

Enabling Actions from NetScreen-Security Manager

After you complete all the configuration in the SRC software, you configure the **thm.py** script—a script that implements the messaging to record problem incidents and identifies the action for the SRC software to take. If the **thm.py** script cannot send an event to the SRC-TMP, it records the event in a file.

In a testing environment, you can use the **thm.sh** script to set up and troubleshoot a configuration that integrates NetScreen-Security Manager into an SRC-managed environment. The **thm.sh** script sets the library paths, redirects debugging output, and executes the **thm.py** script. Do not use the **thm.sh** script in a production environment.

The **thm.py** script requires Python version 2.3. The SMCpython package available in the **SDK+AppSupport+Demos+Samples.tar.gz** file on the Juniper Web site at <https://www.juniper.net/support/csc/swdist-erx/src.html> contains Python version 2.3.

Before you configure scripts:

- Complete all other configuration for the Threat Mitigation Application.
- Verify the location where Python is installed on the system. If you installed Python from the **SDK+AppSupport+Demos+Samples.tar.gz** file on the Juniper Web site at <https://www.juniper.net/support/csc/swdist-erx/src.html>, the default installation directory is */opt/UMC/python*. If you installed Python to a different directory, update the paths in *thm.py* and in *thm.sh* (if you use this file).
- For a production environment, start NetScreen-Security Manager in an environment in which the library path includes the Python libraries.

The **thm.py** script provides configuration properties to allow you to create customized implementations. You can locate the scripts in the */opt/UMC/conf/thma/scripts* directory.

To configure scripts:

1. Edit the *thm.py* file to set the configuration properties.
2. Copy the *thm.py* file and the *thm.sh* file (if you use this file) to the appropriate directory for NetScreen-Security Manager. For the location of this directory, see the NetScreen-Security Manager documentation at <http://www.juniper.net/techpubs/software/management/security-manager/>.

Fields in the thm.py File

The following list describes the fields in the *thm.py* file used by the SRC-TMP.

RECORD URL

- URL of the record interface for the SRC-TMP that stores information received from NetScreen-Security Manager. The interface records information about detrimental traffic in the ATTACK table in the database. The security rules configured in NetScreen-Security Manager determine the type of incidents recorded.
- Value—URL in the form “
http(s):// < user > : < password > @ < host > : < port > /thmp/record”
 - < user > —Client ID
 - < password > —Password associated with the client ID
 - < host > —Hostname or IP address of the server on which the SRC-TMP runs
 - < port > —Port number used by the SRC-TMP on the server
- Guidelines—Enclose the URL in quotation marks because this entry is a Python string.
- Default—RECORD_URL = “ http://admin:secret@127.0.0.1:8080/thmp/record”
- Example—RECORD_URL = “ https://admin:secret@192.0.2.25:8443/thmp/record”

FAIL_DIR

- Pathname to the directory that records incidents that were not successfully sent to the record URL.
- Value—Pathname in the form “ < pathname > ”
- Guidelines—Enclose the pathname in quotation marks because this entry is a Python string.
- Default—FAIL_DIR = “ failedEvents”

FAIL_FILE_LIMIT

- Maximum number of events that will be recorded in the fail directory. If this number is exceeded, the oldest event is deleted to make room for the most recent event. If this number is 0, the script will not add any failed events, check the fail directory for failed events, or spawn the daemon process.
- Value—Integer in the range 0–2147483647
- Default—FAIL_FILE_LIMIT = 100

NUM_RETRIES

- Number of times the script (and daemon process) will retry sending an event to the record URL if the first attempt fails. If the retry limit is reached, the script gives up and writes the event to the fail directory. If the retry limit is reached by the daemon process, it stops trying to send failed events until its next interval. For example, if NUM_RETRIES is 2, then the script will try at most 3 times to send an event to the record URL.
- Value—Integer in the range 0–2147483647
- Default—NUM_RETRIES = 2

DAEMON_INTERVAL

- Amount of time that the daemon process will take between attempts to send events to the fail directory. When first started, the daemon process will wait this number of seconds before trying to send events recorded in the fail directory. If it fails to send any event in the fail directory, it will not try to send any more events for this amount of time.
- Value—Number of seconds in the range 0–604800 (1 week)
- Default—DAEMON_INTERVAL = 30

DEBUG

- Specifies whether or not to print debugging messages.
- Value
 - True—Print messages.
 - False—Do not print messages.
- Guidelines—Set this value to True only for troubleshooting. Set this value to False to minimize the effects on performance.
- Default—DEBUG = True

SEND_XML

- Specifies whether or not to send attack log events to the SRC-TMP as an XML document.
- Value
 - True—The attack log event is sent to the SRC-TMP as an XML document.

- False—The script parses the XML document and posts the relevant data as individual request parameters.
- Guidelines—Set this value to True to minimize CPU resources consumed by this script. Set this value to False to minimize the CPU resources used by the SRC-TMP in recording the attack. Setting this value to False will cause the script to consume approximately 60% more CPU resources.
- Default—SEND_XML = False

BACKGROUND_LOG_FILE

- Name of the file that logs messages for the process that retries sending attack log events. This file is created in the directory specified by FAIL_DIR.
- Value—Filename in the form “ <filename> ”
- Guidelines—Enclose the filename in quotation marks because this entry is a Python string. Set this value to None for no background logging.
- Default—BACKGROUND_LOG_FILE = “ thm.log”

BACKGROUND_LOG_FILE_LIMIT

- Maximum size of the background log file. If this number is exceeded, a sequence number is appended to the filename and a new log file is started.
- Value—Number of bytes in the range 0-2147483647
- Default—BACKGROUND_LOG_FILE_LIMIT = 50000