



We Secure the Internet.

# Check Point® Troubleshooting and Debugging Tools for Faster Resolution

January 24, 2006



## IMPORTANT

**Check Point recommends that customers stay up-to-date with the latest service packs, HFAs and versions of security products, as they contain security enhancements and protection against new and changing attacks.**

In This Section

<i>Mandatory Support Information</i>	<i>page 1</i>
<i>FireWall Common debugging</i>	<i>page 2</i>
<i>Security Server debugging</i>	<i>page 4</i>
<i>VPN debugging</i>	<i>page 5</i>
<i>Provider-1 debugging</i>	<i>page 5</i>
<i>VPN-1 VSX debugging</i>	<i>page 6</i>
<i>ClusterXL debugging</i>	<i>page 6</i>
<i>Connectra debugging</i>	<i>page 6</i>
<i>FireWall-1 GX debugging</i>	<i>page 6</i>
<i>InterSpect debugging</i>	<i>page 7</i>
<i>SNX – SSL Network Extender debugging</i>	<i>page 7</i>
<i>Further Debugging – Memory Diagnostics</i>	<i>page 8</i>

## Mandatory Support Information

The following information is the information that the Customer needs to provide Support when opening a Support Service Request

- 1) Problem Description, provide a detailed description of the issue
- 2) Network Topology Diagram, provide a comprehensive diagram which illustrates the described problem.
- 3) Execute CPINFO on the required Check Point component. To create CPINFO, execute %  
cpinfo -o <Output file>

---

Over and above the information in the Service Request, it is recommended to do basic debugging. The debugging commands can be found in this document.

## Important Comments

- In certain specific scenarios, the debugging commands included in this document may need to be supplemented by more advanced debugging procedures. Advanced procedures should be executed in conjunction with the Check Point Escalation engineers.
- Debugging should only be performed when the described issue can be captured.

## FireWall Common debugging

### Kernel debugging

#### Usage

```
% fw ctl debug -buf [buffer size]
% fw ctl debug [-x] [-m <module>] [+|-] <options | all | 0>
% fw ctl kdebug -f > <output file>
```

To disable the Kernel debugging, execute:

```
% fw ctl debug -buf 0
% fw ctl debug x
```

#### Common Syntax

```
% fw ctl debug -buf 12288
% fw ctl debug -m fw conn drop ld packet if
% fw ctl kdebug -f > <output file>
```

The `ld` option may cause high CPU usage. It is advised to use it for short session debugging only.

To execute the kernel you can also use `fw ctl zdebug` to allocate the buffer (where the buffer can only be 1024).

```
% fw ctl zdebug
% fw ctl kdebug -f > <output file>
```

### User Mode Processes debugging

In This Section

<a href="#">Usage</a>	<a href="#">page 3</a>
<a href="#">Debugging CPD</a>	<a href="#">page 3</a>
<a href="#">Debugging FWM</a>	<a href="#">page 3</a>
<a href="#">Debugging FWD</a>	<a href="#">page 3</a>

---

## Usage

```
% fw debug <process name> <on/off> TDERROR_ALL_ALL=<value 1-5>
```

CPD is treated differently from the other User Mode processes and will be executed differently, see [“Debugging CPD” on page 3](#).

### Debugging CPD

CPD is a high in the hierarchichal chain and helps to execute many services, such as Secure Internal Communcation (SIC), Licensing and status report.

For CPD debug, execute: `% cpd_admin debug on TDERROR_ALL_ALL=5`

The debug file is located under `$CPDIR/log/cpd.elg`

To stop the CPD debug, execute: `% cpd_admin debug off TDERROR_ALL_ALL=1`

### Debugging FWM

The FWM process is responsible for the execution of the database activities of the SmartCenter server. It is; therefore, responsible for Policy installation, Management High Availability (HA) Synchronization, saving the Policy, Database Read/Write action, Log Display, etc.

For FWM debug, execute:

```
% fw debug fwm on TDERROR_ALL_ALL=5
```

```
% fw debug fwm on OPSEC_DEBUG_LEVEL=9
```

The debug file is located under `$FWDIR/log/fwm.elg`

To stop the FWM debug, execute:

```
% fw debug fwm off TDERROR_ALL_ALL=1
```

```
% fw debug fwm off OPSEC_DEBUG_LEVEL=1
```

### Debugging FWD

The FWD process is responsible for logging. It is executed in relation to logging, Security Servers and communication with OPSEC applications.

For FWD debug, execute: `% fw debug fwd debug on TDERROR_ALL_ALL=5`

The debug file is located under `$FWDIR/log/fwd.elg`

To stop the FWD debug, execute: `% fw debug fwd off TDERROR_ALL_ALL=1`

## FireWall Monitor Network Capturing

The FireWall Monitor is responsible for packet flow analysis.

To execute: `% fw monitor -e "accept;" -o <output file>`

---

# Security Server debugging

## Debugging User Authentication

### Usage

Debugging is done on the service itself (in. ahttpd, in. atelnetd, in. aftp. etc.)

```
% fw debug <process name> on TDERROR_ALL_ALL=5
```

The debug file is located under: \$FWDIR/log/ahttpd.elg\* or \$FWDIR/log/aftp.elg\* or \$FWDIR/log/atelnetd.elg\* depending on the service that you are debugging.

## HTTP Security Server

For HTTP Security Server debug, execute:

```
% fw debug in.ahttpd on TDERROR_ALL_ALL=5
```

```
% fw debug in.ahttpd on OPSEC_DEBUG_LEVEL=3
```

The debug file is located under: \$FWDIR/log/ahttpd.elg\*

If more than one HTTP Security Server process is running, execute:

```
% fw kill fwd
```

```
% setenv TDERROR_ALL_ALL=5
```

```
% setenv OPSEC_DEBUG_LEVEL=3
```

```
% fwd -d >& <output file> &
```

**Note** - The setenv commands used above correlate with Unix environment. For other platforms, execute the relevant command.

## SMTP Security Server

To debug the SMTP Security Server, execute:

```
% fw debug in.asmtpd on TDERROR_ALL_ALL=5.
```

The debug file is located under \$FWDIR/log/asmtpd.elg\*

To debug the mdq, execute the following commands:

```
% fw debug mdq on TDERROR_ALL_ALL=5.
```

The debug file is located under \$FWDIR/log/mdq.elg\*

## Debugging Session Authentication

To debug Session Authentication, execute:

```
% fw debug in.assessiond on TDERROR_ALL_ALL=5
```

The debug file is located under: \$FWDIR/log/assessiond.elg\*

## Debugging Client Authentication

For HTTP to port 900, execute:

---

```
% fw debug in.ahclientd on TDERROR_ALL_ALL=5
```

For Telnet to port 259, execute:

```
% fw debug in.aclientd on TDERROR_ALL_ALL=5
```

The debug file is located under: \$FWDIR/log/ahclientd.elg\*

## **VPN debugging**

### **On the Module**

To start, execute:

```
% vpn debug trunc.
```

This command is equivalent to these two commands: `vpn debug on`, `vpn debug ikeon`.

To stop, execute:

```
% vpn debug off; vpn debug ikeoff.
```

The debug file is located under \$FWDIR/log/ike.elg and \$FWDIR/log/vpnd.elg

## **FireWall Monitor for packet flow analysis**

```
% fw monitor -e "accept;" -o <output file>
```

### **Client Side**

The Client side can only run under the root directory (C :/...)

To start, execute:

```
% sc debug on
```

To stop, execute:

```
% sc debug off
```

The debug file is located under `sr_service_tde.log`, under the SecuRemote installation folder, for example: `C:\Program files\CheckPoint\SecuRemote`.

For packet capture from the Client side, execute:

```
% srfw monitor -e "accept;" -o <output file>
```

## **Provider-1 debugging**

### **MDS Level**

Most of the MDS actions are performed by the MDS's `fwm` process, execute:

```
% mdsenv
```

```
% fw debug mds on TDERROR_ALL_ALL=5
```

```
% fw debug mds on OPSEC_DEBUG_LEVEL=9
```

The debug file is located under `/opt/CPsuit-R60/fw1/log/mds.elg`

---

## CMA Level

See [“FireWall Common debugging” on page 2](#).

## VPN-1 VSX debugging

See [“FireWall Common debugging” on page 2](#), either refer to user mode or kernel, as necessary.

## ClusterXL debugging

For ClusterXL debugging for Clustering, Synchronization, High Availability, Fail-over, execute:

```
% cphaprob state
% cphaprob -ia list
% cphaprob -a if
% fw ctl pstat
```

Kernel debug for packet filter analysis

```
% fw ctl debug -buf 12288
% fw ctl debug -m fw conn drop packet if sync
% fw ctl debug -m cluster all
% fw ctl kdebug -f > <output file>
```

## Connectra debugging

For Connectra debugging issues relating to Web, files, Webmail, OWA, iNotes, Citrix, the httpd process should be debugged:

To turn the debug on, under: `$CVPNDIR/conf/httpd.conf` change `LogLevel` to `debug`.

You should execute the process: `cvpnrestart`

The output is located at: `$CVPNDIR/log/httpd.log`

For debugging authentication issues, execute: `Debug cvpnd`

Run: `cvpnd_admin debugset TDERROR_ALL_ALL=5`

To start, execute: `% cvpnrestart`

The debug file is located under `$CVPNDIR/log/cvpnd.elg`

To stop debug, run:

```
% cvpnd_admin debug off
```

## FireWall-1 GX debugging

See [“FireWall Common debugging” on page 2](#).

Kernel debug for packet filter analysis

---

```
% fw ctl debug -buf 12288
% fw ctl debug -m fw conn drop ld packet filter
% fw ctl kdebug -T -f > <output file>
```

## InterSpect debugging

Kernel debug for packet filter analysis

```
% fw ctl debug -buf 12288
% fw ctl debug -m fw conn drop packet if
% fw ctl kdebug -f > <output file>
```

Additional kernel debug options for InterSpect:

- portscan, for port scanning issues
- dynlog, for dynamic logging
- mail, for mail security in the kernel
- sam, for SAM IP address blocking

Kernel debug for Packet Drop, execute:

```
% fw ctl zdebug + drop
```

Kernel debug for SmartDefense TCP Streaming, execute:

```
% fw ctl zdebug + tcpstr + cifs
```

Kernel debug for Dynamic list (SAM), execute:

```
% fw tab -t sam_requests_v2 -u -f
% fw samp
```

## SNX – SSL Network Extender debugging

### Server Side

```
% vpn debug trunc
% vpn debug on slim=5
```

Debug can be found at \$FWDIR/log/vpnd.elg.

You should execute `vpn debug on [DEBUG_TOPIC=5]`. The relevant debug topics are: proxy, rasta, rasta\_protocol and slim.)

### Client Side

For the service:

Type regedit at the command prompt and set:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\cpextender\parameters\dbg_level to 5
```

Open the Command Line interface window and execute:

---

```
% net stop cpextender
% net start cpextender (or kill slimsvc.exe)
```

The debug file is located under:

```
%Program Files%\CheckPoint\SSL Network Extender\slimsvc.log
```

For the ActiveX: (only when using ActiveX with Internet Explorer), type regedit at the command prompt and set the following:

```
% set HKEY_CURRENT_USER\Software\CheckPoint\SSL Network
Extender\parameters\dbg_level to 5
```

The debug file is located under %APPDATA%\Check Point\extender\activex.log.

For the Applet: (when using the Applet version) SNX can be used by Microsoft JVM or by other vendors (SUN, IBM...). To view the Java console when using Microsoft JVM you need to check **Java console enabled (requires restart)** in the **Internet Options Advanced** tab and restart Internet Explorer. You can also switch between the different JVMs (in case you have two or more) in the same tab.

## Further Debugging – Memory Diagnostics

The following utilities applies to all non-Windows systems supported by Check Point:

```
% free
% vmstat 2 10
% sar -k 2 10
% top
% ps -auxw
% cat /proc/meminfo
% cat /proc/slabinfo
```

### Routing information

```
% arp -a
% netstat -ie
% netstat
```