



# RedCastle®

Copyright ©2002–2007 REDGATE Co., Ltd.

---

**Secure OS Solution for Asianux**

## **RedCastle Reference Manual**

**2002-2007 REDGATE Co., Ltd., All rights reserved**

## **Notices**

This manual provides the information about RedCastle® (Secure OS Solution).

### **Copyright**

© 2002 – 2007 REDGATE Co., Ltd. All rights reserved.

Any parts of this manual can't be reproduced, adapted, or translated into other languages without REDGATE's prior written approval unless it is permitted by the copyright.

### **Limitation of Warranty**

This manual is provided "AS IS" and disclaims all expression or implied warranties, including but not limited to the implied warranties or merchantability, fitness for a particular purpose.

### **User Note**

This manual can make the technical inaccuracy or the typing error. It can be changed on regular basis after it is updated. This change is applied to revision. REDGATE Co., Ltd can change or improve the program or product described in this manual.

### **Trademarks**

REDGATE, RedCastle, RedCastle for Asianux, and each logo is a registered trademark of REDGATE Co., Ltd.

# Table of Contents

<b>CHAPTER 1 OVERVIEW</b> .....	<b>1</b>
<b>1.1 Purpose</b> .....	<b>1</b>
<b>1.2 Configuration</b> .....	<b>1</b>
<b>1.3 Main Functions of RedCastle</b> .....	<b>1</b>
<b>CHAPTER 2 REDCASTLE PRODUCT CONCEPT</b> .....	<b>3</b>
<b>2.1 Security Attribute of Subject</b> .....	<b>3</b>
<b>2.2 Secured User &amp; Security Group List</b> .....	<b>3</b>
<b>2.3 root Separation</b> .....	<b>4</b>
<b>CHAPTER 3 MAIN FUNCTIONS</b> .....	<b>6</b>
<b>3.1 MAC (Mandatory Access Control)</b> .....	<b>6</b>
<b>3.1.1 Access Control Rule</b> .....	<b>6</b>
<b>3.1.2 Applying Target</b> .....	<b>7</b>
<b>3.1.3 Operation Mode for Applying MAC</b> .....	<b>7</b>
<b>3.2 ACL (Access Control List)</b> .....	<b>8</b>
<b>3.2.1 ACL Policy and Definition of Object</b> .....	<b>8</b>
<b>3.2.2 Definition of Subject and User Role</b> .....	<b>8</b>
<b>3.2.3 Rule Settings</b> .....	<b>9</b>
<b>3.2.4 Operation Mode</b> .....	<b>9</b>
<b>3.3 SU Control</b> .....	<b>10</b>
<b>3.4 Resetting of Subject Security Attribute</b> .....	<b>11</b>
<b>3.5 Advanced Functions</b> .....	<b>11</b>
<b>3.5.1 Commands Control</b> .....	<b>11</b>
<b>3.5.2 SETUID Control</b> .....	<b>12</b>
<b>3.5.3 KILL Deny</b> .....	<b>13</b>
<b>CHAPTER 4 OPERATION</b> .....	<b>14</b>
<b>4.1 RedCastle Configuration</b> .....	<b>14</b>
<b>4.2 Initial Operation Environment Settings</b> .....	<b>14</b>
<b>4.3 RedCastle Basic Security Policy</b> .....	<b>15</b>
<b>4.3.1 ACL Basic Policy</b> .....	<b>15</b>
<b>4.3.2 Security Attributes Reset List</b> .....	<b>16</b>
<b>4.4 RedCastle Security Functions – Start/Stop</b> .....	<b>16</b>

4.5	How to Apply Main Functions .....	1 7
4.5.1	RedCastle Security Functions – Start/Stop : rgctl Command .....	1 7
4.5.2	RedCastle Operation Environment – rgconf Command.....	1 8
4.5.3	ACL Policy Management – fgroupset Command.....	1 9
<b>CHAPTER 5 BASIC MANAGER .....</b>		<b>2 3</b>
5.1	Introduction .....	2 3
5.2	Connecting and Disconnecting .....	2 3
5.3	Settings .....	2 6
5.3.1	Module .....	2 7
5.3.2	Access Control.....	2 8
5.3.3	Authentication.....	2 9
5.4	ACL Policy Query .....	3 0
<b>CHAPTER 6 OPERATING COMMANDS OF REDCASTLE .....</b>		<b>3 2</b>

## **CHAPTER 1 OVERVIEW**

RedCastle, based on operating system kernel level security technology, both secure your computer at the kernel level and improves the security of your other application-based security systems. By placing the security decisions within the kernel itself, no application can by-pass the security decision-making process. By separating root's power into many different, more limited privileges, and preventing the root user from acquiring more privileges, you no longer need to worry about root exploits in applications since root no longer has any real power.

### **1.1 Purpose**

This manual describes what the security manager must be well aware of with the RedCastle for its appropriate operation.

### **1.2 Configuration**

Chapter 1 describes the purpose, configuration of this manual and main functions of RedCastle briefly.

Chapter 2 describes the important product concept of RedCastle.

Chapter 3 describes the main functions of RedCastle.

Chapter 4 describes how to operate RedCastle properly.

Chapter 5 describes about the Basic Manager for RedCastle management.

Chapter 6 describes the operating commands of RedCastle.

### **1.3 Main Functions of RedCastle**

RedCastle Secure OS for Asianux is divided into two modules; Basic Module and Advanced Module. The Basic Module is provided free of charge to protect the system environment from various security risks. The Basic Module includes the most fundamental security functions to keep your host secure from expectable security vulnerabilities. And the Advanced Module will be charged certain amount of license fee to provide thorough security to your host environment with more flexible Windows based manager module and complete audit trail data management function.

RedCastle is loaded on the server operating system of the security asset and is configured with kernel part doing the Mandatory Access Control and Discretionary Access Control with the application doing the other security functions. The major functions are as follows.

- Reference monitor
- Label-based Mandatory Access Control function
- ACL (Access Control List)-based Discretionary Access Control function
- Role-based Access Control function
- Discretionary Access Control function based on allow/deny list
- Potential violation analysis and corresponding function
- Security log and system log collection & management function
- Security attributes and security policy management function
- Command Line Interface for security management

## CHAPTER 2 REDCASTLE PRODUCT CONCEPT

This chapter describes important concept required to understand and operate RedCastle appropriately.

### 2.1 Security Attribute of Subject

All performances in the system can be described as certain actions conducted upon the objects by the subjects. In these performances, the subjects can be described as processes and the objects would be data such as the files. RedCastle assigns specific security attribute which is composed of security level, security group and security role to every subject by applying MAC model. And it classifies all subjects according to these security attributes. This is just a logical classification done by RedCastle and it does not mean physical classification in the real system. The security attribute of process will be assigned automatically when it is created. Every subject will have its own genuine security attribute and its capable tasks will be determined according to the security attribute. So, which security attribute will be assigned to the processes? The processes will have same security attributes which are defined in the Secured User List.

### 2.2 Secured User & Security Group List

Secured user and security group lists are the most basic data in RedCastle operation and used to determine security attributes of a subject. The security attributes consists of security level, security group and security role. Since the security role will be assigned automatically by the security group, what the security officer need to develop is the secured user and the security group list.

Only the login possible user account which exists in the system can be a secured user. However, root account can not be a secured user. The user account registered in the Secured User List by the Security Officer becomes a secured user. When the Security Officer tries to add new secured user to the List, assigns specific security role (security group) and security level on it. Secured user will be classified into three kinds as the below.

<b>Security Role</b>	<b>Description</b>
<b>SO</b>	Security Officer. RedCastle Administrator. User who can transfer

	to root.
<b>SA</b>	System Administrator. Secured user who can transfer to root.
<b>MU</b>	Secured user who can not transfer to root.

Even though the secured user will be protected by RedCastle, but its account itself is treated as same as usual user for its status. However, when a secured user who has SO or SA role transferred to root by using su command, it makes difference. This will be explained in detail later.

To operate RedCastle, there should be a Security Officer (secured user with SO security role). All works related with RedCastle should be done with the security officer account. For this reason, RedCastle requires one security officer when it is installed at first.

Two security group lists will be provided by default when RedCastle is installed and those are as the below.

<b>Group No.</b>	<b>Description</b>
<b>1</b>	Security group which has SO security role. Only one exists.
<b>2</b>	Security group which has SA security role.

Other security groups will be created or deleted by the Security Officer.

And new security group created by the Security Officer should be under the one of existed security groups. If new security groups are under the SA security group which is No. 2, then those groups will be security groups which have SA security role. If new security groups are under the SO security group which is No. 1, then those security groups will be security groups which have MU security role. As you can see, the security role of new created security group will be determined automatically according to the security role of upper level security group. However, since the SO security group can exist only one, all other security groups created under the SO security group will have MU security role instead of SO security role.

### **2.3 root Separation**

RedCastle classifies all subjects by its security attributes. For this reason, root will be classified into many conditions too. This is only a logical classification done by

RedCastle for its operation and it is not a physical classification in the real system. In RedCastle, root will be classified as the below.

<b>System</b>	<b>RedCastle</b>	<b>Condition</b>
<b>root</b>	<b>SO root</b>	Transferred to root from SO role user through su command.
<b>root</b>	<b>SA root</b>	Transferred to root from SA role user through su command.
<b>root</b>	<b>UXR root</b>	Logged in as root

Only SO root can do all jobs under the RedCastle operation. UXR root can also do many jobs too for it still has strong authority of root even though it is under the RedCastle control. However, it will be limited its authority up to quite level by access control policies applied by RedCastle. The SA root is between the SO root and UXR root in the sense of authority. It might be very hard to define what SA and UXR root can do exactly because it will be determined by policies applied through RedCastle. If no policies applied through RedCastle, then surely UXR root can do most of jobs supposed to do. However, the more policies applied through RedCastle, the less things can be done by UXR root. The important thing is only SO root can do any job under RedCastle operation and all RedCastle operation related works have to be done by SO root.

## CHAPTER 3 MAIN FUNCTIONS

This chapter describes main functions provided by RedCastle. The functions included in the Basic Module will be described first and some of functions included in the Advanced Module will be explained briefly.

### 3.1 MAC (Mandatory Access Control)

MAC classifies and grades the subject and object based on its importance and controls its access according to its security level. In RedCastle, the subject and object will be classified based on its security attributes which is composing of security level, security group, and security role. These security attributes are used to conduct access control. About security attribute of subject is already explained in 2.4. In object case, security attribute can be assigned in the same form of the subject through the command provided.

#### 3.1.1 Access Control Rule

The basic concept of access control rule is to allow subject's access to the object if the subject has higher authority than the object. Surely access will be denied if the subject does not have higher authority than the object. And this authority level is determined by comparing security attributes assigned to the subject and object. MAC will be applied between the subject and object which has same security role. For example, security role SA and MU are very exclusive each other. So, the subject which has security role SA can not access to the object which has security role MU fundamentally. Access control rule is divided into 'read' and 'write' rule.

##### - Read Rule

Read rule allows the access trial only when it satisfies 'subject security attribute  $\geq$  object security attribute' condition. Since the security attribute consists of security level and security group, both of security level and group have to satisfy this condition.

For access trial such as 'file read' and 'file execute' does not require any modifications on an object, these will be classified as 'read access trial' and 'read access rule' will be applied on it. For example, a subject with security level 4 can read a file with security level 5, but can not read a file with security level 2.

**-. Write Rule**

Write rule allows the access trial only when it satisfies ‘subject security attribute = object security attribute’ condition. Since the security attribute consists of security level and security group, both of security level and group have to satisfy this condition.

For access trial such as ‘file edit’ and ‘file delete’ will make modifications on an object, these will be classified as ‘write access trial’ and ‘write access rule’ will be applied on it. For example, a subject with security level 4 can edit a file with security level 4, but can not edit a file with security level 5.

**3.1.2 Applying Target**

**-. File Access Control**

This policy is applying when an object is a file. File access control policy is applied to normal occasions such as read or write files. The security attributes of a subject and an object is compared based on the rule and the access trial will be allowed or denied according to the result of comparison.

**-. Process Kill Control**

In Linux system, only users can terminate their own processes fundamentally but exceptionally, root can terminate all processes also. However, under the control of RedCastle applied with MAC policy, even a root authority can not terminate specific user’s process. The process of secured user will be protected by RedCastle’s MAC function against illegal termination. Only the Security Officer who has security role of SO can terminate all users’ processes.

**-. SU Control**

This function controls security role transfer by using su command between users. For example, a user with security role of MU can not transfer to highly authorized user who has security role of SO. But highly authorized user who has security role of SO can transfer to a user with security role of MU.

**3.1.3 Operation Mode for Applying MAC**

Independent operation mode is provided for applying the ‘MAC’ function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use ‘MAC (Mandatory Access Control)’ function.

	When security violated action occurred, the access will be denied and violation log will be generated.
<b>Warning</b>	Use 'MAC (Mandatory Access Control)' function. When security violated action occurred, the access will be allowed but violation log will be generated.
<b>Off</b>	Do not apply 'MAC (Mandatory Access Control)' function.

### 3.2 ACL (Access Control List)

ACL policy of RedCastle is provided in the hybrid form of RBAC (Role-based Access Control) and DAC (Discretionary Access Control). Whereas MAC applies access control rule based on the security attributes of subject and object, ACL applies access control based on policies. ACL policy is composed of various rules defining rights of subject, object and subject over objects. The Security Officer can define a subject and object applied for ACL policy in various way and also can configure multiple rules. For these reasons, ACL policy can conduct a lot more flexible and wider access controls.

#### 3.2.1 ACL Policy and Definition of Object

An Object is a file or directory to protect. The object can be one or more. The Security Officer develops policies and defines an object by designating a file or directory to protect. ACL policy only shows objects to protect at this state, but by configuring a subject and rule additionally, it becomes complete ACL policy.

#### 3.2.2 Definition of Subject and User Role

In ACL, various applicable subjects are possible. The following table shows those various applicable subjects in ACL.

Classification	Description
User	Normal user existing in the system
Group	All users belong to the group existing in the system.
Security Group	RedCastle security group. All secured users belong to the security group.
User Role	Role-based user subject defined by SO
Any	Any kind of subject

'User Role' is the subject based on its role defined by the Security Officer. The Security Officer can make a 'role' to satisfy expecting purpose and assign a specific applicable

subject to create a new subject. The following list shows those assignable subjects for 'User Role'.

<b>Classification</b>	<b>Description</b>
<b>User</b>	User existing in the system. Up to 8 users can be included.
<b>Group</b>	Group existing in the system. Up to 8 groups can be included.
<b>Security Group</b>	RedCastle security group. Only one can be included. All secured users belong to the security group.
<b>Security Level</b>	RedCastle security level. Only one can be applied. All secured users who have specific security level.

### 3.2.3 Rule Settings

To complete ACL policy, you need to set rules for the subjects and objects defined in the above. Those rules will be configured by assigning rights of which the subjects can execute on the objects. The applicable rights are as follows.

<b>Right</b>	<b>Description</b>
<b>Read</b>	Subject can read an object's contents.
<b>Write</b>	Subject can edit an object's contents.
<b>Execute</b>	Subject can execute an object.
<b>Delete</b>	Subject can delete an object.
<b>Create</b>	Subject can create a new object.
<b>Change Name</b>	Subject can modify an object's name.
<b>Change Owner</b>	Subject can change an object's owner.
<b>Change Attributes</b>	Subject can change an object's 'permission'.

### 3.2.4 Operation Mode

Independent operation mode is provided for applying the 'ACL' function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use 'ACL' function. When security violated action occurred, the access will be denied and violation log will be generated.
<b>Warning</b>	Use 'ACL' function. When security violated action occurred, the access will be

	allowed but violation log will be generated.
<b>Off</b>	Do not use 'ACL' function.

### 3.3 SU Control

RedCastle controls status transfer between users by using 'su' command. For this control, RedCastle provides two functions ; 'root Transfer Control' and 'User Transfer Allow'. And both functions are supported with an independent operation mode.

#### - root Transfer Control

This is function to control status transfer from user to root. Through this function, RedCastle can prevent illegal root authority acquisition by exploiting root password theft. RedCastle only allows root transfer for the user who was registered as a secured user. And among secured users who have security role of SO and SA, it can allow root transfer selectively. However, all other users can't be allowed to have root transfer.

Independent operation mode is provided for applying the 'root Transfer Control' function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use 'root Transfer Control' function. Only secured user who has security role of SO or SA can transfer to root.
<b>Warning</b>	Use 'root Transfer Control' function. Anyone can transfer to root if the person knows the password. However, if the transferred one doesn't have security role of SO or SA, security violation log will be occurred.
<b>Off</b>	Do not use 'root Transfer Control'. Anyone can transfer to root if the person knows the password.

#### - User Transfer Allow

This is function to control status transfer from root to user. Through this function, RedCastle can restrict root authority. Fundamentally, RedCastle does not allow status transfer from root to user. However, when the transfer from root to user is required, it can be allowed only to the users registered in 'SU Allowed User List'. Only the secured user can be registered in the 'SU Allowed User List' and unregistered normal user will

be allowed to transfer from root to user. For this function will not apply to the Security Officer, root with SO (L=1, C=1) can transfer from root to any user status.

Independent operation mode is provided for applying the 'User Transfer Allow' function and it will be applied based on the operation mode as follows.

Operation Mode	Operation
On	Use 'User Transfer Allow' function. It is possible to transfer from root to 'SU Allow User List' registered user.
Off	Do not use 'User Transfer Allow' function. It is impossible to transfer from root to user.

### 3.4 Resetting of Subject Security Attribute

Specific daemon processes registered in 'cron job' such as 'updatedb' should be executed for normal server operation. However, under the control of RedCastle, it could be failed due to the access control rule configured by the Security Officer. To solve this problem, RedCastle provides 'Resetting of Subject Security Attribute'. For example, when you try to execute 'updatedb', by enabling the process to acquire SO authority, complete execution of 'updatedb' will be guaranteed. Also, you can apply the 'Resetting of Subject Security Attribute' function on executable binary files.

Administrator can register files requiring security attribute modification on the 'Resetting List of Subject Security Attribute' and the registered files will have new modified security attribute when it is executed.

### 3.5 Advanced Functions

These are brief description of some security functions included in the Advanced Module. These functions are not available in the Basic Module.

#### 3.5.1 Commands Control

'Commands Control' is a function to restrict execution of commands. The Security Officer can register commands on the 'Commands Control List' to restrict its execution. And those registered commands can be executed only by the Security Officer who has security role of SO. So even the user has root authority, it's not allowed to execute those registered commands without appropriate security role.

Independent operation mode is provided for applying the 'Command Control' function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use 'Commands Control' function. Only the Security Officer can execute this function. If other users try to use this function, security violation log will be generated.
<b>Warning</b>	Use 'Commands Control' function. Other users can execute this function but security violation log will be generated.
<b>Off</b>	Do not use 'Commands Control' function.

Note : All registered commands have its own 'Applicable Mode' and if its applicable mode is 'N', it means can not be executed.

### 3.5.2 SETUID Control

'SETUID control' is a function to restrict 'setuid' program execution.

The security officer can register 'setuid' programs on 'SETUID Control List' to allow its execution. Only those registered 'setuid' programs are allowed to execute and all other 'setuid' programs' execution will be denied. By recognizing registered 'setuid' programs as trusted 'setuid' programs only and allow its execution, it is possible to prevent illegal root authority obtainment through exploiting 'setuid' programs.

In some 'setuid' programs, root process required to be created while those programs are in execution. To enable this, RedCastle use the 'root Process Create Option'. The Security Officer can specify the 'root Process Create Option' when he registers 'setuid' programs in the 'SETUID Control List'. Usually the option value will be set as 'N'.

Independent operation mode is provided for applying the 'SETUID Control' function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use 'SETUID Control' function. Only registered 'setuid' programs on the list can be executed. If a unregistered 'setuid' program executed, security violation log will be generated.
<b>Warning</b>	Use 'SETUID Control' function. Unregistered 'setuid' program can be executed but security violation log will be generated.
<b>Off</b>	Do not use 'SETUID Control' function.

### 3.5.3 KILL Deny

'KILL Deny' function is to prevent process kill through 'kill' command. The Security Officer can register processes in 'KILL Deny List' to protect them from 'kill' command. Those registered processes can be killed only by the Security Officer who has security role of SO. SO, even a root can not kill those registered processes. There are only seven signals to control this function and those are HUP, KILL, TERM, STOP, CONT, USR1, and USR2. (All these seven signals can kill registered processes.)

When the Security Officer registers processes to protect in the 'KILL Deny List', he can designate 'SA Allow'. And If he specified 'SA Allow' into 'Y', it means a user with security role of SA can kill the processes registered in the 'KILL deny List.

Independent operation mode is provided for applying the 'Kill Deny' function and it will be applied based on the operation mode as follows.

<b>Operation Mode</b>	<b>Operation</b>
<b>On</b>	Use 'KILL Deny' function. Only the Security Officer can kill processes on the list. If any other users try to kill processes on the list, security violation log will be occurred.
<b>Warning</b>	Use 'KILL Deny' function. Any user can try to kill processes on the list but security violation log will be occurred.
<b>Off</b>	Do not use 'KILL Deny' function.

## CHAPTER 4 OPERATION

This chapter describes how to operate RedCastle properly. And it will include initial data creation for operation, basic security policy provided by RedCastle, start and stop of RedCastle security functions and operation environment settings.

### 4.1 RedCastle Configuration

RedCastle consists of 5 RPMs and each RPM provides the following functions.

rpm	Description
<b>kmod-redcastle</b>	RedCastle Kernel Module Provide RedCastle main functions
<b>redcastle-app</b>	RedCastle Application Provides Daemon, Commands, and others required for RedCastle operation
<b>redcastle-pam</b>	RedCastle Pluggable Authentication Module
<b>redcastle-data</b>	Data required for RedCastle operation Provides basic policy, environment setting files, and etc.
<b>redcastle-manager</b>	RedCastle Basic Manager GUI manager enables start, stop and operation environment set of RedCastle

### 4.2 Initial Operation Environment Settings

When you install Asianux Server 3, You can find RedCastle setting view in Anaconda. In this view, you can decide whether you want to use the security function provided by RedCastle and select operation mode of RedCastle. The following table describes the operation mode to select.

Operation Mode	Description
<b>Enable</b>	Create the redcastle account for administration of RedCastle. After the installation, RedCastle will be initiated and operated by basic policy provided by RedCastle.
<b>Warning</b>	Work same as Enable mode. However, it will not block any security violations but generate security violation logs.

<b>Disable</b>	Not to use RedCastle. System admin can initiate RedCastle by manual.
----------------	---

If you select RedCastle operation mode as 'Disable', RedCastle will not run on Asianux server system. And if you need to use RedCastle later, you have to create basic data for operation of RedCastle manually.

By using 'mkinitdata', you can create initial data required for RedCastle operation.

```
/usr/share/redcastle/sbin-{kernel version}/mkinitdata -u redcastle -m enable
```

The following table describes about 2 arguments required by 'mkinitdata'.

Argument	Description
<b>-u redcastle</b>	RedCastle admin account. Have to use the account existing in the system.
<b>-m enable warn</b>	RedCastle operation mode. Have to use one of 'enable' or 'warn'.

BY using chkconfig command, add 'redcastle' and 'redmanager' script to the Service List and designate 'runlevel'.

```
/sbin/chkconfig --add redcastle
```

```
/sbin/chkconfig --level 2345 redcastle
```

```
/sbin/chkconfig --add redmanager
```

```
/sbin/chkconfig --level 2345 redmanager
```

### 4.3 RedCastle Basic Security Policy

RedCastle provides basic security policy for secured server operation. The basic security policies provided by RedCastle are as follows.

#### 4.3.1 ACL Basic Policy

RedCastle provides ACL (Access Control List) basic policy as 'Default Policy'. The purpose of basic policy is to prevent system commands and important environment setting files from illegal falsification. The 'Default Policy' consists of four policies and the following table shows its detailed information.

Policy	Target File
	Rule

<b>OS Command Protection</b>	/usr/bin/*, /usr/sbin/*, /sbin/*, /bin/*
	Only allow read and execute to all users. (All users can execute command but can not conduct the tasks such as file creation and delete in the directory.)
<b>Configuration Protection</b>	/etc/*.conf, /etc/hosts, /etc/services, /etc/xinetd.d/*, /etc/pam.d/*
	Only allow read to all users
<b>Tmp Directory Protection</b>	/tmp/*, /var/tmp/*
	All users can conduct any tasks except execute. (It prevents tmp directory from exploitation for hacking.)
<b>Boot File Protection</b>	/boot/*, /boot/grub/*
	Only allow read to all users.

#### 4.3.2 Security Attributes Reset List

Specific daemon processes, registered in cron job such as 'updatedb', must be executed for normal server operation. However, while RedCastle is running, partial process could be failed by access control rules set by the security officer. In this case, you can execute the command with other security attributes using 'Security Attributes Reset' function. The Security Attributes Reset List provided by RedCastle is as follows.

<b>Target</b>	<b>New Security Attributes</b>
<b>/usr/bin/updatedb</b>	By command execution, the security attribute is reset as SO(L=1, C=1).
<b>/usr/sbin/prelink</b>	By command execution, the security attribute is reset as SO(L=1, C=1).

#### 4.4 RedCastle Security Functions – Start/Stop

For RedCastle operation, two scripts ('redcastle' and 'redmanager') are used.

The 'redcastle' script is used to start or stop RedCastle's security function.

/etc/init.d/redcastle start or /etc/init.d/redcastle stop

Only root user is possible to start and only SO root can stop the security function of RedCastle.

RedCastle provides GUI based manager program (running on Windows) as well as CLI. If you want to use the manager program, you have to activate the agent daemon first. The 'redmanager' script is used to start or stop the agent daemon.

`/etc/init.d/redmanager start` or `/etc/init.d/redmanager stop`

If RedCastle's security function is in use, only SO root can use this and if RedCastle's security function is stopped, only root user can use this.

#### 4.5 How to Apply Main Functions

This section explains how to use the essential commands required to know for RedCastle operation. Other commands will be described in the chapter of 'Operating Commands of RedCastle'.

##### 4.5.1 RedCastle Security Functions – Start/Stop : rgctl Command

For RedCastle provided in kernel module form, you have to load RedCastle kernel module onto the kernel to use RedCastle. And if you don't want to use RedCastle, you have to unload RedCastle kernel module from the kernel. The above mentioned script includes all these processes.

The following 'rgctl' command is used to start or stop the security functions of RedCastle in the condition of RedCastle kernel module was loaded onto the kernel.

The following is showing how to use the 'rgctl' command.

`rgctl start|restart [enable|warning] , rgctl stop|info`

Option	Description
<b>start [enable warning]</b>	Start RedCastle security functions. Enabling setting of RedCastle operation mode. If not set, it will apply former operation mode.
<b>restart [enable warning]</b>	Restart RedCastle security functions. It is same as start again after stop.
<b>stop</b>	Stop RedCastle security function.
<b>info</b>	Query RedCastle operation status.

RedCastle operation mode (enable or warning) will be described in the following 'rgconf' command.

#### 4.5.2 RedCastle Operation Environment – rgconf Command

This is a command to manage RedCastle operation environment.

RedCastle has separate operation mode for each security function. You can set each function's operation mode by using 'rgconf' command. The following is showing how to use 'rgconf' command.

rgconf -g|-l|-e

Option	Description
<b>-g</b>	Query present operation environment
<b>-l</b>	Apply new operation environment
<b>-e</b>	Save present operation environment into a file

RedCastle operation environment file is /usr/share/redcastle/data/ips.conf, and this can be created by using 'rgconf -e'. If you want to change operation environment, modify ips.conf file first then apply new operation environment by using 'rgconf -l'.

The following table shows the applicable value for each function's operation mode provided by RedCastle.

Function	ips.conf File	Set Value
<b>MAC</b>	MACMODE	Default : on (on/warn/off)
<b>ACL</b>	ACLMODE	Default : on (on/warn/off)
<b>Command Control</b>	COMMAND	Default : on (on/warn/off)
<b>SETUID Control</b>	SETUID	Default : warn (on/warn/off)
<b>KILL Deny</b>	KILLMODE	Default : on (on/warn/off)
<b>root SU Control</b>	ROOTSU	Default : on (on/off)
<b>User SU Control</b>	USERSU	Default : warn (on/warn/off)

The operation mode used in 'rgctl' command is RedCastle operation mode and relationship with each function's operation mode is as follows.

RedCastle Operation Mode	Functional Operation Mode
enable	Use Functional Operation Mode Possible to change each functional operation mode
warning	All functions operations are in warning mode Impossible to change each functional operation mode

### 4.5.3 ACL Policy Management – fgroupset Command

fgroupset is a command to manage ACL policy. The following shows how to use 'fgroupset' command.

fgroupset -a|-m TEMPLET , fgroupset -d all|POLICY\_ID , fgroupset -i|-p

Option	Description
<b>-a TEMPLET</b>	ACL policy add by using TEMPLET file
<b>-m TEMPLET</b>	ACL policy modify by using TEMPLET file
<b>-d all POLICY_ID</b>	ACL policy delete (if you use all, you can delete all ACL policies)
<b>-i</b>	ACL policy add (Interactive mode)
<b>-p</b>	Policy group manage

If you want to make an ALC policy, you need a policy group first.

The relationship between policy group and ACL policy is same as the one between file and directory. We are using a policy group to manage an ACL policy more efficiently same as we are using a directory to manage a file more efficiently. In the case of ACL basic policy mentioned before, 'Default Policy' will be the policy group and this 'Default Policy' has four ACL policies. To manage the policy group, we are using 'fgroup -p'. By using this, you can add or delete policy group.

The following is the format in ACL policy template file. If you want to add or modify an ACL policy, you have to use correct format of template file. The following example is one of ACL policies provided by RedCastle.

[FGROUP]	← Template start tag
1	← Policy number
OS Command Protection	← Policy name
1	← Number of rules
4	← Number of members
1	← Applicable mode
1	← Policy group number
0:::::rx	← Rule
/bin/*	← Member (1/4)
1	
/sbin/*	← Member (2/4)
1	

/usr/bin/*	← Member (3/4)
1	
/usr/sbin/*	← Member (4/4)
1	
[/FGROUP]	← End template tag

The following table explains each item used in template file.

Item	Value	Description
<b>Template Start Tag</b>	[FGROUP]	Start template file Tag must be used at the start and end of template.
<b>Policy Number</b>	1	ACL policy number 0 must be used to add ACL policy. Use the ACL policy number to modify the policy.
<b>Policy Name</b>	OS Command Protection	ACL policy name
<b>Number of Rules</b>	1	Number of rules used in ACL policy.
<b>Number of Members</b>	4	Number of members to be protected by ACL policy
<b>Applicable Mode</b>	1	Applicable mode of ACL policy 0 – Off ( do not apply ACL policy) 1 – Enable (apply ACL policy) 2 – Warning (apply ACL policy but do not block)
<b>Policy Group Number</b>	1	Number of policy group which has ACL policy.
<b>Rule</b>	0:::::rx	ACL rule. Write as many as 'number of rules'. (Please refer to the below for more detailed information about rule.)
<b>Member</b>	/bin/*	Member to be protected by ACL policy.
	1	Have to exist as many as 'number of rules'. Consists of 'Member name' and 'Automatic trace' options. (Please refer to the below information for more detailed 'Automatic trace' option.)

<b>Template End Tag</b>	[/FGROUP]	The end of template file. Tag must be used at the start and end of template.
-------------------------	-----------	---

An ACL rule consists of six items and each item is classified by colon. The following table shows each item used for composing ACL rules. (Unused item will be blanked)

<b>Item</b>	<b>Description</b>
<b>Subject Classification</b>	Subject class. It will have one of the following 5 numbers. 0 – All subject classes 1 – User 2 – Security group 4 – User role 16 – Group
<b>ID</b>	Subject ID. It will be assigned depends on the subject class at the above. All subjects(0) – not to use User(1) - User ID Security group(2) – Security category ID User role(4) – User role ID Group(16) – Group ID
<b>Name</b>	Subject name. It will be assigned depends on the subject class at the above. All subjects(0) – not to use User(1) – User name Security group(2) – Security category name User role(4) – User role name Group(16) – Group name
<b>Program</b>	Assign a program which can access to an object. If not specified, all programs would have the access right.
<b>IP Address</b>	Assign IP Address which can access to an object. If a specific IP address assigned, the access is possible through the IP address only. If not specified, the access is possible through all IP addresses.
<b>Right</b>	The task subject can do on the object. It would be 8 kinds of tasks (rwxdcnm). If no right has assigned, a subject can not have any access right to a object.

The 'Auto Trace' used in 'Member' is an option to apply ACL policy by automatic update when the i-node number of 'Member' changed. For example, if a member recreated with the same name after it was deleted once, the i-node number of the member can be changed. On the Linux system, if the i-node number is different even though it has same name, it is identified as different file. If you use 'Auto Trace' option, you can keep ACL policy applying by automatic ACL policy updates even in the case of i-node number changing. 'Auto Trace' would have the value of 0 or 1. The value 1 means to use 'Auto Trace' option and 0 means not using the option.

As you can see in the above example, you can use wild card with 'Member name'. Only '\*' will be supported for using wildcard and this will be limited for several cases. The following shows the three supporting wild card formats.

Format	Description
/usr/bin/*	All files in the '/usr/bin' directory
/usr/bin/a*	All files start with 'a' in the '/usr/bin' directory
/usr/bin/*b	All files end with 'b' in the '/usr/bin' directory

Other formats except the above three mentioned, will not be supported. The following table shows the examples for not supporting.

Format	Description
/usr/*/bin	Wild card used in the middle of whole path
/usr/bin/a*b	Wild card used in the middle of file name

RedCastle provides interactive mode for more comfortable ACL policy addition. The interactive mode uses 'fgroupset -i'. IP address input will be skipped in the interactive mode.

## CHAPTER 5 BASIC MANAGER

This chapter describes about the Basic Manager. The Basic Manager is a management tool for RedCastle Basic Module.

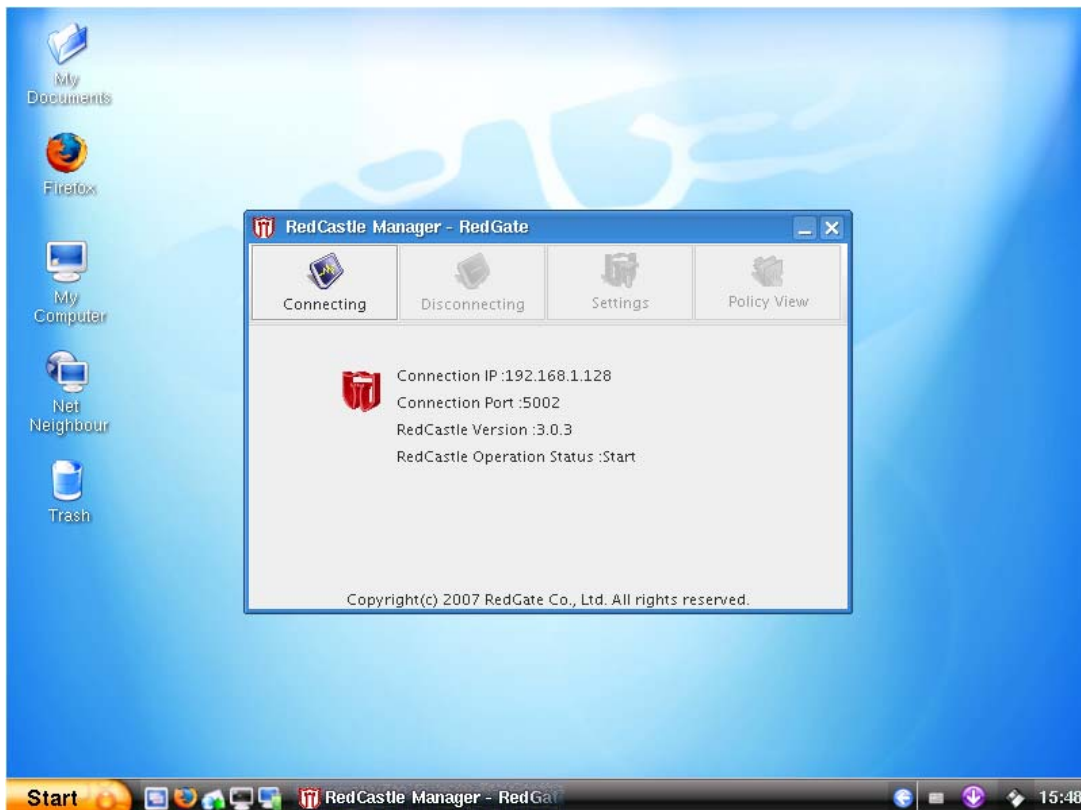
### 5.1 Introduction

Basic manger is a simple GUI tool to help RedCastle management.

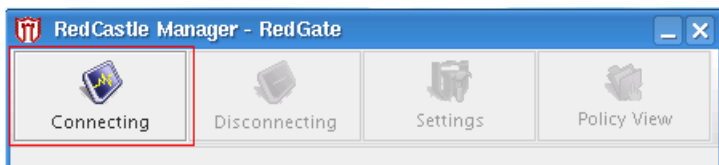
The security officer can use the manager through proper authentication process and by using this manager, he can start/stop the security functions and set operation environment of RedCastle. To use this manager, RedCastle agent daemon should be in operation condition.

### 5.2 Connecting and Disconnecting

By clicking of 'Start Menu → Control Panel → System Configuration → RedCastle Manager', you can run RedCastle basic manager and the following manager main view will be displayed.



Please click the 'Connecting' button in the Manager main view to connect to the Manager.



By clicking the 'Connecting' button, the following login window will be shown.

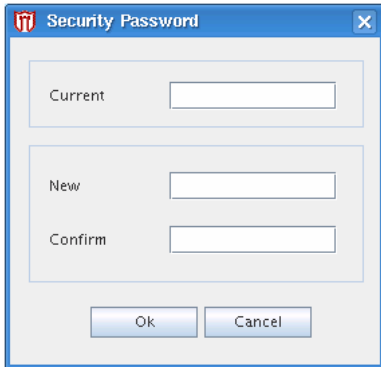


Please enter User ID, Password, and Security Password in the login space. Only RedCastle security officer can login as a user. Other users can not login to the Manager. The Password is a system password of the security officer account and the Security Password is an exclusive password to login RedCastle. The initiating Security Password will be same with the system password.

If your initiating authentication is successful, the following message for security password will be displayed.

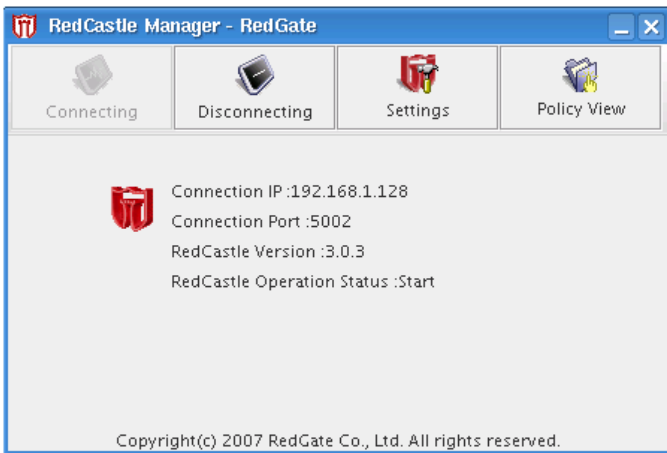


The user must change the Security Password. If you click 'No' button, you can't change the Security Password and can't connect to the Manager. If you click 'Yes' button, the following view will be displayed to change the Security Password.

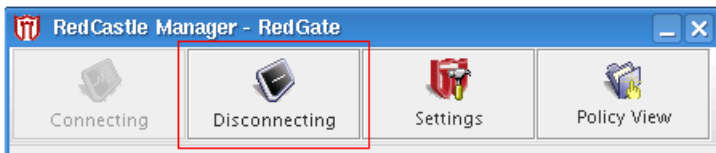


Please enter the current security password and a new security password here. The Security Password can be characters between 6 to 15 and this will include any alphabets, numbers and special marks such as !@#\$. The Security Password should include at least one of each alphabets, numbers and special marks. For example, the Security Password consists only alphabets and numbers will not be allowed.

If your security password changed successfully, you can connect to the manager. The below is the manager module's starting view.

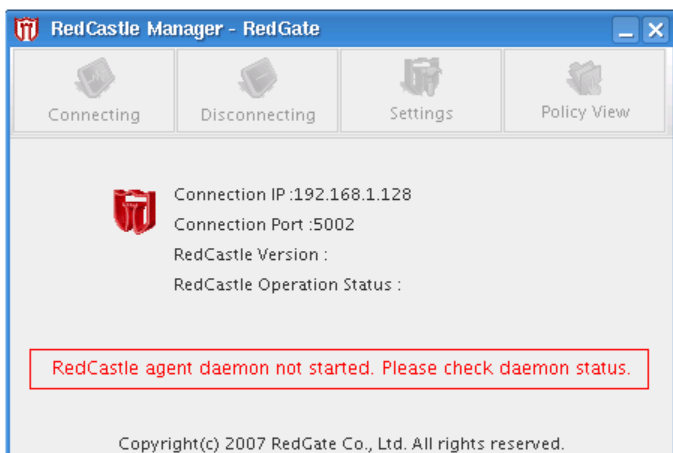


If you want to disconnect with the manager, click the 'Disconnecting' button.

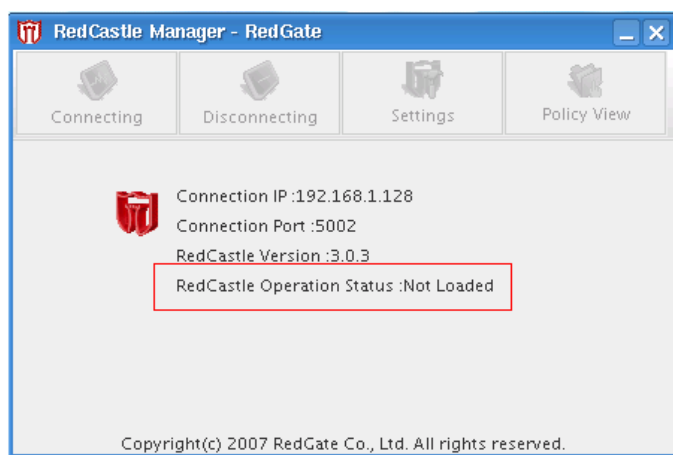


Under the two conditions you can not connect to the manager.

The first condition is that the agent daemon of RedCastle is not in operation. In this case, the below message will be displayed.



If you want to execute the manager in this condition, start RedCastle agent by using '/etc/init.d/redmanager' script. The second condition is that the security function of RedCastle is not in operation. In this case, RedCastle operation status will be displayed as 'Not Loaded' as follows.



If you want to execute the manager in this condition, start the security functions of RedCastle by using '/etc/init.d/redcastle' script.

### 5.3 Settings

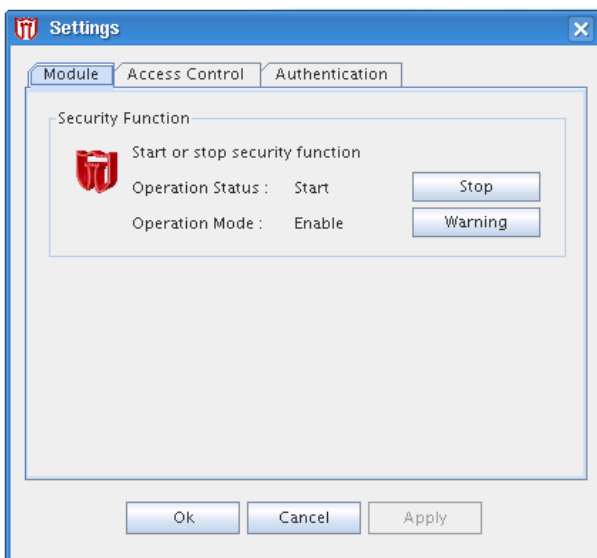
Through the manger, you can start/stop the security functions of RedCastle and configure various functions' operation mode too. If you click the 'Settings' button,

'Settings' view will be displayed. The 'Settings' view has three tabs : 'Module', 'Access Control', and 'Authentication'.

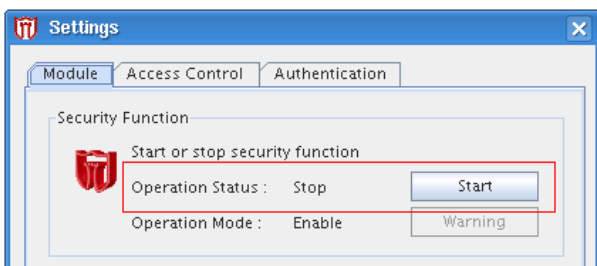


### 5.3.1 Module

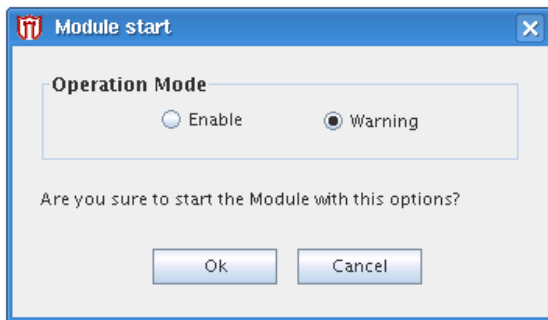
In the 'Module' tab, you can check and modify the operation status and mode of RedCastle security functions. The following figure shows when RedCastle security functions are in start condition.



If you want to stop RedCastle security functions, click the 'Stop' button in 'Operation Status'. If RedCastle security functions are stopped, 'Operation Status' will be changed into Stop condition and the button will be changed as 'Start'. If you want to restart RedCastle security functions, click the 'Start' button.

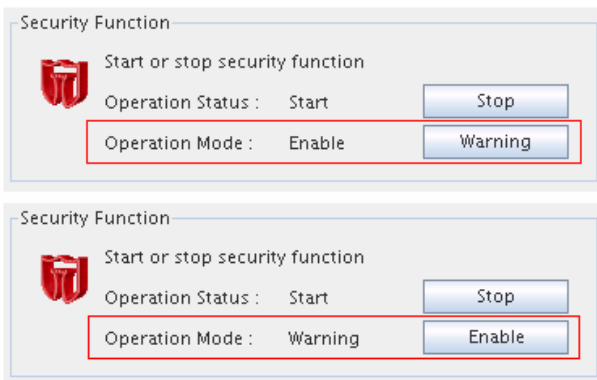


When you start RedCastle security functions, you have to select the operation mode.



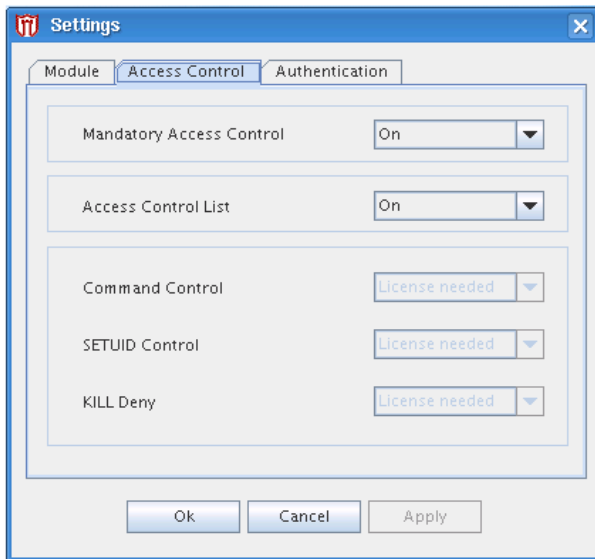
‘Enable’ means to allow the system protection control by RedCastle security functions. If a security violation activity occurs, it will generate security violation log and block the violation activity. ‘Warning’ mode does similar things with ‘Enable’ mode except do not block violation activities. The security officer can check security violations through the security log recorded.

‘RedCastle Operation Mode’ can be changed even in the status of RedCastle security functions in operation. You can change ‘RedCastle Operation Mode’ by clicking the ‘Operation Mode’ button as shown in the below figure. (Present status of the Operation Mode and button sign will be toggled)



### 5.3.2 Access Control

In the ‘Access Control’ tab, you can check and modify the operation mode of every security function provided by RedCastle. In ‘Basic Mode’, you can configure ‘Mandatory Access Control’ and ‘Access Control List’ functions.



The operation mode will be set as one of 'On', 'Warning', and 'Off'. 'On' means to use the function and if a security violation activity occurs, it will generate security violation log and block the violation. 'Warning' does same things as 'On' except do not block violation activities. 'Off' means do not use the function.

Every function's operation mode will be influenced by the 'RedCastle Operation Mode' configured in the 'Module' tab. Only the 'RedCastle Operation Mode' is in 'Enable' status, you can set each function's operation mode. If the 'RedCastle Operation Mode' is in 'Warning' status, all other function's operation mode will be set as 'Warning' automatically and this can not be modified unless the 'RedCastle Operation Mode' becomes 'Enable' again. If the 'RedCastle Operation Mode' is in 'Enable' mode again, all other functions' operation mode will restore the former value.

### 5.3.3 Authentication

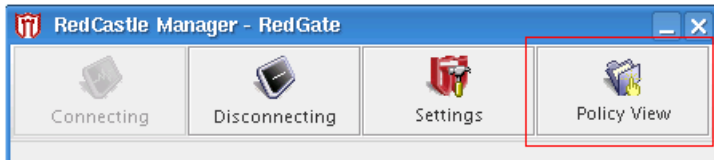
In the 'Authentication' tab, you can configure and modify SU control function provided by RedCastle. Also, you can change a secured user's security password here.



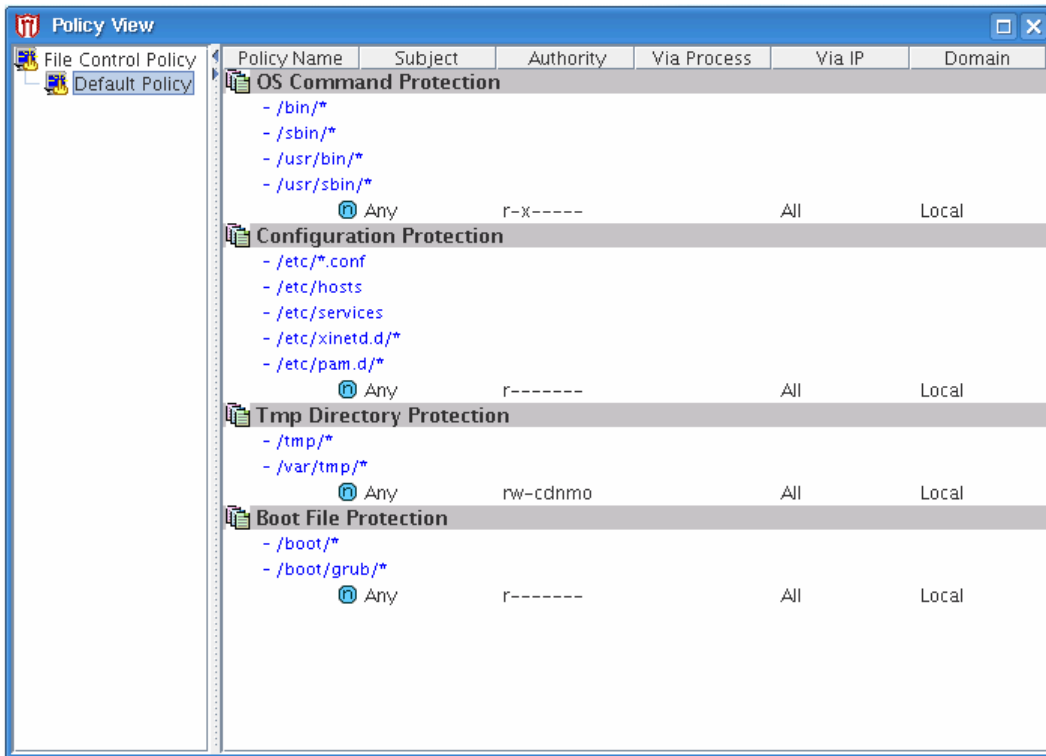
Every function's operation mode in the 'Authentication' tab will be influenced by 'RedCastle Operation Mode' configured in the 'Module' tab.

#### 5.4 ACL Policy Query

You can query ACL policy configured by the Manager. To query the ACL policy, please click the 'Policy View' button in the Manger's main view.



If you click the 'Policy View' button, the following 'Policy View' window will be displayed.



The 'Policy View' is divided into two large views. The left view shows 'ACL Policy Group' and the right view shows 'ACL Policies' belongs to the 'ACL Policy Group'. The above figure shows the basic policy provided by RedCastle. As you can see, there are four ACL policies under an 'ACL Policy Group' named as the 'Default Policy'. For each ACL policy, you can check its policy name, member and rule.

Through the Manger, only you can query ACL policy and can not set ACL policy. If you want to set ACL policy, you have to use the command provided by RedCastle.

## CHAPTER 6 OPERATING COMMANDS OF REDCASTLE

This chapter describes about the commands provided by RedCastle.

The following table shows classified commands which will be explained in the chapter.

Classification	Commands
Operation	rgctl , rgconf
Secured User Management	addmu , delmu , getmu , modmu , addcat , delcat , getcat , modcat
ACL Policy Management	fgroupget , fgroupset
User Role Management	uroleassign , uroleadd , uroledel , uroleget
Security Log	rclogd , seclog
Security Attributes	getmf , setmf , psx
Security Password	chpw , conv_rcpwd
Others	alcmd , sosd , suctl

All commands provide help message and version information through '-h' and '-v' options. Since these two options are quite basic and applicable to the all commands, separate description will be skipped.

**NAME**

**addcat - Add new security group on the Security Group List**

**SYNOPSIS**

**addcat -p PARENT\_CID -n NAME**

**DESCRIPTION**

addcat is a command to add a new security group on the Security Group List. The Security Group List has tree format. So, if you want to add a new security group, it has to be under the existing security group. RedCastle provides only two security groups and they are 'ROOT' and 'System Admins'. There is only one 'Root' security group which has security role of SO. All added security groups under the 'ROOT' security group will have security role of MU. The 'System Admins' security group has security role of SA and all added security groups under this one will have security role of SA. The argument of addcat command is the name of newly added security group and the parent security group ID. The newly added security groups will be assigned its ID by the added order.

**OPTIONS****-n NAME**

Security group name

**-p PARENT\_CID**

Parent security group ID that new security group will be belonged to.

**SEE ALSO**

**delcat, getcat, modcat**

## **NAME**

**addmu - Add new security user on the Security User List**

## **SYNOPSIS**

**addmu [-r SO|SA|MU [-c CID]] [-l LEVEL] [-s Y|N] USER**

## **DESCRIPTION**

addmu is a command to add new secured users to the secured user list. The arguments of the command are user name and security attributes the user will have. Only a user who can login with shell and home directory can be a secured user and root account can not be added as a secured user. The security attributes consist of security role, security group and security level. And this will be assigned by using '-r', '-c', '-l' options consecutively.

The security role will be assigned as one of SO, SA and MU by using '-r' option. If you do not use this option, MU role is applied automatically.

The security group will be assigned as CID by using '-c' option. CID is security group ID in the existing Security Group List and it can be confirmed with getcat command.

If you do not use this option, default CID is applied automatically according to the designated security role by -r option.

By using '-l' option, you can designate the security level. It has one of from 1 to 7, and 1 is the highest level, and 7 is the lowest. If you do not use this option, level 7, the lowest, is applied automatically.

By using '-s' option, you can set root transfer control option. Root transfer control is a function provided by RedCastle to restrict user transfer to root through su command exploitation. This option can be set by a user who has the security role of SO or SA only and for users with MU security role, this option is ignored. In this option, Y means allowance of root transfer, and N means denial of root transfer. If you do not set this option, N is applied automatically by default. If a user has SO security role, this option is ignored and Y is applied automatically by default.

## **OPTIONS**

### **-r SO|SA|MU**

Security role which is one of user security attributes.

### **-c CID**

Security group which is one of user security attributes.

### **-l LEVEL**

Security level which is one of user security attributes.

**-s Y|N**

Root Transfer Control Option.

**USER**

User account that will be added.

**SEE ALSO**

**delmu, getmu, modmu, getcat**

## **NAME**

**alcmd - Security Attribute Reset List management tool**

## **SYNOPSIS**

**alcmd -a|-d|-g|-m**

## **DESCRIPTION**

alcmd is a command to manage the Security Attributes Reset List. If you want to add a new item, you have to assign command name, execution role, and new security attributes. If a root, assigned in the list with execute role, executes a command, security attributes of the command will be reset as the one assigned in the list.

A command has to be registered with its full path and it should be executed through its full path. The execution role is security role to execute the listed command and it will have a security role of SA, UXR or ALL. SA means reset security attributes when a root with SA security role executes. ALL means a root with SA or UXR security role.

To assign new security attributes, designate security role, security level and security group ID. The security role will have one of SO, SA, and UXR. The security level will have a value between 1 to 7 and the security group ID will be designated as same as the security group's ID existing in the Security Group List. The security group can be checked by using 'getcat' command. The security level and security group can be assigned based on the security role.

## **OPTIONS**

**-a**

Add new command on the Security Attribute Reset List.

**-d**

Delete a command from the Security Attribute Reset List.

**-g**

Query the Security Attribute Reset List

**-m**

Modify attributes of registered command.

## **SEE ALSO**

**getcat**

**NAME**

**chpw – Change secured user’s security password**

**SYNOPSIS**

**chpw**

**DESCRIPTION**

chpw is a command to change security password. Security password is a password to use in RedCastle manager program authentication. If a user registered as a secured user, a security password will be set as a system password automatically. A secured user can change his security password by using this command. The security password will be used in RedCastle manager program authentication.

**SEE ALSO**

**conv\_rcpwd**

**NAME**

**conv\_rcpwd – Initialize security passwords of all users**

**SYNOPSIS**

**conv\_rcpwd**

**DESCRIPTION**

conv\_rcpwd is a command to initialize the security password. By using this command, All secured users' security password can be initialized as user's system passwords.

**SEE ALSO**

**chpw**

**NAME**

**delcat - Delete a security group from the Security Group List**

**SYNOPSIS**

**delcat CID**

**DESCRIPTION**

delcat is a command to delete a security group from the Security Group List. However, if there is a secured user in the security group to delete, it can't be deleted. And you can't delete the default two security group ( 'Root' and 'System Admins').

**OPTIONS****CID**

Security group ID that will be deleted.

**SEE ALSO**

**addcat, getcat, modcat**

**NAME**

**delmu - Delete a security user from the Security User List**

**SYNOPSIS**

**delmu -a|USER**

**DESCRIPTION**

delmu is a command to delete a secured user from the secured user list. You can delete all secured users except secured users with SO security role by using '-a' option.

**OPTIONS**

**-a**

Delete all security users except the security users with SO security role.

**USER**

Security user name that will be deleted

**SEE ALSO**

**addmu, getmu, modmu**

**NAME**

**fgroupget – Query ACL policies and rules**

**SYNOPSIS**

**fgroupget [-g POLICY\_ID]**

**DESCRIPTION**

Fgroupget is a command to query ACL policies. By using fgroupget command, you can query basic information of all ACL policies. Also, you can do detailed query on a specific ACL policy. Through this detailed query, you can query members and rules.

**OPTIONS**

**-g POLICY\_ID**

Policy ID that will be queried rules

**SEE ALSO**

**fgroupset**

## **NAME**

**fgroupset - ACL policy management tool**

## **SYNOPSIS**

**fgroupset -a|-m TEMPLATE**

**fgroupset -d all|POLICY\_ID**

**fgroupset -i|-p**

## **DESCRIPTION**

fgroupset is a command to manage ACL policies. Adding ACL policies can be done by using interactive mode ('-i' option). Also, you can add or modify ACL policies by using template file. Please refer to the Manual for detailed description.

## **OPTIONS**

**-a TEMPLATE**

Add a policy on the ACL Policy List.

**-d all|POLICY\_ID**

Delete a policy from the ACL Policy List.

**-m TEMPLATE**

Modify the policy.

**-i**

Add a policy on the ACL Policy List. (Interactive mode)

**-p**

Manage the Policy Group List.

## **SEE ALSO**

**fgroupget**

**NAME**

**getcat – Query the Security Group List**

**SYNOPSIS**

**getcat [CID]**

**DESCRIPTION**

getcat is a command to query the security group list. The items you can query by using this command are security group ID, security group name, and its hierarchic structure.

**OPTIONS****CID**

Security group ID that will be queried.

**SEE ALSO**

**addcat, delcat, modcat**

**NAME**

**getmf - Query file security attributes**

**SYNOPSIS**

**getmf [-R] [FILE]**

**DESCRIPTION**

Getmf is a command to query file security attributes. Output of the command is same as the one of 'ls' command and the security attributes will be displayed additionally. If a file has the assigned security attributes, it will be shown with '+' sign at the end of the file permission. The security attributes consists of security role, security level and security group ID. The security role will be one of 'SO', 'SA', and 'MU' and it is assigned based on the file's security group automatically. A file without security attributes will be displayed with security role of UX, security level of '0', and security group ID of '0'.

**OPTIONS****FILE**

Filename that will be queried its security attribute

**-R**

List subdirectories recursively

**SEE ALSO**

**setmf**

**NAME**

**getmu – Query the Secured User List**

**SYNOPSIS**

**getmu [USER]**

**DESCRIPTION**

getmu is a command to query the Secured User List. By using this command, you can check the security attributes of a secured user.

**OPTIONS****USER**

User name that will be queried

**SEE ALSO**

**addmu, delmu, modmu**

## **NAME**

**lastlog – Query the most recent logins of users**

## **SYNOPSIS**

**lastlog [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][--u USER][--r HOST]**

## **DESCRIPTION**

Lastlog is a command to query the most recent login information of users. The recent login information show the user name, login terminal, recent login time and login host.

If you use '-s' and '-p' option, you can see a specific part of queried result. PG\_SIZE specified by the '-s' option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the '-p' option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use '-f' and '-t' option, you can also specify the period that will be queried. The basic form of 'FROM\_DATE' and 'TO\_DATE' is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use 'FROM\_DATE' only, it will query the log data during period specified 'FROM\_DATE' to the present. And if you use 'TO\_DATE' only, it will query the log data during period from the oldest time to the specified 'TO\_DATE'.

## **OPTIONS**

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

### **-f FROM\_DATE -t TO\_DATE**

The period that will be queried.

### **-u USER**

User name that will be queried.

### **-r HOST**

Host name that will be queried.

## **NAME**

**loginc – Query the current logged in users**

## **SYNOPSIS**

**loginc [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][-u USER][-r HOST]**

## **DESCRIPTION**

loginc is a command to query the current logging users. Logging users data shows user name, login terminal, login time and login host name.

If you use '-s' and '-p' option, you can see a specific part of queried result. PG\_SIZE specified by the '-s' option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the '-p' option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use '-f' and '-t' option, you can also specify the period that will be queried. The basic form of 'FROM\_DATE' and 'TO\_DATE' is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use 'FROM\_DATE' only, it will query the log data during period specified 'FROM\_DATE' to the present. And if you use 'TO\_DATE' only, it will query the log data during period from the oldest time to the specified 'TO\_DATE'.

## **OPTIONS**

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

### **-f FROM\_DATE -t TO\_DATE**

The period that will be queried.

### **-u USER**

User that will be queried.

### **-r HOST**

Host name that will be queried.

## **NAME**

**loginf – Query a specific user’s login fail log**

## **SYNOPSIS**

**loginf [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][-u USER]**

## **DESCRIPTION**

loginf is a command to query a specific user’s login fail log data. Login fail log data shows user name, login terminal, login fail time and login host name.

If you use ‘-s’ and ‘-p’ option, you can see a specific part of queried result. PG\_SIZE specified by the ‘-s’ option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the ‘-p’ option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use ‘-f’ and ‘-t’ option, you can also specify the period that will be queried. The basic form of ‘FROM\_DATE’ and ‘TO\_DATE’ is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use ‘FROM\_DATE’ only, it will query the log data during period specified ‘FROM\_DATE’ to the present. And if you use ‘TO\_DATE’ only, it will query the log data during period from the oldest time to the specified ‘TO\_DATE’.

## **OPTIONS**

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

### **-f FROM\_DATE -t TO\_DATE**

The period that will be queried.

### **-u USER**

User that will be queried.

## NAME

**loginh – Query the user’s login history log**

## SYNOPSIS

**loginh [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][-u USER][-r HOST][-c login|boot]**

## DESCRIPTION

loginh is a command to query a specific user’s login history data. Login history data consists of user name, login terminal, login time, logout time, and login host name. And this will be shown from the most recent data.

If you use ‘-s’ and ‘-p’ option, you can see a specific part of queried result. PG\_SIZE specified by the ‘-s’ option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the ‘-p’ option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use ‘-f’ and ‘-t’ option, you can also specify the period that will be queried. The basic form of ‘FROM\_DATE’ and ‘TO\_DATE’ is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use ‘FROM\_DATE’ only, it will query the log data during period specified ‘FROM\_DATE’ to the present. And if you use ‘TO\_DATE’ only, it will query the log data during period from the oldest time to the specified ‘TO\_DATE’.

## OPTIONS

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

### **-f FROM\_DATE –t TO\_DATE**

The period that will be queried.

### **-u USER**

User that will be queried.

### **-r HOST**

Host name that will be queried.

**-c login|boot**

Log types. Login means logs about the history of user's login and boot means logs about system booting.

**NAME**

**mkinit – Create RedCastle runtime initial data**

**SYNOPSIS**

**mkinit**

**DESCRIPTION**

mkinit is a command to verify RedCastle data before you are initiating the security function of Redcastle. The command should be used before you load RedCastle's security functions onto the kernel. And the command should be used in redcastle script to start/stop the security functions of RedCastle only.

**SEE ALSO**

**mkinitdata**

**NAME**

**mkinitdata – Create and delete RedCastle initial data**

**SYNOPSIS**

**mkinitdata -u USER -m enable|warn**

**mkinitdata -d**

**DESCRIPTION**

mkinitdata is a command to create the initial data of RedCastle. The command is used only once after the installation of RedCastle. When you install Asianux Server 3, if you configure in Anaconda to use RedCastle, you don't need to use this command since RedCastle creates the initial data automatically. By using this command, you can designate RedCastle administrator and operation mode. And you should use this command before you load RedCastle's security functions onto the kernel.

**OPTIONS**

**-d**

Delete RedCastle initial data.

**-m enable|warn**

RedCastle operation mode. It has one of enable or warn.

**-u USER**

RedCastle administrator. User account that can login with login shell and home directory.

**SEE ALSO**

**mkinit**

## **NAME**

**modcat - Modify the security group attribute**

## **SYNOPSIS**

**modcat CID -p NEW\_PARENT\_CID|-n NEW\_NAME**

## **DESCRIPTION**

Modcat is a command to modify the security group attribute. Modifiable attributes of the security group are security group name and its hierarchic structure. If you want to change its hierarchic structure, you have to be careful on two things. The first one is that the security group can not move to its own sub security group. And the second one is that if a security group moves, its own sub security groups will move together.

You can not change the hierarchic structure of two default security groups ( 'ROOT' and 'System Admins.')

provided by RedCastle. Only you can change its security group name.

## **OPTIONS**

### **CID**

Security group ID that will be modified.

### **-n NEW\_NAME**

New name for the security group to be modified

### **-p NEW\_PARENT\_CID**

New parent security group ID. Security group will be belonged to this new parent security group.

## **SEE ALSO**

**addcat, delcat, getcat**

## **NAME**

**modmu - Modify security attributes of a secured user**

## **SYNOPSIS**

**modmu [-r SO|SA|MU] [-c CID] [-I LEVEL] [-s Y|N] USER**

## **DESCRIPTION**

modmu is a command to change the security attributes of a secured user. Please refer to the 'addmu' command for more detailed description of modifiable security attributes.

If you don't use '-c' option while you are modifying the security role, the security group will be changed into the default security group by its security role automatically. If you want to modify security group only, it only can be modified to the security group with same security role. If the modifying security group's security role is different from the present one, you have to use '-r' option.

## **OPTIONS**

**-r SO|SA|MU**

New security role

**-c CID**

New security group ID.

**-I LEVEL**

New security level

**-s Y|N**

Root Transfer Control Option

## **USER**

A secured user to be modified its attribute

## **SEE ALSO**

**addmu, delmu, getmu, getcat**

## **NAME**

**psx - Query the security attribute of processes**

## **SYNOPSIS**

**psx [-aef]**

## **DESCRIPTION**

psx is a command to query the security attribute of processes. Output result of the command will be same as the output of 'ps' command and also can confirm the security attributes too. The security attribute is composing of security role, security level and security group ID. The security role will be one of 'SO', 'SA', 'UXR', 'MSO', 'MSA', 'MU', and 'UX'. If the security role of process is one of 'SO', 'SA', and 'UXR', then it means root process. And if the security role is one of 'MSO', 'MSA', 'MU', and 'UX', then it means user process. All processes except security role of 'UXR' and 'UX' are processes of secured users. The normal user's process security level and security group ID will be expressed in 0, 0.

## **OPTIONS**

**-a**

Select all processes except session leaders and processes not associated with a terminal.

**-e**

Select all processes.

**-f**

Full-format listing.

**NAME**

**rclogd – RedCastle Log Daemon**

**SYNOPSIS**

**rclogd**

**DESCRIPTION**

rclogd is log daemon of RedCastle. Log daemon collects security violation logs and system logs generated during operation. If you want to initiate the security functions of RedCastle, the log daemon should be in running status. If you start or stop the security functions of RedCastle by using redcastle script, log daemon will also start or stop its running automatically.

## NAME

**rgconf - RedCastle Environment management tool**

## SYNOPSIS

**rgconf -c|-e|-g|-l|**

## DESCRIPTION

rgconf is a command to manage operation environment of RedCastle. RedCastle has independent operation mode by each function. By using rgconf command, you can set each function's operation mode. The following table shows each function's operation mode and possible setting values.

Function	ips.conf	Setting Value
MAC	MACMODE	Default Setting : on (on/warn/off)
ACL	ACLMODE	Default Setting : on (on/warn/off)
root SU Control	ROOTSU	Default Setting : on (on/off)
User SU Control	USERSU	Default Setting : warn (on/warn/off)
Command Control	COMMAND	Default Setting : on (on/warn/off)
SETUID Control	SETUID	Default Setting : warn (on/warn/off)
KILL Deny	KILLMODE	Default Setting : on (on/warn/off)

There is another separate operation mode for RedCastle operation different from the each function's operation mode. By using rgctl command, you can set RedCastle operation mode. Based on the RedCastle operation mode, each function's operation mode will work as follows.

RedCastle Operation Mode	Operation Mode by Function
enable	Use Operation Mode by Function. Each Operation Mode by Function can be changed.
warning	All functions will be operated in warning mode. Each Operation Mode by Function can not be changed.

To modify each function's operation mode, ips.conf file will be used. To do this, create ips.conf file by using '-e' option first. And change the value of the operation mode you want to modify in ips.conf file, then apply the modified value by using '-l' option.

## OPTIONS

**-c**

Compare the current environment with environment file.

**-e**

Export the current environment to environment file.

**-g**

Query the current environment.

**-l**

Apply the new environment. Firstly, edit environment file.

## **NAME**

**rgctl - RedCastle security module management tool**

## **SYNOPSIS**

**rgctl start [enable|warning]**

**rgctl restart [enable|warning]**

**rgctl stop|info|sync**

## **DESCRIPTION**

Rgctl is a command to start/stop the security function of RedCastle and query the operation status of RedCastle. When you try to start/stop the security function of RedCastle, you can set the operation mode of RedCastle. This will be one of enable or warning operation mode and if you do not select one of these, former operation mode will be applied.

## **OPTIONS**

### **info**

Query RedCastle operation status.

### **restart [enable|warning]**

Restart RedCastle security functions. It can be set the operation mode and if not set the mode, start the current operation mode or the earlier operation mode.

### **start [enable|warning]**

Start RedCastle security functions. It can be set the operation mode and if not set the mode, start the earlier operation mode.

### **stop**

Stop RedCastle security functions.

### **sync**

Synchronize file security attributes to file.

## NAME

**seclog – Query the security log**

## SYNOPSIS

**seclog [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][-y LOG\_TYPE][-l LOG\_LEVEL][-m KEYWORD]**

## DESCRIPTION

seclog is a command to query security log. Security log includes all logs generated during the operation of RedCastle. Security log consists of log generating time, log collect time, log generating location and message.

If you use '-s' and '-p' option, you can see a specific part of queried result. PG\_SIZE specified by the '-s' option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the '-p' option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use '-f' and '-t' option, you can also specify the period that will be queried. The basic form of 'FROM\_DATE' and 'TO\_DATE' is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use 'FROM\_DATE' only, it will query the log data during period specified 'FROM\_DATE' to the present. And if you use 'TO\_DATE' only, it will query the log data during period from the oldest time to the specified 'TO\_DATE'.

LOG\_TYPE specified by '-y' option shows a specific location in which security log is generated. You can query log data which are generated from the specific module only by using LOG\_TYPE. In doing this, 1 means kernel module, 2 means agent daemon, 3 means log daemon, 4 means RedCastle commands, 5 means IP table, and 7 means security logs generated from RedCastle PAM module.

LOG-LEVEL specified by '-l' option shows the level of security log. You can query a specific level's log data only by using LOG\_LEVEL. In doing this, 1 means Info, 2 means Notice, 3 means Warning, 4 means Error, and 5 means Critical level.

## OPTIONS

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

**-f FROM\_DATE -t TO\_DATE**

The period that will be queried.

**-y LOG\_TYPE**

Security log types that will be queried.

**-l LOG\_LEVEL**

Security log level that will be queried.

**-m KEYWORD**

Query logs that include specific KEYWORD string.

**NAME**

**setmf – File security attribute management tool**

**SYNOPSIS**

**setmf [-R] -c CID -I LEVEL FILE**

**setmf -r|-Rr FILE**

**DESCRIPTION**

setmf is a command to assign or delete file security attributes. The security attributes consists of security role, security level and security group. To assign security attributes to a file, you have to designate security level and security group ID. The security level will have a value between 1 to 7 and the security group ID will be designated as same as the security group's ID existing in the Security Group List. The security group can be checked by using 'getcat' command. The security role will be assigned based on the security group automatically.

**OPTIONS****-c CID**

CID is security group ID in the existing Security Group List and it can be confirmed with getcat command.

**FILE**

File that will be set its security attribute

**-I LEVEL**

Security level. It has one of from 1 to 7, and 1 is the highest level, and 7 is the lowest.

**-r**

Delete security label. If you use this option, other options will be ignored.

**-R**

Set subdirectories recursively.

**SEE ALSO**

**getmf, getcat**

**NAME**

**sosd – RedCastle Agent Daemon**

**SYNOPSIS**

**sosd [-p PORT]**

**DESCRIPTION**

sosd is agent daemon of RedCastle. The agent daemon takes care of communications with the RedCastle Manager module. In this communication with the Manger, port No. 5002 will be used basically. If you want to use the other ports, you can use '-p' option to do it. By using redmanager script, you can start or stop the agent daemon.

**OPTIONS****-p PORT**

Communication port number with RCManager. It is set 5002 by default.

**NAME**

**suctl - SU Permit User List management tool**

**SYNOPSIS**

**suctl -a|-d USER**

**suctl -g|-i**

**DESCRIPTION**

suctl is a command to manage the SU Permit User List. RedCastle provides 'User Transfer Allow' function to restrict user status transfer from 'root user' to 'secured user' by using su command. Only root with SO security role can transfer to any users and root with SA or UXR security role can not transfer to the user. If you need user status transfer from root to secured user, you have to add the user into the SU Permit User List. Only Redcastle secured users can be added into the SU Permit User List and the secured user with SO security role can not be registered.

**OPTIONS****-a USER**

Security user account that will be added on the SU Permit User List. The Security user with SA or MU role only can be added on the list. Security user can confirm with getmu command.

**-d USER**

Delete a USER from the SU Permit User List.

**-g**

Query the SU Permit User List.

**-i**

Delete all users from the SU Permit User List.

**SEE ALSO**

**getmu, rgconf**

## **NAME**

**syslog – Query system log**

## **SYNOPSIS**

**syslog [-s PG\_SIZE][-p PG\_NUM][-f FROM\_DATE -t TO\_DATE][-I LOG\_HOST][-a APPLICATION][-m KEYWORD]**

## **DESCRIPTION**

syslog is a command to query system log.

If you use '-s' and '-p' option, you can see a specific part of queried result. PG\_SIZE specified by the '-s' option is number of lines that will be shown. If you do not specify the line numbers, it will be set as 50 in default. PG\_NUM specified by the '-p' option is number of pages that will be shown. Whole queried results will be divided by PG\_SIZE and this will compose each page. If you do not specify page numbers, it will show the first page. And if you use '-f' and '-t' option, you can also specify the period that will be queried. The basic form of 'FROM\_DATE' and 'TO\_DATE' is YYYYMMDD, and you can extend this to query specific time by using the form of YYMMDD[hh[mm[ss]]]. If you use 'FROM\_DATE' only, it will query the log data during period specified 'FROM\_DATE' to the present. And if you use 'TO\_DATE' only, it will query the log data during period from the oldest time to the specified 'TO\_DATE'.

## **OPTIONS**

### **-s PG\_SIZE**

The number of lines that will be shown.

### **-p PG\_NUM**

The number of pages that will be shown.

### **-f FROM\_DATE -t TO\_DATE**

The period that will be queried.

### **-I LOG\_HOST**

Query logs that has generated in specific hosts.

### **-a APPLICATION**

Query logs that has generated in specific applications.

**-m KEYWORD**

Query logs that include KEYWORD string.

**NAME**

**uroleadd - Add new user role on the User Role List**

**SYNOPSIS**

**uroleadd ROLE\_NAME**

**DESCRIPTION**

Uroleadd is a command to create a new user role. The user role name have to be less than 23 characters. And if you want to configure complete user role, use uroleassign command to assign attribute to the user role.

**OPTIONS****ROLE\_NAME**

User role name that will be added.

**SEE ALSO**

**uroledele, uroleget**

## NAME

**uroleassign - User role attribute management tool**

## SYNOPSIS

**uroleassign -a ROLE\_NAME -u UID|-g GID|-c CID|-l LEVEL**

**uroleassign -d ROLE\_NAME -u UID|-g GID|-c CID|-l LEVEL**

## DESCRIPTION

Uroleassign is a command to manage user role attribute. To configure complete user role, use this command for assigning attribute to the user role. New subject can be created by assigning attribute to the user role. The following attributes can be assigned to the security role.

<b>Security Attribute(Subject)</b>	<b>Description</b>
<b>User</b>	<b>User existing in the system. Up to 8 people can be included.</b>
<b>Group</b>	<b>Group existing in the system. Up to 8 group can be included.</b>
<b>Security Group</b>	<b>RedCastle Security Group. Only one possible. All users belong to Security Group.</b>
<b>Security Level</b>	<b>RedCastle Security Level. Only one possible. All secured users have specific security levels.</b>

## OPTIONS

### **-a ROLE\_NAME**

Add the attribute to the user role named ROLE\_NAME. To add the attribute, use the options -u, -g, -c and -l.

### **-c CID**

Security Group. CID is the existing security group ID in the Security Group List and the security group can be query with getcat command.

### **-d ROLE\_NAME**

Delete the attribute from user role named ROLE\_NAME. To delete the attribute, use the options -u, -g, -c and -l.

### **-g GID**

Group ID. GID means Group ID of the existing user.

**-I LEVEL**

Security Level. It has one of from 1 to 7.

**-u UID**

User ID. UID means user id of the existing user.

**SEE ALSO**

**uroleget, getcat, getmu**

**NAME**

**uroledel - Delete user role from the User Role List**

**SYNOPSIS**

**uroledel ROLE\_NAME**

**DESCRIPTION**

Uroledel is a command to delete Security Role from User Role List.

**OPTIONS****ROLE\_NAME**

User role name that will be deleted

**SEE ALSO**

**uroleadd, uroleget**

**NAME**

**uroleget - Query the User Role List**

**SYNOPSIS**

**uroleget**

**DESCRIPTION**

Uroleget is a command to query User Role List.

**SEE ALSO**

**uroleadd, uroledele**