.x86_64
kernel-3.13.7-200.fc20.x86_64
kernel-3.13.9-200.fc20.x86_64
kernel-modules-extra-3.11.10-301.fc20.x86_64
kernel-modules-extra-3.13.7-200.fc20.x86_64
kernel-modules-extra-3.13.9-200.fc20.x86_64
kexec-tools-2.0.4-18.fc20.x86_64
keyutils-1.5.9-1.fc20.x86_64
keyutils-libs-1.5.9-1.fc20.x86_64
khmeros-base-fonts-5.0-17.fc20.noarch
khmeros-fonts-common-5.0-17.fc20.noarch
kmod-15-1.fc20.x86_64
```

```
kmod-libs-15-1.fc20.x86_64
kpartx-0.4.9-56.fc20.x86_64
krb5-libs-1.11.5-4.fc20.x86_64
langtable-0.0.24-1.fc20.noarch
langtable-data-0.0.24-1.fc20.noarch
langtable-python-0.0.24-1.fc20.noarch
lcms2-2.6-1.fc20.x86_64
ldns-1.6.16-6.fc20.x86_64
less-458-7.fc20.x86_64
lesstif-0.95.2-6.fc20.x86_64
leveldb-1.12.0-5.fc20.x86_64
libabw-0.0.2-1.fc20.x86_64
libacl-2.2.52-4.fc20.x86_64
libaio-0.3.109-8.fc20.x86_64
libao-1.1.0-8.fc20.x86_64
libarchive-3.1.2-7.fc20.x86_64
libassuan-2.1.0-2.fc20.x86_64
libasyncns-0.8-6.fc20.x86_64
libatasmart-0.19-5.fc20.x86_64
libattr-2.4.47-3.fc20.x86_64
libavc1394-0.5.3-14.fc20.x86_64
libbasicobjects-0.1.0-20.fc20.x86_64
libblkid-2.24.1-1.fc20.x86_64
libbluray-0.5.0-2.fc20.x86_64
libbsd-0.6.0-2.fc20.x86_64
libcacard-1.6.2-1.fc20.x86_64
libcanberra-0.30-4.fc20.x86_64
libcanberra-gtk2-0.30-4.fc20.x86_64
libcanberra-gtk3-0.30-4.fc20.x86_64
libcap-2.22-7.fc20.x86_64
libcap-ng-0.7.3-6.fc20.x86_64
libcdio-0.90-3.fc20.x86_64
libcdio-paranoia-10.2+0.90+1-1.fc20.x86_64
libcdr-0.0.14-5.fc20.x86_64
libcgroup-0.38-8.fc20.x86_64
libchamplain-0.12.5-1.fc20.x86_64
libchamplain-gtk-0.12.5-1.fc20.x86_64
libchewing-0.3.5-1.fc20.x86_64
libcmis-0.4.1-3.fc20.x86_64
libcollection-0.6.2-20.fc20.x86_64
libcom_err-1.42.8-3.fc20.x86_64
libcomps-0.1.4-4.fc20.x86_64
libconfig-1.4.9-5.fc20.x86_64
```

```
libcroco-0.6.8-3.fc20.x86_64
libcue-1.4.0-3.fc20.x86_64
libcurl-7.32.0-8.fc20.x86_64
libdaemon-0.14-6.fc20.x86_64
libdb-5.3.28-1.fc20.x86_64
libdb-utils-5.3.28-1.fc20.x86_64
libdhash-0.4.3-20.fc20.x86_64
libdmapsharing-2.9.24-1.fc20.x86_64
libdmx-1.1.3-2.fc20.x86_64
libdnet-1.12-12.fc20.x86_64
libdrm-2.4.52-1.fc20.x86_64
libdv-1.0.0-16.fc20.x86_64
libdvdnav-4.2.1-1.fc20.x86_64
libdvdread-4.2.1-1.fc20.x86_64
libe-book-0.0.3-1.fc20.x86_64
libedit-3.1-2.20130601cvs.fc20.x86_64
libeot-0.01-1.fc20.x86_64
liberation-fonts-common-1.07.3-2.fc20.noarch
liberation-mono-fonts-1.07.3-2.fc20.noarch
liberation-sans-fonts-1.07.3-2.fc20.noarch
liberation-serif-fonts-1.07.3-2.fc20.noarch
libertas-usb8388-firmware-20140317-37.gitdec41bce.fc20.noarch
libetonyek-0.0.3-1.fc20.x86_64
libevdev-0.6-4.fc20.x86_64
libevent-2.0.21-3.fc20.x86_64
libexif-0.6.21-6.fc20.x86_64
libexttextcat-3.4.3-2.fc20.x86_64
libffado-2.1.0-4.fc20.x86_64
libffi-3.0.13-5.fc20.x86_64
libfontenc-1.1.1-4.fc20.x86_64
libfprint-0.5.1-1.fc20.x86_64
libfreehand-0.0.0-3.fc20.x86_64
libgadu-1.12.0-0.3.rc2.fc20.x86_64
libgcc-4.8.2-7.fc20.x86_64
libgcrypt-1.5.3-2.fc20.x86_64
libgdata-0.14.1-1.fc20.x86_64
libgdither-0.6-7.fc20.x86_64
libgee-0.12.0-1.fc20.x86_64
libgee06-0.6.8-2.fc20.x86_64
libgexiv2-0.7.0-1.fc20.x86_64
libglade2-2.6.4-10.fc20.x86_64
libgnomekbd-3.6.0-3.fc20.x86_64
libgnome-keyring-3.10.1-1.fc20.x86_64
```

```
libgomp-4.8.2-7.fc20.x86_64
libgpg-error-1.12-1.fc20.x86_64
libgphoto2-2.5.3-6.fc20.x86_64
libgpod-0.8.3-1.fc20.x86_64
libgsf-1.14.29-1.fc20.x86_64
libgtop2-2.28.5-1.fc20.x86_64
libgudev1-208-16.fc20.x86_64
libgusb-0.1.6-2.fc20.x86_64
libgweather-3.10.1-1.fc20.x86_64
libgxps-0.2.2-8.fc20.x86_64
libhangul-0.1.0-7.fc20.x86_64
libhbaapi-2.2.9-3.fc20.x86_64
libhbalinux-1.0.16-2.fc20.x86_64
libical-1.0-4.fc20.x86_64
libICE-1.0.8-6.fc20.x86_64
libicu-50.1.2-10.fc20.x86_64
libidn-1.28-2.fc20.x86_64
libiec61883-1.2.0-10.fc20.x86_64
libieee1284-0.2.11-15.fc20.x86_64
libimobiledevice-1.1.5-3.fc20.x86_64
libini_config-1.0.0.1-20.fc20.x86_64
libipa_hbac-1.11.4-3.fc20.x86_64
libiptcdata-1.0.4-10.fc20.x86_64
libiscsi-1.9.0-4.fc20.x86_64
libjpeg-turbo-1.3.0-2.fc20.x86_64
libkkc-0.3.3-1.fc20.x86_64
libkkc-common-0.3.3-1.fc20.noarch
libkkc-data-0.2.7-2.fc20.x86_64
liblangtag-0.5.2-1.fc20.x86_64
libldb-1.1.16-4.fc20.x86_64
liblouis-2.5.3-2.fc20.x86_64
liblouis-python3-2.5.3-2.fc20.noarch
libmash-0.2.0-11.fc20.x86_64
libmbim-1.5.0-1.20130815git.fc20.x86_64
libmcpp-2.7.2-11.fc20.x86_64
libmetalink-0.1.2-4.fc20.x86_64
libmnl-1.0.3-6.fc20.x86_64
libmodman-2.0.1-7.fc20.x86_64
libmount-2.24.1-1.fc20.x86_64
libmpcdec-1.2.6-12.fc20.x86_64
libmspack-0.4-0.2.alpha.fc20.x86_64
libmspub-0.0.6-4.fc20.x86_64
libmtp-1.1.6-2.fc20.x86_64
```

```
libmusicbrainz5-5.0.1-8.fc20.x86_64
libmwaw-0.2.0-1.fc20.x86_64
libmx-1.4.7-9.fc20.x86_64
libndp-1.2-1.fc20.x86_64
libnetfilter_conntrack-1.0.4-1.fc20.x86_64
libnfnetlink-1.0.1-3.fc20.x86_64
libnice-0.1.4-2.fc20.x86_64
libnice-gstreamer1-0.1.4-2.fc20.x86_64
libnl-1.1.4-3.fc20.x86_64
libnl3-3.2.24-1.fc20.x86_64
libnl3-cli-3.2.24-1.fc20.x86_64
libnm-gtk-0.9.9.0-9.git20140123.fc20.x86_64
libnotify-0.7.6-1.fc20.x86_64
liboath-2.4.0-2.fc20.x86_64
liboauth-1.0.1-2.fc20.x86_64
libodfgen-0.0.4-1.fc20.x86_64
libofa-0.9.3-23.fc20.x86_64
libogg-1.3.0-6.fc20.x86_64
liborcus-0.5.1-5.fc20.x86_64
libosinfo-0.2.9-1.fc20.x86_64
libpaper-1.1.24-7.fc20.x86_64
libpath_utils-0.2.1-20.fc20.x86_64
libpcap-1.5.3-1.fc20.x86_64
libpciaccess-0.13.2-1.fc20.x86_64
libpeas-1.9.0-2.fc20.x86_64
libpinyin-1.0.0-1.fc20.x86_64
libpinyin-data-1.0.0-1.fc20.x86_64
libpipeline-1.2.4-2.fc20.x86_64
libplist-1.10-2.fc20.x86_64
libpng-1.6.3-3.fc20.x86_64
libproxy-0.4.11-7.fc20.x86_64
libproxy-mozjs-0.4.11-7.fc20.x86_64
libpurple-2.10.9-1.fc20.x86_64
libpwquality-1.2.3-1.fc20.x86_64
libqmi-1.6.0-1.fc20.x86_64
libquvi-0.9.2-1.fc20.x86_64
libquvi-scripts-0.9.20131104-1.fc20.noarch
LibRaw-0.15.4-1.fc20.x86_64
libraw1394-2.1.0-2.fc20.x86_64
libref_array-0.1.3-20.fc20.x86_64
libreoffice-calc-4.2.3.2-3.fc20.x86_64
libreoffice-core-4.2.3.2-3.fc20.x86_64
libreoffice-draw-4.2.3.2-3.fc20.x86_64
```

```
libreoffice-graphicfilter-4.2.3.2-3.fc20.x86_64
libreoffice-impress-4.2.3.2-3.fc20.x86_64
libreoffice-opensymbol-fonts-4.2.3.2-3.fc20.noarch
libreoffice-pdfimport-4.2.3.2-3.fc20.x86_64
libreoffice-ure-4.2.3.2-3.fc20.x86_64
libreoffice-writer-4.2.3.2-3.fc20.x86_64
librepo-1.6.0-1.fc20.x86_64
libreport-2.2.1-1.fc20.x86_64
libreport-anaconda-2.2.1-1.fc20.x86_64
libreport-cli-2.2.1-1.fc20.x86_64
libreport-fedora-2.2.1-1.fc20.x86_64
libreport-filesystem-2.2.1-1.fc20.x86_64
libreport-gtk-2.2.1-1.fc20.x86_64
libreport-plugin-bugzilla-2.2.1-1.fc20.x86_64
libreport-plugin-kerneloops-2.2.1-1.fc20.x86_64
libreport-plugin-logger-2.2.1-1.fc20.x86_64
libreport-plugin-reportuploader-2.2.1-1.fc20.x86_64
libreport-plugin-ureport-2.2.1-1.fc20.x86_64
libreport-python-2.2.1-1.fc20.x86_64
libreport-python3-2.2.1-1.fc20.x86_64
libreport-web-2.2.1-1.fc20.x86_64
libreswan-3.8-1.fc20.x86_64
librsvg2-2.40.1-1.fc20.x86_64
libsamplerate-0.1.8-5.fc20.x86_64
libsane-hpaio-3.13.11-4.fc20.x86_64
libsecret-0.16-1.fc20.x86_64
libselinux-2.2.1-6.fc20.x86_64
libselinux-python-2.2.1-6.fc20.x86_64
libselinux-utils-2.2.1-6.fc20.x86_64
libsemanage-2.1.10-14.fc20.x86_64
libsemanage-python-2.1.10-14.fc20.x86_64
libsepol-2.1.9-2.fc20.x86_64
libshout-2.2.2-10.fc20.x86_64
libsigc++20-2.3.1-3.fc20.x86_64
libsilc-1.1.10-10.fc20.x86_64
libSM-1.2.1-6.fc20.x86_64
libsmbclient-4.1.6-1.fc20.x86_64
libsmi-0.4.8-12.fc20.x86_64
libsndfile-1.0.25-8.fc20.x86_64
libsolv-0.4.1-1.gitbcedc98.fc20.x86_64
libsoup-2.44.2-1.fc20.x86_64
libspectre-0.2.7-3.fc20.x86_64
libsrtp-1.4.4-10.20101004cvs.fc20.x86_64
```

```
libss-1.42.8-3.fc20.x86_64
libssh2-1.4.3-8.fc20.x86_64
libsss_idmap-1.11.4-3.fc20.x86_64
libsss_nss_idmap-1.11.4-3.fc20.x86_64
libstdc++-4.8.2-7.fc20.x86_64
libsysfs-2.1.0-15.fc20.x86_64
libtalloc-2.1.0-3.fc20.x86_64
libtar-1.2.11-27.fc20.x86_64
libtasn1-3.3-2.fc20.x86_64
libtdb-1.2.12-2.fc20.x86_64
libteam-1.9-1.fc20.x86_64
libtevent-0.9.20-1.fc20.x86_64
libthai-0.1.19-2.fc20.x86_64
libtheora-1.1.1-9.fc20.x86_64
libtiff-4.0.3-14.fc20.x86_64
libtomcrypt-1.17-21.fc20.x86_64
libtommath-0.42.0-3.fc20.x86_64
libtool-ltdl-2.4.2-23.fc20.x86_64
libudisks2-2.1.2-2.fc20.x86_64
libunistring-0.9.3-9.fc20.x86_64
libunwind-1.1-3.fc20.x86_64
libusb-0.1.5-3.fc20.x86_64
libusbx-1.0.18-1.fc20.x86_64
libuser-0.60-3.fc20.x86_64
libuser-python-0.60-3.fc20.x86_64
libutempter-1.1.6-3.fc20.x86_64
libuuid-2.24.1-1.fc20.x86_64
libv4l-1.0.0-1.fc20.x86_64
libvdpau-0.7-1.fc20.x86_64
libverto-0.2.5-3.fc20.x86_64
libvisio-0.0.31-1.fc20.x86_64
libvisual-0.4.0-14.fc20.x86_64
libvorbis-1.3.4-1.fc20.x86_64
libvpx-1.3.0-3.fc20.x86_64
libwacom-0.8-1.fc20.x86_64
libwacom-data-0.8-1.fc20.noarch
libwayland-client-1.2.0-3.fc20.x86_64
libwayland-cursor-1.2.0-3.fc20.x86_64
libwayland-server-1.2.0-3.fc20.x86_64
libwbclient-4.1.6-1.fc20.x86_64
libwebp-0.3.1-2.fc20.x86_64
libwnck3-3.4.7-1.fc20.x86_64
libwpd-0.9.9-1.fc20.x86_64
```

```
libwpg-0.2.2-5.fc20.x86_64
libwps-0.2.9-5.fc20.x86_64
libwvstreams-4.6.1-10.fc20.x86_64
libX11-1.6.1-1.fc20.x86_64
libX11-common-1.6.1-1.fc20.noarch
libXau-1.0.8-2.fc20.x86_64
libxcb-1.9.1-3.fc20.x86_64
libXcomposite-0.4.4-4.fc20.x86_64
libXcursor-1.1.14-2.fc20.x86_64
libXdamage-1.1.4-4.fc20.x86_64
libXdmcp-1.1.1-5.fc20.x86_64
libXevie-1.0.3-7.fc20.x86_64
libXext-1.3.2-2.fc20.x86_64
libXfixes-5.0.1-2.fc20.x86_64
libXfont-1.4.7-1.fc20.x86_64
libXft-2.3.1-5.fc20.x86_64
libXi-1.7.2-2.fc20.x86_64
libXinerama-1.1.3-2.fc20.x86_64
libxkbcommon-0.3.1-1.fc20.x86_64
libxkbfile-1.0.8-4.fc20.x86_64
libxklavier-5.4-1.fc20.x86_64
libxml2-2.9.1-2.fc20.x86_64
libxml++-2.37.1-1.fc20.x86_64
libxml2-python-2.9.1-2.fc20.x86_64
libXmu-1.1.1-5.fc20.x86_64
libXp-1.0.2-2.fc20.x86_64
libXpm-3.5.10-5.fc20.x86_64
libXrandr-1.4.1-2.fc20.x86_64
libXrender-0.9.8-2.fc20.x86_64
libXres-1.0.7-2.fc20.x86_64
libxslt-1.1.28-5.fc20.x86_64
libXt-1.1.4-7.fc20.x86_64
libXtst-1.2.2-2.fc20.x86_64
libXv-1.0.9-2.fc20.x86_64
libXvMC-1.0.8-2.fc20.x86_64
libXxf86dga-1.1.4-2.fc20.x86_64
libXxf86misc-1.0.3-7.fc20.x86_64
libXxf86vm-1.1.3-2.fc20.x86_64
libzapojit-0.0.3-2.fc20.x86_64
linux-atm-libs-2.5.1-8.fc20.x86_64
linux-firmware-20140317-37.gitdec41bce.fc20.noarch
lklug-fonts-0.6-10.20090803cvs.fc20.noarch
lldpad-0.9.46-3.fc20.x86_64
```

```
llvm-libs-3.3-4.fc20.x86_64
lockdev-1.0.4-0.11.20111007git.fc20.x86_64
logrotate-3.8.7-1.fc20.x86_64
lohit-assamese-fonts-2.5.3-2.fc20.noarch
lohit-bengali-fonts-2.5.3-4.fc20.noarch
lohit-devanagari-fonts-2.94.0-1.fc20.noarch
lohit-gujarati-fonts-2.92.2-1.fc20.noarch
lohit-kannada-fonts-2.5.3-4.fc20.noarch
lohit-malayalam-fonts-2.92.0-1.fc20.noarch
lohit-oriya-fonts-2.5.4.1-2.fc20.noarch
lohit-punjabi-fonts-2.5.3-3.fc20.noarch
lohit-tamil-fonts-2.5.3-2.fc20.noarch
lohit-telugu-fonts-2.5.3-3.fc20.noarch
lpsolve-5.5.2.0-7.fc20.x86_64
lrzsz-0.12.20-34.fc20.x86_64
lsof-4.87-3.fc20.x86_64
lua-5.2.2-5.fc20.x86_64
lua-expat-1.2.0-6.fc20.x86_64
lua-json-1.3.2-2.fc20.noarch
lua-lpeg-0.12-2.fc20.x86_64
lua-socket-2.1-0.2.rc1.fc20.x86_64
lvm2-2.02.103-5.fc20.x86_64
lvm2-libs-2.02.103-5.fc20.x86_64
lzo-2.06-5.fc20.x86_64
m17n-contrib-1.1.14-3.fc20.noarch
m17n-db-1.6.4-3.fc20.noarch
m17n-lib-1.6.4-10.fc20.x86_64
mactel-boot-0.9-9.fc20.x86_64
mailcap-2.1.42-1.fc20.noarch
make-3.82-19.fc20.x86_64
man-db-2.6.5-2.fc20.x86_64
marisa-0.2.4-4.fc20.x86_64
mcelog-1.0-0.11.f0d7654.fc20.x86_64
mcpp-2.7.2-11.fc20.x86_64
mdadm-3.3-4.fc20.x86_64
meanwhile-1.1.0-12.fc20.x86_64
media-player-info-17-4.fc20.noarch
memtest86+-5.01-1.fc20.x86_64
mesa-dri-drivers-10.0.4-1.20140312.fc20.x86_64
mesa-filesystem-10.0.4-1.20140312.fc20.x86_64
mesa-libEGL-10.0.4-1.20140312.fc20.x86_64
mesa-libgbm-10.0.4-1.20140312.fc20.x86_64
mesa-libGL-10.0.4-1.20140312.fc20.x86_64
```

```
mesa-libglapi-10.0.4-1.20140312.fc20.x86_64
mesa-libGLU-9.0.0-4.fc20.x86_64
mesa-libwayland-egl-10.0.4-1.20140312.fc20.x86_64
mesa-libxatracker-10.0.4-1.20140312.fc20.x86_64
microcode_ctl-2.1-3.fc20.x86_64
mlocate-0.26-4.fc20.x86_64
mobile-broadband-provider-info-1.20120614-4.fc20.noarch
ModemManager-1.1.0-2.git20130913.fc20.x86_64
ModemManager-glib-1.1.0-2.git20130913.fc20.x86_64
mokutil-0.7-1.fc20.x86_64
mousetweaks-3.10.0-1.fc20.x86_64
mozilla-filesystem-1.9-10.fc20.x86_64
mozjs17-17.0.0-8.fc20.x86_64
mpfr-3.1.2-4.fc20.x86_64
mtdev-1.1.4-1.fc20.x86_64
mtools-4.0.18-4.fc20.x86_64
mutter-3.10.4-1.fc20.x86_64
mutter-wayland-3.10.4-1.fc20.x86_64
mythes-1.2.3-6.fc20.x86_64
nautilus-3.10.1-3.fc20.x86_64
nautilus-extensions-3.10.1-3.fc20.x86_64
nautilus-sendto-3.8.1-1.fc20.x86_64
ncurses-5.9-12.20130511.fc20.x86_64
ncurses-base-5.9-12.20130511.fc20.noarch
ncurses-libs-5.9-12.20130511.fc20.x86_64
neon-0.30.0-2.fc20.x86_64
net-snmp-libs-5.7.2-17.fc20.x86_64
nettle-2.7.1-3.fc20.x86_64
net-tools-2.0-0.15.20131119git.fc20.x86_64
NetworkManager-0.9.9.0-33.git20131003.fc20.x86_64
NetworkManager-glib-0.9.9.0-33.git20131003.fc20.x86_64
NetworkManager-l2tp-0.9.8.6-1.fc20.x86_64
NetworkManager-openconnect-0.9.8.0-2.fc20.x86_64
NetworkManager-openvpn-0.9.9.0-0.1.git20140128.fc20.x86_64
NetworkManager-openvpn-gnome-0.9.9.0-0.1.git20140128.fc20.x86_64
NetworkManager-pptp-0.9.8.2-3.fc20.x86_64
NetworkManager-pptp-gnome-0.9.8.2-3.fc20.x86_64
NetworkManager-vpnc-0.9.8.2-2.fc20.x86_64
NetworkManager-vpnc-gnome-0.9.8.2-2.fc20.x86_64
newt-0.52.16-2.fc20.x86_64
newt-python-0.52.16-2.fc20.x86_64
nhn-nanum-fonts-common-3.020-9.fc20.noarch
nhn-nanum-gothic-fonts-3.020-9.fc20.noarch
```

```
nmap-ncat-6.40-2.fc20.x86_64
nm-connection-editor-0.9.9.0-9.git20140123.fc20.x86_64
notify-python-0.1.1-25.fc20.x86_64
nspr-4.10.4-1.fc20.x86_64
nss-3.16.0-1.fc20.x86_64
nss-mdns-0.10-13.fc20.x86_64
nss-softokn-3.16.0-1.fc20.x86_64
nss-softokn-freebl-3.16.0-1.fc20.x86_64
nss-sysinit-3.16.0-1.fc20.x86_64
nss-tools-3.16.0-1.fc20.x86_64
nss-util-3.16.0-1.fc20.x86_64
ntfs-3g-2014.2.15-1.fc20.x86_64
ntfsprogs-2014.2.15-1.fc20.x86_64
numactl-libs-2.0.9-1.fc20.x86_64
oddjob-0.31.5-1.fc20.x86_64
oddjob-mkhomedir-0.31.5-1.fc20.x86_64
opencc-0.4.3-2.fc20.x86_64
openconnect-5.02-1.fc20.x86_64
openjpeg-libs-1.5.1-8.fc20.x86_64
openldap-2.4.39-2.fc20.x86_64
openssh-6.4p1-3.fc20.x86_64
openssh-clients-6.4p1-3.fc20.x86_64
openssh-server-6.4p1-3.fc20.x86_64
openssl-1.0.1e-37.fc20.1.x86_64
openssl-libs-1.0.1e-37.fc20.1.x86_64
open-vm-tools-9.4.0-1.fc20.x86_64
open-vm-tools-desktop-9.4.0-1.fc20.x86_64
openvpn-2.3.2-4.fc20.x86_64
opus-1.1-1.fc20.x86_64
orc-0.4.18-1.fc20.x86_64
orca-3.10.2-1.fc20.noarch
os-prober-1.58-4.fc20.x86_64
p11-kit-0.20.2-1.fc20.x86_64
p11-kit-trust-0.20.2-1.fc20.x86_64
PackageKit-0.8.17-1.fc20.x86_64
PackageKit-command-not-found-0.8.17-1.fc20.x86_64
PackageKit-glib-0.8.17-1.fc20.x86_64
PackageKit-gstreamer-plugin-0.8.17-1.fc20.x86_64
PackageKit-gtk3-module-0.8.17-1.fc20.x86_64
PackageKit-yum-0.8.17-1.fc20.x86_64
PackageKit-yum-plugin-0.8.17-1.fc20.x86_64
pakchois-0.4-9.fc20.x86_64
paktype-naqsh-fonts-4.1-2.fc20.noarch
```

```
pam-1.1.8-1.fc20.x86_64
pam_pkcs11-0.6.2-11.fc20.x86_64
pango-1.36.1-3.fc20.x86_64
pangomm-2.34.0-2.fc20.x86_64
paps-0.6.8-27.fc20.x86_64
paps-libs-0.6.8-27.fc20.x86_64
paratype-pt-sans-fonts-20101909-3.fc20.noarch
parted-3.1-13.fc20.x86_64
passwd-0.79-2.fc20.x86_64
passwdqc-1.3.0-1.fc20.x86_64
passwdqc-lib-1.3.0-1.fc20.x86_64
pciutils-3.2.1-1.fc20.x86_64
pciutils-libs-3.2.1-1.fc20.x86_64
pcmciautils-018-5.fc20.x86_64
pcre-8.33-4.fc20.x86_64
pcsc-lite-libs-1.8.10-1.fc20.x86_64
perl-5.18.2-289.fc20.x86_64
perl-Carp-1.26-245.fc20.noarch
perl-constant-1.27-292.fc20.noarch
perl-Encode-2.54-2.fc20.x86_64
perl-Exporter-5.68-293.fc20.noarch
perl-File-Path-2.09-292.fc20.noarch
perl-File-Temp-0.23.01-4.fc20.noarch
perl-Filter-1.49-5.fc20.x86_64
perl-Getopt-Long-2.42-1.fc20.noarch
perl-HTTP-Tiny-0.034-4.fc20.noarch
perl-libs-5.18.2-289.fc20.x86_64
perl-macros-5.18.2-289.fc20.x86_64
perl-Module-CoreList-3.03-289.fc20.noarch
perl-parent-0.228-1.fc20.noarch
perl-PathTools-3.40-291.fc20.x86_64
perl-Pod-Escapes-1.04-289.fc20.noarch
perl-podlators-2.5.1-291.fc20.noarch
perl-Pod-Perldoc-3.20-7.fc20.noarch
perl-Pod-Simple-3.28-292.fc20.noarch
perl-Pod-Usage-1.63-4.fc20.noarch
perl-Scalar-List-Utils-1.31-293.fc20.x86_64
perl-Socket-2.013-1.fc20.x86_64
perl-Storable-2.45-2.fc20.x86_64
perl-Text-ParseWords-3.29-3.fc20.noarch
perl-threads-1.92-1.fc20.x86_64
perl-threads-shared-1.46-1.fc20.x86_64
perl-Time-HiRes-1.9726-1.fc20.x86_64
```

```
perl-Time-Local-1.2300-291.fc20.noarch
perl-version-0.99.07-1.fc20.x86_64
pinentry-0.8.1-11.fc20.x86_64
pinentry-gtk-0.8.1-11.fc20.x86_64
pixman-0.30.0-3.fc20.x86_64
pkcs11-helper-1.10-2.fc20.x86_64
pkgconfig-0.28-3.fc20.x86_64
plymouth-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-core-libs-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-graphics-libs-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-plugin-label-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-plugin-two-step-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-scripts-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-system-theme-0.8.9-3.2013.08.14.fc20.x86_64
plymouth-theme-charge-0.8.9-3.2013.08.14.fc20.x86_64
pm-utils-1.4.1-26.fc20.x86_64
policycoreutils-2.2.5-3.fc20.x86_64
policycoreutils-python-2.2.5-3.fc20.x86_64
polkit-0.112-2.fc20.x86_64
polkit-pkla-compat-0.1-3.fc20.x86_64
poppler-0.24.3-3.fc20.x86_64
poppler-data-0.4.6-4.fc20.noarch
poppler-glib-0.24.3-3.fc20.x86_64
poppler-utils-0.24.3-3.fc20.x86_64
popt-1.16-2.fc20.x86_64
portaudio-19-17.fc20.x86_64
ppp-2.4.5-33.fc20.x86_64
pptp-1.8.0-1.fc20.x86_64
procps-ng-3.3.8-16.fc20.x86_64
protobuf-2.5.0-5.fc20.x86_64
protobuf-c-0.15-8.fc20.x86_64
protobuf-compiler-2.5.0-5.fc20.x86_64
psacct-6.6.1-7.fc20.x86_64
psmisc-22.20-3.fc20.x86_64
pth-2.0.7-21.fc20.x86_64
pulseaudio-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-gdm-hooks-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-libs-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-libs-glib2-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-module-bluetooth-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-module-x11-4.0-9.gitf81e3.fc20.x86_64
pulseaudio-utils-4.0-9.gitf81e3.fc20.x86_64
pyatspi-2.10.0-1.fc20.noarch
```

```
pycairo-1.8.10-7.fc20.x86_64
pygobject2-2.28.6-11.fc20.x86_64
pygobject3-3.10.2-1.fc20.x86_64
pygobject3-base-3.10.2-1.fc20.x86_64
pygpgme-0.3-8.fc20.x86_64
pygtk2-2.24.0-8.fc20.x86_64
pygtk2-libglade-2.24.0-8.fc20.x86_64
pykickstart-1.99.48-1.fc20.noarch
pyliblzma-0.5.3-10.fc20.x86_64
pyparted-3.9-4.fc20.x86_64
pytalloc-2.1.0-3.fc20.x86_64
python-2.7.5-11.fc20.x86_64
python3-3.3.2-11.fc20.x86_64
python3-beaker-1.5.4-8.fc20.noarch
python3-brlapi-0.6.0-9.fc20.x86_64
python3-cairo-1.10.0-6.fc20.x86_64
python3-dbus-1.2.0-1.fc20.x86_64
python3-gobject-3.10.2-1.fc20.x86_64
python3-libs-3.3.2-11.fc20.x86_64
python3-mako-0.7.3-2.fc20.noarch
python3-markupsafe-0.18-1.fc20.x86_64
python3-pyatspi-2.10.0-1.fc20.noarch
python3-speechd-0.8-7.fc20.x86_64
python-augeas-0.4.1-4.fc20.noarch
python-blivet-0.23.9-1.fc20.noarch
python-caribou-0.4.13-1.fc20.noarch
python-chardet-2.0.1-7.fc20.noarch
python-cssselect-0.9.1-1.fc20.noarch
python-cups-1.9.65-1.fc20.x86_64
python-decorator-3.4.0-3.fc20.noarch
python-hawkey-0.4.12-1.fc20.x86_64
python-iniparse-0.4-9.fc20.noarch
python-inotify-0.9.4-4.fc20.noarch
python-IPy-0.75-6.fc20.noarch
python-javapackages-3.4.1-1.fc20.noarch
python-kitchen-1.1.1-5.fc20.noarch
python-libcomps-0.1.4-4.fc20.x86_64
python-librepo-1.6.0-1.fc20.x86_64
python-libs-2.7.5-11.fc20.x86_64
python-lxml-3.3.3-1.fc20.x86_64
python-meh-0.27-1.fc20.noarch
python-nss-0.14.0-2.fc20.x86_64
python-ntplib-0.3.1-2.fc20.noarch
```

```
python-pwquality-1.2.3-1.fc20.x86_64
python-pyblock-0.53-5.fc20.x86_64
python-pycurl-7.19.3-1.fc20.x86_64
python-six-1.4.1-1.fc20.noarch
python-slip-0.6.0-1.fc20.noarch
python-slip-dbus-0.6.0-1.fc20.noarch
python-sssdconfig-1.11.4-3.fc20.noarch
python-urlgrabber-3.9.1-32.fc20.noarch
pytz-2012d-5.fc20.noarch
pyxattr-0.5.1-4.fc20.x86_64
pyxdg-0.25-2.fc20.noarch
qemu-guest-agent-1.6.2-1.fc20.x86_64
qpdf-libs-5.1.1-1.fc20.x86_64
qrencode-libs-3.4.2-1.fc20.x86_64
raptor2-2.0.9-2.fc20.x86_64
rasqal-0.9.30-2.fc20.x86_64
rdist-6.1.5-57.fc20.x86_64
readline-6.2-8.fc20.x86_64
realmd-0.14.6-4.fc20.x86_64
redhat-menus-12.0.2-7.fc20.noarch
redland-1.0.16-4.fc20.x86_64
reiserfs-utils-3.6.21-9.fc20.x86_64
rest-0.7.90-5.fc20.x86_64
rhino-1.7R4-7.fc20.noarch
rhythmbox-3.0.2-1.fc20.x86_64
rng-tools-4-4.fc20.x86_64
rootfiles-8.1-16.fc20.noarch
rpm-4.11.2-2.fc20.x86_64
rpm-build-libs-4.11.2-2.fc20.x86_64
rpm-libs-4.11.2-2.fc20.x86_64
rpm-python-4.11.2-2.fc20.x86_64
rp-pppoe-3.11-4.fc20.x86_64
rsync-3.1.0-2.fc20.x86_64
rtkit-0.11-8.fc20.x86_64
rygel-0.20.3-1.fc20.x86_64
samba-client-4.1.6-1.fc20.x86_64
samba-common-4.1.6-1.fc20.x86_64
samba-libs-4.1.6-1.fc20.x86_64
sane-backends-1.0.24-7.fc20.x86_64
sane-backends-drivers-scanners-1.0.24-7.fc20.x86_64
sane-backends-libs-1.0.24-7.fc20.x86_64
satyr-0.13-1.fc20.x86_64
sbc-1.2-1.fc20.x86_64
```

```
seahorse-3.10.2-1.fc20.x86_64
sed-4.2.2-6.fc20.x86_64
seed-3.8.1-2.fc20.x86_64
selinux-policy-3.12.1-149.fc20.noarch
selinux-policy-targeted-3.12.1-149.fc20.noarch
setools-libs-3.3.7-41.fc20.x86_64
setroubleshoot-3.2.17-1.fc20.x86_64
setroubleshoot-plugins-3.0.59-1.fc20.noarch
setroubleshoot-server-3.2.17-1.fc20.x86_64
setup-2.8.71-2.fc20.noarch
setuptool-1.19.11-7.fc20.x86_64
sg3_utils-libs-1.37-2.fc20.x86_64
sgpio-1.2.0.10-12.fc20.x86_64
shadow-utils-4.1.5.1-8.fc20.x86_64
shared-mime-info-1.2-1.fc20.x86_64
shim-0.7-1.fc20.x86_64
shim-unsigned-0.7-1.fc20.x86_64
shotwell-0.15.1-1.fc20.x86_64
sil-abyssinica-fonts-1.200-6.fc20.noarch
sil-mingzat-fonts-0.020-3.fc20.noarch
sil-nuosu-fonts-2.1.1-7.fc20.noarch
sil-padauk-fonts-2.8-5.fc20.noarch
skkdic-20131114-7.T1121.fc20.noarch
slang-2.2.4-11.fc20.x86_64
smc-fonts-common-6.0-7.fc20.noarch
smc-meera-fonts-6.0-7.fc20.noarch
snappy-1.1.0-2.fc20.x86_64
socat-1.7.2.3-1.fc20.x86_64
sos-3.0-23.fc20.noarch
sound-theme-freedesktop-0.8-3.fc20.noarch
soundtouch-1.4.0-8.fc20.x86_64
sox-14.4.1-4.fc20.x86_64
speech-dispatcher-0.8-7.fc20.x86_64
speex-1.2-0.18.rc1.fc20.x86_64
spice-glib-0.23-2.fc20.x86_64
spice-gtk3-0.23-2.fc20.x86_64
spice-vdagent-0.15.0-1.fc20.x86_64
splunk-6.0.2-196940.x86_64
sqlite-3.8.4.2-2.fc20.x86_64
sssd-1.11.4-3.fc20.x86_64
sssd-ad-1.11.4-3.fc20.x86_64
sssd-client-1.11.4-3.fc20.x86_64
sssd-common-1.11.4-3.fc20.x86_64
```

```
sssd-common-pac-1.11.4-3.fc20.x86_64
sssd-ipa-1.11.4-3.fc20.x86_64
sssd-krb5-1.11.4-3.fc20.x86_64
sssd-krb5-common-1.11.4-3.fc20.x86_64
sssd-ldap-1.11.4-3.fc20.x86_64
sssd-proxy-1.11.4-3.fc20.x86_64
startup-notification-0.12-7.fc20.x86_64
stoken-libs-0.5-1.fc20.x86_64
stunnel-4.56-3.fc20.x86_64
sudo-1.8.8-1.fc20.x86_64
sushi-3.10.0-1.fc20.x86_64
symlinks-1.4-8.fc20.x86_64
syslinux-4.05-7.fc20.x86_64
syslinux-extlinux-4.05-7.fc20.x86_64
system-config-printer-libs-1.4.4-1.fc20.noarch
system-config-printer-udev-1.4.4-1.fc20.x86_64
systemd-208-16.fc20.x86_64
systemd-libs-208-16.fc20.x86_64
systemd-python-208-16.fc20.x86_64
systemd-python3-208-16.fc20.x86_64
sysvinit-tools-2.88-14.dsf.fc20.x86_64
tabish-eeyek-fonts-1.0-5.fc20.noarch
taglib-1.9.1-2.fc20.x86_64
tar-1.26-31.fc20.x86_64
tcpdump-4.5.1-1.fc20.x86_64
tcp_wrappers-7.6-76.fc20.x86_64
tcp_wrappers-libs-7.6-76.fc20.x86_64
teamd-1.9-1.fc20.x86_64
telepathy-farstream-0.6.1-1.fc20.x86_64
telepathy-filesystem-0.0.2-6.fc20.noarch
telepathy-gabble-0.18.2-1.fc20.x86_64
telepathy-glib-0.22.0-1.fc20.x86_64
telepathy-haze-0.8.0-1.fc20.x86_64
telepathy-idle-0.2.0-1.fc20.x86_64
telepathy-logger-0.8.0-3.fc20.x86_64
telepathy-mission-control-5.16.1-1.fc20.x86_64
telepathy-salut-0.8.1-4.fc20.x86_64
thai-scalable-fonts-common-0.5.0-7.fc20.noarch
thai-scalable-waree-fonts-0.5.0-7.fc20.noarch
tigervnc-license-1.3.0-14.fc20.noarch
tigervnc-server-minimal-1.3.0-14.fc20.x86_64
time-1.7-44.fc20.x86_64
tmpwatch-2.11-4.fc20.x86_64
```

```
totem-3.10.1-1.fc20.x86_64
totem-mozplugin-3.10.1-1.fc20.x86_64
totem-nautilus-3.10.1-1.fc20.x86_64
totem-pl-parser-3.10.0-1.fc20.x86_64
traceroute-2.0.19-4.fc20.x86_64
tracker-0.16.4-2.fc20.x86_64
tree-1.6.0-11.fc20.x86_64
trousers-0.3.11.2-3.fc20.x86_64
ttmkfdir-3.0.9-40.fc20.x86_64
tzdata-2014b-1.fc20.noarch
tzdata-java-2014b-1.fc20.noarch
udisks2-2.1.2-2.fc20.x86_64
unbound-libs-1.4.21-3.fc20.x86_64
unzip-6.0-12.fc20.x86_64
upower-0.9.23-2.fc20.x86_64
urw-fonts-2.4-18.fc20.noarch
usb_modeswitch-1.2.7-3.fc20.x86_64
usb_modeswitch-data-20131113-1.fc20.noarch
usbmuxd-1.0.8-10.fc20.x86_64
usbredir-0.6-5.fc20.x86_64
usbutils-007-2.fc20.x86_64
usermode-1.111-4.fc20.x86_64
ustr-1.0.4-15.fc20.x86_64
util-linux-2.24.1-1.fc20.x86_64
vconfig-1.9-13.fc20.x86_64
vim-common-7.4.179-1.fc20.x86_64
vim-enhanced-7.4.179-1.fc20.x86_64
vim-filesystem-7.4.179-1.fc20.x86_64
vim-minimal-7.4.179-1.fc20.x86_64
vinagre-3.10.2-1.fc20.x86_64
vino-3.10.1-1.fc20.x86_64
vlgothic-fonts-20130607-2.fc20.noarch
vpnc-0.5.3-20.svn457.fc20.x86_64
vpnc-script-0.5.3-20.svn457.fc20.noarch
vte3-0.34.9-1.fc20.x86_64
wavpack-4.70.0-1.fc20.x86_64
webkitgtk3-2.2.6-1.fc20.x86_64
webrtc-audio-processing-0.1-5.fc20.x86_64
wget-1.14-12.fc20.x86_64
which-2.20-6.fc20.x86_64
wireless-tools-29-10.1.fc20.x86_64
wireshark-1.10.6-2.fc20.x86_64
wireshark-gnome-1.10.6-2.fc20.x86_64
```

```
wpa_supplicant-2.0-8.fc20.x86_64
wqy-zenhei-fonts-0.9.46-11.fc20.noarch
wvdial-1.61-8.fc20.x86_64
xcb-util-0.3.9-3.fc20.x86_64
xdg-user-dirs-0.15-2.fc20.x86_64
xdg-user-dirs-gtk-0.10-3.fc20.x86_64
xdg-utils-1.1.0-0.22.rc2.fc20.noarch
xfsprogs-3.1.11-2.fc20.x86_64
xkeyboard-config-2.10.1-1.fc20.noarch
xl2tpd-1.3.1-14.fc20.x86_64
xml-common-0.6.3-40.fc20.noarch
xmlrpc-c-1.32.5-1903.svn2451.fc20.x86_64
xmlrpc-c-client-1.32.5-1903.svn2451.fc20.x86_64
xorg-x11-drv-ati-7.2.0-3.20131101git3b38701.fc20.x86_64
xorg-x11-drv-evdev-2.8.2-1.fc20.x86_64
xorg-x11-drv-fbdev-0.4.3-10.fc20.x86_64
xorg-x11-drv-intel-2.21.15-5.fc20.x86_64
xorg-x11-drv-mga-1.6.2-8.fc20.x86_64
xorg-x11-drv-modesetting-0.8.0-2.fc20.x86_64
xorg-x11-drv-nouveau-1.0.9-2.fc20.x86_64
xorg-x11-drv-openchrome-0.3.3-2.fc20.x86_64
xorg-x11-drv-qxl-0.1.1-3.fc20.x86_64
xorg-x11-drv-synaptics-1.7.4-5.fc20.x86_64
xorg-x11-drv-vesa-2.3.2-10.fc20.x86_64
xorg-x11-drv-vmmouse-13.0.0-6.fc20.x86_64
xorg-x11-drv-vmware-13.0.1-3.20131207gita40cbd7b.fc20.x86_64
xorg-x11-drv-wacom-0.23.0-5.fc20.x86_64
xorg-x11-fonts-ISO8859-1-100dpi-7.5-9.fc20.noarch
xorg-x11-fonts-ISO8859-1-75dpi-7.5-9.fc20.noarch
xorg-x11-fonts-Type1-7.5-9.fc20.noarch
xorg-x11-font-utils-7.5-18.fc20.x86_64
xorg-x11-glamor-0.5.1-3.20140115gitfb4d046c.fc20.x86_64
xorg-x11-server-common-1.14.4-7.fc20.x86_64
xorg-x11-server-utils-7.7-2.fc20.x86_64
xorg-x11-server-Xorg-1.14.4-7.fc20.x86_64
xorg-x11-utils-7.5-12.fc20.x86_64
xorg-x11-xauth-1.0.7-4.fc20.x86_64
xorg-x11-xinit-1.3.2-9.fc20.x86_64
xorg-x11-xkb-utils-7.7-8.fc20.x86_64
xpdf-3.03-8.fc20.x86_64
xulrunner-27.0-1.fc20.x86_64
xz-5.1.2-8alpha.fc20.x86_64
xz-libs-5.1.2-8alpha.fc20.x86_64
```

```
yajl-2.0.4-3.fc20.x86_64
yelp-3.10.1-1.fc20.x86_64
yelp-libs-3.10.1-1.fc20.x86_64
yelp-xsl-3.10.1-1.fc20.noarch
yum-3.4.3-137.fc20.noarch
yum-langpacks-0.4.3-1.fc20.noarch
yum-metadata-parser-1.1.4-9.fc20.x86_64
yum-utils-1.1.31-20.fc20.noarch
webrtc-audio-processing-0.1-5.fc20.x86_64
wget-1.14-12.fc20.x86_64
which-2.20-6.fc20.x86_64
wireless-tools-29-10.1.fc20.x86_64
wireshark-1.10.6-2.fc20.x86_64
wireshark-gnome-1.10.6-2.fc20.x86_64
wpa_supplicant-2.0-8.fc20.x86_64
wqy-zenhei-fonts-0.9.46-11.fc20.noarch
wvdial-1.61-8.fc20.x86_64
xcb-util-0.3.9-3.fc20.x86_64
xdg-user-dirs-0.15-2.fc20.x86_64
xdg-user-dirs-gtk-0.10-3.fc20.x86_64
xdg-utils-1.1.0-0.22.rc2.fc20.noarch
xfsprogs-3.1.11-2.fc20.x86_64
xkeyboard-config-2.10.1-1.fc20.noarch
xl2tpd-1.3.1-14.fc20.x86_64
xml-common-0.6.3-40.fc20.noarch
xmlrpc-c-1.32.5-1903.svn2451.fc20.x86_64
xmlrpc-c-client-1.32.5-1903.svn2451.fc20.x86_64
xorg-x11-drv-ati-7.2.0-3.20131101git3b38701.fc20.x86_64
xorg-x11-drv-evdev-2.8.2-1.fc20.x86_64
xorg-x11-drv-fbdev-0.4.3-10.fc20.x86_64
xorg-x11-drv-intel-2.21.15-5.fc20.x86_64
xorg-x11-drv-mga-1.6.2-8.fc20.x86_64
xorg-x11-drv-modesetting-0.8.0-2.fc20.x86_64
xorg-x11-drv-nouveau-1.0.9-2.fc20.x86_64
xorg-x11-drv-openchrome-0.3.3-2.fc20.x86_64
xorg-x11-drv-qxl-0.1.1-3.fc20.x86_64
xorg-x11-drv-synaptics-1.7.4-5.fc20.x86_64
xorg-x11-drv-vesa-2.3.2-10.fc20.x86_64
xorg-x11-drv-vmmouse-13.0.0-6.fc20.x86_64
xorg-x11-drv-vmware-13.0.1-3.20131207gita40cbd7b.fc20.x86_64
xorg-x11-drv-wacom-0.23.0-5.fc20.x86_64
xorg-x11-fonts-ISO8859-1-100dpi-7.5-9.fc20.noarch
xorg-x11-fonts-ISO8859-1-75dpi-7.5-9.fc20.noarch
```

```
xorg-x11-fonts-Type1-7.5-9.fc20.noarch
xorg-x11-font-utils-7.5-18.fc20.x86_64
xorg-x11-glamor-0.5.1-3.20140115gitfb4d046c.fc20.x86_64
xorg-x11-server-common-1.14.4-7.fc20.x86_64
xorg-x11-server-utils-7.7-2.fc20.x86_64
xorg-x11-server-Xorg-1.14.4-7.fc20.x86_64
xorg-x11-utils-7.5-12.fc20.x86_64
xorg-x11-xauth-1.0.7-4.fc20.x86_64
xorg-x11-xinit-1.3.2-9.fc20.x86_64
xorg-x11-xkb-utils-7.7-8.fc20.x86_64
xpdf-3.03-8.fc20.x86_64
xulrunner-27.0-1.fc20.x86_64
xz-5.1.2-8alpha.fc20.x86_64
xz-libs-5.1.2-8alpha.fc20.x86_64
yajl-2.0.4-3.fc20.x86_64
yelp-3.10.1-1.fc20.x86_64
yelp-libs-3.10.1-1.fc20.x86_64
yelp-xsl-3.10.1-1.fc20.noarch
yum-3.4.3-137.fc20.noarch
yum-langpacks-0.4.3-1.fc20.noarch
yum-metadata-parser-1.1.4-9.fc20.x86_64
yum-utils-1.1.31-20.fc20.noarch
zd1211-firmware-1.4-9.fc20.noarch
zeitgeist-libs-0.9.14-2.fc20.x86_64
zenity-3.8.0-3.fc20.x86_64
zip-3.0-9.fc20.x86_64
zlib-1.2.8-3.fc20.x86_64
```