Secure Web

Hardware Sizing Guide

Table of Contents

List of Figures

1. Introduction

This document aims to provide guidelines for choosing hardware for Secure Web. It gives the reader an overview about performance critical points when planning the proxy environment in mid- and enterprise level scenarios.

After reading this document, the reader should be able to fully understand the key points of performance relevant parts of server infrastructure and plan and design accordingly. It also gives a good understanding of tools needed to measure and monitor performance metrics.

The guidelines have been written for hardware appliances, but all recommendations and advices also apply to virtual infrastructure.

2. Sizing Guide

Sizing hardware and virtual appliances for production use of Secure Web is not always a straight-forward task. Various factors, like network topology, number of concurrent users and the enabled features of Secure Web must be taken into account to properly determine hardware requirements. The following tables should give a rough overview about the required hardware to operate Secure Web.

Anti-Virus scanning, SSL Intercept and Caching are key features regarding performance of an appliance. Please note that the more you get to the limits of a specific platform, the more these features will have an impact on the overall performance. In this case, we strongly suggest to switch to the next higher platform or to think about creating a cluster of Secure Web appliances. We suggest to evaluate both your Internet bandwidth as well as concurrent users and choose the highest platform. For example, in an environment with a 10 MBit/s Internet connection and 250 users we suggest to go with the RS-400 appliance even though the bandwidth can still be covered with the DS-100.

The first chart compares the appliances against the available downstream Internet bandwidth.

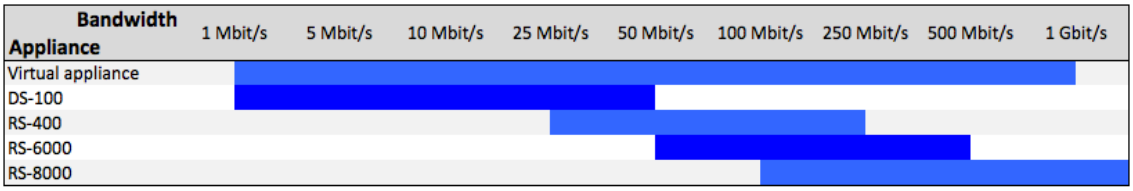

Figure 2.1. Sizing by Internet bandwidth

The second chart compares the appliances against the number of concurrent users using the proxy system.

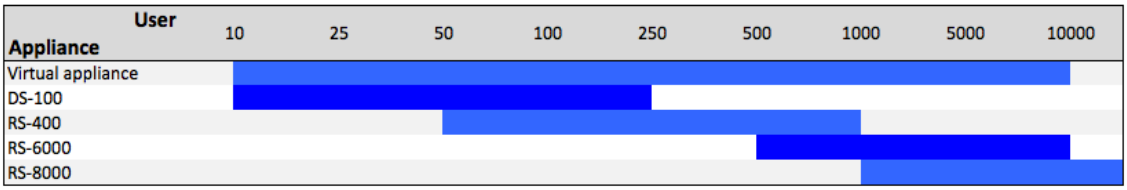

Figure 2.2. Sizing by concurrent users

3. CPU

Secure Web is designed to take full advantage of todays CPU architecture. Multiple threads of execution are a key point of the proxy engines, which makes sure that the available CPU power is utilized as good as possible.

For good performance characteristics of the Secure Web proxy engine, CPU power is not the main characteristic though. As a rule of thumb, the higher the number of parallel connections is to be expected, the more CPU cores can be used by the proxy engines, because the connection count can be fully distributed across all cores.

CPU power is mainly important for:

- Anti virus scanning
- Deep archive inspection
- SSL interception

For both anti virus scanning and deep archive inspection, CPU power is needed to decompress archives and analyse file contents. SSL interception, thus de- and encrypting SSL traffic on the fly, is a CPU heavy operation because of the complex mathematics used by the encryption algorithms.

3.1. Measurement

CPU utilization can easily be monitored on a Linux machine with use of commands like vmstat or top. The important processes to look for are *sweb* for the HTTP and FTP proxy engines as well as *smail* for the e-mail relevant proxies.

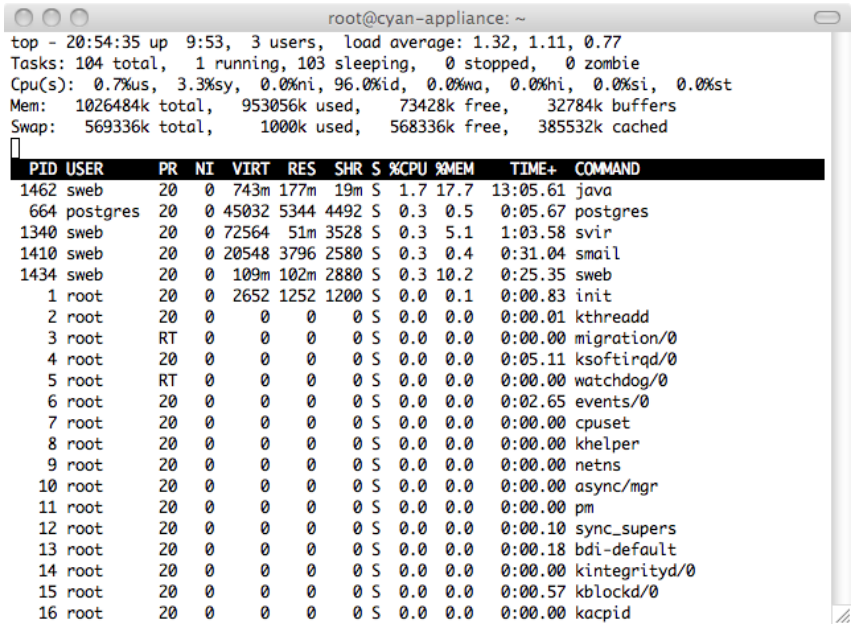

Figure 3.1. Screenshot from Linux "top"

Be carefull with judging of CPU performance using the *load average* values from eg *top*. The *load average* contains a lot of performance metrics, including harddisk I/O, and should only be used to get an overview about the system status as a whole. High values do not necessarely indicate a performance problem. Comparing the *load average* value to other systems should also be avoided, because the numbers can not be carried over to another system.

Thread counts can be queried with help of *ps* or by using *netstat* to look for open connections to the proxies.

```
$ ps -efL|grep bin/sweb|wc -l
12
```

In the examples above, 12 threads for the Secure Web process *sweb*, which provides the HTTP and FTP proxy functionality, is the initial thread count that is used for internal processing. Every connection that is accepted by the proxy engine is assigned to an additional thread.

```
$ netstat -nlat|grep 8080
tcp        0      0 0.0.0.0:8080            0.0.0.0:*               LISTEN
tcp        0      0 10.1.254.250:8080       10.1.3.15:55336         ESTABLISHED
tcp        0      0 10.1.254.250:8080       10.1.3.15:55337         ESTABLISHED
```

There is a limit in the Linux kernel of around 60.000 parallel TCP connections that can be made on one IP address of the proxy machine. In this example, there are two active connections to the HTTP proxy.

4. RAM

RAM is important if a high number of parallel connections is to be expected or if memory caching is to be used in the HTTP and FTP proxy engines. All RAM that is not used by processes is used for in-kernel buffers and caching of eg file system relevant data, so it is a good idea not to assign all memory to Secure Web processes.

Initially, the Secure Web HTTP and FTP proxy engine *sweb* allocates approximately 100 MB of RAM for the URL database.

Each connection that is handled by the Secure Web proxy engines takes 128 Kbyte of RAM. During processing, buffers and data structures of about 64 Kbytes add up to at least 192 Kbyte of RAM used per active connection.

Each parallel anti virus scan operation needs 2 MB of RAM.

If caching for HTTP and FTP protocols should be enabled, it's a good advice to dedicate a good amount of free RAM to the memory cache. The settings therefor can be found in the administrative interface of Secure Web under -Proxy Settings_ → *Web Proxy* → *Cache*.

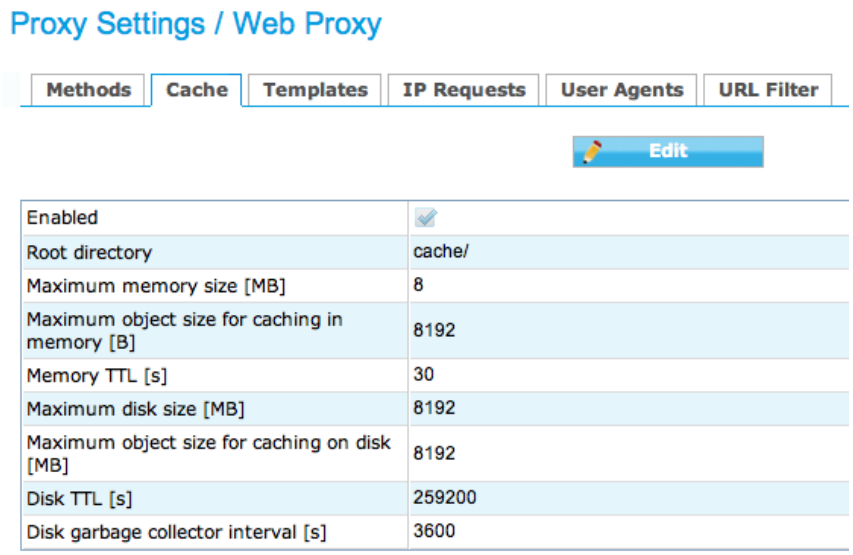

Figure 4.1. Screenshot of Secure Web Cache Settings

This example shows the default configuration of memory and disk cache after installing Secure Web. It only dedicates 8 MB of RAM and only consideres files of up to 8192 bytes size (8 Kbyte) to be put in the memory cache.

On a machine with 8 GB of RAM, which is commonly found in todays IT infrastructure, an administrator can easily dedicate 1 GB or more of RAM for memory caching and raise the maximum object size to up to 1 MB of data.

The data structures needed for the disk cache can easily take up to 1 GB of RAM, depending on the size of the disk cache, and is tied to the amount of file objects. A typical cache of 60 GB contains up to 350.000 files on disk and takes almost 350 Mbyte of RAM for the cache index. 1 cached file on disk is about 1 KByte of data needed in the cache index in memory.

The amount of files can be reduced by raising the minimum object size. There is no dedicated setting for this, as all files that are not put into the memory cache are stored in the disk cache instead. So raising the maximum object size of the memory cache also defines a higher minimum objects size of the disk cache.

4.1. Measurement

RAM usage can easily be checked by tools like *top*. The *%MEM* column of the relevant process, *sweb* for the HTTP/FTP proxy engine and *smail* for the e-mail proxies, will tell you the memory consumption based on the total amount of memory available on the system. *RES* contains the memory that is wired and used actively by the process.

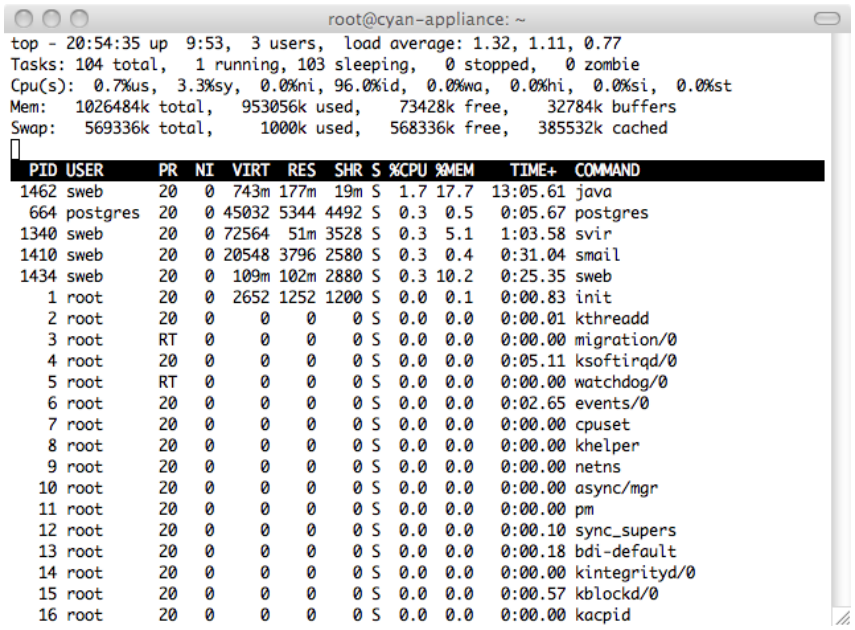

Figure 4.2. Screenshot from Linux "top"

Please note that on 32-bit systems, only up to 3 GB of memory can be used by the user mode processes. 1 GB is reserved for the kernel address space if a kernel without PAE (physical address extension) is used. PAE brings a performance penalty though and should be avoided. A 64-bit installation should be considered if high amount of memory usage is an issue.

5. Harddisk

Harddisk performance is a cruical point when designing the hardware infrastructure for the Secure Web environment.

On the operating system side, do not expect much disk I/O. Beside writing log files, the operating system will not have to much disk access.

When it comes to disk caching and anti virus scanning though, disk I/O is the main performance characteristic to take care of. The more parallel connections are to be expected to the HTTP and FTP proxy, the more disk I/O is needed to store and deliver from the disk cache.

When choosing a harddisk for caching, take the following characteristics into account:

- SATA 7k, avg. 60 MB/sec, avg. 9.5 ms average latency, about 100 IOPS
- SAS 15k, up to 200 MB/sec, avg. 3.5 ms average latency, about 300 IOPS
- SSD, up to 400 MB/sec read/write performance, almost no latency, up to 8.600 IOPS

For a high performance cache system, a SSD is certainly a good choice to be used for the sole purpose of caching. A SAS 15k disk of 147 GB of size is usually a good tradeoff between speed and costs though. **Do not share the cache disk with anything else but dedicate a single disk for caching**.

Please note that RAID may be good providing fault tolerance for the operating system and Secure Web installation, but **must be avoided** for the cache disk. RAID controllers can not keep up with IOPS needed for cache operations and fault tolerancy is certainly not needed for this kind of data.

5.1. Mesurement of disk activity

Disk I/O can be checked and monitored with tools like *atop* (press D to show disk statistics after launching atop). Please note that per-process disk I/O accounting is only provided by Linux kernels 2.6.18 and higher.

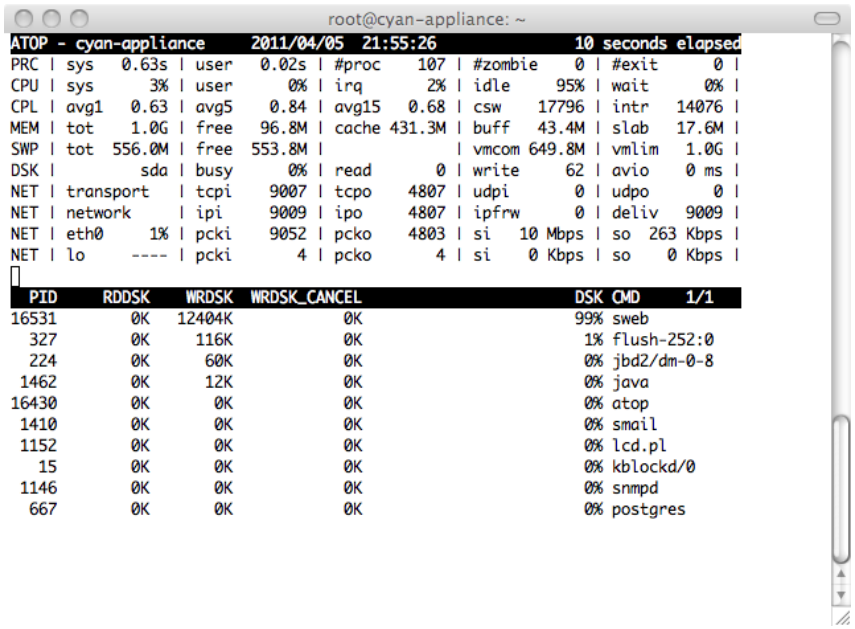

Figure 5.1. Screenshot from Linux "atop"

In the *CPU* line, the time percentage wasted by the CPU waiting for I/O operations to complete are shown in the *wait* column. The higher the value, the more CPU time is lost because the underlying storage system can't keep up feeding data.

The *DSK* line on top show I/O summaries of all disks found on the system. It displays the read and write operations on the given harddisk in the monitored time interval as well as the average I/O time and a *busy* percentage. A busy percentage of 70% or higher is considered critical and will have a noticeable impact on the machines performance.

The block on the bottom shows a list of processes, ordered by disk consumption from top to bottom, and the amound of Kbytes transfered to and from the disk.

Disk performance can be tested with *hdparm*, which measures the raw read rate of the selected hard disk:

```
$ hdparm -t /dev/sda
/dev/sda:
 Timing buffered disk reads:  380 MB in  3.01 seconds = 126.34 MB/sec
```

5.2. Troubleshooting

If disk I/O is too high and causes performance problems, check your cache settings and set the disk cache to only save files of a certain minimum size. This can be achieved by raising the maximum object size for caching in memory to a higher value. Files that are not cached in memory are cached on disk instead.

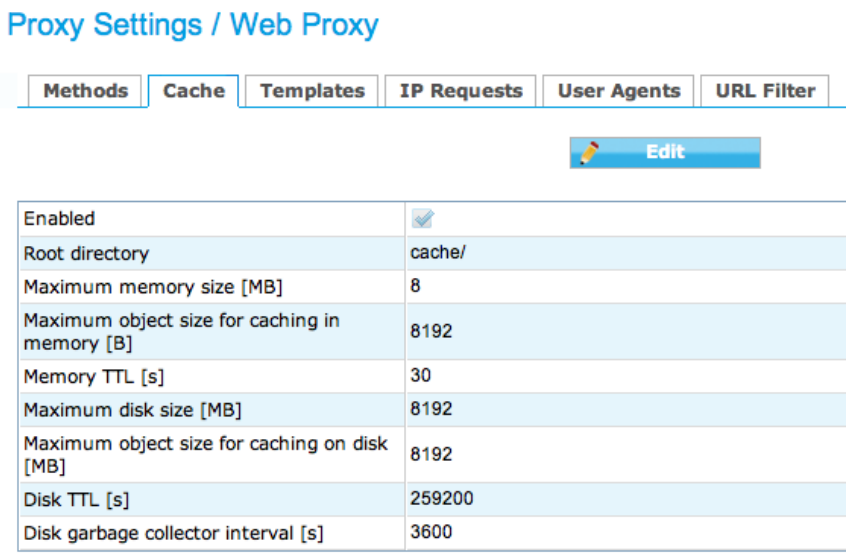

Figure 5.2. Screenshot of Secure Web Cache Settings

Disk garbage collector interval can also be raised to be run in longer intervals, to reduce the stress on the harddisk and leave more I/O operations for the proxy cache engine to store and deliver data.

If other processes on the proxy machine take up too much disk I/O, like periodic tasks compressing log files or creating backups, consider moving these jobs to times where proxy usage is low.

6. Filesystem

The choice of file system is very important for disk caching and anti virus scanning. A filesystem must be capable of efficiently handling a large amount of files in the disk cache. Numbers of up to 1.000.000 files are common in caches with 60 GB to 100 GB of size.

CYAN Networks recommends the usage of ext3 for caching because of its good performance and stability. We suggest to use the following options for the file system for best performance:

- dir_index: directory indexes speed up access of directories with large amounts of files
- journal_data_writeback: only use the journal for metadata
- data=writeback: only use the journal for metadata
- noatime: deactivates the last access timestamp on files
- nodiratime: deactivates the last access timestamp on directories

Disabling the journal for data runs the risk of data corruption after crashes, but greatly boosts the performance of the filesystem. Since cache data is considered non-critical, this should be considered when looking for performance enhancements.

During tests we saw bad performance and strange behaviours from JBD (Journaling Block Device) when used with the ext4 filesystem. Because of this, we can not recommend the usage of ext4 for production use yet.

Filesystems like btrfs, reiserfs4, xfs and jfs have not been thoroughly tested by CYAN Networks and may bring similar results or even better performance when tuned accordingly. Xfs and jfs is known to have very good performance.

6.1. Measurement

File system performance can be tested with a tool like *bonnie++*, which is also used for stress testing file systems to detect bugs in high load environments.

6.2. Troubleshooting

Ext3 file system options can be queried with *tune2fs*:

```
$ tune2fs -l /dev/sda1
tune2fs 1.41.11 (14-Mar-2010)
Filesystem volume name:   <none>
Last mounted on:          <not available>
Filesystem UUID:          e5f4854b-e810-4158-bb15-eea1aec11e0c
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
Filesystem features:      dir_index filetype sparse_super
Default mount options:    journal_data_writeback
```

If the file system feature *dir_index* is not present, it can be enabled with *tune2fs -O dir_index /dev/sda1*.

journal_data_writeback can be enabled with help of *tune2fs -o journal_data_writeback /dev/sda1*.

Please choose the appropiate harddisk for the above commands and back up your data before changing any filesystem parameter, as it may render your parition useless if used incorrectly.

The mount options for the paritions can be found in /etc/fstab:

```
/dev/sda1       /cache          ext3    noatime,nodiratime,data=writeback       0
```

noatime and *nodiratime* can be set without fear of risking your data. *data=writeback* needs to above described filesystem option *journal_data_writeback* to work correctly.

7. Database

The database system is solely used for the reporting subsystem (CRS) and as such must be designed to take a huge amount of data for reporting.

For test and evaluation environments, CYAN Networks ships packages which set up a local database using PostgreSQL and configure the reporting system automatically to use the database.

For production environments, a local database on the Secure Web machine must not be used for running the reporting database. Disk I/O during report data feeding and generation as well as database maintenance tasks like vacuum is too high to allow a coexistence with Secure Web.

As a rule of thumb, a single user performs 1.200 request per day on average, which means that every user adds 1.200 rows into the database per day. A single row in the database comes down to 500 bytes of storage needed to store the actual data. The final database partition should leave a fair amount of disk space left free for database maintenance jobs, like *vacuum* on PostgreSQL databases.

7.1. Recommendations

CYAN networks recommends PostgreSQL to be used as the database system for reporting if free and open source software (OSS) is favored. PostgreSQL should be configured to use as much memory as possible for sort operations, to speed up ORDER and GROUP queries.

Whenever possible and supported by expert IT staff, an enterprise database like Microsoft SQL, Oracle or IBM DB2 (Oracle and DB2 support with Secure Web 2.1) should be considered.

With database aggregation enabled in CRS, the size of the database can be reduced by roughly 80%, with the tradeoff that all data is aggregated on an hourly basis and URL path information is lost in this process. Disk space usage can be reduced and report generation times enhanced considerabely.

The reporting database is delivered without indexes created. Because of the high number of different reports, creating indexes for all possible combinations would significantly grow the database in size and slow down the INSERT performance considerabely. Please consult the database administrators to tune the database accordingly and create indexes where necessary.

8. Contact information

Please contact me for any questions or comments you have:

Markus Cserna m.cserna@cyan-networks.com +43 (1) 33933-203

Appendix A. Contact data

A.1. How to contact our sales department

Tel.: +43 (1) 33933-0

Email: sales@cyan-networks.com

A.2. How to contact our support department

Tel.: +43 (1) 33933-333

Email: support@cyan-networks.com

A.2.1. Getting Support

In case you should have any technical problems, or questions and would like to get support from our team, we kindly ask you to provide us with the following information:

- Description of your question or problem
- The version information of the product:
  - The version information of Secure Web can be found after logging into the Web Admin Interface in the top part of the screen:

Figure A.1. Version information of the Secure Web

  - The version information of the Reporting System can be found after login in the top part of the screen of the Web Admin Interface:

Figure A.2. Version information of the Reporting System

  - All the information contained in the screen found in menu *Services* / *Services* / *Overview*
- In the case authentication is activated, provide us with the method in place (via Windows Agent, via Linux Agent, etc.)
- The deployment method of the Appliance (Out-of-line, In-Line, DMZ)
- The operation mode of the Appliance (dedicated mode, transparent mode)

- Information about the environment (proxy cascades that are used, firewalls and gateways involved in the infrastructure that are of relevance to the Appliance)

The appliance interface provides the possibility to create a support package that includes the configuration and log files of the system. This package can help us to track down the issue easier and faster. Please attach this package to your e-mail.

In order to create a support pack, navigate to menu *Appliances / Maintenance / Support* and click on the *Download* button. You may select the files you want to provide to our support team and then download a package, which we kindly ask you to send to our support email address.

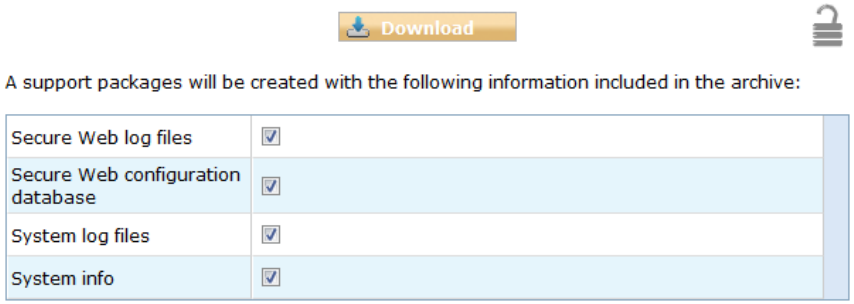

Figure A.3. Support Package

Further documentation about the product as well as technical white papers that describe certain use cases can be found in our documentation repository on our homepage:

http://www.cyan-networks.com/documentation