



Yessong Johng
Jim Coon
Craig Jacquez

IBM i5/OS Intrusion Detection System

Introduction

The Intrusion Detection System (IDS), introduced in IBM® i5/OS®, is a system that notifies you of attempts to hack into, disrupt, or deny service to the system. Prior to IDS, the i5/OS took some protective measures against the types of intrusions described here. However, with the new IDS support, the i5/OS system can now *tell* you about the intrusions.

This IBM Redpaper describes the following types of intrusions on the i5/OS system that are caught, audited, and, in many cases, discarded—before they become a threat:

- ▶ “Attacks” on page 2
 - “IP fragments” on page 2
 - “Malformed packets” on page 3
 - “SYN floods” on page 3
 - “Internet Control Message Protocol (ICMP) redirect messages” on page 3
 - “Perpetual echo” on page 3
 - “Restricted IP options” on page 4
 - “Restricted IP protocols” on page 4
- ▶ “Scans” on page 4
- ▶ “Traffic regulation anomalies for TCP and UDP” on page 5

Throughout this Redpaper, the term *hacker* is used to describe computer attackers. Occasionally, the term *intruder* or *perpetrator* is also used. Whatever the terminology, those described by such terms may be amateurs who are just playing around; intellectual types who may “hack” just for the challenge (or because they are disgruntled employees); or they may be professionals.

The Internet is a vast playground that can attract unsophisticated hackers who try to disrupt or deny service to both targeted and random IP addresses. Viruses, floods, ping-of-death attempts, “Smurf” attacks, User Datagram Protocol (UDP) “fraggle” attacks and so on, are designed just for the purposes of disruption and Denial of Service (DoS). But these types of attacks are commonplace by now, and are on the decline.

However, there are also types of sophisticated hackers: the ethical variety and the malicious criminal. The “ethical hacker” looks for vulnerabilities in a company’s security defenses and

suggests how to plug the holes. In contrast, the “malicious hacker” wants to exploit those vulnerabilities and steal information in such a way that the victim may never be aware of the infiltration. The malicious hacker’s intent is to “own” your system, gaining access using stealthy scanning techniques and then using trojan horses or root kits to wreak havoc or steal information.

It is interesting to note where these two types of hackers have crossed paths in the past. Sometimes the ethical hacker publicly exposes a vulnerability before the provider of the software has had a chance to come up with a fix. In some cases, this exposure has been intentional, either for publicity or to precipitate a quick solution to the problem. Whatever the reason, this practice opens a window of opportunity for the malicious hacker. Typically, during such a window of opportunity, a virus might be launched, or— worse yet—a company’s secrets might be compromised.

The i5/OS system has long detected attempts to disrupt and deny service. With IDS, there is now notification that these potential attacks have taken place. Additionally, other types of intrusions are now monitored and handled. Some events may be just legitimate attempts to connect to the system. It is up to a systems administrator or the person monitoring the security audit journal to decide whether attempts are legitimate or otherwise.

This is just the beginning of intrusion detection on the i5/OS system. This paper describes the Intrusion Detection System currently offered on the i5/OS system.

Intrusion types

This section describes the most common types of intrusion: attacks, scans, and traffic regulation anomalies.

Attacks

Attacks may or may not be malicious. As mentioned, IDS is notified of various events and some of these notifications may be false alarms. A description of these attacks is offered here.

IP fragments

Datagrams that are too big to be transmitted over a network are broken down into fragments. The fragmentation process involves tacking on an IP header to each piece of the fragmented datagram, setting the “More Fragments” (MF) flag, and providing the offset of where this fragment lies within the original datagram. This information, along with a fragment identification number and the length of data in the fragment itself, is used by the target in reassembling the original datagram.

On the i5/OS system, fragments that IDS is notified about fall into three categories:

- ▶ Fragments that, when reassembled, would be greater than 64 K in size and, therefore, too large (see “Malformed packets”).
- ▶ Fragments that are less than 576 bytes in length.
- ▶ Fragments with an offset of less than 256 bytes. (This does not mean the fragment itself is less than 576 bytes. This may be an attempt to overlay data in the first fragment.)

In the case of a fragment that is too large, the intent may be to crash or hang a system. In the other two cases, where the fragments are smaller, the intent may be a malicious attempt to slip through a firewall; but then again, it could just be a normal case of packet retransmission. At any rate, a packet is not forwarded to the next layer until it is completely reassembled.

Malformed packets

Malformed packets may be designed to cause a system to crash or hang. They are detected by the TCP/IP stack in the following instances:

- ▶ When a checksum is wrong
- ▶ For a destination port of 0
- ▶ When a packet size, including fragments, is greater than 64 K
- ▶ When both SYN and FIN are set (indicating that a client is attempting to establish a connection, but has no more data to send)

The TCP/IP stack notifies IDS of these malformed packets and then, in most cases, discards them.

SYN floods

SYN floods are an attempt to tie up system resources and deny service. They occur when the TCP/IP three-part handshake does not complete. An attacker will initiate a connection attempt to a host (first part of the handshake: SYN) and provide a spoofed source address for the host to acknowledge (second part of the handshake: SYN/ACK), and then leave the host waiting on an acknowledgment (third part of the handshake: ACK) that will never come from the spoofed address. This ties up system resources.

On the i5/OS system, these incomplete connections are queued. After the queue limit is exceeded, the oldest incomplete connection attempts are dropped from the backlog one at a time. Each time one such connection is dropped, the TCP/IP stack notifies IDS of the possible flood situation.

Internet Control Message Protocol (ICMP) redirect messages

The Internet Control Message Protocol (ICMP) is used for sending out-of-band messages concerning network operations. There are many types of ICMP messages (see Request for Comment (RFC) 792).

An ICMP type 5 message is a “redirect” message. This message may be sent by a router to a host on the same subnet to indicate a more optimal route for packets sent by that host to some other target host in another network. The message will indicate that there is another gateway (that is, router) on the same subnet to which the packets may be sent and forwarded more efficiently. The original packet from the host which precipitated the ICMP redirect message from the router is forwarded anyway and the host does not have to honor the ICMP redirect message from the router. That is, it can choose to ignore ICMP redirect messages.

ICMP redirect messages can be used maliciously by a hacker for Man-in-the-Middle (MITM) attacks. In this type of an attack, the hacker, posing as a router, sends an ICMP redirect message to a host indicating that all future traffic must be directed his way as the more optimal route to the intended destination.

On the i5/OS system, the TCP/IP stack will notify IDS in the event of any ICMP redirect message whether the intent is valid or not, and whether or not the stack has been configured to ignore such messages.

Perpetual echo

The UDP “fraggle” attack is an annoying DoS attack involving UDP echo port 7. An attacker sends a UDP echo request to an IP broadcast address and provides a spoofed source address for all the targets to echo back responses. The spoofed source address, which is not the hacker’s address, becomes the victim of a potentially large amount of network traffic. If the source port is also port 7, then a “perpetual echo” results.

On the i5/OS system, any UDP request to destination port 7 is signaled by the TCP/IP stack to IDS as a possible perpetual echo attack. IDS then checks the source port to determine if the packet truly is a perpetual echo attempt.

Restricted IP options

An IP header may contain the Loose Source and Record Route (LSRR) option traditionally used by “traceroute” to map out a network’s topology. This option has been used by network administrators to determine why two hosts on a network are not communicating, or to specify alternate routes to relieve network congestion. A hacker may try to use LSRR to get through firewalls. By specifying LSRR and a hop that is reachable both by the hacker and private IP addresses, the hacker may reach what was previously thought to be a protected IP address.

On the i5/OS system, the TCP attribute IPSRCRTG (IP source routing) may be set to either On or Off through the **CHGTCPA** command. If IPSRCRTG is On, the packet is forwarded if it can be. If IPSRCRTG is Off and the system is not the destination of the packet, the packet is discarded. At any rate, any datagrams with IP options are signaled by the TCP/IP stack to IDS as possible suspicious events.

Restricted IP protocols

The IP protocols most often used are ICMP, TCP, UDP, and IGMP. Other protocols listed as part of the Internet Assigned Numbers Authority (IANA) may be used in an attempt to gain back door entry into a system.

On the i5/OS system, unrecognized IP protocols are signaled by the TCP/IP stack to IDS and handed off to Raw support. If there is no application listening on the Raw port, the packet is discarded. If, however, there is an application listening on the port, this could be the back door that the perpetrator is trying to access. The restricted IP protocol event logged in the security audit journal must alert a systems administrator to the possibility of such a rogue application.

Scans

Scanning entails sending a datagram to a system in order to determine what the listening ports are. After the open ports are discovered, the hacker tries to discover the weaknesses and gain access to the system.

On the i5/OS system, the TCP/IP stack signals IDS when connection attempts to non-listening ports are made (that is, “undemuxable SYNs”) or when a connection attempt is made in which the source address is the same as the target address (which could be a “spoofing” attempt).

Scans may be innocent attempts at connections to a server that may be down, making the resulting attention event a false alarm. However, they may be of interest if they come in at a very high rate or a very slow rate.

The high rate variety may be quick attempts at gathering information or attempts to deny service. They are more readily identifiable in the system logs than slow scans. The slow, “stealthy” variety are of more interest. A perpetrator may be seeking information about what ports to probe, what operating system is running, and so forth.

Hackers may scan from an Internet cafe, a library, and so on; in short, a “disposable” source. If tracked down in a log, the source IP address will no longer be valid. Also, by the time a suspicious IP address is noticed in a log, the hacker may have already gained access to the system, having sneaked in under the radar, and stolen valuable information.

Traffic regulation anomalies for TCP and UDP

Traffic regulation anomalies are events that cover TCP established connections or UDP transmissions. Their purpose is to single out an inordinate number of connections to a certain range of addresses/ports/applications. The UDP variety, being connectionless, is tougher to monitor than the TCP variety. These anomalies may indicate a DoS attack or be used to monitor certain connections and usage of certain applications on a system.

Setup for IDS notification on i5/OS

To enable IDS on the i5/OS system, here are a few steps that generally have to be performed only once:

1. Enable Quality of Service (QoS). The i5/OS system currently uses the QoS server to push its intrusion detection policies down to the network level. Figure 1 shows QoS being enabled through the **CHGTCPA** command.

```
Change TCP/IP Attributes (CHGTCPA)

Type choices, press Enter.

IP time to live (hop limit) . . . 64          1-255, *SAME, *DFT
IP QoS enablement . . . . . *YES        *SAME, *TOS, *YES, *NO
IP dead gateway detection:
  Enablement . . . . . *YES          *SAME, *DFT, *NO, *YES
  Interval . . . . . 2              1-60
ARP cache timeout . . . . . 15       1-1440, *SAME, *DFT
Enable ECN . . . . . *NO           *SAME, *YES, *NO
Network file cache:
  Enablement . . . . . *YES        *DFT, *CLEAR, *SAME, *YES, *NO
  Cached file timeout . . . . . 300  *NOMAX,30-604800 sec (1week)
  Cache size . . . . . 10          10-100000 megabytes
Log protocol errors . . . . . *NO   *SAME, *YES, *NO

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
```

Figure 1 Enable QoS by changing TCP attributes

2. Enable i5/OS system security auditing in order to have intrusions show up in the system security audit journal. This is done by changing the system value of QAUDCTL to allow for auditing (*AUDLVL), using the **WRKSYSVAL QAUDCTL** command and taking option **2**, as shown in Figure 2.

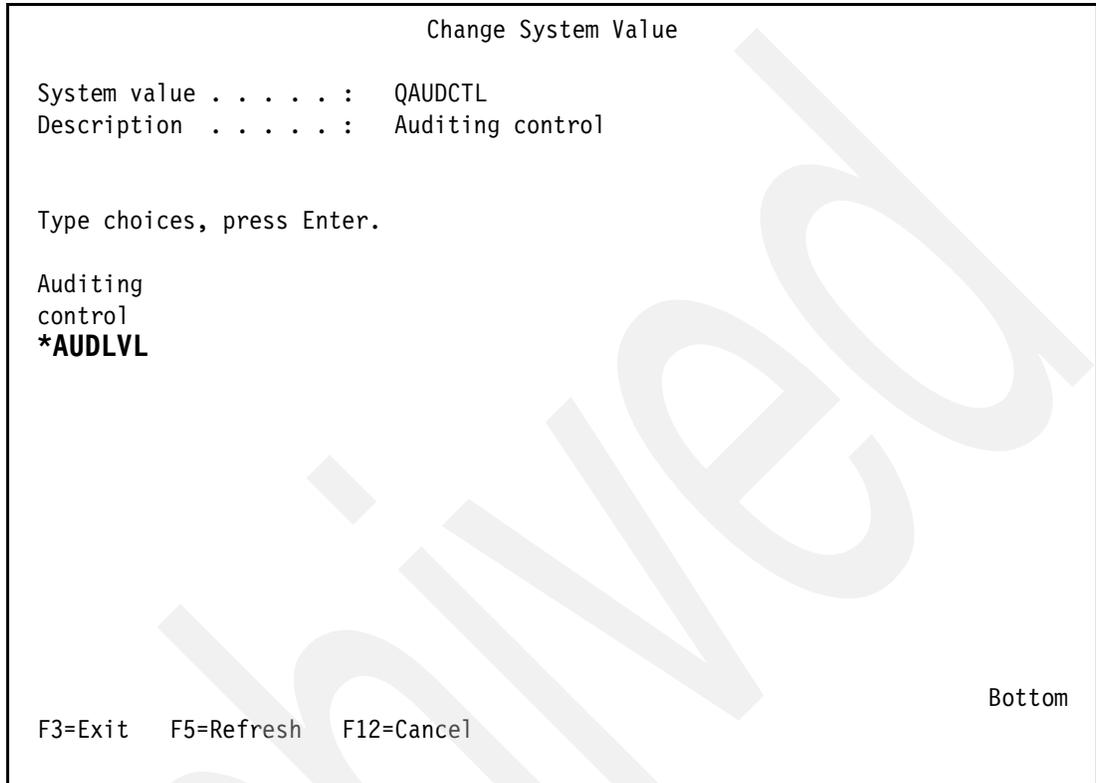


Figure 2 Turn auditing on (*AUDLVL)

3. Also, either one of the two system values—QAUDLVL or QAUDLVL2—should allow for auditing “attention events” (*ATNEVT. This can be accomplished using the **WRKSYSVAL** command on either QAUDLVL or QAUDLVL2 and taking option 2 to change or add the option, as shown in Figure 3.

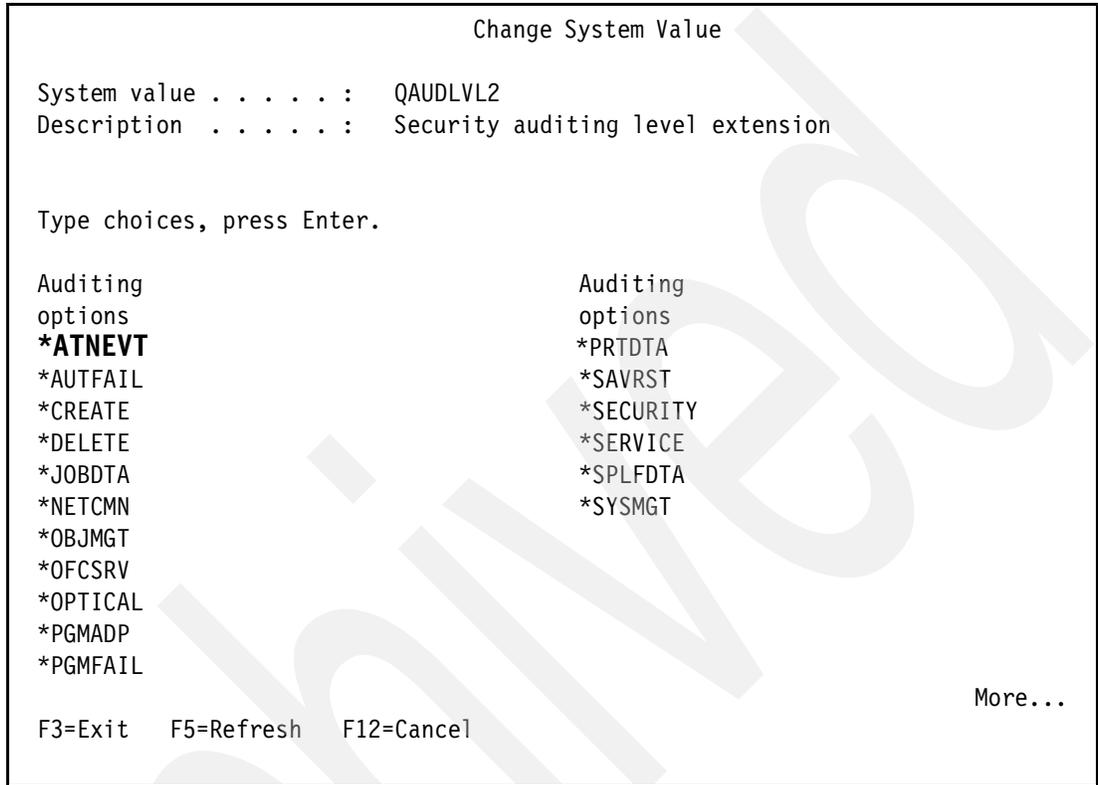


Figure 3 Allow auditing of attention events (*ATNEVT)

Note: If *ATNEVT is not set as an option in QAUDLVL, then *AUDLVL2 must be set in QAUDLVL in order to direct the system to interrogate QAUDLVL2 for any auditing options not covered in QAUDLVL.

4. A viable policy file, IDSPOLICY.CONF, must exist at /QIBM/USERDATA/OS400/QOS/ETC; otherwise, there will be no IDS policies to load. The conditions and actions in the policy file must be created by the user. To facilitate the creation of conditions and associated actions, a commented out sample policy is provided in the initial version of IDSPOLICY.CONF. A sample of a viable IDS policy file is shown in Figure 4.

Note: A copy of the IDS policy file is shipped in /QIBM/PRODDATA/OS400/QOS and loaded into /QIBM/USERDATA/OS400/QOS/ETC at installation time. The system administrator should edit the file in /QIBM/USERDATA/OS400/QOS/ETC to create conditions and actions that will be loaded to the network layer and enable IDS.

```

Edit File: /qibm/userdata/os400/qos/etc/idspolicy.conf
Record :      1  of      13 by  8      Column :      1      59 by  74
Control :

CMD .....1.....2.....3.....4.....5.....6.....7.....+
*****Beginning of data*****
  ibm-idsConditionAuxClass      idscond1
  {
  ibm-idsConditionType          ATTACK
  ibm-idsAttackType             FLOOD
  ibm-idsLocalPortRange         1-65535
  ibm-idsLocalHostIPAddress     2-9.000.000.000-8
  ibm-policyIdsActionName       idsact1
  }
  ibm-idsActionAuxClass         idsact1
  {
  ibm-idsActionType             ATTACK
  ibm-idsMaxEventMessage        25
  }
  *****End of Data*****

F2=Save  F3=Save/Exit  F12=Exit  F15=Services  F16=Repeat find
F17=Repeat change  F19=Left  F20=Right

```

Figure 4 A viable IDS policy file (idspolicy.conf) example

The sample in Figure 4 consists of one condition and one associated action. Normally, there will be several conditions covering all types of intrusions. An action may be referenced by more than one condition.

Note: In its most primal form, an IDS policy consists of at least one condition and an associated action. There may be multiple conditions and multiple associated actions. There may also be multiple conditions associated with one action.

5. Finally, in order to load the IDS policies, the QoS server should be started, as shown in Figure 5. This step will need to be done more than once if policy changes are made.

```
Command Entry                                     Request level: 1
Previous commands and messages:
(No previous commands or messages)

Type command, press Enter.
===> STRTCPSVR *QOS

F3=Exit   F4=Prompt   F9=Retrieve   F10=Include detailed messages
F11=Display full   F12=Cancel   F13=Information Assistant   F24=More keys

Bottom
```

Figure 5 Start the QoS server

To have the QoS server start automatically with TCP, select the **QoS Server Properties** by right-clicking **QoS** when using the iSeries™ Navigator and check the box next to **Start QoS server when TCP/IP is started**.

Note: To reload the IDS policy file after changes have been made, the QoS server needs to be ended and restarted. To end the QoS server, type the following command and press Enter:

```
ENDTCPSVR *QOS
```

IDS policy file

The IDS policy file is used to define the intrusions that will be audited. It is set up with *directives* or keywords for both conditions and actions. The various directives with their range of values are given in Example 1.

The example shows the prolog of the shipped IDS policy file (/QIBM/PRODDATA/OS400/QOS/IDSPOLICY.CONF). Following the definition of the directives is Table 2 on page 12, which shows both the condition and action directives.

Example 1 IDS policy directives defined in terms of their possible values

```
# IDS Policy File Keywords and Values:
#
# ibm-idsConditionAuxClass      <condition name> (max of 31 characters)
# {
#   ibm-idsConditionType:      ATTACK | TR | SCAN_GLOBAL | SCAN_EVENT
#   ibm-idsAttackType:         MALFORMED_PACKET | FLOOD | OUTBOUND_RAW |
#                               ICMP_REDIRECT | PERPETUAL_ECHO | IP_FRAGMENT |
#                               RESTRICTED_IP_OPTIONS | RESTRICTED_IP_PROTOCOL
#   ibm-idsLocalPortRange:     <from-port>[:<to-port>] (ports range from 1 to 65535)
#   ibm-idsRemotePortRange:    <from-port>[:<to-port>] (ports range from 1 to 65535)
#   ibm-idsProtocolRange:      <from-protocol>[:<to-protocol>] (protocols range from 1 to 255)
#                               (See www.iana.org/assignments/protocol-numbers)
#   ibm-idsIPOptionRange:      <from-option>[:<to-option>] (options range from 1 to 255)
#                               (See www.iana.org/assignments/ip-parameters)
#   ibm-idsLocalHostIPAddress: 1-All local addresses
#                               2-<IPv4Address>-<PrefixMaskLength>
#                               3-<IPV4Address1><-IPV4Address2>
#   ibm-idsRemoteHostIPAddress: 1-All remote addresses
#                               2-<IPv4Address>-<PrefixMaskLength>
#                               3-<IPV4Address1><-IPV4Address2>
#   ibm-policyIdsActionName    <action name> (max of 31 characters)
# }
# ibm-idsActionAuxClass        <action name> (max of 31 characters)
# {
#   ibm-idsActionType:         ATTACK | TR | SCAN_GLOBAL | SCAN_EVENT
#   ibm-idsStatInterval:       n          (Default is 60, max is 4294967295)
#                               where n is the interval length in minutes to collect
#                               IDS statistics
#   ibm-idsMaxEventMessage:    n          (Default is 5)
#                               where n is the maximum number of attack event
#                               messages to be audited per interval specified
#                               with ibm-idsStatInterval.
#   ibm-idsTRtcpTotalConnections: n
#                               where n is the total number of connections
#                               allowed for a listening server application.
#   ibm-idsTRtcpPercentage:    n          (Default is 100)
#                               where n represents anything in the range
#                               of 0 - 100%
#   ibm-idsTRtcpLimitScope:    PORT | PORT_INSTANCE (Default is PORT_INSTANCE)
#                               PORT specifies that traffic regulation parameters
#                               apply to the aggregate of all sockets bound to
#                               the target port (i.e., regardless of IP address)
#                               PORT_INSTANCE specifies that traffic regulation
#                               parameters apply to each socket bound to the
#                               target port individually (i.e., IP address range taken into account)
#   ibm-idsFSInterval:         n          (Default is 1 minute)
#                               where n is the interval in minutes for monitoring fast
#                               scanning attacks (maximum value is 1440)
#
```

```

# ibm-idsFSThreshold:      n      (Default is 5)
#                          where n is the fast scanning threshold (maximum value is 64)
# ibm-idsSSInterval:      n      (Default is 120 minutes)
#                          where n is the interval in minutes for monitoring slow
#                          scanning attacks (maximum value is 1440)
#                          NOTE: This interval must be greater than the fast scan
#                          interval. However, a value of 0 can be specified
#                          to indicate that no slow scan interval exists (to
#                          "turn off" slow scan processing).
# ibm-idsSSThreshold:     n      (Default is 10)
#                          where n is the slow scanning threshold (maximum value is 64)
#                          NOTE: This threshold must be greater than the fast scan
#                          threshold. However, a value of 0 can be specified
#                          to indicate that no slow scan threshold exists (to
#                          "turn off" slow scan processing).
# }

```

Note: Here it may be pointed out that the directive `ibm-idsMaxEventMessage` can gate the number of attention events that are written to the security audit journal. If the value is reasonable (for example, 25) in the action associated with, say, a “malformed packet” attack condition, only that small number of Intrusion Monitor (IM) records are cut in the audit journal.

If, however, the value of this action directive is large, the security audit journal could be “flooded” by IM records during a packet storm. A value of 0 for `ibm-idsMaxEventMessage` implies no limit to the number of audit journal entries that can be generated as a result of executing the corresponding action.

Given the IDS directives in Example 1 on page 10, it is now worth discussing how to construct meaningful conditions and actions. For example, it should be obvious that `ibm-idsFSInterval` does not go with an `ibm-idsActionType` of `ATTACK` since it applies to either an action type of `SCAN_EVENT` (preferred) or `SCAN_GLOBAL` (which is interpreted as a `SCAN_EVENT` internally).

For this reason, Table 2 on page 12 is presented as a “cheat sheet” for you to use when constructing conditions and their associated actions. Table 1 provides the key information to interpreting the IDS policy file directives in Table 2 on page 12.

Table 1 Key to IDS policy file directive in Table 2 on page 12

<ul style="list-style-type: none"> o - optional r - required x - not supported i - ignored d - depends on type of attack 	<ul style="list-style-type: none"> TR - (Traffic Regulation) SE - SCAN_EVENT SG - SCAN_GLOBAL AT - ATTACK 	<ul style="list-style-type: none"> MP - MALFORMED_PACKET FL - FLOOD OR - OUTBOUND_RAW IR - ICMP_REDIRECT PE - PERPETUAL_ECHO IF - IP_FRAGMENT RO - RESTRICTED_IP_OPTIONS RP - RESTRICTED_IP_PROTOCOL
---	---	--

Table 2 IDS policy file directives

	ibm-idsConditionType				ibm-idsActionType							
	TR	SE	SG ^a	AT	MP	FL	OR	IR	PE	IF	RO	RP
Condition Directives:												
ibm-idsLocalPortRange^b	o	r	i	o	o	o	x	o	o	o	o	o
ibm-idsRemotePortRange	o	o	i	o	o	o	x	o	o	o	o	o
ibm-idsProtocolRange	r	x	i	d	x	x	x	x	x	x	x	r
ibm-idsIPOptionRange	x	x	i	d	x	x	x	x	x	x	r	x
ibm-idsLocalHostIPAddress	r	r	i	o	o	o	x	o	o	o	o	o
ibm-idsRemoteHostIPAddress	o	o	i	o	o	o	x	o	o	o	o	o
ibm-policyIdsActionName	r	r	r	r	r	r	x	r	r	r	r	r
Action Directives:												
ibm-idsActionType	r	r	r	r								
ibm-idsStatInterval	o	i	i	o								
ibm-idsMaxEventMessage	o	o	o	o								
ibm-idsTRtcpTotalConnections	r	x	x	x								
ibm-idsTRtcpPercentage	r	x	x	x								
ibm-idsTRtcpLimitScope	o	x	x	x								
ibm-idsTRudpQueueSize	o	x	x	x								
ibm-idsFSInterval^c	x	o	o	x								
ibm-idsFSThreshold	x	o	o	x								
ibm-idsSSInterval	x	o	o	x								
ibm-idsSSThreshold	x	o	o	x								

- a. The TCP/IP stack can only detect single scan events. IDS keeps a tally of scan events and is better able to determine when a global scan has occurred.
- b. If no local port (range) is given, the condition applies to all local ports.
- c. The scan action directives (ibm-idsFSInterval, ibm-idsFSThreshold, ibm-idsSSInterval, ibm-idsSSThreshold) will be assigned the default values if not specifically assigned values in the policy file.

Note: Directives that do not appear in Table 2 (ibm-idsMessageDest, ibm-ICMPRedirect, ibm-idsNotification, ibm-idsLoggingLevel, ibm-idsTypeActions, ibm-idsSensitivity, ibm-idsScanExclusion) are ignored.

Examples of IDS policy conditions and actions

This section provides a few examples of IDS policy conditions and actions depending on attack types.

Flood

Consider the condition and action for a flood attack in Figure 6.

```
Edit File: /qibm/userdata/os400/qos/etc/idspolicy.conf
Record :      1  of      13 by  8          Column :      1   59 by  74
Control :

CMD .....1.....2.....3.....4.....5.....6.....7.....+
*****Beginning of data*****
ibm-idsConditionAuxClass      idscond1
{
ibm-idsConditionType          ATTACK
ibm-idsAttackType             FLOOD
ibm-idsLocalPortRange         1-65535
ibm-idsLocalHostIPAddress     2-9.000.000.000-8
ibm-policyIdsActionName       idsact1
}
ibm-idsActionAuxClass         idsact1
{
ibm-idsActionType             ATTACK
ibm-idsMaxEventMessage        25
}
*****End of Data*****

F2=Save   F3=Save/Exit  F12=Exit  F15=Services  F16=Repeat find
F17=Repeat change  F19=Left  F20=Right
```

Figure 6 IDSPOLICY.CONF file example: Flooding attack detection

This condition has the name “idscond1”. It describes a flood attack on any one of the local ports 1 through 65535 to the range of addresses 9.0.0.0 to 9.255.255.255. If any intrusions fit this description, then the action named “idsact1” is taken.

The action simply states that an event should be signaled, provided that 25 events that call out this action have not already been signaled. (And 25 IM records should then be seen in the audit journal.)

In the case of a flood attack, after IDS is notified by the TCP/IP stack, the system most probably is under attack. Floods are only signaled after an aging connection attempt (SYN) is dropped from the queue of all incomplete connection attempts. The attack is signaled immediately.

Traffic regulation

Consider the Traffic Regulation (TR) condition/action combination in Example 2 on page 14.

Example 2 IDSPOLICY.CONF file example: Traffic regulation

```
ibm-idsConditionAuxClass    idscond2
{
ibm-idsConditionType        TR
ibm-idsLocalPortRange       80
ibm-idsProtocolRange        6
ibm-idsRemoteHostIPAddress  2-9.124.1.0-24
ibm-policyIdsActionName     idsact2
}
ibm-idsActionAuxClass       idsact2
{
ibm-idsActionType           TR
ibm-idsStatInterval         10
ibm-idsTRtcpTotalConnections 1000
ibm-idsTRtcpPercentage      10
ibm-idsMaxEventMessage      50
}
```

The condition and associated action shown in Example 2 indicate that an attention event is generated when the number of established TCP connections to port 80 (HTTP server) from the range of remote IP addresses 9.124.1.0 to 9.124.1.255, over a certain period of time (10 minutes), has either exceeded a preset limit (1000) of connections, or exceeded a preset percentage (10%) of the total number of established connections to the system.

At the end of the interval, internal counts are reset and a new statistical interval begins. Note that notification occurs only when the conditions are met. Nothing is done dynamically to remedy the situation.

Scan

A sample condition/action combination is given for scan events in Example 3.

Example 3 IDSPOLICY.CONF file example: Scanning activities monitoring

```
ibm-idsConditionAuxClass    idscond3
{
ibm-idsConditionType        SCAN_EVENT
ibm-idsLocalPortRange       26-136
ibm-idsRemoteHostIPAddress  2-9.0.0.0-8
ibm-policyIdsActionName     idsact3
}
ibm-idsActionAuxClass       idsact3
{
ibm-idsActionType           SCAN_EVENT
ibm-idsFSInterval           1
ibm-idsFSThreshold          5
ibm-idsSSInterval           120
ibm-idsSSThreshold          10
ibm-idsMaxEventMessage      50
}
```

As shown in Example 3, a scan event will be signaled to the audit journal if the following conditions are met: a connection attempt is made to non-listening ports (that is, undemuxable SYNs) 26 to 136 from remote IP addresses in the range of 9.0.0.0 to 9.255.255.255 and such attempts number 5 or more for a 1 minute interval (fast scan), or such attempts take place at the rate of 10 every 120 minutes (slow scan). To distinguish between a fast scan and a slow scan, internal counts for both fast scans and slow scans are reset when a threshold is met.

Also, the internal count for fast scans is reset when the fast scan interval expires. Similarly, the internal count for slow scans is reset when the slow scan interval expires. For non-listening ports, any connection attempt may be of interest to a system administrator, especially the slow scan case where a perpetrator may be trying to bypass defenses and sneak in “under the radar.”

Intrusion Monitor entries

When intrusions are signaled from the IDS task, an entry is made in the security audit journal. An example of an Intrusion Monitor (IM) entry is given in Figure 7.

```

Display Journal Entry

Object . . . . . :                               Library . . . . . :
Member . . . . . :                               Minimized entry data : *NONE
Incomplete data . . : No
Sequence . . . . . : 46500
Code . . . . . : T - Audit trail entry
Type . . . . . : IM - Intrusion monitor

Entry specific data
Column *...+....1....+....2....+....3....+....4....+....5
00001 'P2006-06-25-18.11.41.0911761105 169909.5.175.141 '
00051 '                                     027549.10.229.77'
00101 '                                     TR-TCP0233 '
00151 '                                     á B Ja V( ýâ; B '
00201 'M OnÿÐ Ø Ø zÅ K ©vŌ '

Bottom

Press Enter to continue.

F3=Exit F6=Display only entry specific data
F10=Display only entry details F12=Cancel F24=More keys

```

Figure 7 Traffic Regulation (TR) TCP event denoted by Intrusion Monitor (IM) entry in the security audit journal (QSYS/QAUDJRN)

Referring to the Intrusion Monitor audit record layout in Table 3 on page 16, the audit journal entry in Figure 7 describes a Traffic Regulation event:

- ▶ With a “P” for “potential” intrusion (they are all potential intrusions)
- ▶ Occurring on 6/25/2006 at 18:11:41.091176
- ▶ Detected by internal code point 1105
- ▶ And, though we cannot see it, within the address family of IPv4 (0x02)
- ▶ To local port 16990 and local IP address 9.5.175.141
- ▶ From remote port 2754 and remote IP address 9.10.229.77
- ▶ Of the Traffic Regulation variety depicting an established TCP connection

- ▶ With an event correlator of 233 (used for debugging)
- ▶ And a suspected packet consisting of a 2-byte non-displayable length and up to 1000 bytes, which can best be viewed in hex format by pressing F11

If the intrusion in Figure 7 on page 15 appears over and over again from the same remote IP address, a system administrator could create an IP filter rule denying any more input from that address. IP filtering is not discussed in this paper, but is one preventive measure that a system administrator could take to deny access to suspicious clients. (See the i5/OS system Information Center: **Networking** → **Networking security** → **IP filtering and network address translation**).

Table 3 Intrusion Monitor audit record

Offset	Field	Format	Description
1			Heading fields common to all entry types.
610	Entry Type	Char(1)	The type of entry. P Potential intrusion event detected
611	Time of Event	TIMESTAMP	Timestamp of when the event was detected in SAA® timestamp format.
637	Detection Point Identifier	Char(4)	This is a unique identifier for the processing location that detected the intrusion event. This field is intended for use by service personnel.
641	Local Address Family	Char(1)	Local IP address family associated with the detected event.
642	Local Port Number	Zoned(5,0)	Local port number associated with the detected event.
647	Local IP Address	Char(46)	Local IP address associated with the detected event.
693	Remote Address Family	Char(1)	Remote address family associated with the detected event.
694	Remote Port Number	Zoned(5,0)	Remote port number associated with the detected event.
699	Remote IP Address	Char(46)	Remote IP address associated with the detected event.
745	Probe Type Identifier	Char(6)	Identifies the type of probe used to detect the potential intrusion. Possible values include: <ul style="list-style-type: none"> ▶ ATTACK – attack action detected event ▶ TR-TCP, TR-UDP– Traffic Regulation event ▶ SCANG – scan global action detected event ▶ SCANE– scan event action detected event
751	Event Correlator	Char(4)	Unique identifier for this specific intrusion event. This identifier can be used to correlate this audit record with other intrusion detection information.

755	Event type	Char(8)	Identifies the type of potential intrusion that was detected. The possible values are: <ul style="list-style-type: none"> ▶ MALFPKT – malformed packet ▶ FLOOD – flood event ▶ ICMPRED – ICMP (Internet Control Message Protocol) redirect ▶ PERPECH – perpetual echo ▶ IPFRAG – IP fragment ▶ RESTOPT - restricted IP options ▶ RESTPROT – restricted IP protocol
763	Reserved	Char(20)	Reserved for future use
783	Suspected Packet	Char(1002) ¹	This is a variable length field which may contain up to the first 1000 bytes of the IP packet associated with the detected event. This field contains binary data and should be treated as though it has a CCSID of 65535.

Intrusion Monitor entries in the audit journal may be viewed by entering the following command:

```
DSPJRN QAUDJRN ENTTYP(IM)
```

This command can be specified with multiple parameters, which can help limit the number of IM entries returned. For example, the most recent entries could be viewed by specifying a value for the FROMTIME parameter that was after the time that the last DSPJRN command was run. A CL program could be written to exploit these capabilities and return only the latest entries that have been added to the audit journal. This is an exercise left to the reader.

Verifying IDS policy implementation

To verify whether your IDS policy is active, complete the following steps:

1. Verify whether jobs QTOQRAGENT and QTOQSRVR are active in subsystem QSYSWRK.
2. If QTOQRAGENT is not active, review its job log to help identify policy directive errors.

To verify whether your IDS directives are correct, do the following:

1. Send datagrams that match the IDS policy directives that you are monitoring.
2. Review the audit journal for IM type entries.

Using nmap 4.11, we generate datagrams to send to our i5/OS v5r4 host for IDS policy verification.

Note: For more information regarding nmap, go to:

<http://www.insecure.org/nmap/>

This protocol scan example is capable of generating attack and TCP traffic regulation notifications.

```
nmap -s0 i5oshostname
```

This example is using a SYN-scan directly to our i5/OS host to generate IDS policy notifications.

```
nmap -sS i5oshostname
```

This example uses a “zombie” with a stealth idle scan to generate scan event notifications from an IDS policy. But note that our notification logs only show the source as the zombie host.

```
nmap -P0 -sI zombiehost i5oshostname
```

To assist in mining the audit journal for intrusion monitoring notifications, use the following i5/OS commands:

- ▶ CPYAUDJRNE IM

This will copy im entries, if any, to qtemp/qauditim.

- ▶ RUNQRY *NONE QAUDITIM

This will query the im entries from qtemp/qauditim.

Tips and techniques

The following are a few tips and techniques regarding running i5/OS IDS:

- ▶ It is important to verify that IDS policies are continually maintained.
- ▶ Remember, inbound datagrams may be stopped by active packet filtering rules, thus preventing them from reaching IDS for processing.
- ▶ Insure that your policy action directive criteria are met. Thresholds and intervals can be overlooked when attempting to identify why datagram traffic did not trigger a policy directive notification.
- ▶ Note that only TCP/IP scans to non-listening ports and spoofing attempts are signaled for IDS processing.

i5/OS intrusion detection and prevention - a summary

The Intrusion Detection System on the i5/OS system is an integrated, host-based, highly secure and yet flexible, notification system. Potential intrusions are presented as events in the system security audit journal. These potential intrusions may signify that a firewall is not doing its job and may need reprogramming by a network administrator.

On the other hand, the intrusions may signify that the firewall is doing its job, within its limitations, and that the host-based IDS is catching intrusions that have sneaked through the firewall. At any rate, the i5/OS system IDS can stand alone as an intrusion detection system or be used in conjunction with a firewall for even greater security and peace of mind.

This Redpaper has primarily dealt with intrusion detection. With i5/OS system IDS, IBM has extended its capabilities in network security to place it at the vanguard of the IT industry.

The team that wrote this IBM Redpaper

This Redpaper was produced by a team of specialists through a short-term residency at the International Technical Support Organization, Rochester Center.



Yessong Johng is an IBM Certified IT Specialist at the IBM International Technical Support Organization, Rochester Center. He started his IT career at IBM as a S/38 Systems Engineer in 1982 and has been with S/38, AS/400, and iSeries for 20 years. He writes extensively and develops and teaches IBM classes worldwide on the areas of IT Optimization whose topics include Linux, AIX®, and Windows® implementations on System i™ platform. His other areas of expertise include TCP/IP, networking security, HA, and SAN.



Jim Coon has been an IBM Software Engineer for 22 years. He started out working on the IBM System 36 in networking and communications. He then migrated to the AS/400® and now the iSeries. In addition to networking and communications, he has worked on retail support, Digital Certificate Manager, and now Intrusion Detection.



Craig Jacquez has been involved with IBM midrange systems since the late 1970s. Working with application development, networking and systems integration across many platforms. Craig holds the following technical certifications: IBM eServer-Certified Specialist for Technical Solutions, Client Access, WebSphere®, Domino®, eBusiness, and Linux®.

Thanks to the following people for their contributions to this project:

Dave Christenson
Christopher Gloe
Rick Hemmer
Brian Jongekryg
Scott McCreadie
Tim Seeger
Daryl Woker
IBM Rochester

Archived

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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 and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.



Send us your comments in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:
ibm.com/redbooks
- ▶ Send your comments in an email to:
redbook@us.ibm.com
- ▶ Mail your comments to:
IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400 U.S.A.



Trademarks

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

Redbooks (logo) ™
iSeries™
i5/OS®
AIX®

AS/400®
Domino®
IBM®
System i™

SAA®
WebSphere®

The following terms are trademarks of other companies:

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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