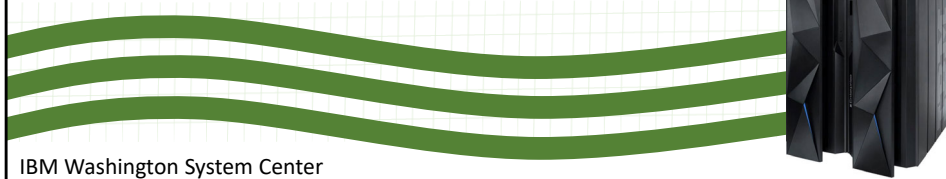


Securing and Encrypting Network Traffic
with z/OS Communications Server and
Policy Agent

Security Workshop

Traffic Regulation Management Daemon
(TRMD)



IBM Washington System Center
IBM Technical Sales Support

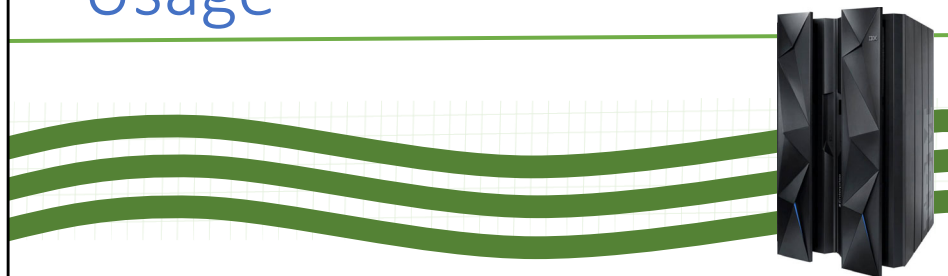
Trademarks

- **The following are Registered Trademarks of the International Business Machines Corporation in the United States and/or other countries.**
 - IBM
 - z/OS
- **The following are trademarks or registered trademarks of other companies.**
 - Microsoft is a registered trademark of Microsoft Corporation in the United States and other countries.
- All other products may be trademarks or registered trademarks of their respective companies.
- Refer to www.ibm.com/legal for further legal information.
- OSA-Express Features
- There are many different types of OSA-Express features. In this document where OSA is mentioned it refers to an OSA port on any of the OSA-Express features unless a specific OSA-Express feature is mentioned.

Agenda

- TRMD's Role in Policy Agent
- Implementing TRMD
 - Prerequisites
 - Setting the Timezone variable
 - Initializing and stopping TRMD
 - Format of the Environment Variable file or dataset
- Generating Formatted Reports for IDS with the 'trmdstat' Command

TRMD's Role in Policy Usage

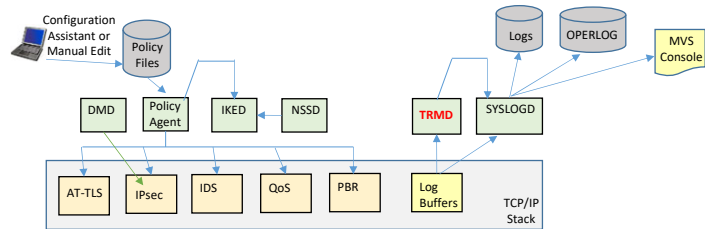


009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 4

Lots of Different Policy Types and Started Tasks



- Policy Agent
 - Installs Policies into the TCP/IP stack
- TCP/IP Stack
 - Enforces the Policies
- DMD (Defense Manager Daemon)
 - ipsec command can be used to install temporary IP Filter rules.
- IKED (Internet Key Exchange Daemon)
 - Required for IPsec Dynamic VPN (Virtual Private Network) Tunnels
- NSSD (Network Security Server Daemon)
 - Required for IKEv2
 - Provides central RACF certificate repository for remote IKED applications
 - Provides DataPower access to RACF
- **TRMD (Traffic Regulation Management Daemon)**
 - **Required for log messages to syslogd for IP Filter, IPsec, and IDS**
- SyslogD
 - Recommended for logging

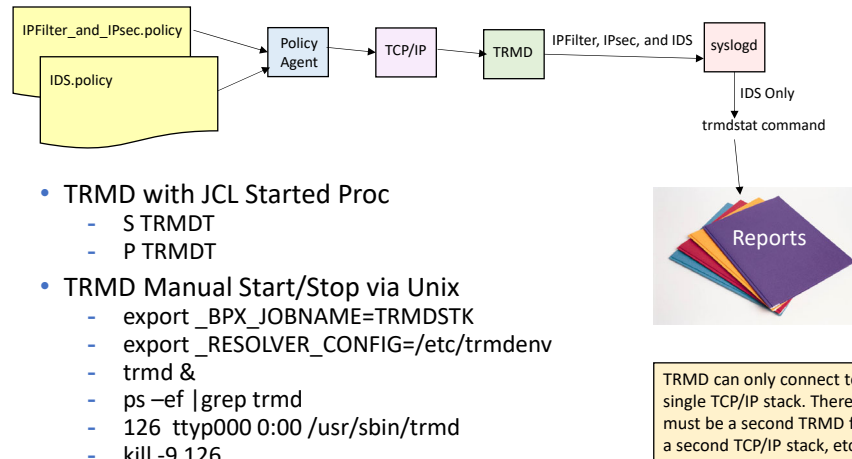
009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 5

Configuration Assistant can really help getting the whole environment setup.

TRMD (Traffic Regulation Management Daemon)



009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 6

Only a few Policy Agent policy types require the services of TRMD:

- IDS
- IP Filtering
- IPsec (all forms: manual, dynamic pre-shared key, dynamic RSA signature mode)

TRMD captures messages from the TCP/IP stack that are related to these types of policies. TRMD then writes the messages to the SYSLOG Daemon.

- Because TRMD has affinity to only one TCP/IP stack, you may have multiple instances of TRMD running. As a result, you can take advantage of Syslog Daemon Isolation to split the TRMD messages into separate log files.

The Traffic Regulation Management Daemon (trmd) receives information (statistics) from the TCP/IP stack (TRM function) which it then logs into SYSLOGD.

Can be started from the z/OS Unix shell or as started procedure; preferred method is with a started procedure.

Note that UNIX forks the task and appends a digit if the jobname is shorter than 8 characters in length. ("s trmd" becomes "trmd1")

Parameters:

-d n : specifies level of debug information (0-3)

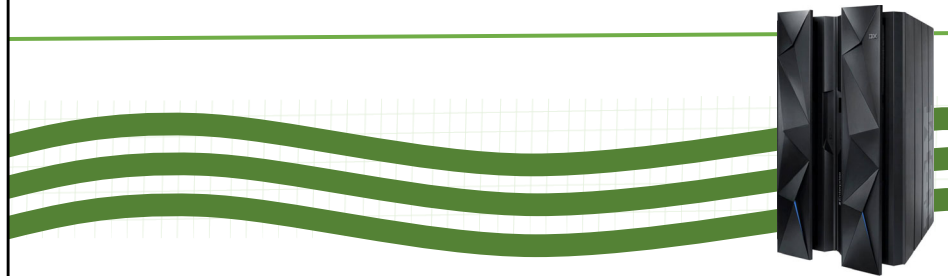
Sample procedure shipped in hlq.SEZAINST(TRMD)

Should be started after TCP/IP, the policy agent and the syslog daemon.

If you fail to start TRMD, the messages generated will be queued up in the TCP/IP stack for a time and a dump can capture them.

However, the only sure way of obtaining the messages is to use TRMD to write them to SYSLOGD.

Implementing TRMD



009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 7

TRMD Prerequisites

- APF Authorization:
 - SYS1.TCPIP.SEZALOAD
- UID=0 or Authorization to BPX.SUPERUSER

```
//*
//*TRMD EXEC PGM=IKJEFT01
//*SYSTSPRT DD SYSOUT=*
//*SYSTSIN DD *
//* SETROPTS CLASSACT(STARTED)
//* SETROPTS RACLIST(STARTED)
//* SETROPTS GENERIC(STARTED)
//* ADDUSER TRMD DFLTGRP(OMVSGRP) OMVS(UID(nn) HOME('/'))
//* RDEFINE STARTED TRMD.* STDATA(USER(TRMD))
//* SETROPTS RACLIST(STARTED) REFRESH
//* SETROPTS GENERIC(STARTED) REFRESH
//*
//* Permit access to BPX.SUPERUSER
//* PERMIT BPX.SUPERUSER CLASS(FACILITY) ID(TRMD) ACCESS(READ)
//*
```

- Associated TCP/IP stack must be initialized
- Resolver Config to locate the TCP/IP Stack must be correct prior to initialization
- Logging Timestamps:
 - Event detection in UTC time (Greenwich Mean Time)
 - Recording time (according to Timezone variable)

009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 8

TRMD runs as an APF-authorized program.

The user ID associated with TRMD must be defined with a UID of 0, or must be permitted to become a superuser by having READ access to the BPX.SUPERUSER resource in the FACILITY class.

- See the EZARACF member of SEZAINST for sample RACF commands for TRMD.

The resolver configuration file is used to determine the stack that TRMD will use.

- Ensure that the RESOLVER_CONFIG environment variable is correctly set before starting TRMD.
- Otherwise the default resolver search sequence is used, which may not be what you want.
- A separate instance of TRMD must be run for each TCP/IP stack.

The Log records written by TRMD contain 2 timestamps:

- A timestamp generated when the event was detected by the stack. This timestamp is generated by the stack and is always Coordinated Universal Time (UTC).
- A timestamp that is generated when the syslogd record ID is created. This timestamp is dependent on the setting of the TZ environment variable at the time that TRMD is started.
 - If the installation wants this timestamp to be based on UTC, then ensure the TZ environment variable is properly set (for example, export TZ=0) before starting TRMD

.If running multiple instances of TRMD, consider using the syslogd -u option when starting syslogd. The -u option causes the jobname of the application writing the log record to be included in the log record.

The TCP/IP stack must be running before TRMD can be started.

TRMD can be started from the z/OS shell or as a started task.

TRMD Sample Procedure in hlq.SEZAINST(TRMD)

```
//TRMD      PROC
//TRMD      EXEC PGM=EZATRMD,REGION=4096K,TIME=NOLIMIT,
//          PARM=('POSIX(ON) ALL31(ON)',
//          'ENVAR("LIBPATH=/usr/lib")/')
// * - To pass parameters to TRMD, specify them after the final slash
// * on the PARM statement. For example:
// * // PARM=('POSIX(ON) ALL31(ON) /-d 1')
// *
// *** Examples for specifying configuration data sets
// *
// * Example 1: TCPIP.DATA in partitioned data set
// * // PARM=('POSIX(ON) ALL31(ON)',
// * // 'ENVAR("RESOLVER_CONFIG=//'SYS1.TCPPARMS(TCPDATA)')')
// *
// * Example 2: TCPIP.DATA in HFS file
// * // PARM=('POSIX(ON) ALL31(ON)',
// * // 'ENVAR("RESOLVER_CONFIG=/etc/resolv.conf")')
// *
// * Example 3: Specification of data sets via STDENV DD statement
// * // 'ENVAR("_CEE_ENVFILE=DD:STDENV")')
// *
//STDENV    DD DUMMY
//SYSPRINT  DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSIN     DD DUMMY
//SYSERR    DD SYSOUT=*
//SYSOUT    DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//CEEDUMP   DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
```

Resolver_Config MVS

Resolver_Config Unix

- STDENV stored in MVS (V, VB) or in a unix file
 - Remember STDENV must not be in Fixed Block dataset.

```
RESOLVER_CONFIG=// 'SYS1.CS.TCPPARMS(DATIA)'
TZ=EST5EDT
```

009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 9

TRMD has only one parameter available to it: "-d"

You might enable -d1, -d2, or -d3 for debugging purposes, but you would not usually run with debug mode enabled.

TRMD must find the TCP/IP job name with which it should be associated. It uses the TCPIPJOBNAME value from the TCPIP.DATA file. The TCPIP.DATA file used can be controlled by setting the RESOLVER_CONFIG environment variable. See the three examples within the JCL.

You may also wish to set the TZ variable if you are not using the CEEPRM member for this purpose.

- If using the STDENV method, be careful to allocate any MVS repository for it as a VB file.

Class Example: TRMD Proc

```
//TRMDT      PROC      DATA=DAT&CL1.A,
//          CS=SYS1
//          *          CS=USER
//TRMD      EXEC PGM=EZATRMD,REGION=4096K,TIME=NOLIMIT,
//          PARM=('POSIX(ON) ALL31(ON)',
//          'ENVAR("RESOLVER CONFIG=//'&CS'.CS.TCPPARMS (&DATA) '")',
//          'TZ=EST5EDT"7')
//          *          "TZ=EST5EDT"/ -d1'
//STDENV    DD DUMMY
//SYSPRINT  DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//SYSIN     DD DUMMY
//SYSERR    DD SYSOUT=*
//SYSOUT    DD SYSOUT=*,DCB=(RECFM=F,LRECL=80,BLKSIZE=80)
//CEEDUMP   DD SYSOUT=*,DCB=(RECFM=FB,LRECL=132,BLKSIZE=132)
```

Using MVS System Symbols
"&CL1" = MVS SYSNAME in SYS1.PARMLIB(IEASYMnn)

LE Environment Variables on EXEC Statement

- If executing at MVS2, above JCL resolves to:

```
//TRMDT      PROC
//TRMDT      PROC      DATA=DAT2A,
//          CS=SYS1
//TRMD      EXEC PGM=EZATRMD,REGION=4096K,TIME=NOLIMIT,
//          PARM=('POSIX(ON) ALL31(ON)',
//          'ENVAR("RESOLVER CONFIG=//'SYS1'.CS.TCPPARMS (DAT2A) '")',
//          'TZ=EST5EDT"7')
```

009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 10

You can set up these environment variables by coding them on the EXEC statement, as you see in this example. Alternatively you can set up these environment variables by placing them in a Standard Environment definition that resides either in an MVS dataset or a unix file.
At the bottom of the screen you see how the variables that we have used resolve when the procedure is executed.

trmdstat Command

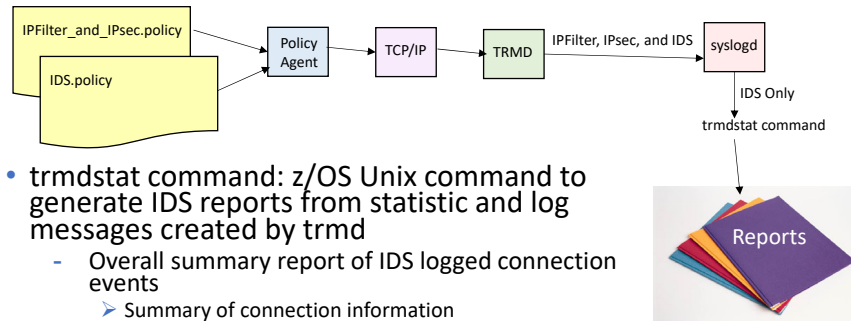


009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 11

Create IDS Reports



- **trmdstat command:** z/OS Unix command to generate IDS reports from statistic and log messages created by trmd
 - Overall summary report of IDS logged connection events
 - Summary of connection information
 - Summary of connection information for a specified host
 - Detailed reports of IDS logged events
 - Details of connection information
 - Details of connection information for a specified host
 - Examples:
 - Reports of logged intrusions defined in the ATTACK policy
 - Reports of logged intrusions defined in the TCP policy
 - Reports of logged intrusions defined in the UDP policy
 - Reports of statistics events

009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 12

The trmdstat command is provided to read the statistics and logging messages about IDS activity only from the syslog daemon output.

- The command operates only against IDS actions -- not against IP Filtering or IPsec
- It writes reports to the z/OS Unix shell screen or its output may be piped into a separate unix file.

trmdstat Syntax

```
>>---trmdstat---| Report Option |---+-----log_filename---<
+---| Report Content |---| Filter |---| Global |---+

Report Options:
+---+I---+
|-----+
+---+A---+
+---+C---+
+---+E---+
+---+G---+
+---+I---+
+---+N---+
+---+Q---+
+---+T---+
+---+U---+
+---+D---+
+---+S---+

Report Content:
|-----+D---+
+---+E---+
+---+S---+

Filter:
|-----+i initial time-----+
+---+f final time-----+
| +---+p 1-65535 -----+
+---+-----+
+---+p port range-----+
+---+h ip address-----+
+---+j stack name-----+
+---+k ip address-----+
+---+a ip address-----+
+---+t ip address-----+
+---+c correlator-----+
+---+n interface_name-----+

Global:
+---+d 0 ---+
|-----+
+---+d n ---+
```

-A Displays the attack summary.
-C Displays the connection summary.
-F Displays the flood summary.
-G Displays the Global TCP Stall summary.
-I Displays the IDS Overall Summary Report.
-N Displays the scan summary.
-Q Displays the TCP Queue Size summary
-T Displays the TCP TR summary.
-U Displays the UDP TR summary.
-D Displays detailed information.
-E Specifies the TCP extended summary report.
-S Displays statistics summary.

Use the trmdstat command to give a consolidated view of the log messages written out by the Traffic Regulation Management daemon (TRMD).

Consult the IP System Administrator's Guide for specifics on the command and views of the various types of output.

Attack Detail Report

```
trmdstat -A -D /tmp/tstlog.log
trmdstat for z/OS CS V1Rn      Wed Nov  8 09:55:36 2008

Log Time Interval   : Aug 21 08:32:09 - Aug 21 10:32:09
Stack Time Interval : Aug 21 14:32:09 - Aug 21 14:32:09
TRM Records Scanned : 71
Port Range          : ALL

ATTACK Events

Packets Discarded
Attack Date and Time  Dst IpAddr  Src IpAddr  Dst Port  Src Port  Correlator  ProbeID
-----
Malf 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 0 0 82334 04010009
IPFr 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 0 0 82336 04030001
IPOP 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 0 0 82338 04050001
PRTO 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 0 0 82339 04060001
Perp 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 13001 10001 82342 04080001
ICMP 8/21/2008 14:32:9.53 51.52.53.54 41.42.43.44 12001 10001 82337 04040009

Packets would have been Discarded
Attack Date and Time  Dst IpAddr  Src IpAddr  Dst Port  Src Port  Correlator  ProbeID
-----
ORAW 8/21/2008 14:32:9.54 41.42.43.44 71.72.73.74 0 0 87999 04020001

TRMD Started          : Aug 21 10:32:09
```

009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 14

This report will be displayed when the -A and -D options are specified on the trmdstat command. It will display the contents of attack event records.

Attack: Indicates the ATTACK type causing the packet to be discarded or would have been discarded if "TypeActions Limit" had been specified in the policy:

- Malf - Malformed Packet
- ORaw - OutBound Raw
- IPFr - IP Fragment
- ICMP - ICMP
- IPop - IP Options
- PRTO - IP Protocol error
- Flod - Flood
- NoID - Not identified

Date and Time: Indicates the date and time contained in the record written at the time of the attack.

Dst IpAddr: Indicates the destination IP address of the attack.

Src IpAddr: Indicates the source IP address of the attack.

Dst Port: Indicates the destination port number.

Src Port: Indicates the source port number.

Correlator: Indicates the trace correlator.

ProbeID: Indicates the probe ID.

End of Topic



009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 15

End of Topic



009_ZCS301_TRMD

© Copyright IBM Corporation 2023

Page 16