



## Security Package Version 3.00 Includes fully featured Firewall!

Released: 31. Dec. 2001

Last Updated: 14. Feb 2003



At the moment all out-of-the-box SUN/Cobalt RaQs are vulnerable to a wide range of exploits. Not all of them can be fixed by just applying the latest OS patches from Cobalt. Even then, there are suggested changes to configuration files which will further enhance the security aspects of any Cobalt RaQ. Nonetheless keeping up to date on the OS patches is highly recommended for any Cobalt RaQ owner.

To prepare and to defend your valuable business assets against hacks we assembled comprehensive, [multi-layered Intrusion Detection and Prevention software](#) which offers the best integrated protection available for SUN/Cobalt RaQs.

Up to date we installed version 1 and version 2 of our Security Package on well over 900 SUN/Cobalt RaQ3's and RaQ4's and not a single server secured that way has been hacked again. The even better news is that Security Package 3.1.0 is out, which is by far improved in all regards and now also includes a **fully featured and rock solid firewall solution**.

The security package we offer creates several lines of defence which protect your business assets and those of your customers. The question, which only you can answer, is: **Can you afford to do without?**



### **This is what we offer:**

When establishing an Intrusion Detection System Process, a defence in depth process concentrating on software, networks, and hardware is the key to success as [SANS claims](#). Our approach to that follows these lines and creates several layers of protection.

In short this includes the following software packages (detailed description follows):

- Installation of all missing patches
- Improved configuration files
- Installation of OpenSSH
- Installation of our custom built Firewall
- Installation of Port Sentry in "HoneyPot"-mode
- Installation of LCAP to prevent loading of kernel modules
- Installation of Logwatch
- Installation of FCheck (similar to Tripwire)
- Installation of automated CHKROOTKIT

All these programs are available as OpenSource software and have been modified by us to suit the specific needs and environmental conditions on a SUN/Cobalt RaQ.

## A) Enhancing server integrity:

### 1.) Updating the server with all the latest patches

This is what **Autopatch** does. Upon installation of the Security Package it connects to SUN/Cobalt and if there are any patches which your server is missing, then it'll install them automatically. Once that's done an email is sent to the Admin email account and lists all newly installed patches.

#### Usage information:

**Autopatch** is installed in `/home/security/Updater` and its only configurable option is the email address where it sends mails to. This defaults to "admin".

If you ever want to run Autopatch again to install one or more missing patches, then ssh into your server as admin and do the following steps:

```
su -  
(enter your admin password again)  
cd /home/security/  
./Updater
```

That will run the Autopatch utility and will commence with the installation of all missing patches. Your server **might** reboot at the end of the run of Autopatch if one (or more) patches require a reboot.

To change the email address where all Security Package related tools send email to just edit the following file: `/home/security/admin-email.cfg`

### 2.) Improvements of existing services and configuration files

Upon installation of the Security Package we go through several important configuration files and apply small tweaks which will enhance the reliability and stability of services. We will also make sure that all important services are properly configured to offer potential attackers the least leeway in regards to intrusions.

### 3.) Installation of OpenSSH

OpenSSH is a Telnet replacement and does away with the unencrypted transmission of usernames, passwords and data. So when you use SSH to connect to the server, then nobody can eavesdrop on your communication to gather valuable information.

#### Usage information:

The configuration files for SSH are located in `/etc/ssh/` and the most important one of it is `sshd_config`. That one is for the SSH daemon which listens on port 22 for connections. SSH is configured to only accept connections with SSH protocol 2. User **root** is not allowed to log in, so you need to log in as **admin**.

You need to use an SSH enabled client for the connection from a remote computer. For Microsoft Windows you can either use [PUTTY](#) or [SecureCRT](#).

OpenSSH includes also a client application on your server with which you can connect to other remote machines. Just type `ssh -ladmin <IP-ADDRESS>` to establish a connection to another RaQ running SSH to login there as user **admin**.

Another included command is **scp**, which stands for “secure copy”. With that you can copy files to and from the server in fully encrypted fashion. The syntax to send a file from a remote Linux server to your RaQ could be like this:

```
scp FILENAME admin@yoursite.com:/home/sites/home/web/
```

This will send file FILENAME to the server yoursite.com. You will be asked for the password of user admin. The file will be put into the directory /home/sites/home/web/ provided that user admin is allowed to write files into that directory.

There is also tool available for Microsoft Windows machines called [WinSCP](#). It looks similar to WS\_FTP and uses SCP to securely up- and download files to and from your server.

For more information on the scp command look at the manpages for this command with `man scp`.

## **B) Perimeter Defence:**

### **4.) Installation of our custom built Firewall for SUN/Cobalt RaQs**

We adapted Godot's [gShield-1.5.6](#) to the specific requirements needed for securing a SUN/Cobalt RaQ. Our adapted version of gShield includes many back ports from the recent IPTABLES driven 2.7.X series of gShield and is very easy to configure.

The firewall determines the network settings all by itself, even when you change your network settings. The server administrator can enable and disable services from one single configuration file. If you ever want to block certain IP-addresses or IP-address ranges, then just add them to the "blacklist" to ban them permanently.

**Any strict Firewall contains the risk that you lock yourself out of the server if you make the wrong modifications. We tried to keep this risk down to the minimum by a straightforward set-up and configuration process.**

If ever something goes wrong and you lock yourself out, then you can reboot your server from the front panel. The Firewall will be started five minutes after the server is up and running. During that period of time you can log in and disable the start-up procedure of the firewall by setting a switch in the configuration file from YES to NO. This allows you to fix any problems you might encounter.

The time remaining until firewall initialisation will be shown on the LCD display on the RaQs front panel, which will also give you a reminder if you forgot to enable the Firewall by accident.

## Usage information:

All firewall related files are installed in [/home/security/firewall](#) and the main start-up script is called [gShield.rc](#)

The following commands are available:

[/home/security/firewall/gShield.rc start](#)  
[/home/security/firewall/gShield.rc stop](#)  
[/home/security/firewall/gShield.rc restart](#)

All firewall options can be configured in the main configuration file [gShield.conf](#). In there you can define if specific services like HTTP, FTP, SMTP, MySQL (and others) are available to the outside world or not. Be sure to take a look at this file and its comments.

The firewall will be started 5 minutes after the server has been (re)started. This is achieved through a line in [/etc/rc.d/rc.local](#) which will call [/home/security/rc\\_security.sh](#) which is the central start-up scrip for all security package related services.

Should you ever lock yourself out of the server by improperly configuring the firewall, then reboot the machine from the front panel. Once the server is up again you have 5 minutes to login before the firewall kicks in. To prevent the firewall from starting edit the file [/home/security/init.firewall](#) as user root and set the switch **STARTFIREWALL** in there from **YES** to **NO**.

## How to block certain IP-addresses and networks with the firewall:

Go to the directory [/home/security/firewall](#) and edit the file [blacklist](#)

Put in any offending IP address or if you want to block an entire network, then put in the proper network address.

### Examples:

66.33.22.10 Will block just this single IP-address  
66.33.22.0/32 Will block the entire network from 66.33.22.1 up to 66.33.22.255  
66.33.0.0/24 Will block the entire network from 66.33.0.1 up to 66.33.255.255

[Always put in IP-addresses and not hostnames!](#)

Be sure to read the README in the firewall directory which goes a little further into details.

## 5.) Portsentry operates in “Honeypot”-mode

This is part of the Intrusion Detection Process. A Portscan is a very nice way for a system administrator to check the health and status of his server. However, nobody

else except the system administrator has a legitimate reason to run a Portscan on your server to probe it for open ports and services. In fact a Portscan is often the first stage of an intrusion attempt.

So we not only make sure that the alarm goes off when a burglar rattles your fence, but we also pull in the drawbridge to block his access to the server.

We achieve this through carefully crafted holes in the Firewall behind which **Portsentry** is listening for connections. Portsentry only listens on about half a dozen ports which are usually not used by any services.

**Chances that a legitimate user locks himself out are less than zero.**

Any Portscan on ports between 1-1023 will run into at least one of the honeypots we laid out. When that happens, Ipchains kicks in and adds a rule to the Firewall which will deny access to the burglar which triggered your defences.

Once that happens an alert email is sent out to the Admin email account of the server.

#### **Usage information:**

Portsentry is located in [/home/security/portsentry/](#) and is launched at server start-up through [/home/security/rc\\_security.sh](#)

To restart Portsentry you can use the command [/home/security/portsentry/start.sh](#)

Portsentry listens in UDP and TCP mode on the following **ports: 6, 8, 12, 26, 135, 136, 900 and 911**. If someone connects to those (usually unused) ports, then the connection will be dropped and a firewall rule will be generated to block the originating IP address. Those firewall rules are reset once per day at 11:01 p.m.

Portsentry is only able to listen on the above ports when the Firewall operates in Honeypot-mode, which is set to YES by default in [/home/security/firewall/gShield.conf](#)

## **C) Inner Defence Layer**

If ever someone manages to get through the outer layer of defences (by exploiting a yet unknown vulnerability in a running service), then we sure want to know about it. Therefore our Security Package closely monitors the file system for changes and limits the damage which a hacker can do to the system. We achieve this through the following steps:

### **6.) Installation of LCAP to prevent loading of kernel modules (LKM's)**

Linux kernel versions 2.2.11 and greater include the idea that you can load modules with additional capability into the kernel. Like network drivers or SCSI support.

Generally this is a good idea. However, a couple of root-kits available in the Internet allow intruders to load Linux Kernel Modules (so called LKMs) into the kernel, which

will then effectively hide all hacker processes and files which the intruder unleashes. Even user "root" will then be unable to do anything against this.

So we install LCAP to effectively removed the ability to load further kernel modules once all legitimate kernel modules have been loaded upon start-up of the server.

Any attacker which wants to load a kernel module will have to delete the LCAP and has to reboot the server forcefully for the change to take effect. Any such reboot and system change will not go unnoticed.

### **Usage information:**

LCAP will be started at server start-up through a call in [/home/security/rc\\_security.sh](#)

## **7.) Installation of Logcheck**

Logcheck checks your server's logfiles for unusual system events. If it detects something worthy of interest it sends an email to a specified email address. Usually the admin email account of a RaQ.

Logcheck will report to you who logs into your machine by SSH (or Telnet, if enabled) and by FTP. It will also report failed login attempts, unusual system events (like when the server or an individual service reboots).

Logcheck is able to distinguish between intrusion alerts, normal system events and unusual system events.

Based on this emailed reports you will always be up to date about everything going on at the server, without having to sort through hundreds of megabytes of logfiles.

### **Usage information:**

Logcheck is installed in [/home/security/logcheck/](#) and is started through the command [/home/security/logcheck/logcheck.sh](#)

A cronjob for user **root** will run it every 15 minutes. The email address to which Logcheck sends its reports is defined in [/home/security/admin-email.cfg](#)

## **8.) Installation of FCheck (similar to Tripwire)**

[FCheck](#) and has been written by Michael A. Gumienny and is an integrity checker written in Perl. Upon server start-up and twice per day it will check vital directories and system files for modifications, additions and deletions. Any modifications will be reported in a detailed email to the server administrator.

Nobody will be able to install (or modify) system files without the system administrators knowledge.

### Usage information:

FCHECK is installed in `/home/security/fcheck/` and the command to start it from the command line is `/home/security/fcheck/fcheck`

The following common options exist:

`./fcheck -a` (Automatic run, shows output on screen)  
`./fcheck -ca` (Regenerates the file-state database)

Automatic run of FCHECK and mailing of the output to user **admin** (as defined in `/home/security/admin-email.cfg`) can be triggered through the command `./frun.sh`

That command is executed upon server start-up through `/home/security/rc_security.sh` and twice per day at 10 a.m. and 10 p.m.

The FCHECK configuration file `fcheck.cfg` contains the information which directories are included in the check and which ones are specifically excluded. You can adapt this to your own needs if you want to extend or reduce the scanning.

## 9.) Installation of automated CHKROOTKIT

[Chkrootkit](#) is a diagnostic tool which will scan all vital system binaries to check if they have been replaced with tampered versions. Many rootkits the hackers use will do just that. Additionally this tool will check if your network card(s) are in promiscuous mode (aka.: sniffing the network), it will check if your logfiles have been manipulated and it will check for hidden processes.

However, the hidden process check can and will sometimes report hidden processes when there are none. Please be aware of these *\*false\** alarms which will happen mostly when you're running many dynamic processes. Like Apache or MySQL. Why does it happen? Chkrootkit compares the processes in the `/proc/` directory with those shown by the command `"ps"`. If both outputs don't match, then it'll sound an alert. However, the comparison takes a few moments and if a process ends (naturally) during the comparison, then that will cause a false alarm.

The diagnostic output of Chkrootkit will be emailed to the admin account of your RaQ.

### Usage information:

CHKROOTKIT is installed in `/home/security/chkrootkit/` and the command to run it manually with output redirected to the screen is `./chkrootkit`

To run it non-interactively with sending the output by email to **admin** use the command `./check.sh`

The output email address can be specified in `/home/security/admin-email.cfg`

CHKROOTKIT is called automated through a cronjob of user **root** at 11:02 p.m.