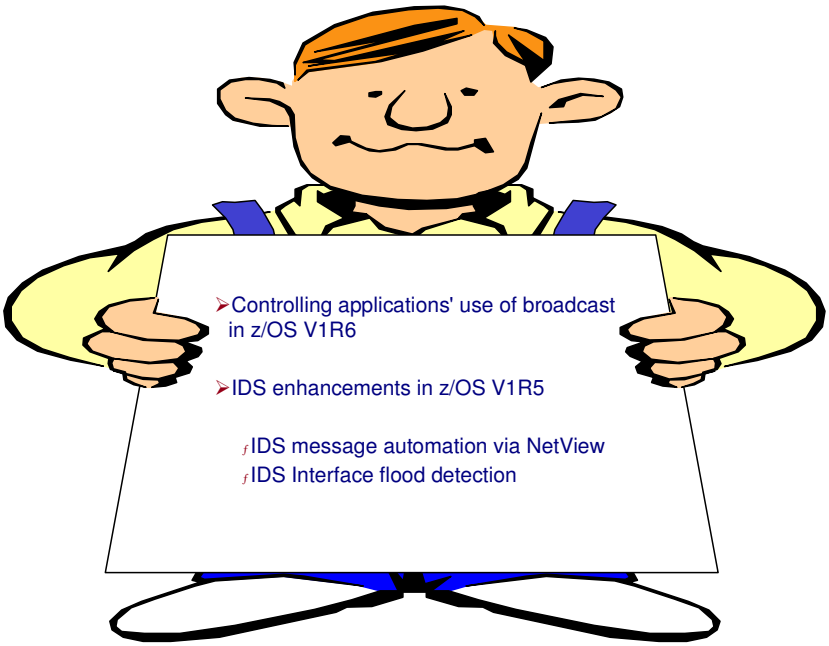
Communications Server z/OS V1R5 and V1R6 Technical Update

zOS CS Security: Controlling Broadcast Applications Intrusion Detection Services

Topics



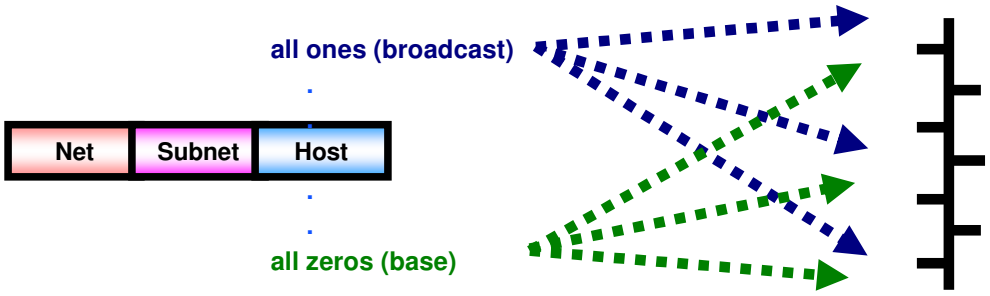
- Controlling applications' use of broadcast in z/OS V1R6
- IDS enhancements in z/OS V1R5
  - IDS message automation via NetView
  - IDS Interface flood detection

SAF control of broadcast applications

Background Information

- IPv4 IP addresses are divided into a network portion, an optional subnetwork portion and a host portion. Normally, UDP and RAW datagrams are delivered to the single peer system identified by the destination IP address.

- However, when the destination address is a (sub)network base address (host portion is all zeros) or a (sub)network broadcast address (host portion is all ones), the datagram is delivered to every peer system in that (sub)network.



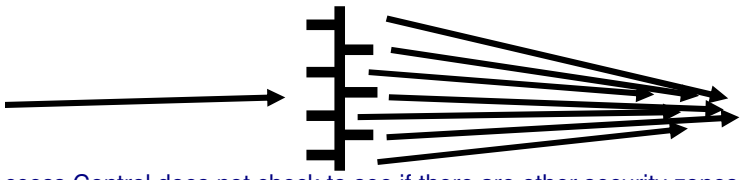
Background Information

- There are several network management and data sharing applications that use broadcast.
  - Routing protocols
  - Remote procedure calls
  - Dynamic IP address assignment

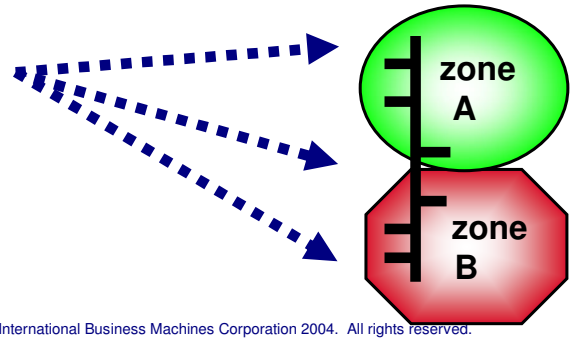
- Socket semantics require that an application set the SO_BROADCAST option on before attempting to send a datagram to a base or broadcast address. This protects the application from accidentally sending a datagram to many systems.

Potential Misuse of Broadcast Addresses

- Some network flood attack programs use broadcast addresses as traffic amplifiers.



Network Access Control does not check to see if there are other security zones defined within the scope of a destination (sub)network address and whether the user is permitted to send a datagram to all of these security zones.



Socket Option Access Control

- Socket Option Access Control is designed to give system administrators the ability to assign permission for z/OS users to set selected socket options using a SAF security server.
- Access control is provided for the SOL_SOCKET level, SO_BROADCAST option in V1R6.
- The socket option to be protected is represented by the resource name:
  EZB.SOCKOPT.*sysname.tcpname*.SO_BROADCAST.

  RDEFINE SERVAUTH EZB.SOCKOPT.*.*.SO_BROADCAST UACC(NONE)

- When this profile is defined, users of any program setting this option will require READ permission. Access to the option is also allowed if the security server indicates there is no profile covering this resource.

  PERMIT EZB.SOCKOPT.*.*.SO_BROADCAST CLASS(SERVAUTH) -
  ACCESS(READ) ID(OMPROUT SNTPD)

- Multilevel security environment considerations:
  - This profile is required. Access to the option is denied if the security server indicates there is no profile covering this resource.
  - All SERVAUTH class profiles must have security labels. This profile may safely use the SYSNONE security label.

  RALTER SERVAUTH EZB.SOCKOPT.*.*.SO_BROADCAST SECLABEL(SYSNONE)

Socket Option Access Control

- Conditional access lists are supported for profiles covering socket option access control resources.

  PERMIT EZB.SOCKOPT.*.*.SO_BROADCAST CLASS(SERVAUTH) -
  ACCESS(READ) ID(*) WHEN(PROGRAM(ORPCINFO))

  PERMIT EZB.SOCKOPT.*.*.SO_BROADCAST CLASS(SERVAUTH) -
  ACCESS(READ) ID(NETADMIN) WHEN(PROGRAM(ORPCINFO))

- TCP/IP programs known to set the SO_BROADCAST socket option include:
  - omproute,
  - orouted,
  - dhcp,
  - binlsd,
  - and sntpd, when invoked with the -b option.

- Additionally, any programs that use the clnt_broadcast() service in the SUN rpc libraries, or the send_pkt(sock, pkt, addr, broadcast) service in the NCS rpc library with the broadcast parameter set, require permission to the SO_BROADCAST socket option. The following TCP/IP programs use RPC services that require permission to broadcast:
  - rpcinfo, when invoked with the -b option;
  - orpcinfo, when invoked with the -b option

Socket Option Access Control

- To use program names in conditional access lists, the program must be loaded into a controlled environment from a program controlled dataset. TCP/IP applications are distributed in the <tcpip>.SEZALOAD load library. To program control this dataset you must add it to the ** profile in the PROGRAM class:

  RALTER PROGRAM ** ADDMEMBER('TCPIP.SEZALOAD'//NOPADCHK)

- The program name listed in the conditional access list must be the name the program is invoked by. Most TCP/IP applications are invoked by an ALIAS name rather than the MODULE name. The following table lists TCP/IP applications that send broadcast datagrams:

LOAD MODULE	ALIAS	NOTES
EZAORRTE	OMPROUTE	
EZBROUTD	OROUTED	
EZATDHSD	DHCPSD	
EZATDLSD	BINLSD	
EZASNTPD	SNTPD	
EZARPCIN	RPCINFO	TSO: RPCINFO -b ...
EZARORNP	ORPCINFO	USS: [o]rpcinfo -b ...

Intrusion Detection Services

IP-based security technology overview and introduction

e-business

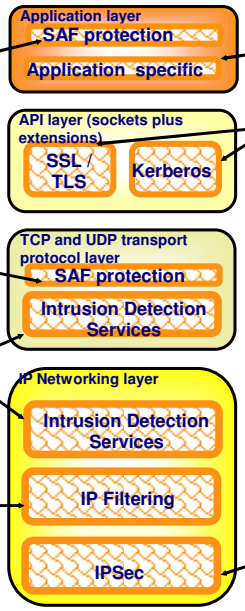
Protect the system

z/OS CS TCP/IP applications use SAF to authenticate users and prevent unauthorized access to datasets, files, and SERVAUTH protected resources..

The SAF SERVAUTH class is used to prevent unauthorized user access to TCP/IP resources (stack, ports, networks)

Intrusion detection services protect against attacks of various types on the system's legitimate (open) services. IDS protection is provided at both the IP and transport layers.

IP filtering blocks out all IP traffic that this systems doesn't specifically permit.



Protect data in the network

Examples of application protocols with built-in security extensions are SNMPv3, DNS, and OSPF.

Both Kerberos and SSL/TLS are located as extensions to the sockets APIs and applications have to be modified to make use of these security functions. Both SSL/TLS and Kerberos are connection-based and only applicable to TCP (stream sockets) applications, not UDP.

IPSec resides at the networking layer and is transparent to upper-layer protocols, including both transport layer protocol and application protocol.

Intrusion threat - protecting against attacks on your system or your legitimate (open) services

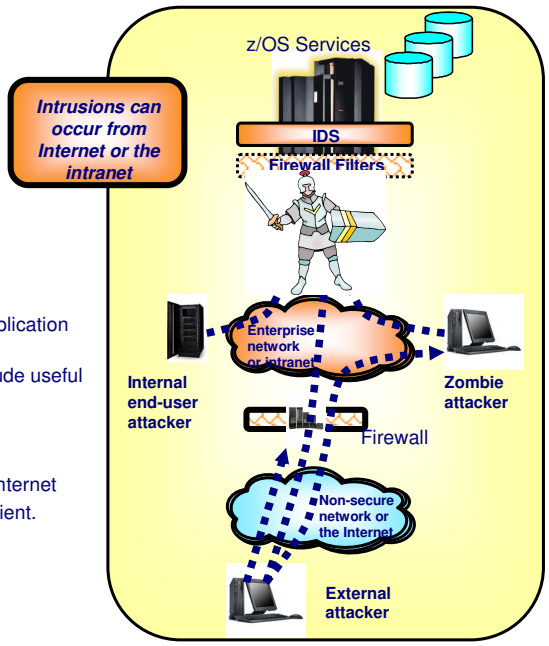
e-business

What is an intrusion?

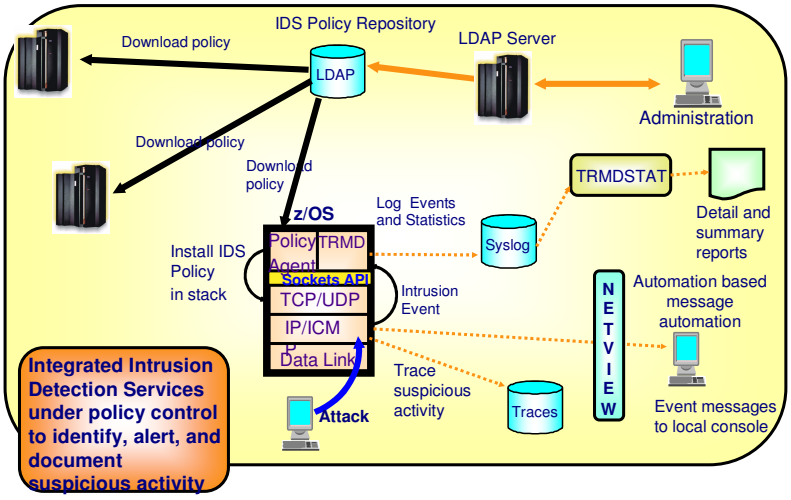
- Information Gathering
  - Network and system topology
  - Data location and contents
- Eavesdropping/Impersonation/Theft
  - On the network/on the host
  - Base for further attacks on others
    - Amplifiers
    - Robot or zombie
- Denial of Service
  - Attack on availability
    - Single Packet attacks - exploits system or application vulnerability
    - Multi-Packet attacks - floods systems to exclude useful work

Attacks can occur from Internet or intranet

- Firewall can provide some level of protection from Internet
- Perimeter Security Strategy *alone* may not be sufficient.
  - Considerations:
    - Access permitted from Internet
    - Trust of intranet



Intrusion Detection Services Overview



Events detected
- –Scans, Attacks Against Stack, Flooding (both TCP and UDP)

Defensive methods
- –Packet discard, limit connections

Reporting
- –Logging, event messages to local console, IDS packet trace
- –Notifications to NetView

Security Policy Stored in LDAP

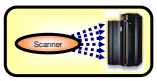
z/OS IDS broadens intrusion detection coverage:
- –Ability to evaluate inbound encrypted data - IDS applied after decryption on the target system
- –Avoids overhead of per packet evaluation against table of known attacks - IDS policy checked after attack detected
- –Detects statistical anomalies real-time - target system has stateful data / internal threshholds unavailable to external IDSs
- –Policy can control prevention methods on the target, such as connection limiting and packet discard

IDS Event types

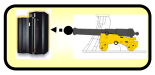
e-business

- Scan detection and reporting
  - Intent of scanning is to map the target of the attack (Subnet structure, addresses, masks, addresses in-use, system type, op-sys, application ports available, release levels)
    - TCP port scans
    - UDP port scans
    - ICMP scans
      - Sensitivity levels for all scans can be adjusted to control number of false positives recorded.
- Attack detection, reporting, and prevention
  - Intent is to crash or hang the system (Single or multiple packet)
    - Malformed packet events
    - Inbound fragment restrictions
    - IP option restrictions
    - IP protocol restrictions
    - ICMP redirect restrictions
    - Flooding events (SYN flood detections, physical interface flood detection added in z/OS V1R5)
    - Outbound raw restrictions
    - UDP perpetual echo
- Traffic regulation for TCP connections and UDP receive queues
  - Could be intended to flood system OR could be an unexpected peak in valid requests
    - UDP backlog management by port
      - Packets discard
    - TCP total connection and source percentage management by port
      - Connection limiting

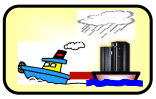
Scan



Attack



Flooding



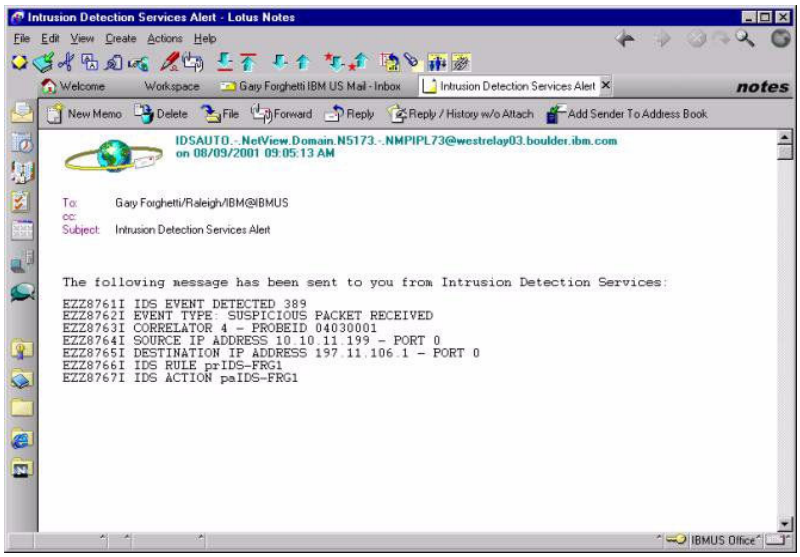
IDS actions and message automation

- Options
  - Event logging
    - –Syslogd - Number of events per attack subtype recorded in a five minute interval is limited
    - –Local Console - Recording suppression provided if quantity of IDS console messages reach policy-specified thresholds
  - Statistics
    - –Syslogd - Normal, Exception
  - IDS packet trace
    - –Activated after attack detected
      - •Number of packets traced for multi-packet events are limited
      - •Amount of data trace is configurable (header, full, byte count)
- All IDS events recorded in syslog and console messages, and packet trace records have probeid and correlator
  - Probeid identifies the specific event detected
  - Correlator allows events to be matched with corresponding packet trace records

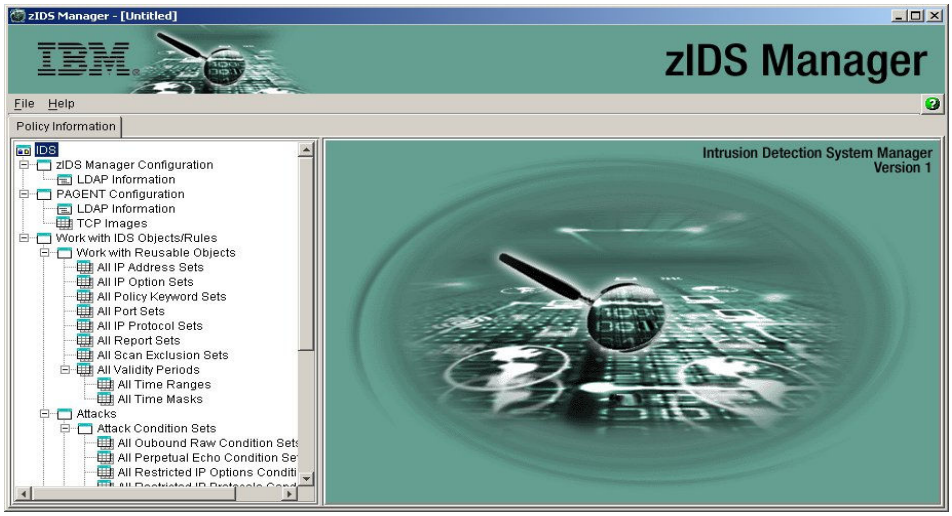
- Console message can drive message automation
  - MPF message suppression can suppress message output to system console
  - Example automation actions:
    - –Route message to NetView console(s)
    - –email notification to security administrator
    - –Run trmdstat and attach output to email
  - Selectors
    - NetView clists: http://www.ibm.com/support/all_download_drivers.html Search: idsauto be ID

IDSAUTO in NetView

Example of an email sent by the NetView IDSAUTO solution as the result of an Intrusion event being detected by the IDS component of z/OS TCP/IP:



Defining IDS Policy for z/OS - using Windows or Linux with the zIDS Manager



zIDSManager "as is" Web Tool: to implement LDAP policies.
Available at: http://www-3.ibm.com/software/network/commserver/downloads
Needs Java runtime 1.3 or 1.4.
Available at: http://java.sun.com/j2se/1.4.1/download.html

Enhanced flood detection

- Prior to z/OS V1R5 individual events (for example, malformed packet, requests to an unbound port, queue full conditions, etc.) are detected and handled.
- IDS can provide notification about the individual instances but the discarded packets may be a symptom of a larger problem such as a flood.
- The only flood type currently detected by IDS Attack support is a SYN flood.
- z/OS V1R5 adds interface flood detection support as part of IDS flood detection:
  - A high percentage of discarded packets on a physical interface may indicate the interface is under attack.
  - Using the information already detected by IDS, track the discard rate by physical interface to determine if there is a potential attack
  - Notify the customer that a possible interface flood condition is occurring if the discard rate exceeds a specified limit.
  - Provide information to help determine the potential cause of the interface flood
  - Allow the customer to specify policy criteria used to detect an interface flood
  - Provide detection support without a large performance impact

IDS traffic regulation may also detect some flood situations but it's prime function to help manage the utilization of resources.

Interface flood detection - notes

NOTES

- The term 'discarded packets' used here includes not only packets discarded because they are malformed or destined to an unbound port, it also includes packets not processed because queues were full or IDS traffic regulation limits were exceeded, etc.
- The Physical Interface Flood detection support attempts to identify flooding conditions by looking at the percentage of discards occurring on an interface and then provide information to help identify the cause of the flood -- for example, the previous hop that is involved . This support provides the detection of a potential flood event, it does not add any defensive action beyond what is currently provided by the stack.
- By detecting flooding on an interface bases, we can start to understand the source of the problem (especially if only a single interface is affected) and start a trace back to the origin of the problem. If the flood is only occurring on a single interface, the customer may be able to vary that interface off-line.
- This support is activated through the IDS ATTACK Flood policy that currently exists. The IDS ATTACK policy can only be specified in LDAP.

How it works

- Policy related to interface flood detection
  - part of Attack Flood support
  - 2 new actions attributes provided
    - ibm-idsIfcFloodMinDiscard (default 1000)
    - ibm-idsIfcFloodPercentage (default 10)
- For each interface, counts are tracked for
  - The number of inbound packets that arrived over the physical interface
  - The number of these packets that are discarded
- When the specified number of discards (ibm-idsIfcFloodMinDiscard) is hit:
  - If it took longer than 1 minute to accumulate the discards, doesn't qualify as a flood condition
  - If the discards occurred in a minute or less:
    - the discard rate is calculated for the interval :
      - # discards during the interval / # inbound packets for the interval
    - if the discard rate equals or exceeds the specified threshold, an interface flood condition exists
- Once an interface flood is detected, this data is collected and evaluated for the interface at 1 minute intervals. The interface flood is considered ended if the discards for a subsequent interval:
  - Fall below the minimum discard value OR
  - Discard rate for the interval is less than or equal to 1/2 of the specified threshold

The minimum discard count (ibm-idsIfcFloodMinDiscard) is the minimum number of discards that must occur in a 1 minute interval before considering a condition an interface flood.

The minimum discard count is checked to avoid false flood detections in cases where there is low inbound activity on an interface. For example, if only 100 packets were received during the last 1 minute interval and 25 were discarded, the rate of discard would appear very high (i.e. 25%). However, this would not normally be considered a flood. Having the capability to specify the minimum number of discards that must occur in the interval helps to prevent false flood detections and limits the number of times that IDS needs to check for an interface flood condition .

How it works - notes

NOTES

- The minimum discard count (ibm-idsIfcFloodMinDiscard) is the minimum number of discards that must occur in a 1 minute interval before considering a condition an interface flood.

- The minimum discard count is checked to avoid false flood detections in cases where there is low inbound activity on an interface. For example, if only 100 packets were received during the last 1 minute interval and 25 were discarded, the rate of discard would appear very high (i.e. 25%). However, this would not normally be considered a flood. Having the capability to specify the minimum number of discards that must occur in the interval helps to prevent false flood detections and limits the number of times that IDS needs to check for an interface flood condition .

How it works - example

- Assume the IDS flood policy specifies:
  - ibm-idsIfcFloodMinDiscard:2000
  - ibm-idsIfcFloodPercentage:10
- The activity for interface X is as shown in the table below:

time

time interval	inbound cnt	discard cnt	discard rate	notes
> 1 min	13,000	2000	N/A	took longer than a minute to see the minimum discard count, so not a flood and discard rate not calculated
< 1 min	30,000	2000	6.6%	not a flood, rate <10%
< 1 min	20,000	2000	10%	***interface flood start detected.*** Run 1 minute timer until flood end detected
1 min	40,000	3000	7.5%	flood condition still exists, reset 1 minute timer.
1 min	50,000	2500	5%	***Interface flood end detected.*** Discard rate <= half of policy specified rate.

If the number of discards fell below 2000 for the 4th or 5th interval shown on the chart, this would have also caused an interface flood end to be detected.

Information provided about an interface flood

- When an interface flood is detected:
  - An 'EZZ8761I IDS EVENT DETECTED' group message is written to the console.
  - An 'EZZ8654I TRMD ATTACK Interface flood start' message is written to syslogd.

  These messages identify the interface experiencing the potential flood.
  The syslogd record will also show the discard count and discard rate that triggered the flood detection.

- Once a flood is detected, information is gathered about subsequent discarded packets on the interface:
  - discard category
  - protocol
  - source MAC of prior hop if LCS or OSA QDIO (with microcode support)

  This information is shown in the records written to the IDS trace and will also be used to determine the most frequent discard category, protocol and source MAC seen during the interface flood condition.

- If the interface flood continues for more than 5 minutes, an 'EZZ8656I TRMD ATTACK Interface flood continues' message is written to syslogd every 5 minutes.

- When the interface flood ends:
  - An 'EZZ8655I TRMD ATTACK Interface flood end' is written to syslogd
  - An 'EZZ8761I IDS EVENT DETECTED' group message is written to the console.

Note: Console messages and syslogd messages are only written if requested by the IDS Attack Flood policy.

The source MAC address of the prior hop is available for:

LCS devices

OSA QDIO with microcode level that supports providing the source MAC address

If the source MAC address is available, it may help in tracking back one step closer to the source of the attack.

IDS trace records may also include the source IP address from the outer IPSec header if the packet had been received as IPsec tunnel mode. In this case, the source IP address could be a gateway or firewall that could allow someone to trace this closer to the source

Interface flooding - notes

NOTES

- Note: Console messages and syslogd messages are only written if requested by specifying the ibm-idsNotification attribute in the IDS Attack Flood policy (additional details in IDS overview section)..
- The source MAC address of the prior hop is available for:
  - LCS devices
  - OSA QDIO with microcode level that supports providing the source MAC address
- If the source MAC address is available, it may help in tracking back one step closer to the source of the attack.
- IDS trace records may also include the source IP address from the outer IPSec header if the packet had been received as IPsec tunnel mode. In this case, the source IP address could be a gateway or firewall that could allow someone to trace this closer to the source than even the prior hop.
- Flood data is not tracked by source IP address since a malicious attacker will usually spoof the source address. Tracking a large number of spoofed IP address could consume a large amount of storage and magnify the effect of an attack.

Information provided about an interface flood (continued)

- The "Interface flood continues" and the "Interface flood end" messages provide the following information intended to help determine the type and source of the flood. The information is cumulative from the time the interface flood started until the time the record was generated.
  - Interface name and an IP address associated with the interface
  - Correlator
  - Probe ID
  - Number of discards
  - Overall discard rate
  - Duration of the flood
  - Most frequently seen:
    - discard category/percent
    - protocol/percent
    - prior hop source MAC address (where available)/percent
  - If the prior hop source MAC address is available, the most frequently seen discard category and protocol for the above source MAC is also supplied
  - The source IP address from the last discarded packet and number of times since another source IP address was seen.

Collection of the detail information needed to determine the 'Most frequently seen' data does not start until and interface flood is actually detected. The counts do not include the initial discards that contributed to the interface flood detection.

Along with the most frequent discard category, protocol and source MAC address, the percent of times the value was seen in the discards is also shown. For example, if total discard count for the flood was 10,000 and 7000 of these discards were UDP packets, the most frequent protocol would be UDP and accounted for 70% of the discards.

Because the source IP address can be easily spoof by a malicious attacker, tracking source IP information could consume large amounts of storage and create storage shortages. Therefore, only information on the

Interface flood discard categories

NOTES

- Storage - storage could not be obtained to process the packet. Note: storage shortages can indicate a problem in the system other than an inbound packet flood.
- CheckSum - packet had checksum error
- Malform - malformed packet
- Dest - destination not found. For example, the port is not active or is reserved, the matching socket is not available, no listeners for the RAW protocol.
- Firewall - packet rejected by firewall
- MedHdr - bad media header
- Forward - packet is not for us but could not be forwarded. For example, forwarding prevented because the header is bad, IPCONFIG NODATAGRAMFWD specified or ASSORTEDPARMS NOFWD specified.
- QOSPol - packet dropped due to QOS policy
- IDSPol - packet dropped due to IDS policy
- NETACC - packet dropped due to NetAccess or MLS checks
- OtherPol - packet dropped due to other configuration policy
- Queue - queue limit (other than those specified by IDS) prevented queuing the packet for processing. Possible queues include the SYN queue, the reassembly queue, the UDP or RAW receive queues.
- OtherSyn - SYN problems other than SYN queue full
- State - state mismatch
- Misc - Miscellaneous reasons not listed above. For example, TCP packet outside of TCP window, duplicate fragments found during packet reassembly.

Note: the discard count used by interface flood support includes 'discard' categories beyond what is shown in the netstat devlink stats. For example, destination not found, packet dropped due to QOS, IDS, MLS or firewall policy,

The RAW queues poses a unique challenge since a RAW protocol may have multiple listeners. Obviously if all listener queues are full, this could indicate a flood situation and the packet will be included in the discard count. If a single queue is full, it may be that one of the listening application is slow and counting this as a 'discard' could result in false flood conditions. But only factoring in situations where all listener queues are full may understate the problem, since new listeners may have just been added and still have a low queue. Ping is a good example of this problem. If response times are slowing down, as may occur during a flood, more and more users may issue pings to check status. If the flood was a result of a ICMP reply flood, all queues may never be full due to new pings being issued and the 'problem' icmp replies would not be counted as a discard.

The approach taken to address this is that if there are

RAW listeners - notes

NOTES

- The RAW queues poses a unique challenge since a RAW protocol may have multiple listeners. Obviously if all listener queues are full, this could indicate a flood situation and the packet will be included in the discard count. If a single queue is full, it may be that one of the listening application is slow and counting this as a 'discard' could result in false flood conditions. But only factoring in situations where all listener queues are full may understate the problem, since new listeners may have just been added and still have a low queue. Ping is a good example of this problem. If response times are slowing down, as may occur during a flood, more and more users may issue pings to check status. If the flood was a result of a ICMP reply flood, all queues may never be full due to new pings being issued and the 'problem' icmp replies would not be counted as a discard.

- The approach taken to address this is that if there are more than half the listener queues are full, this will be counted as a discard for interface flood detection purposes.

Example of Interface flood end syslogd record


```
EZZ8655I TRMD ATTACK Interface flood end: timestamp,
ifcname=ifcname,dipaddr=dipaddr,correlator=correlator,duration=duration,
discardcnt=discardcnt,discardp=discardp,mfproto=mfproto,mfprotop=mfprotop,
mfcat=mfcat,mfcatp=mfcatp,mfsrcmac=mfsrcmac,mfsrcmacp=mfsrcmacp,
smmfproto=smmfproto,smmfprotop=smmfprotop,smmfcat=smmfcat,smmfcatp=smmfcatp,
lastsip=lastsip,sipcnt=sipcnt,
probeid=probeid,sensorhostname=sensorhostname
```

"Translation" hints:

- 'p' suffix -> 'percentage' for example: discardp -> discard percentage
- 'mf' prefix -> 'most frequent': mfproto -> most frequent protocol
- 'smmf' -> 'source MAC most frequent': smmfproto -> for the reported source MAC, the most frequent protocol

Or better yet, use trmdstat to format the record

Note: the interface flood continue record (EZZ8656I) contains the same information as the interface flood end syslogd record. Information in both of these records is cumulative from the time the interface flood started until the time the record was generated.

The 'most frequent' data is tracked from the time the interface flood is detected until the interface flood ends. The counts do not include the initial discards that contributed to the interface flood detection.

trmdstat formatted example

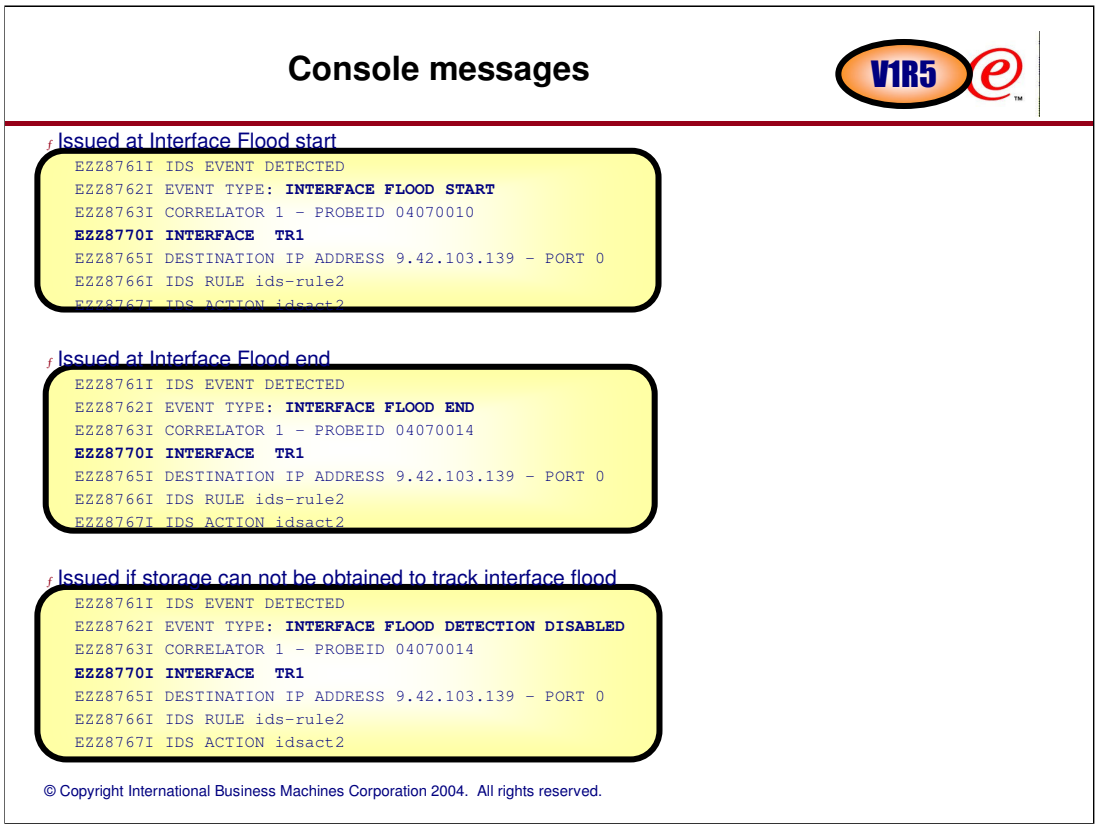
- Interface flood data is also formatted by trmdstat.

```
trmdstat -FD  /tmp/syslog.miscids
trmdstat for z/OS CS V1R5          Wed Feb 26 12:20:07 2003

Stack Name           : ALL
Log Time Interval    : Dec 10 13:54:13  - Dec 10 15:48:13
Stack Time Interval  : Dec 10 13:54:04  - Dec 10 15:47:49
TRM Records Scanned  : 162
Port Range           : ALL

                                        SYN FLOOD  Events

   Date and Time        IP Address      Port  Type  SYNsRecvd   FirstAck  SYNsDiscd  SYNsTimeO   Duration  Correlator
---------------------- --------------- ------ ---  ---------- ---------- ---------- ---------- ---------- ----------
12/10/2002 15:46:29.18 0.0.0.0             23   E                                                            23

                                     Interface FLOOD  Events

   Date and Time/          Interface       Type  Duration    Discard   Correlator/ ----------------Most Frequent----------------------
   Last  Last Source IP/                                     Count/    ProbeID     -----Overall-----   -------Source MAC Data-------
   Count   Dest Address                                      Percent               Proto/  Category/      SrcMAC/  Proto/  Category/
                                                                                   Percent Percent        Percent  Percent Percent

12/10/2002 13:54:04.68     MYHOME2          E                  100         1
           9.42.105.71                                         100  04070010
           9.42.105.113
12/10/2002 13:59:09.77     MYHOME2          C        303       3501         1       55      Dest           N/A       0   unknown
    3402   9.42.105.71                                         100  04070011       97       97             0       0         0
           9.42.105.113
12/10/2002 14:01:12.30     MYHOME2          X        425       3901         1       55      Dest           N/A       0   unknown
    3802   9.42.105.71                                         100  04070014       97       97             0       0         0
           9.42.105.113
```


The interface flood report is 132 characters wide. If viewing the report online, make sure a screen width of at least 132 is used.

Policy defaults and changes

- If Attack flood policy is specified, SYN flood and interface flood detection support is provided.
- If a customer is currently running with IDS flood policy, no policy change is needed to get the interface flood detection support using the interface flood default action attributes.
- New LDAP class object and action attributes can be specified if the defaults are not wanted.
  - new object class: ibm-idsFloodAttackActionsAuxClass
  - new action attributes:
    - ibm-idsIfcFloodMinDiscard - minimum number of discards that must occur in a 1 minute interval. Default 1000.
    - ibm-idsIfcFloodPercentage - minimum discard rate. Default is 10%.



The EZZ8770I message that identifies the interface has been added to the EZZ8761I message group when the IDS Event is related to an interface flood event.

If storage can not be obtained to track the interface flood, interface flood detection is disabled for the specified interface. This problem can occur only when we tried to initiate interface flood detection for the interface (either at interfaces start processing or when a new flood policy is installed). Once storage shortages have been relieved, interface flood detection can be restarted by stopping and restarting the interface or by removing the flood policy and then restarting it.

Console messages - notes

NOTES

- The EZZ8770I message that identifies the interface has been added to the EZZ8761I message group when the IDS Event is related to an interface flood event.

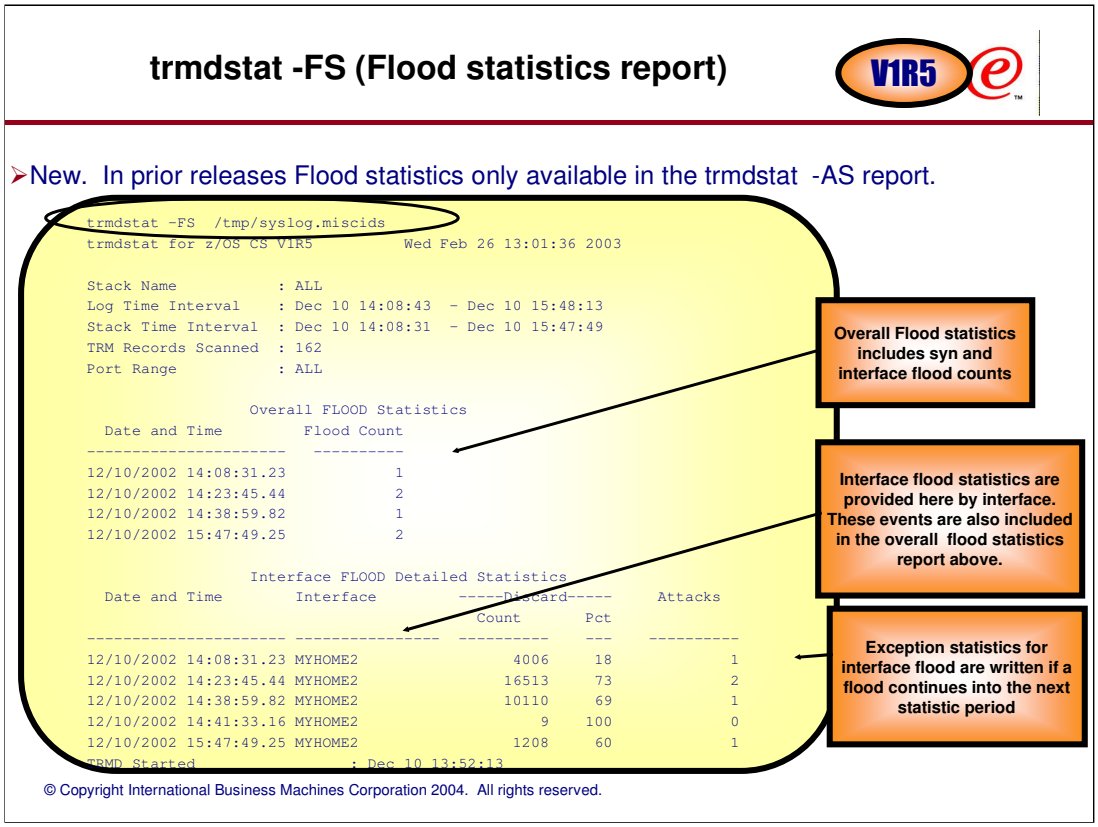
- If storage can not be obtained to track the interface flood, interface flood detection is disabled for the specified interface. This problem can occur only when we tried to initiate interface flood detection for the interface (either at interfaces start processing or when a new flood policy is installed). Once storage shortages have been relieved, interface flood detection can be restarted by stopping and restarting the interface or by removing the flood policy and then restarting it.

trmdstat -I (IDS summary report)

The IDS summary report (trmdstat -I) extended to report interface flood totals

```
trmdstat -I  /tmp/syslog.info
trmdstat for z/OS CS V1R5          Tue Feb 25 18:13:25 2003

Stack Name           : ALL
Log Time Interval    : Dec 17 14:24:18  - Feb 25 20:03:31
Stack Time Interval  : Dec 17 14:23:56  - Feb 25 20:03:27
TRM Records Scanned  : 69
Port Range           : ALL

TCP - Traffic Regulation
------------------------------------------------
Connections would have been refused :          0
Connections refused                 :          0

......

ATTACK Detection
------------------------------------------------
Packet would have been discarded    :          0
Packet discarded                    :          0

FLOOD Detection
------------------------------------------------
Accept queue expanded               :          0
SYN flood start                     :          0
SYN flood end                       :          0
Interface flood start               :          4
Interface flood end                 :          4
```

trmdstat -FS (Flood statistics report)



For consistency with other flood report options, the -FS was added.

The overall statistics report counts include both syn flood and interface flood events.

Interface flood statistics are also available by interface. The interface flood statistics will be shown after the overall flood statistic. Note that the counts shown in the interface flood statistics are also included in the overall flood.

In this sample report, the last overall flood statistic record included a syn flood event. Therefore, one more flood event is shown in the overall flood statistics than in the interface flood statistics report.

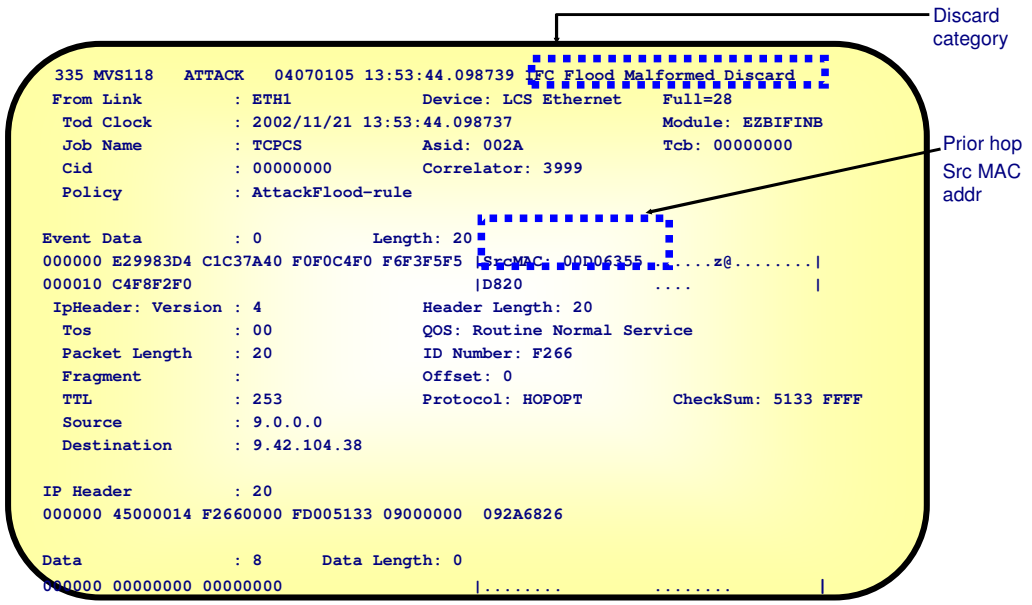
Normally exception stats are written for a flood only if a flood start is detected in the statistics interval. For an interface flood, an interface flood exception stats

trmdstat -FS - notes

NOTES

- For consistency with other flood report options, the -FS was added.
  - The overall statistics report counts include both syn flood and interface flood events.
  - Interface flood statistics are also available by interface. The interface flood statistics will be shown after the overall flood statistic. Note that the counts shown in the interface flood statistics are also included in the overall flood.
- In this sample report, the last overall flood statistic record included a syn flood event. Therefore, one more flood event is shown in the overall flood statistics than in the interface flood statistics report.
- Normally exception stats are written for a flood only if a flood start is detected in the statistics interval. For an interface flood, an interface flood exception stats record for the flooding interface will also be written if an interface flood is active when a new statistics interval starts.
- The trmdstat -AS report will still report overall flood statistics for compatibility with prior releases.

IDS trace sample



Event data is uses to show the prior hop source ip address , if available. The event data fields is a general usage field that is displayed in hex, EBCDIC and ASCII. The source MAC information is most readable in the EBCDIC format.

IDS trace Event data can also include the source IP address from the outer IPSec header if the packet had been received as IPsec tunnel mode. In this case, the source IP address could be a gateway or firewall that could allow someone to trace this closer to the source than even the prior hop.

IDS trace sample - notes

NOTES

- Event data is uses to show the prior hop source ip address , if available. The event data fields is a general usage field that is displayed in hex, EBCDIC and ASCII. The source MAC information is most readable in the EBCDIC format.
- IDS trace Event data can also include the source IP address from the outer IPSec header if the packet had been received as IPsec tunnel mode. In this case, the source IP address could be a gateway or firewall that could allow someone to trace this closer to the source than even the prior hop.

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

© Copyright International Business Machines Corporation 2004. All rights reserved.