# z/OS Communications Server Security: Protecting z/OS from the Network

## SHARE Session 1725

**Lin Overby**
**overbylh@us.ibm.com**

# Trademarks and notices

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

- AIX 7
- AnyNet 7
- AS/400 7
- Candle 7
- CICS 7
- CICSPlex 7
- CICS/ESA 7
- DB2 7
- DB2 Connect™
- DPI 7
- DRDA 7
- e business(logo) 7
- ESCON 7
- eServer™
- ECKD™
- FFST™
- GDDM 7
- GDPS 7
- HiperSockets™
- IBM 7
- Infoprint 7
- IMS™
- IP PrintWay™
- iSeries™
- Language Environment 7
- MQSeries 7
- MVS™
- MVS/ESA™
- NetView 7
- OS/2 7
- OS/390 7
- Parallel Sysplex 7
- PrintWay™
- PR/SM™
- pSeries 7
- RACF 7
- Redbooks™
- Redbooks (logo)™
- S/390 7
- System/390 7
- ThinkPad 7
- Tivoli 7
- Tivoli (logo) 7
- VM/ESA 7
- VSE/ESA™
- VTAM 7
- WebSphere 7
- xSeries 7
- z/Architecture™
- z/OS 7
- z/VM 7
- zSeries 7

- 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 trademarks of Microsoft Corporation in the United States, other countries, or both.
- Intel, Intel Inside (logos), MMX and Pentium 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 trademark of Linus Torvalds in the United States, other countries, or both.
- Red Hat is a trademark of Red Hat, Inc.
- SUSE® LINUX Professional 9.2 from Novell®
- Other company, product, or service names may be trademarks or service marks of others.
- This information is for planning purposes only. The information herein is subject to change before the products described become generally available.
- All statements regarding IBM future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

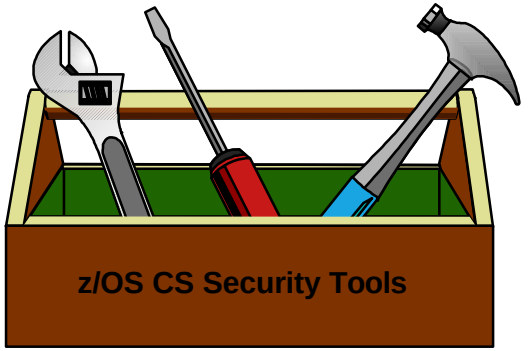
All performance data contained in this publication was obtained in the specific operating environment and under the conditions described and is presented as an illustration. Performance obtained in other operating environments may vary and customers should conduct their own testing.

Refer to www.ibm.com/legal/us for further legal information.

# Topics

- z/OS Communications Server security roles and objectives
- Overview of TCP/IP security functions
  - ➤Focus on primary TCP/IP security tools to protect z/OS and TCP/IP resources from the network
- System and resource protection
  - ➤IP packet filtering
  - ➤Intrusion Detection Services
  - ➤Syslogd Protection
  - ➤SAF SERVAUTH class protection of TCP resources

**z/OS CS provides a variety of network security functions to pick and choose.**

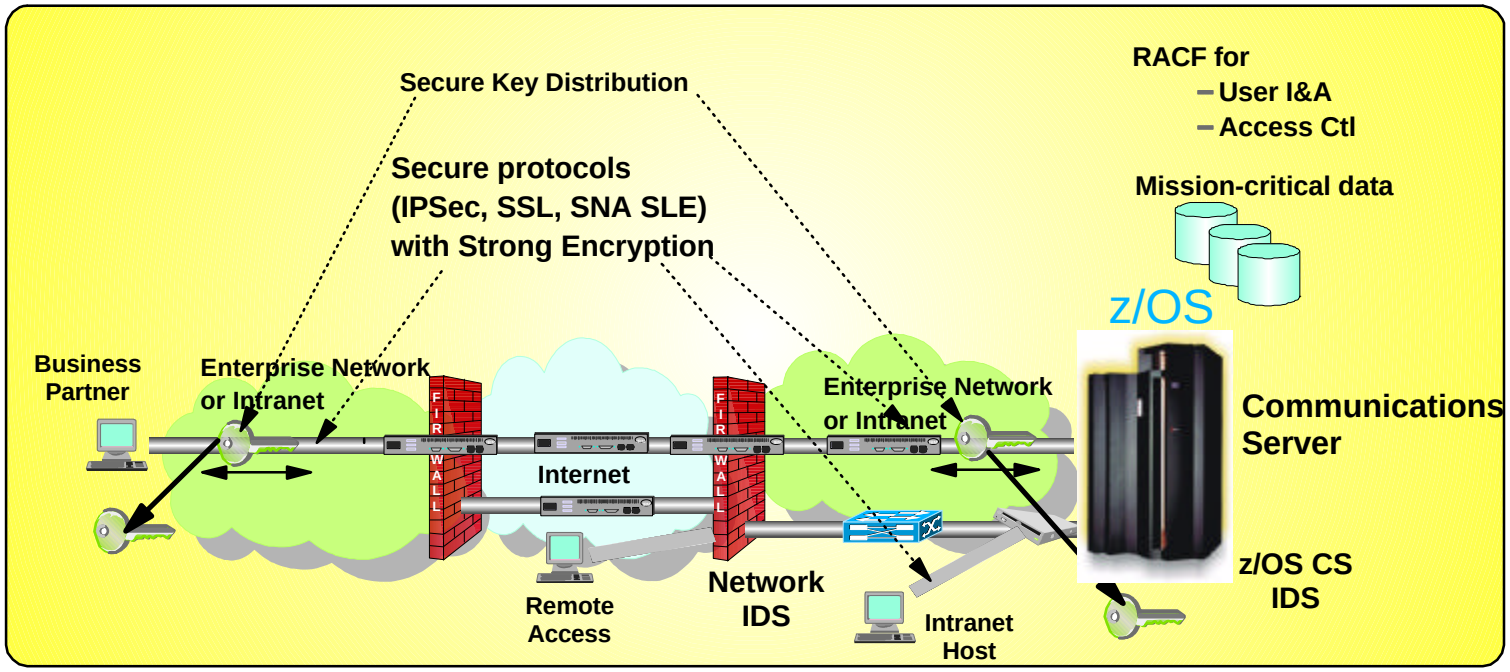



zSeries and z/OS base security characteristics and functions

Objective: Concepts, why, what, selection, and planning

# z/OS Communications Server Security Roles and Objectives

- ✓ Secure access to both TCP/IP and SNA applications
- ✓ Focus on end-to-end security and self-protection
- ✓ Exploits strengths of zSeries hardware and software



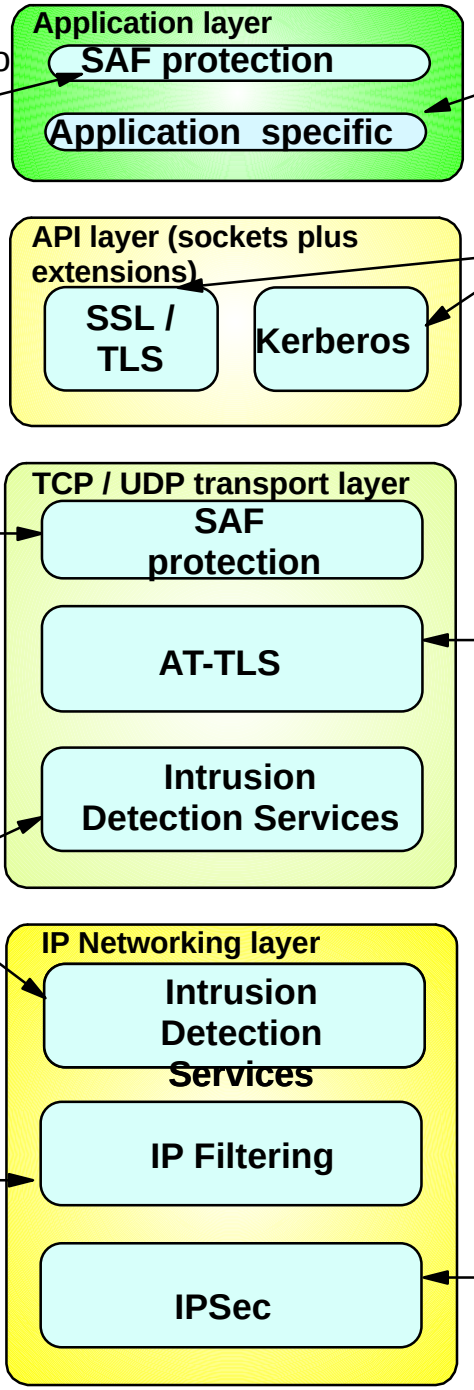

- **Protect data and other resources on the system**
  - ***System availability***
    - Protect system against unwanted access and denial of service attacks from network
  - ***Identification and authentication***
    - Verify identity of users
  - ***Access control***
    - Protect data and other system resources from unauthorized access

- **Protect data in the network using cryptographic security protocols**
  - ***Data Origin Authentication***
    - Verify that data was originated by claimed sender
  - ***Message Integrity***
    - Verify contents were unchanged in transit
  - ***Data Privacy***
    - Conceals cleartext using encryption

# Protocol Stack View of TCP/IP Security Functions

**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..




**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.

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

AT-TLS is TCP/IP stack service that provides SSL/TLS services at the TCP transport layer and is transparent to upper-layer protocols. It is avaliable to TCP applications in all programming languages except PASCAL.

**New in z/OS V1R7**

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

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

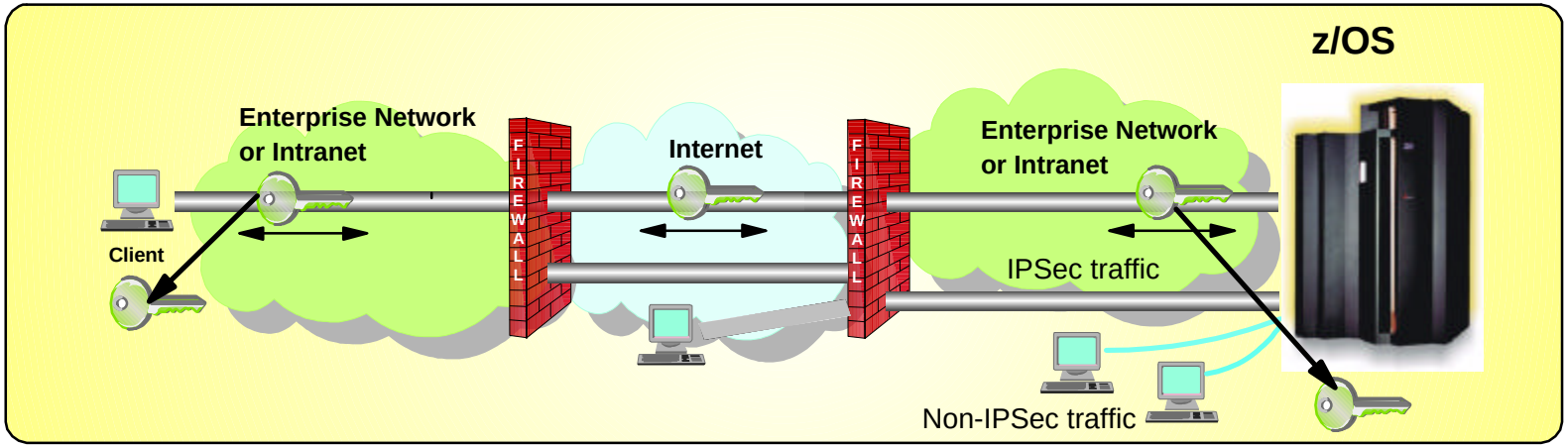
# Steps to Protect TCP/IP / System Resources

1. Blocking unwanted traffic
   - ►Solution: IP packet filtering
2. Protecting against malicious or accidental attacks on your system or your legitimate (open) services
   - ►Solution: intrusion detection services
3. Making sure UNIX system services log data is available when you need it to analyze past events
   - ►Solution: syslogd isolation
4. Controlling user access to TCP/IP resources on the system
   - ►Solution: SAF SERVAUTH class protection using protection of SERVAUTH resources

# z/OS Communications Server Resource Protection

- **IP packet filtering**
- **Intrusion Detection Services**
- **Syslogd availability and integrity**
- **SAF SERVAUTH class protection of TCP resources**

# Background - IP Packet Filtering
# z/OS IP security support




- Prior to z/OS V1R7, IP security packaged with Firewall Technologies
  - ►TCP/IP IPSec and IP filtering support
    - –Communications Server
  - ►IKE daemon and configuration
    - –Integrated Security Services
- In z/OS V1R7, complete IPSec, IP filtering, and IKE solution part of z/OS Communications Server
  - ►Alternative to Firewall Technologies
    - –New IKE daemon and configuration
  - ►Services
    - –IP filtering
    - –Manual IPSec
    - –Dynamic IPSec (IKE)
    - –Filter directed logging to syslogd

- **z/OS Communications Server IP security covers:**
  - ►IP filtering
  - ►IPSec

# Firewall Technologies

- Starting in z/OS V1R8, Firewall Technologies is no longer available

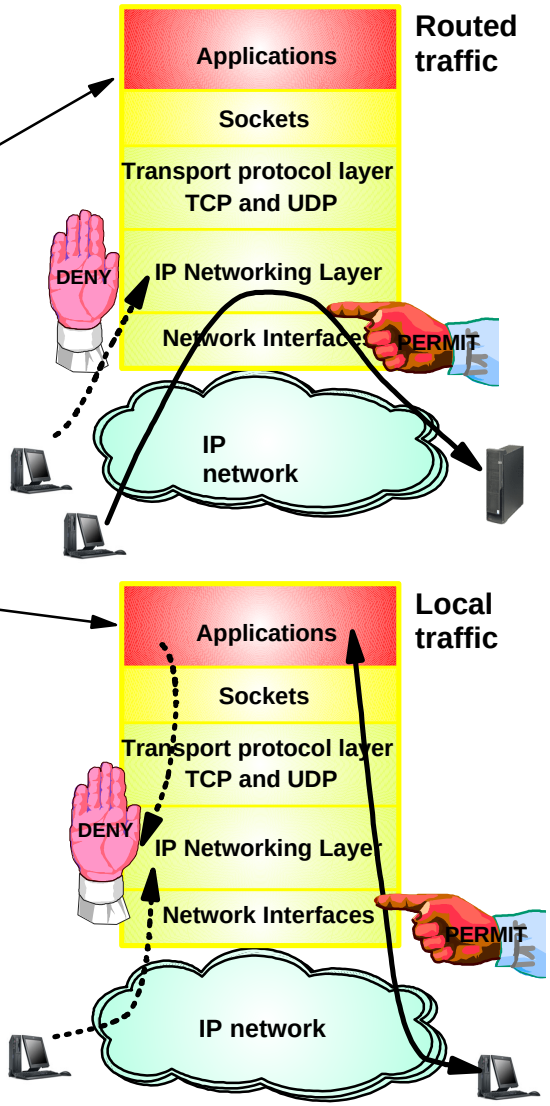
Announced Feb 15th, 2005

- z/OS V1.7 is the last z/OS release to include the Firewall Technologies component of the Integrated Security Services element.
- Many Firewall Technologies functions have been stabilized for some time and can be replaced using comparable or better functions provided by or planned for Communications Server, notably,
  - IPSec
  - IP packet filtering
  - In addition, a functionally rich downloadable tool is planned to replace the IPSec and IP Filtering configuration GUI support.
- The following functions will be removed without replacement:
  - FTP Proxy services
  - Socks V4 services
  - Network Address Translation (NAT)
  - RealAudio support

# IP Packet Filtering Basics

**Packet filtering at IP Layer**

- Filter rules defined to match on inbound and outbound packets based on:
  - packet information
    - IP address, port, protocol
  - network attributes
    - direction, link security
  - time
- Used to control
  - traffic being routed
  - access at server
    - "Personal firewall" on z/OS
- Possible actions
  - 1. Permit
  - 2. Deny
  - 3a. Permit with manual IPSec
  - 3b. Permit with dynamic IPSec
  - Log
    - in combination with other actions

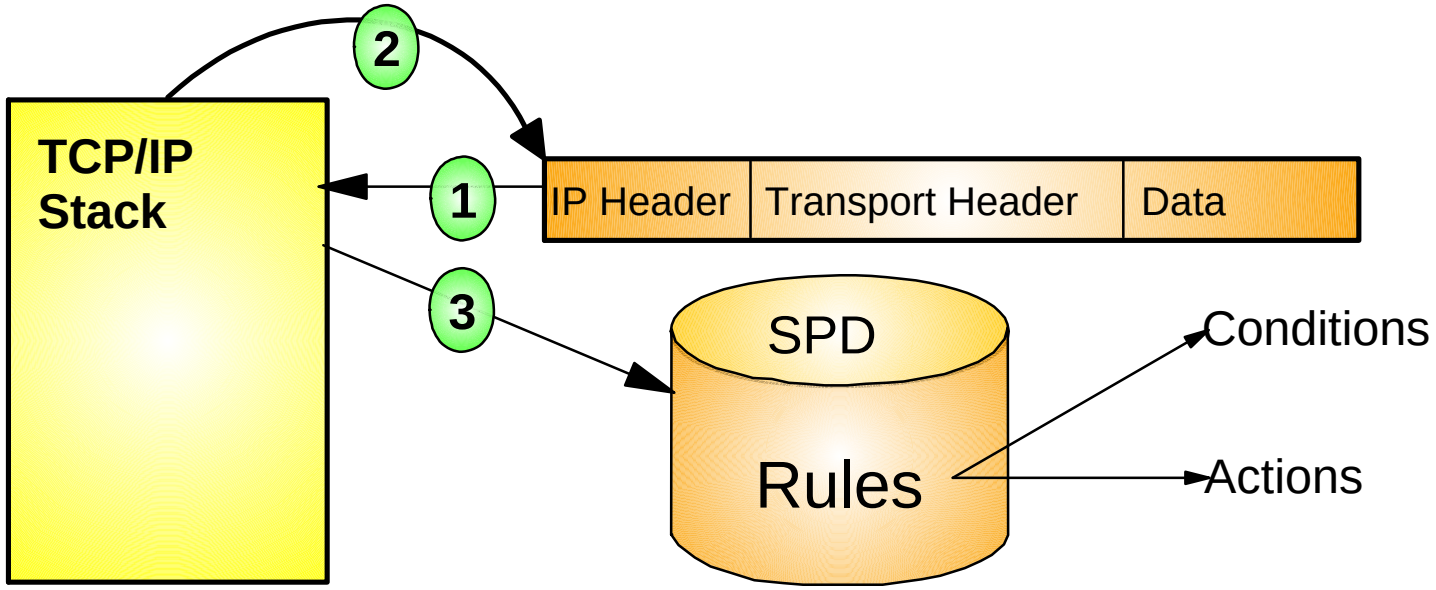


# IP Packet Filtering Processing Overview

1. **Inbound or outbound IP packet arrives**

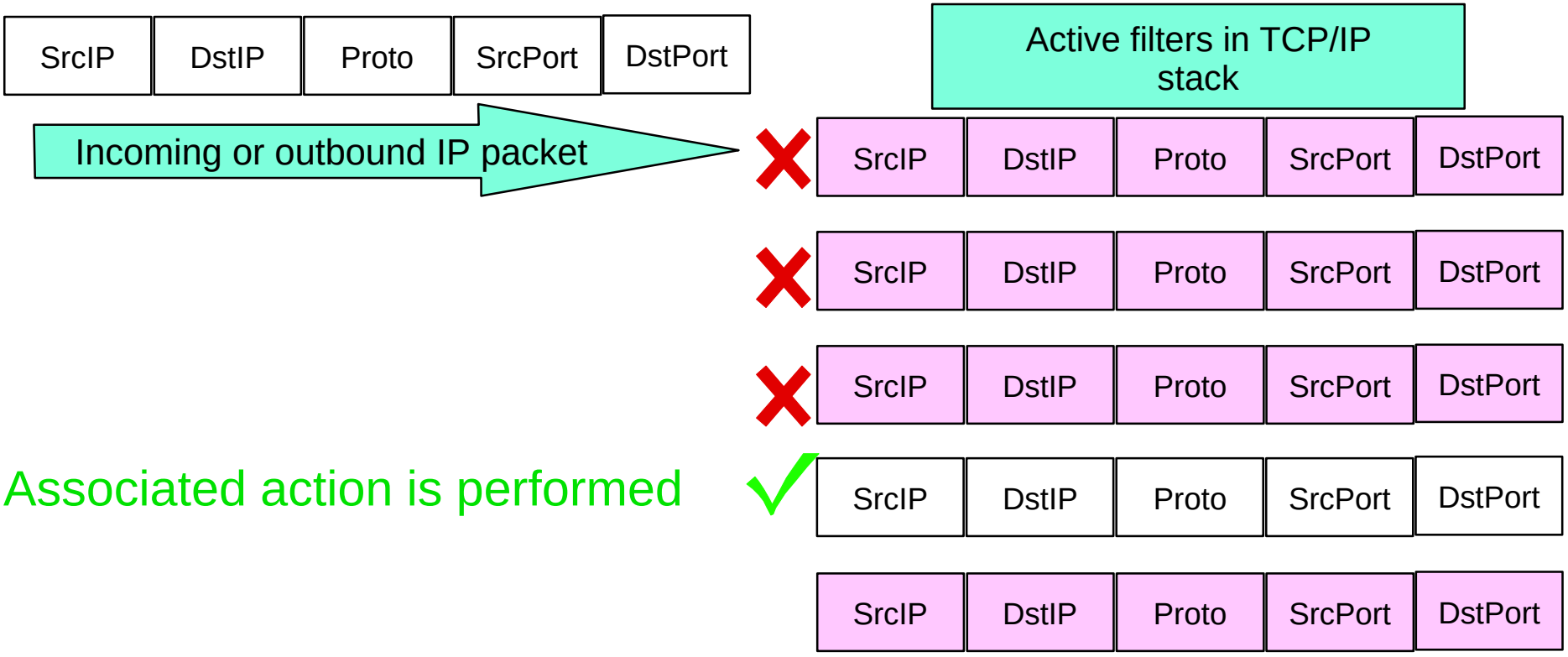
2. **Consult filter rules in a Security Policy Database (SPD)**
   - ►Rules have conditions and actions

3. **Apply action of matching rule to packet**
   - ►Deny
   - ►Permit
   - ►Permit with additional processing applied



# IP Filtering Concepts
# Filter Matching

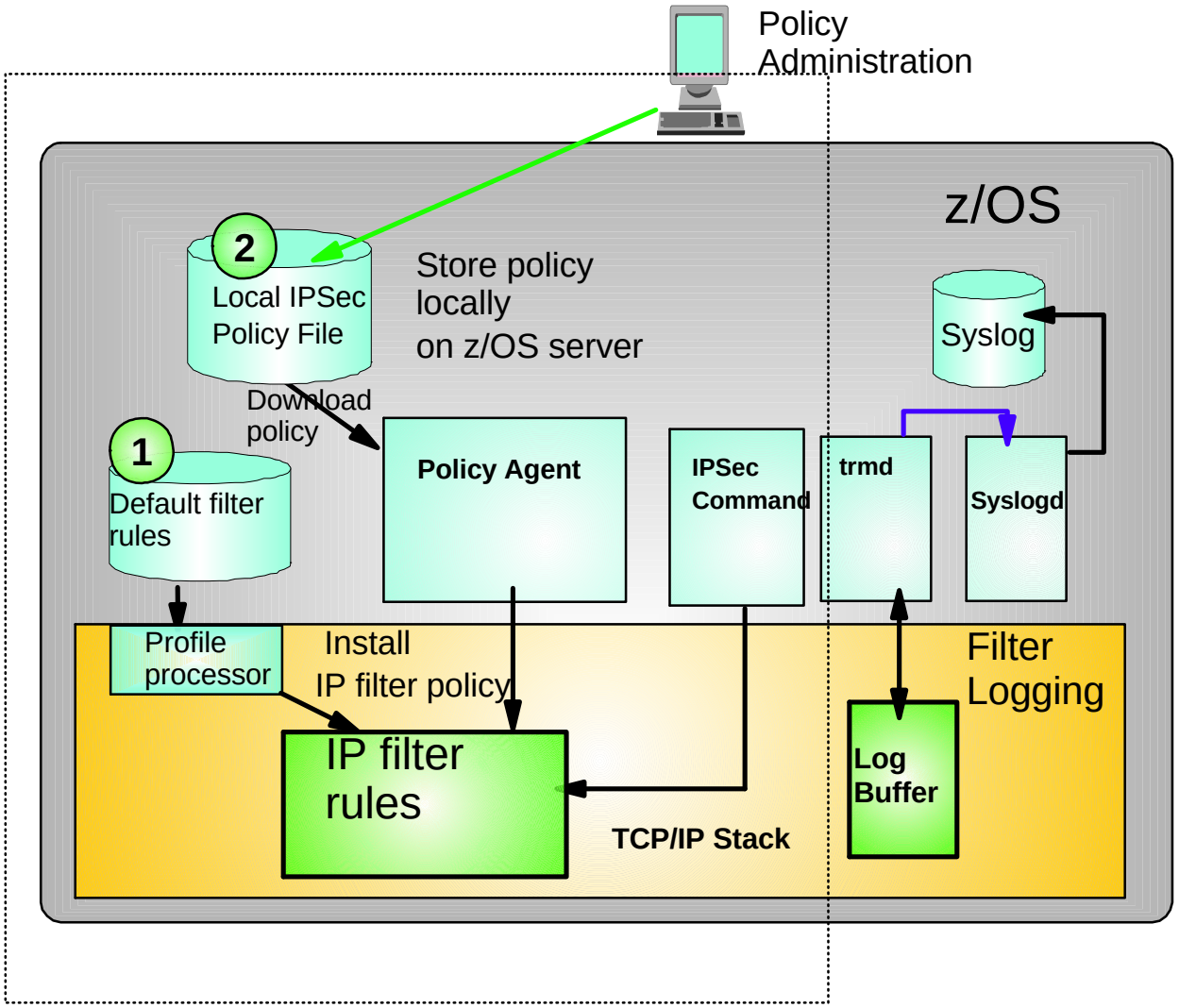
1. Filters are searched in the order they were configured
2. Each rule is inspected, from top to bottom, for a match
3. If a match is found, the search ends and the action is performed



# IP packet filtering conditions

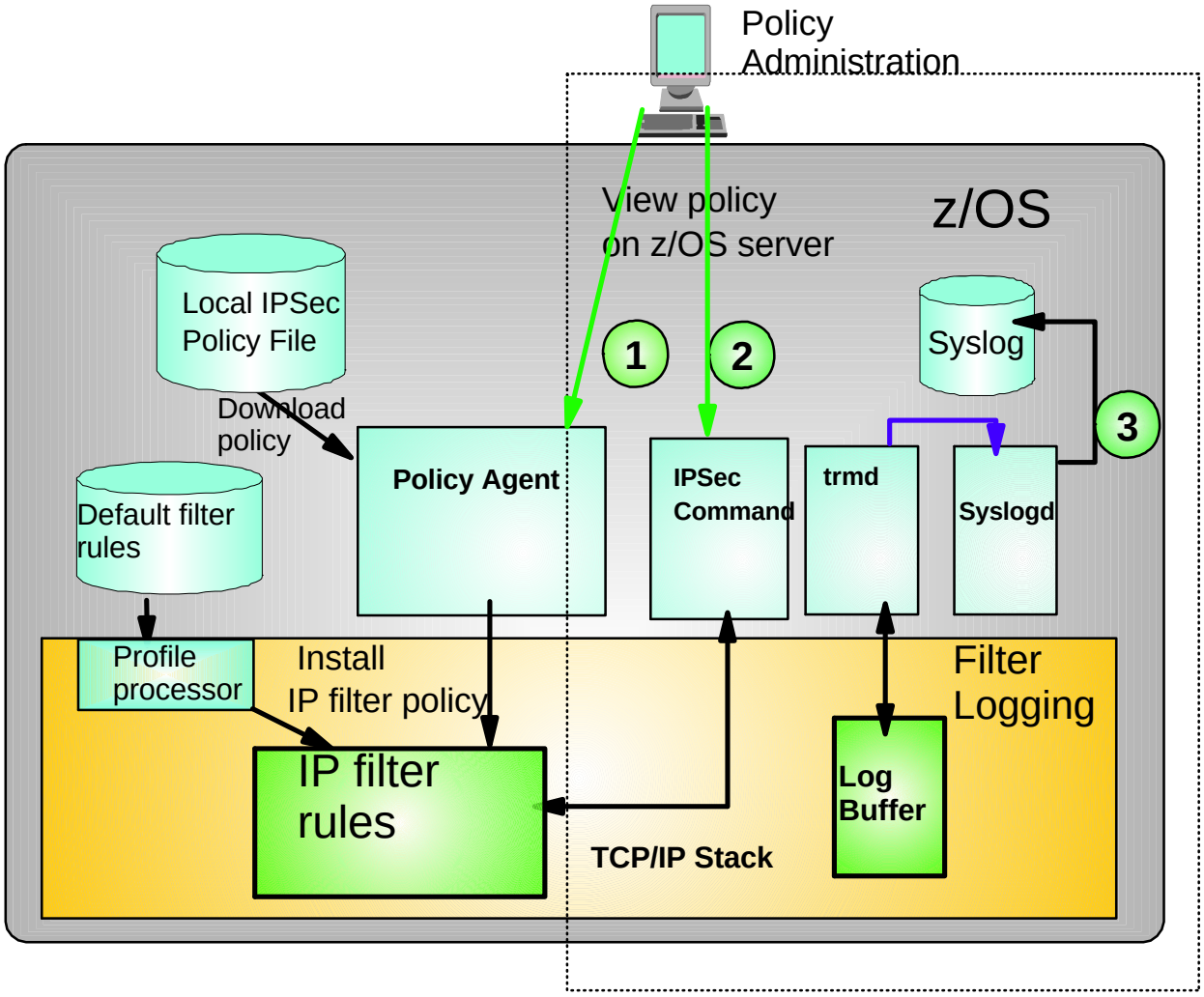
| Criteria | Description |
|---|---|
| **From packet** | |
| Source address | Source IP address in IP header of packet |
| Destination address | Destination IP address in IP header of packet |
| Protocol | Protocol in the IP header of packet (TCP, UDP, OSPF, etc.) |
| Source port | For TCP and UDP, the source port in the transport header of packet |
| Destination port | For TCP and UDP, the destination port in the transport header of packet |
| ICMP type and code | For ICMP, type and code in the ICMP header of packet |
| OSPF type | For OSPF, type located in the OSPF header of packet |
| **Network attributes** | |
| Direction | Direction of packet. |
| Routing | Packet is local if source or destination IP address exists on local host, otherwise it is routed |
| Link security class | A virtual class that allow you to group interfaces with similar security requirements. Non-VIPA addresses can be assigned a security class. Packets inherit the security class of the interface over which packet is sent/received. |
| **Time condition** | |
| Time, Day, Week, Month | Indicates when filter rule is active |

# IP Packet Filtering Policy




1. Default rules in TCP/IP profile
   - Provides filter rule protection when main policy agent filter rules are not active
     - Customize rules to handle window between TCP/IP stack initialization and policy agent filter rule load
     - Quick reload via IPSec command shuts down network access to services
2. Policy agent flat file
   - Main filter rule policy
     - More robust filter rule capability
     - Provides base for IPSec policy

# IP Filtering Monitoring and Logging




1. Pasearch command
   - ►Display policy agent filter policy
2. IPSec command
   - ►Display active filters
3. Filter logging
   - ►trmd logs via syslogd for stack based on filter rule actions

# Defining IP Packet Filtering Policy

✓ See sessions 3914 and 3907 for more information on configuring IP packet filtering policy

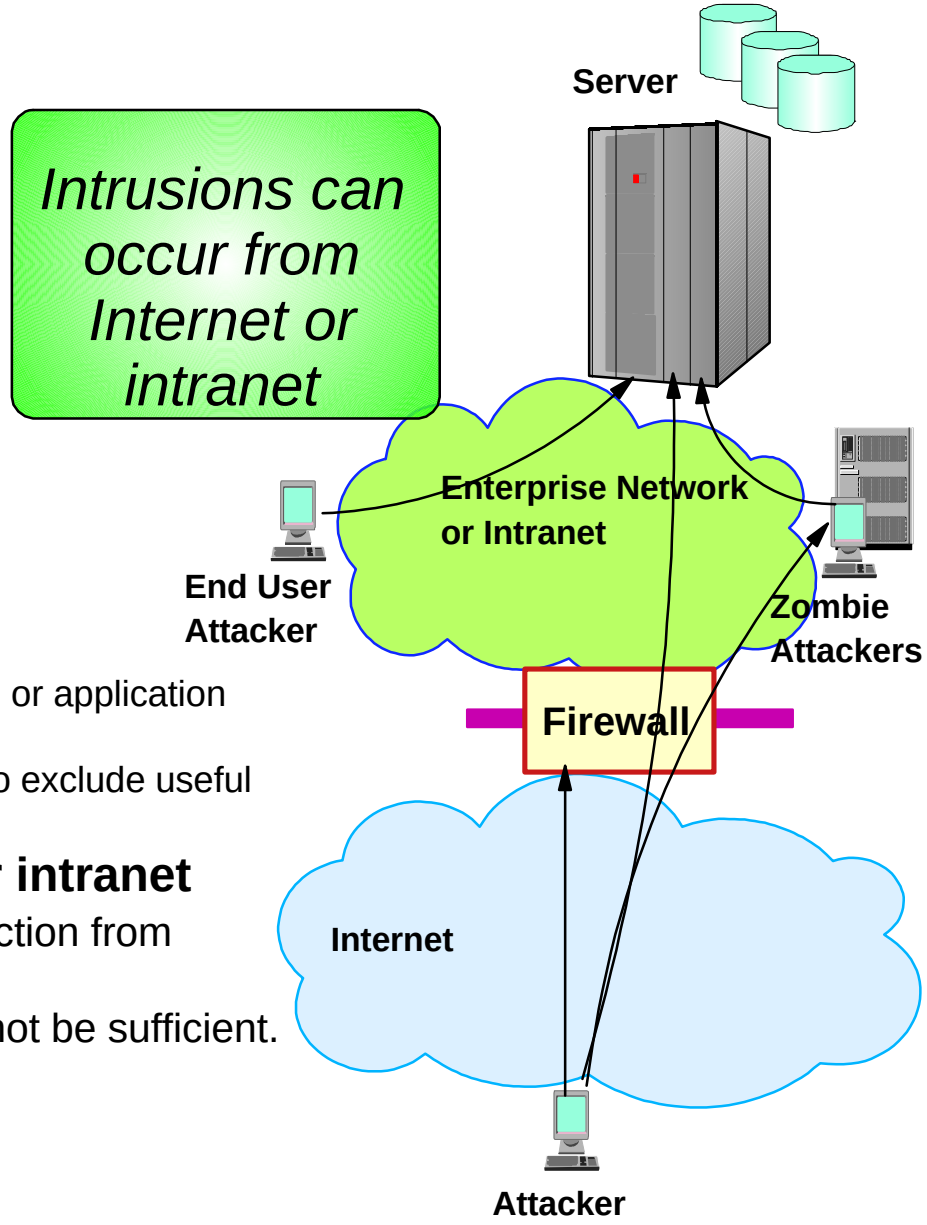

- **In z/OS V1R8 the Policy Agent configuration tools are combined into one tool to manage policies for:**
  - ►AT-TLS
  - ►IPSec and IP packet filtering
  - ►IDS
  - ►QoS
- **Common approach for all policy types:**
  - ►Master copy stored in binary file format (on workstation or file server)
  - ►Text-based configuration files to be parsed by Policy Agent are created and transferred to z/OS

- V1R7 - Network Security Configuration Assistant
  - ►Configured policy for IPSec, IP packet filtering, and AT-TLS
- V1R8 - (Name change) Configuration Assistant for z/OS Communications Server
  - ►Adds policy configuration support for IDS and QoS

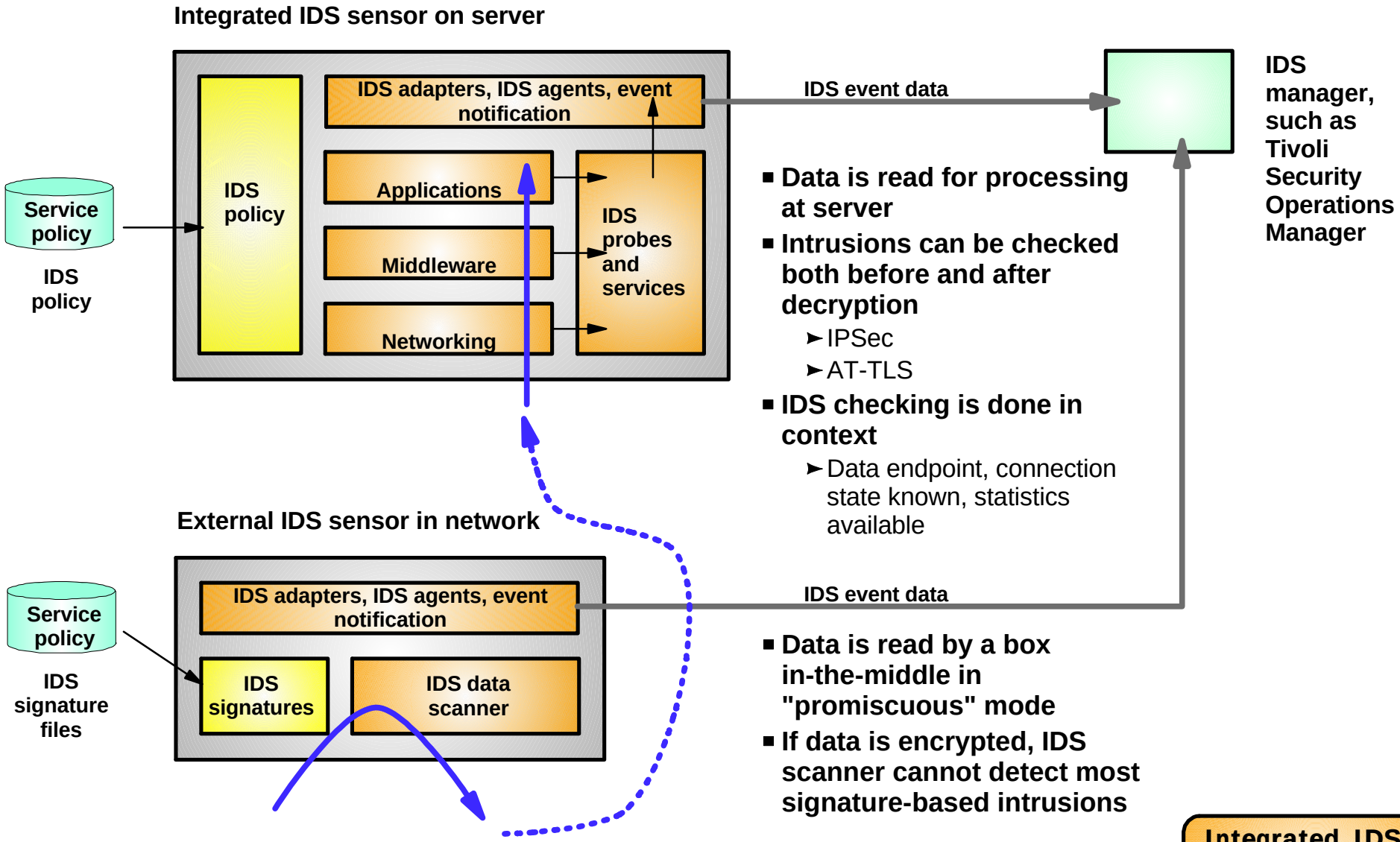
Downloadable policy configuration tool:
http://www-1.ibm.com/support/docview.wss?rs=206&uid=swg21181698

# The Intrusion Threat

- **What is an intrusion?**
  - Information Gathering
    - Network and system topology
    - Data location and contents
  - Eavesdropping/Impersonation/Theft
    - On the network/on the host
    - Based 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



# Integrated vs. External Intrusion Detection Concepts

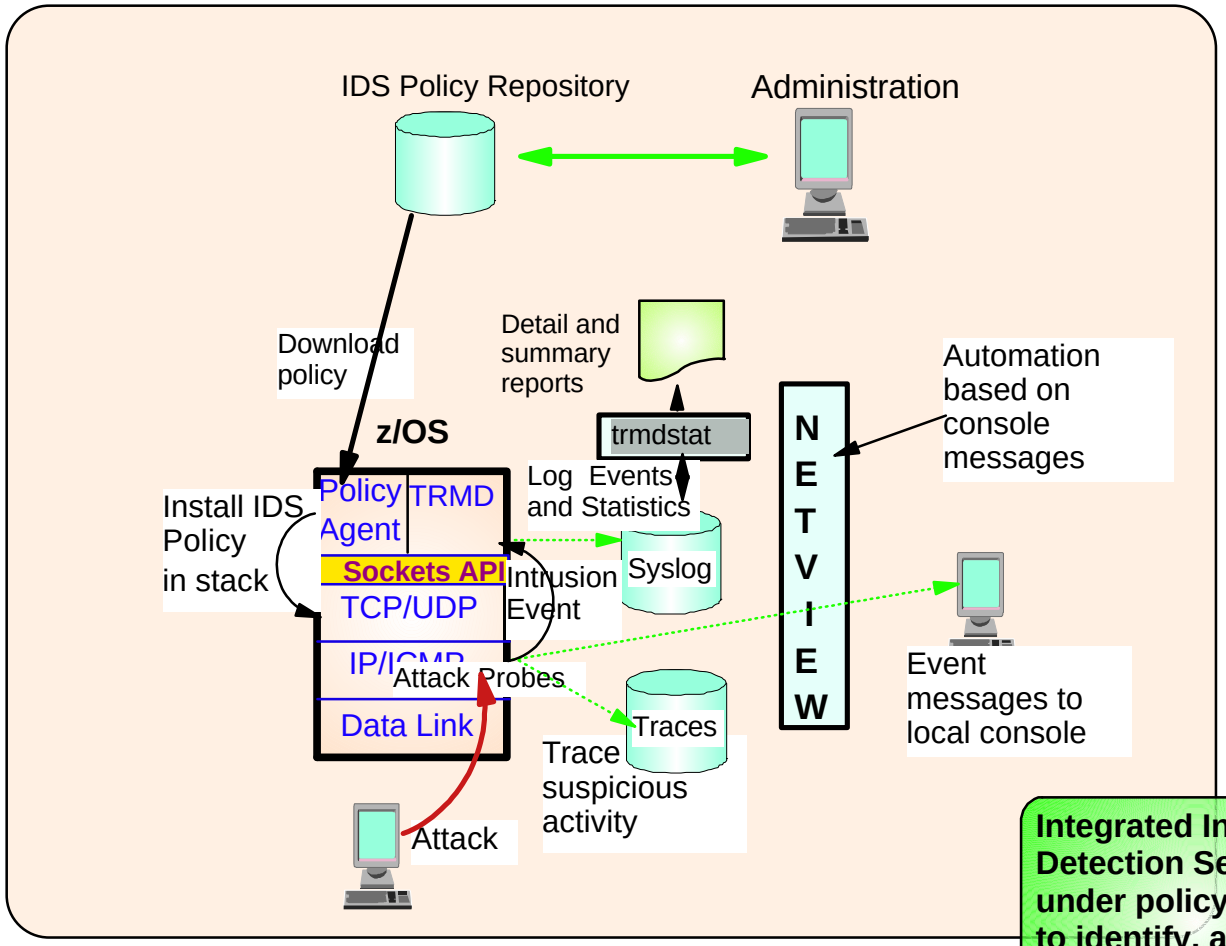



- Data is read for processing at server
- Intrusions can be checked both before and after decryption
  - IPSec
  - AT-TLS
- IDS checking is done in context
  - Data endpoint, connection state known, statistics available

- Data is read by a box in-the-middle in "promiscuous" mode
- If data is encrypted, IDS scanner cannot detect most signature-based intrusions

**Integrated IDS on z/OS complements external IDS technologies**

# 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 Tivoli NetView and Tivoli Security Operations Manager

**IDS Policy Repositories**
- LDAP
- Flat file support as LDAP alternative
  - **New in V1R8**

**Integrated Intrusion Detection Services under policy control to identify, alert, and document suspicious activity**

**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

# Intrusion Event Types Supported

- 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
- Attack detection, reporting, and prevention
  - ➤ Intent is to crash or hang the system
    - – Single or multiple packet
- Traffic regulation for TCP connections and UDP receive queues
  - ➤ Could be intended to flood system OR could be an unexpected peak in valid requests

# Scanning... the prelude to the attack

- z/OS IDS definition of a scanner
  - Source host that accesses multiple unique resources (ports or interfaces) over a specified time period
    - Installation can specify via policy number of unique events (Threshold) and scan time period (Interval)
- Categories of scan detection supported
  - Fast scan
    - Many resources rapidly accessed in a short time period (less than 5 minutes)
      - ✓ usually less than five minutes, program driven
  - Slow scans
    - Different resources intermittantly accessed over a longer time period (many hours)
      - ✓ scanner trying to avoid detection
- Scan events types supported
  - ICMP scans
  - TCP port scans
  - UDP port scans

# Scan Policy Overview

Scan policy provides the ability to:

- Obtain notification and documentation of scanning activity
  - Notify the installation of a detected scan via console message or syslogd message
  - Trace potential scan packets
- Control the parameters that define a scan:
  - The time interval
  - The threshold
- Reduce level of false positives
  - Exclude well known "legitimate scanners" via exclusion list
    - e.g. network management
  - Specify a scan sensitivity level
    - by port for UDP and TCP
    - highest priority rule for ICMP

# Scan Event Counting and Scan Sensitivity

- Scan sensitivity determines whether a scan event is "countable"

| Sensitivity (from policy) | Normal Event | Possibly Suspicious Event | Very Suspicious Event |
|---|---|---|---|
| Low | | | Count |
| Medium | | Count | Count |
| High | Count | Count | Count |

- Countable scan events count against an origin source IP address
  - Total number of countable events for all scan event types is compared to policy thresholds
    - If threshold exceeded for a single IP address, policy-directed notification and documentation is triggered
- Balance between detecting every scan and limit overhead
  - Reserve low ports not explicitly in use to allow configuration of low sensitivity on low ports for both UDP and TCP

# Attacks Against The TCP/IP Stack

- The system already silently defends itself from many attacks against the TCP/IP stack.
- IDS adds capability to control recording intrusion events and supporting documentation.
- IDS adds controls to detect and disable uncommon or unused features which could be used in an attack.

# Attack Categories

- Malformed packet events
  - ➤ Detects packets with incorrect or partial header information
- Inbound fragment restrictions
  - ➤ Detects fragmentation in first 256 bytes of a datagram
- IP protocol restrictions
  - ➤ Detects use of IP protocols you are not using that could be misused
- IP option restrictions
  - ➤ Detects use of IP options you are not using that could be misused
- UDP perpetual echo
  - ➤ Detects traffic between UDP applications that unconditionally respond to every datagram received
- ICMP redirect restrictions
  - ➤ Detects receipt of ICMP redirect to modify routing tables.
- Outbound RAW socket restrictions
  - ➤ Detects z/OS RAW socket application crafting invalid outbound packets
- Flood Events
  - ➤ Detects flood of SYN packets from "spoofed" sources
  - ➤ Detects high percentage of packet discards on a physical interface

# Attack Policy Overview

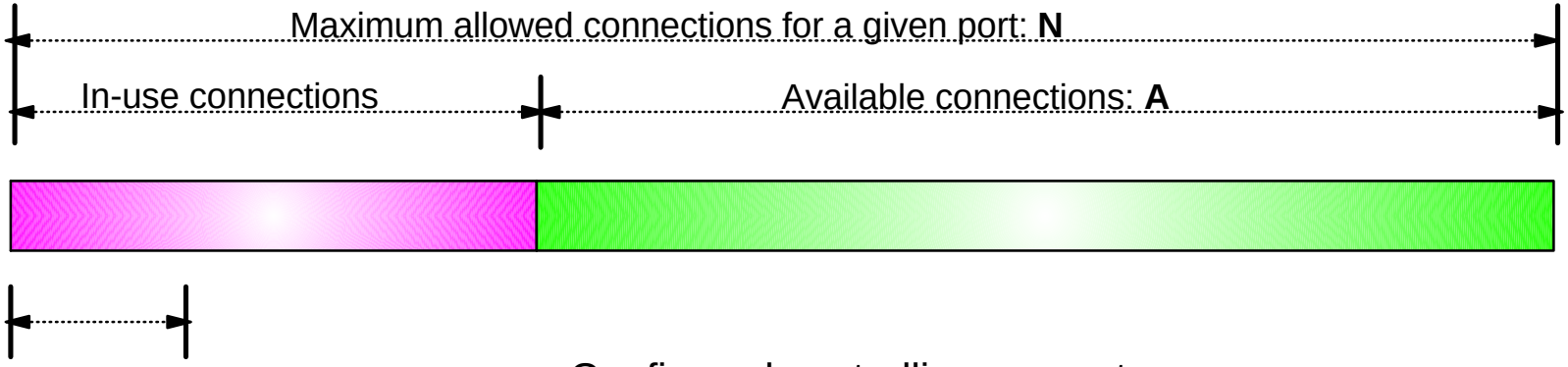
Attack policy provides the ability to:

- Control attack detection for one or more attack categories independently
- Obtain notification and documentation of attacks
  - ➤ Notify the installation of a detected attack via console message or syslogd message
  - ➤ Trace potential attack packets
- Allows request for attack statistics on time interval basis
  - ➤ Normal or Exception
- Control defensive action when attack is detected

# Traffic Regulation for TCP

- Allows control over number of inbound connections from a single host
  - Can be specified for specific application ports
    - Especially for forking applications
  - Independent policies for multiple applications on the same port
    - e.g. telnetd and TN3270
- Connection limit expressed as
  - Port limit for all connecting hosts
  - Individual limit for a single host
- Fair share algorithm
  - Connection allowed if specified individual limit per single remote IP address does not exceed percent of available connections for the port
    - All remote hosts are allowed at least one connection as long as port limit has not been exceeded
      - ✓ QoS connection limit used as override for concentrator sources (web proxy server)

# TCP connection regulation algorithm




If a new connection request is received and A=0, the request is rejected.

If a new connection request is received and A>0 and the request is from a source that already has connections with this port number (in this example: IP@x), then:

If X+1 < P*A then
    Allow the new connection
Else
    Deny the new connection

Purpose: If close to the connection limit, then a given source IP address wil be allowed only a small number of the in-use connections..

# Regulation algorithm example

| Total Allowed | Available | 10% | 20% | 30% | 40% |
|---|---|---|---|---|---|
| 100 | 80 | 8 | 16 | 24 | 32 |
| 100 | 60 | 6 | 12 | 18 | 24 |
| 100 | 40 | 4 | A 8 | 12 | 16 |
| 100 | 20 | 2 | 4 B | 6 | 8 |
| 100 | 10 | 1 | 2 | 3 | 4 |

A If we currently have 60 connections (40 available), the controlling percentage is 20%, and a source IP address tries to establish it's connection number 6, it will be allowed.

B If the number of connections in use rise to 80 (20 available), the controlling percentage is again 20%, and the same source IP address tries to establish it's connection number 6, it will be rejected.

# Traffic Regulation for UDP

- Allows control over length of inbound receive queues for UDP applications
  - ➤ Can be specified for specific application ports
- Before TR for UDP, UDP queue limit control was requested globally for all queues
  - ➤ UDPQueueLimit ON | OFF in TCP/IP Profile
- If neither TR UDP or UDPQueueLimit is used, a stalled application or a flood against a single UDP port could consume all available buffer storage
  - ➤ TR UDP supercedes UDPQueueLimit specification
- TR UDP queue limit expressed as abstract queue length
  - ➤ VERY SHORT
  - ➤ SHORT
    - For applications that consistently receive data at higher rates than can be processed
  - ➤ LONG
  - ➤ VERY LONG
    - Useful for fast applications with bursty arrival rates

# z/OS IDS recording actions

**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

# Tivoli Support for IDS Events

- Tivoli NetView z/OS V5R1, PTF UA11043, provides local z/OS management support for IDS
  - NetView provides ability to trap IDS messages from the system console or syslog and take predefined actions based on IDS event type such as:
    - Route IDS messages to designated NetView consoles
    - email notifications to security admistrator
    - Run trmdstat and attach output to email
    - Issue pre-defined comands
- Tivoli Security Operations Manager provides enterprise-wide management support for IDS
  - Automated aggregation and correlation of events, logs, and vulnerabilities
    - Broad device support for multi-vendor environments, including security, network, host, and applications
    - Support includes processing for z/OS Communications Server syslog messages for IDS events
  - Automates policy and regulatory compliance
    - Policy and Regulatory based policy monitoring and reporting

# Defining IDS Policy

✓See session 3978 for more information on configuring IDS policy


- **In z/OS V1R8 the Policy Agent configuration tools are combined into one tool to manage policies for:**
  - ►AT-TLS
  - ►IPSec and IP packet filtering
  - ►IDS
  - ►QoS
- **Common approach for all policy types:**
  - ►Master copy stored in binary file format (on workstation or file server)
  - ►Text-based configuration files to be parsed by Policy Agent are created and transferred to z/OS

**Note:** IDS policies may now be stored in a text file, just as the other policy types. There is no requirement for LDAP.

- V1R7 - Network Security Configuration Assistant
  - ►Configured policy for IPSec, IP packet filtering, and AT-TLS
- V1R8 - (Name change) Configuration Assistant for z/OS Communications Server
  - ►Adds policy configuration support for IDS and QoS

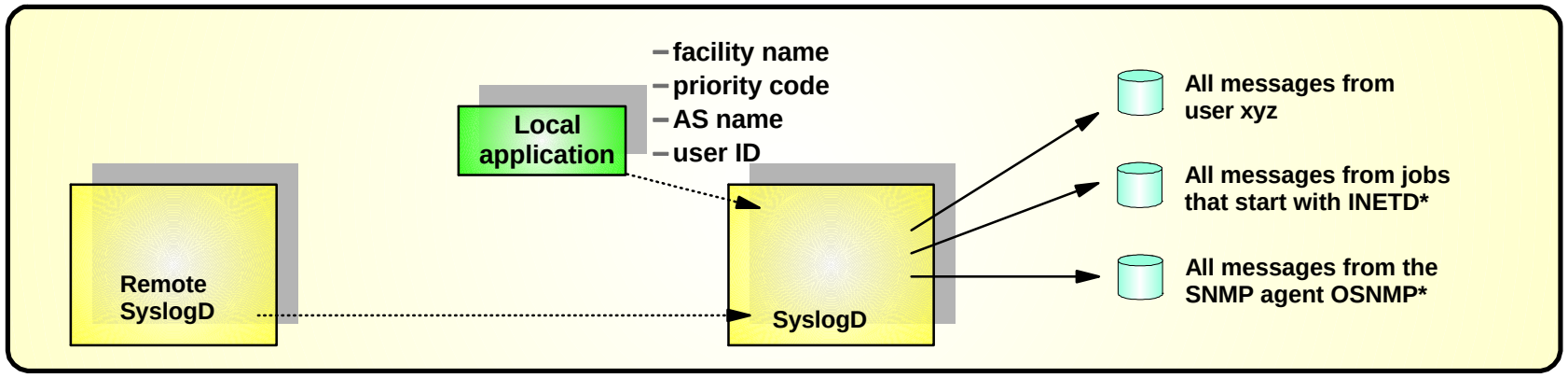
Downloadable policy configuration tool:
http://www-1.ibm.com/support/docview.wss?rs=206&uid=swg21181698

Note: Policy in LDAP can continue to be configured with the zIDS Configuration Manager

# Syslogd Data Integrity and Availability

- **Syslogd integrity and availability goals:**
  - Prevent loss of important system log records due to flooding
    - From network
    - From runaway or malicious applications
  - Keep system log records separate from application log records
    - Ability to audit integrity of syslogd messages

- **z/OS syslogd security controls provide:**
  - Protection from local z/OS users
    - Additional controls to direct syslogd messages to syslogd destinations based on Userid and/or Jobname
    - UserID/Jobname can be traced in log for audit
  - Protection from the network
    - Additional controls to direct syslogd messages to syslogd destinations based on hostname or source IP address (IPv4 or IPv6) **New in z/OS V1R8**
    - Syslogd configuration can turn off reception of log messages via UDP port
      - ✓ Does Not Limit Ability To Send
    - IP filtering can be used to selectively receive syslogd messages from the network



# Controlling Write Access to Syslogd

- Syslogd processing is controlled by configuration file called
  - /etc/syslog.conf
  - defines logging rule conditions and output destinations
    - e.g. files, userids, SMF, syslogd at remote hosts
- Logging rule conditions
  - facility , priority
    - Provided by the application
  - userid , jobname
    - Provided by the system for local logging
  - hostname or IP address
    - Used for syslogd messages received from the IP network
- Conditions specified together as
  - userid. jobname.facility.priority
    - userid and jobname can be wildcarded by '*'

Example
In this example, INETD followed by some other character is the job name of inetd. # Write all messages with priority err or higher from jobname INETD*
which specify facility "daemon" to the log file daemon.

```
*.INETD*.daemon.err*          /var/log/%Y/%m/%d/daemon
```

# Controlling Syslogd Operations

*Syslogd message audit trail by userid and jobname*
*....and syslogd protection from network*

syslogd [-f conffile] [-i][-u[-c[-d][-m markinterval] [-p logpath]
syslogd recognizes the following options:

-f Configuration file name.

**-i** Do not receive messages from the IP network.

**-u** For records received over the AF_UNIX socket (most messages generated on the local system), include the user ID and job name in the record. In this case, a forward slash, the user ID, and the job name will follow the local host name for messages received over the AF_UNIX socket. The forward slash, which immediately follows the local host name, can be used to determine whether or not the user ID and job name is being recorded. If not recorded, a blank immediately follows the local host name. When user ID or job name is not available, N/A will be written in the corresponding field.

-c Create log files and directories automatically.

-d Run syslogd in debugging mode

-m Number of minutes between mark messages.

-p Path name of z/OS UNIX character device for the datagram socket. The default value is /dev/log.

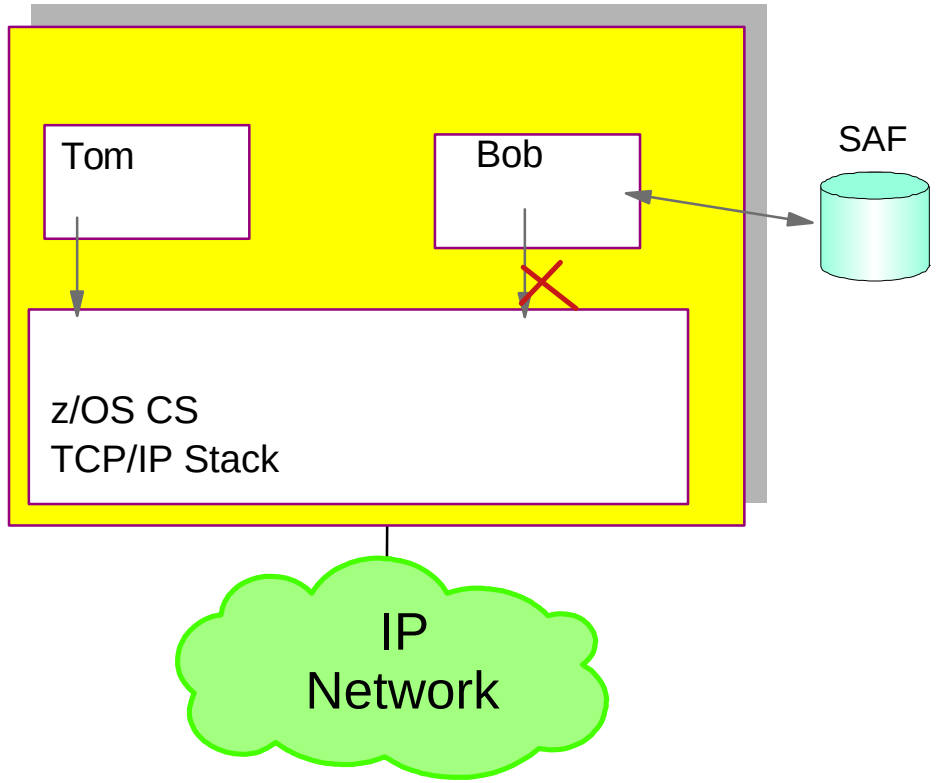
# Protecting Resources on the System with SAF

- All the "traditional" SAF protection of datasets, authorized functions, etc. on a z/OS system applies to TCP/IP workload just as it applies to all other types of workload.
- The SERVAUTH resource class is used to specifically define and protect a number of TCP/IP unique resources
- General SERVAUTH profile format:

  - **`EZB.resource_category.system_name.jobname.resource_name`**
    - EZB designates that this is a TCP/IP profile
    - resource_category is capability area to be controlled e.g. TN3270, Stack Access, etc.
    - system_name is the name of the system - can be wildcarded
    - jobname is the jobname associated with the resource access request - can be wildcarded
    - optional resource_name - one or more qualifiers to indicate name of resource to be protected - can be wildcarded
- To protect one of the supported TCP/IP resources, you define a SERVAUTH profile with universal access NONE and you then permit users to have READ access to the resources
- If using OEM security packages, beware of the differences between defined/not defined resource actions

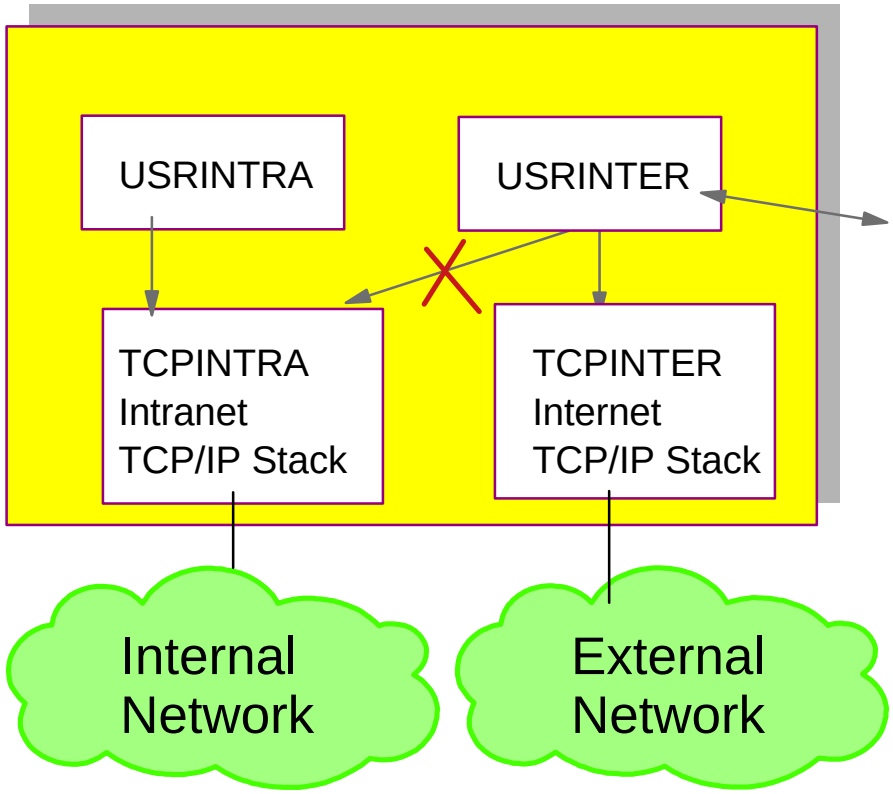
# Stack Access Control

Limits Local Users' Access to TCP/IP Services




- ***Objective is to protect use of TCP resources***
  - **All resources that can be accessed through a specific TCP/IP on z/OS**
    - z/OS CS TCP/IP stack is defined as a SAF resource
  - **Access to stack via TCP or UDP socket allowed if user permitted to SAF resource (SERVAUTH class)**
    - EZB.STACKACCESS.sysname.tcpname
      - ✓ Define the stack resource with UACC(NONE) and permit groups or individual users to allow them to use the TCP/IP stack (open a socket, get hostname or host ID))

# Stack Access Control with Multiple Stacks

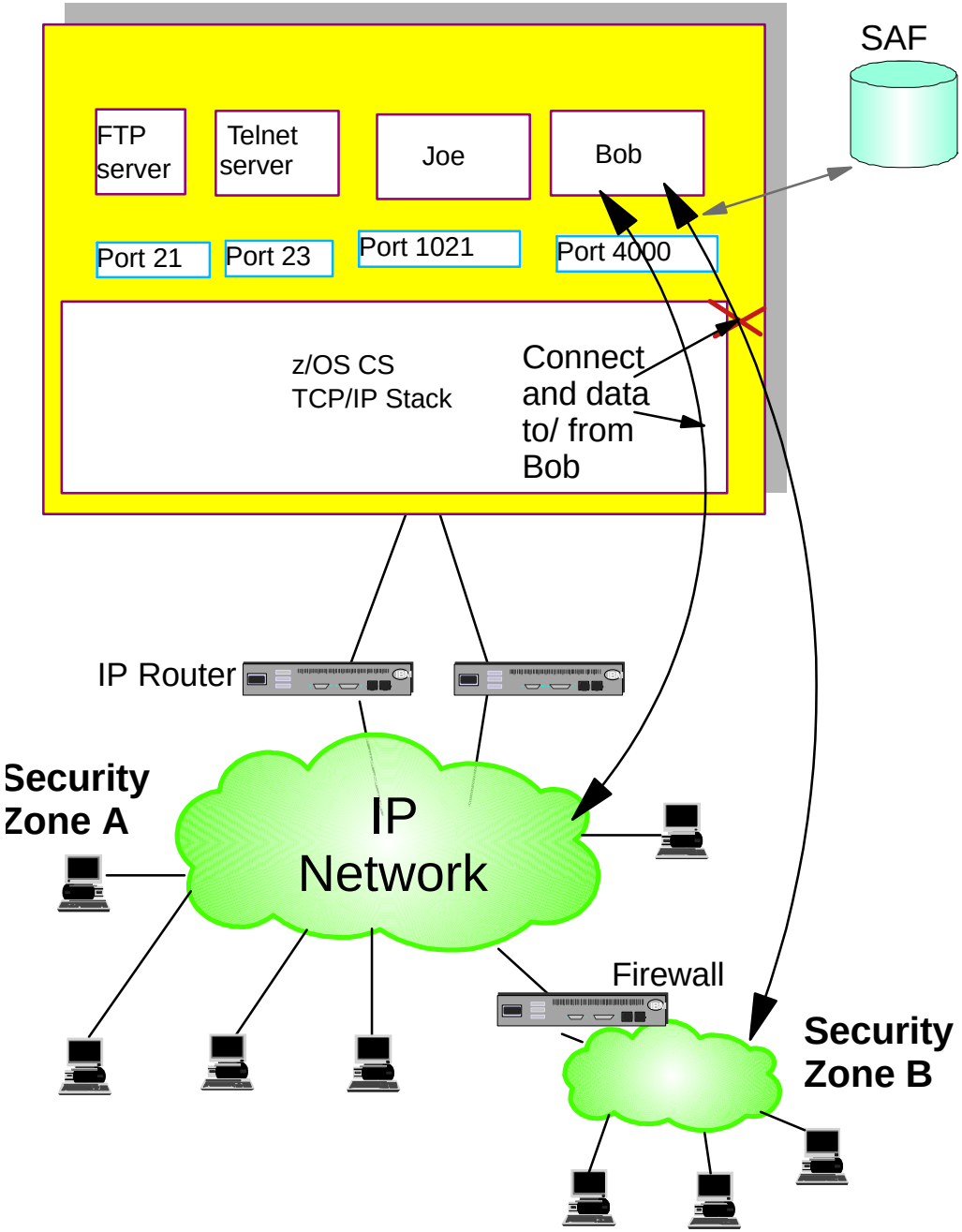



- **With multiple stacks in an LPAR, one must use the Common INET PFS.**
  - **Security-zone specific services must be started using stack affinity so these services only are available in the appropriate security zone.**
- **Use of stacks can be controlled using SAF resources ir the SERVAUTH class:**
  - **EZB.STACKACCESS.LPAR1.TCPINTRA - UACC(None)**
  - **EZB.STACKACCESS.LPAR1.TCPINTER - UACC(NONE)**
- **Then permit the specific started task user IDs to these resources:**
  - **User USRINTRA with READ to EZB.STACKACCESS.LPAR1.TCPINTRA**
  - **User USREXTRA with READ to EZB.STACKACCESS.LPAR1.TCPINTER**

If you are using multiple stacks in z/OS and each of the stacks are connected to different security zones, then you should use stack access control to ensure that only intended z/OS-resident user IDs can use the individual stacks and reach the individual network security zones.

# Network Access Control

## Limit Local Users' Access to Network Segments / Security Zones




- **Controls local user access to network resources**
  - **Network considered a resource**
    - Network/Subnet/Specific host
- **Controls local user ability to explicitly bind to a local address**
- **Allows management of Security Zones**
  - **Via NETACCESS statement In TCP/IP Profile**
    - NETACCESS statement allows grouping of network resources
- **Access to security zone allowed if user permitted to SAF resource (SERVAUTH class)**
    - EZB.NETACCESS.sysname.tcpname.zonename
- **Config to specify whether inbound, outbound, or both directions are subject to Network Access checking**

**TCP/IP Profile definitions:**

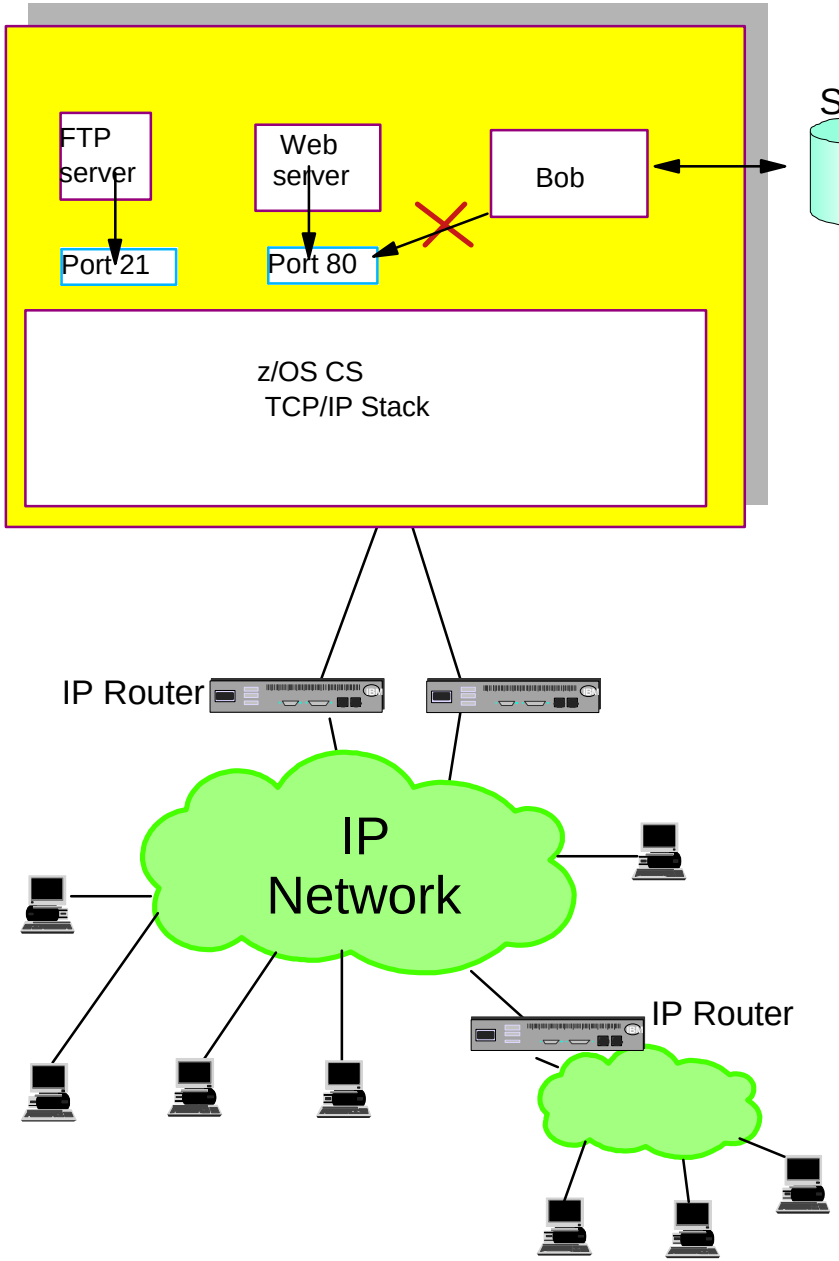
```
NETACCESS  INBOUND OUTBOUND
  9.67.40.0          255.255.248.0        ZONEB
  9.67.0.0/16                             ZONEA
  Default            0                    WORLD
ENDNETACCESS
```

**SERVAUTH resources:**

EZB.NETACCESS.sysname.stackname.ZONEA
EZB.NETACCESS.sysname.stackname.ZONEB
EZB.NETACCESS.sysname.stackname.WORLD

# Port Access Control

## Limit Local Users' Access to non-Ephemeral Ports




- ***Objective is to protect server ports on z/OS through normal SAF controls***
  - **Function enabled**
    - Via SAF Keyword on PORT or PORTRANGE
      - ✓ PORT 80 TCP * SAF WEBSRV
  - **Access to port allowed if user permitted to SAF resource (SERVAUTH class)**
    - EZB.PORTACCESS.sysname.tcpname.SAFname
      - ✓ EZB.PORTACCESS.sysname.tcpname.WEBSRV
    - Universal access is set to NONE, and the started task user ID of the Web Server task is permitted READ access to the resource. Only this user ID can bind to the specified port number.
- **Additional port reservation controls**
  - **Access to port not permitted for any user**
    - Via RESERVED Keyword On PORT Or PORTRANGE
  - **RESTRICTLOWPORTS**
    - Client applications that need to be able to bind to low-ports are APF authorized clients (LPR, LPQ, LPRM, and RSH)

# Other SERVAUTH Protections

- Command protection
  - ipsec
  - pasearch
  - netstat
- Network management APIs
  - packet trace
  - real-time SMF data
  - connection data
- Application control
  - broadcast socket options
  - IPv6 advanced socket APIs
- Other resource restrictions
  - FTP dump and site commands
  - HFS file access via FTP
  - fast response cache accelerator (FRCA) page load
  - snmp subagent access
  - dvipa modification control

See appendix for a complete list of resource types that can be protected using the SERVAUTH profiles.

# z/OS Communications Server Security
# TCP/IP System Protection Features

- Protecting system resources and data from the network
  - IP packet filtering
    - Controls traffic that is allowed to enter and leave the system
  - Integrated Intrusion Detection Services
    - Detects, records, and defends against scans, stack attacks, flooding
  - Syslogd isolation
    - Makes sure UNIX system services log data is available when needed
  - SAF protection of z/OS resources
    - z/OS CS application access to data sets and files
    - SERVAUTH class protection for TCP resources

# z/OS Communications Server Security

Appendix
SERVAUTH Resources

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| TN3270 server access control | Controls ability to access TN3270 server based on SAF userid associated with TLS authenticated X.509 client certificate | EZB.TN3270.sysname.tcpname.PORTxxxxx | OS/390 V2R8 |
| FTP server access control | Controls ability to access FTP server based on SAF userid associated with TLS authenticated X.509 client certificate | EZB.FTP.sysname.ftpdaemonname.PORTxxxxx | z/OS V1R2 |
| DCAS server access control | Controls ability to access DCAS server based on SAF userid associated with TLS authenticated X.509 client certificate | EZB.DCAS.cvtsysname | OS/390 V2R10 PTF |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| TCP stack access control | Controls user ability to open socket and get host name or host ID | EZB.STACKACCESS.sysname.tcpname | OS/390 V2R10 |
| TCP local port access control | Controls user ability to bind to a non-ephemeral TCP or UDP port | EZB.PORTACCESS.sysname.tcpname.port_safname | OS/390 V2R10 |
| TCP netaccess access control | Controls local users inbound and outbound access to network resources. Controls local user access to local IP address when explicitly binding to local interface (or using job-specific source IP addresses) | EZB.NETACCESS.sysname.tcpname.security_zonename | OS/390 V2R10 |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| Netstat command access control | Provides ability to restrict Netstat usage | EZB.NETSTAT.sysname.tcpname.netstat_option | z/OS V1R2 |
| Policy Agent command control | Provides ability to restrict pasearch command usage by policy type | EZB.PAGENT.sysname.tcpname.policy_type | z/OS V1R2 |
| FTP SITE command control | Provides ability to restrict usage of SITE DUMP and DEBUG commands (commands generate large amount of output) | EZB.FTP.sysname.ftpdname.SITE.DUMP<br>EZB.FTP.sysname.ftpdname.SITE.DEBUG | z/OS V1R2 |
| SNMP agent control | Provides ability to control usage of SNMP subagents that connect to the TCP/IP SNMP agent | EZB.SNMPAGENT.sysname.tcpname | z/OS V1R2 |
| MODDVIPA utility program control | Provides ability to restrict usage of MODDVIPA utility program (creates new DVIPA on system) | EZB.MODDVIPA.sysname.tcpname | z/OS V1R2 |
| Fast Response Cache Accelerator (FRCA) Access Control | Provides ability of user to create FRCA cache (FRCA used by webservers for caching static web pages in the stack) | EZB.FRCAACCESS.sysname.tcpname | z/OS V1R4 |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| TCP connection information service access control | Provides ability to restrict access to the TCP connection information using TCP connection information service intended for network management applications. | EZB.NETMGMT.sysname.tcpname.SYSTCPCN | z/OS V1R5 |
| Real-time SMF information service access control | Provides ability to restrict access to select real-time SMF records accessable using the SMF information service intended for network management applications. | EZB.NETMGMT.sysname.tcpname.SYSTCPSM | z/OS V1R5 |
| TCP/IP packet trace service access control | Provides ability to restrict access to select real-time packet trace records accessable using the TCP/IP packet trace service intended for network management applications. | EZB.NETMGMT.sysname.tcpname.SYSTCPDA. | z/OS V1R5 |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| FTP HFS access control | Provides ability to generally restrict FTP user access to HFS. | EZB.FTP.sysname.ftpdaemonname.ACCESS.HFS | z/OS V1R5 |
| Broadcast access control | Provides ability to control whether an application is permitted to set the SO_BROADCAST socket option needed to send broadcast datagrams. | EZB.SOCKOPT.sysname.tcpname.SO_BROADCAST | z/OS V1R6 |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| IPv6 Advanced Socket API access control | Provides ability to control whether an application is permitted to set IPv6 advanced socket API options:<br>IPv6_NEXTHOP<br>IPv6_TCLASS<br>IPv6_RTHDR<br>IPV6_HOPOPTS<br>IPV6_DSPOPTS<br>IPV6_RTHDRDSTOPT<br>IPV6_PKTINFO<br>IPV6_HOPLIMIT | EZB.SOCKOPT.sysname.tcpname.IPV6_NEXTHOP<br>EZB.SOCKOPT.sysname.tcpname.IPV6_TCLASS<br>EZB.SOCKOPT.sysname.tcpname.IPV6_RTHDR<br>EZB.SOCKOPT.sysname.tcpname.IPV6_HOPOPTS<br>EZB.SOCKOPT.sysname.tcpname.IPV6_DSTOPTS<br>EZB.SOCKOPT.sysname.tcpname.IPV6_RTHDRDSTOPTS<br>EZB.SOCKOPT.sysname.tcpname.IPV6_PKTINFO<br>EZB.SOCKOPT.sysname.tcpname.IPV6_HOPLIMIT | V1R7 |
| AT-TLS early stack access control | Controls applications ability to open a socket before AT-TLS policy is loaded into the TCP/IP stack | EZB.INITSTACK.sysname.tcpname | V1R7 |

# SERVAUTH Profiles

| Function | Description | SERVAUTH Profile | Req'd Release |
|---|---|---|---|
| CIM provider access control | Provides ability to restrict access to CIM data | EZB.CIMPROV.sysname.tcpname | V1R7 |
| ipsec command access control | Provides ability to control ipsec command usage | EZB.IPSECCMD.sysname.tcpname.command_type | V1R7 |
| FTP z/OS UNIX file system access control | Provides ability to generally control FTP user access to the z/OS UNIX file system | EZB.FTP.sysname.ftpdaemonname.ACCESS.HFS | V1R7 |

# For More Information....

| URL | Content |
|---|---|
| http://www.ibm.com/servers/eserver/zseries | IBM eServer zSeries Mainframe Servers |
| http://www.ibm.com/servers/eserver/zseries/networking | Networking: IBM zSeries Servers |
| http://www.ibm.com/servers/eserver/zseries/networking/technology.html | IBM Enterprise Servers: Networking Technologies |
| http://www.ibm.com/software/network/commserver | Communications Server product overview |
| http://www.ibm.com/software/network/commserver/zos/ | z/OS Communications Server |
| http://www.ibm.com/software/network/commserver/z_lin/ | Communications Server for Linux on zSeries |
| http://www.ibm.com/software/network/ccl | Communication Controller for Linux on zSeries |
| http://www.ibm.com/software/network/commserver/library | Communications Server products - white papers, product documentation, etc. |
| http://www.redbooks.ibm.com | ITSO redbooks |
| http://www.ibm.com/software/network/commserver/support | Communications Server technical Support |
| http://www.ibm.com/support/techdocs/ | Technical support documentation (techdocs, flashes, presentations, white papers, etc.) |
| http://www.rfc-editor.org/rfcsearch.html | Request For Comments (RFC) |